

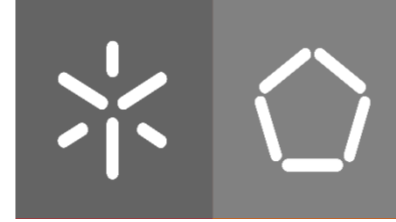


**Utilização de Técnicas de *Machine Learning*
na Previsão de Movimentos Incorretos de
Trabalhadores em Contexto Industrial**

Cristiana Raquel Novais Fernandes

Universidade do Minho
Escola de Engenharia





Universidade do Minho
Escola de Engenharia

Cristiana Raquel Novais Fernandes

**Utilização de Técnicas de *Machine Learning* na
Previsão de Movimentos Incorretos de
Trabalhadores em Contexto Industrial**

Dissertação de mestrado
Mestrado Integrado em Engenharia e Gestão de
Sistemas de Informação

Trabalho realizado sob orientação de:
Professor Doutor Paulo Alexandre Ribeiro Cortez

DIREITOS DE AUTOR E CONDIÇÕES DE UTILIZAÇÃO DO TRABALHO POR TERCEIROS

Este é um trabalho académico que pode ser utilizado por terceiros desde que respeitadas as regras e boas práticas internacionalmente aceites, no que concerne aos direitos de autor e direitos conexos.

Assim, o presente trabalho pode ser utilizado nos termos previstos na licença abaixo indicada.

Caso o utilizador necessite de permissão para poder fazer um uso do trabalho em condições não previstas no licenciamento indicado, deverá contactar o autor, através do RepositóriUM da Universidade do Minho.

Licença concedida aos utilizadores deste trabalho



Atribuição-NãoComercial

CC BY-NC

<https://creativecommons.org/licenses/by-nc/4.0/>

AGRADECIMENTOS

A todos aqueles que tornaram estes anos inesquecíveis e a todos os que me apoiaram ao longo deste percurso, que agora termina, dirijo os meus agradecimentos.

Ao Professor Doutor Paulo Cortez, agradeço pelo acompanhamento, pelos ensinamentos e disponibilidade demonstrados no decorrer deste último ano.

Ao André Pilastrri e a todos os colegas da equipa CCG/EPMQ, agradeço o companheirismo e entre-ajuda, com um especial agradecimento ao Luís Matos por todo o conhecimento com ele adquirido. Agradeço também aos parceiros da Fraunhofer, nomeadamente ao Duarte Folgado e Maria Lua, pelo apoio na fase inicial deste trabalho.

À minha mãe e irmã, agradeço por me acompanharem neste percurso com o carinho e amor de sempre. As conquistas sabem melhor se forem partilhadas, e esta eu partilho-a especialmente convosco.

Aos amigos que fiz e que levo comigo para sempre, termino com a certeza de que o meu futuro continuará a ser vivido e partilhado convosco.

Ao Diogo, obrigada pelo apoio incansável e por celebrares o dobro de mim as minhas conquistas e por partilhares comigo as tuas.

Resta a saudade do que foi vivido e o entusiasmo no futuro que ainda está por viver.

This work is a result of the project STVgoDigital - Digitalization of the T&C sector (POCI-01-0247-FEDER-046086), supported by COMPETE 2020, under the PORTUGAL 2020 Partnership Agreement, through the European Regional Development Fund (ERDF).

DECLARAÇÃO DE INTEGRIDADE

Declaro ter atuado com integridade na elaboração do presente trabalho académico e confirmo que não recorri à prática de plágio nem a qualquer forma de utilização indevida ou falsificação de informações ou resultados em nenhuma das etapas conducente à sua elaboração.

Mais declaro que conheço e que respeitei o Código de Conduta Ética da Universidade do Minho.

Resumo

Hoje em dia, as indústrias de manufatura ainda enfrentam dificuldades para aplicar os métodos tradicionais de avaliação do risco de *work-related musculoskeletal disorder* (WMSD) em aplicações práticas devido ao esforço que é necessário para a recolha contínua de dados de métodos observacionais. Uma solução interessante passa por adotar *Inertial Measurement Units* (IMU), que podem ser utilizados para automatizar a recolha e processamento de dados, dando apoio a ergonomistas e profissionais de saúde ocupacional. Neste trabalho, é feita uma comparação entre vários algoritmos de *Machine Learning* para a previsão do movimento, incluindo: três métodos de *Long Short-Term Memory* (LSTM) - uma LSTM normal com apenas uma *layer*, uma *Stacked LSTM* e um *Sequence to Sequence* (Seq2Seq); e três métodos de regressão - um *Multiple Linear Regression*, um *Random Forest* e um *Support Vector Machine*. O objetivo é prevenir movimentos problemáticos que podem surgir durante movimentos repetitivos de trabalho. O sistema proposto inclui a aplicação inicial do *Madgwick orientation filter* para transformar os dados em bruto dos sensores inerciais, numa série temporal de orientação de ângulo único, de forma a monitorizar o ângulo de abdução/adução do braço. O modelo Seq2Seq LSTM alcançou os melhores resultados, sendo posteriormente avaliado considerando 11 conjuntos de dados de seres humanos e dois procedimentos de avaliação (treino e teste com dados de uma só pessoa e utilizando dados obtidos de múltiplas pessoas). Estas avaliações demonstraram um excelente potencial do modelo preditivo desenvolvido para antecipar movimentos problemáticos. Em trabalho futuro, espera-se integrar tal modelo num protótipo de exoesqueleto capaz de bloquear movimentos perigosos.

Palavras-Chave: *Work-related Musculoskeletal Disorders*; Fusão de sensores; Regressão; *Deep Learning*.

Abstract

Nowadays, manufacturing industries still face difficulties applying traditional work-related musculoskeletal disorder risk (WMSD) assessment methods in practical applications due to the high effort for continuous data collection of observational methods. An interesting solution is to adopt inertial motion capture systems, namely Inertial Measurement Units (IMU), which can be used to automate data collection and processing, supporting ergonomists and occupational health professionals. In this work, distinct Machine Learning algorithms were compared for a single angle orientation time series prediction, including: three Long Short-Term Memory (LSTM) methods – a one layer, a stacked layer and a Sequence to Sequence (Seq2Seq) model; and three non-deep learning methods – a Multiple Linear Regression, a Random Forest and a Support Vector Machine. The goal is to prevent problematic movements that can arise during repetitive working movements. The proposed system includes an initial Madgwick filter to merge the raw inertial sensor data into a single angle orientation time series, aiming to monitor the abduction/adduction angle of the arm. As the Seq2Seq LSTM achieved the best results, this model was further evaluated for WMSD prevention by considering 11 human subject datasets and two evaluation procedures (single person and multiple person training and testing). The main goal was to achieve a model with excellent capability to anticipate problematic movements and capable of being integrated into a solution for WMSDs, like exoskeletons.

Keywords: Work-related Musculoskeletal Disorders; Sensor Fusion; Regression; Deep Learning.

Índice

Resumo	I
Abstract	III
Lista de Figuras	V
Lista de Tabelas	VII
Lista de Abreviaturas, Siglas e Acrónimos	VIII
1 Introdução	1
1.1 Enquadramento e Motivação	1
1.2 Formulação do Problema	2
1.3 Objetivos e Resultados Esperados	2
1.4 Abordagem Metodológica	3
1.5 Organização do Documento	6
2 Revisão da Literatura	7
2.1 Estratégia de Pesquisa Bibliográfica	7
2.2 Mapa Conceitual	8
2.3 Machine Learning	8
2.4 Deep Learning	16
2.5 Work-related Musculoskeletal Disorders (WMSDs)	20
2.6 Human Motion Capture	22
2.6.1 Sensor Fusion Para Estimativa da Orientação	24
2.6.2 Ergonomic Risk Assessment Tools	26
2.7 Previsão de Movimentos com Base em Dados Inerciais	28
2.8 Ferramentas Tecnológicas	31
3 Caso de Estudo: Previsão de Movimentos Prejudiciais aos Trabalhadores com Base em Dados Inerciais	33
3.1 Compreensão do Negócio	34
3.2 Compreensão dos Dados	35
3.3 Preparação dos Dados	37
3.4 Modelação	40
3.5 Avaliação	43
3.6 Implementação	49
4 Conclusões	50
4.1 Síntese do Trabalho Realizado	50
4.2 Contributos	51

4.3	Trabalho Futuro	52
	Referências Bibliográficas	54
a	Anexos	63
a.1	Reconstrução Angular dos Movimentos após o Madgwick orientation filter.	63
a.2	Código Python Utilizado para a Implementação dos Modelos	64
a.3	Parâmetros dos Modelos Utilizados	64
a.4	Pontos de Análise Para Cada Um dos Participantes	66
a.5	Avaliação das Experiências	66

Lista de Figuras

Figura 1	Metodologia CRISP-DM tradicional (adaptado de Wirth and Hipp, 2000).	5
Figura 2	Metodologia CRISP-DM adaptada (adaptado de Taifi, 2020).	5
Figura 3	Mapa conceitual.	8
Figura 4	Enquadramento concetual (adaptado de Taifi, 2020).	9
Figura 5	Fases genéricas de um modelo de <i>Machine Learning</i> (retirado de Alzubi et al., 2018).	11
Figura 6	Arquitetura de uma <i>Decision Tree</i> (retirado de Charbuty and Mohsin Abdula-zeez, 2021).	12
Figura 7	Arquitetura do algoritmo <i>Random Forest</i> (retirado de Verikas et al., 2016).	13
Figura 8	Arquitetura do algoritmo <i>XGBoost</i> (retirado de Wang et al., 2019).	13
Figura 9	Arquitetura do algoritmo <i>LightGBM</i> (retirado de Ju et al., 2019).	14
Figura 10	Neurónio de McCulloch-Pitts	17
Figura 11	Ilustração de uma ANN (retirado de Dertat, 2017).	18
Figura 12	Comparação entre RNN e FNN (retirado de Eliasy and Przychodzen, 2020).	18
Figura 13	Ilustração de uma memory cell (retirado de Yu et al., 2019).	19
Figura 14	Incidência de WMSDs numa indústria automóvel portuguesa (adaptado de Guerreiro et al., 2017).	21
Figura 15	Sistema Vicon (retirada de Esteves, 2007).	23
Figura 16	Sensores inerciais (retirada de Pereira, 2016).	23
Figura 17	Ilustração do processo de <i>sensor fusion</i> .	24
Figura 18	Funcionamento do <i>Madgwick orientation filter</i> (retirado de Cirillo et al., 2016).	26
Figura 19	Colocação dos IMUs em cada um dos participantes e sistema de referência (retirado de Santos et al., 2020b).	36
Figura 20	<i>Workflow</i> do processamento dos dados.	38
Figura 21	Exemplo dos dados recolhidos pelo IMU 3 colocado no braço direito.	39
Figura 22	Exemplo da reconstrução angular do movimento de abdução/adução do braço para o participante A depois de utilizar o Madgwick orientation filter.	40
Figura 23	Arquitetura do Stacked LSTM (adaptado de Dertat, 2017).	41
Figura 24	Arquitetura do LSTM Seq2Seq (adaptado de Dertat, 2017; Xiang et al., 2020).	42
Figura 25	Primeira fase de modelação.	42
Figura 26	Segunda fase de modelação, através de um procedimento de avaliação que utiliza dados de múltiplas pessoas para treino e teste.	43

- Figura 27 Exemplo do movimento do participante A; a linha tracejada horizontal demarca o nível de risco de WMSDs imposto pela norma ISO 11226; o círculo vermelho representa os pontos de análise que antecedem um movimento que ultrapassa o risco de WMSDs. 45
- Figura 28 Comparação dos resultados obtidos para os três pontos críticos do participante A para $h = 1$ e $h = 25$. 46
- Figura 29 Reconstrução angular do movimento de abdução/adução de todos os participantes depois de utilizar o Madgwick orientation filter. A vermelho estão assinalados os pontos de análise. 63

Lista de Tabelas

Tabela 1	Fatores de risco associadas às WMSDs (retirado de Santos et al., 2020a).	20
Tabela 2	Algoritmos de <i>sensor fusion</i> disponíveis no AHRS (retirado do website AHRS ¹).	25
Tabela 3	Revisão da Literatura.	29
Tabela 4	Ferramentas tecnológicas.	32
Tabela 5	Sumário dos seis sets de movimentos realizados.	36
Tabela 6	Síntese dos dados utilizados.	37
Tabela 7	Resultados para cada modelo (primeira fase de modelação, $h = 1$, os melhores valores a negrito).	44
Tabela 8	Resultados das previsões obtidas pelo LSTM Seq2Seq para a pessoa A com três pontos de risco (t_r).	45
Tabela 9	Scores para a antecipação de perigo do ombro, considerando os dados de treino do participante A ($\tau = 1$).	48
Tabela 10	Scores para a antecipação de perigo do ombro, considerando os dados de treino de vários participantes ($\tau = 1$).	49
Tabela 11	Pontos que antecedem um momento que ultrapassa o risco de WMSDs para cada um dos participantes.	66

Lista de Abreviaturas, Siglas e Acrônimos

<i>AHRS</i>	<i>Attitude and Heading Reference System.</i>
<i>ANN</i>	<i>Artificial Neural Network.</i>
<i>CCG</i>	<i>Centro de Computação Gráfica.</i>
<i>CRISP-DM</i>	<i>Cross Industry Standard Process for Data Mining.</i>
<i>DL</i>	<i>Deep Learning.</i>
<i>DNN</i>	<i>Deep Neural Network.</i>
<i>EKF</i>	<i>Extended Kalman Filter.</i>
<i>FNN</i>	<i>Feed-Forward Neural Network.</i>
<i>IA</i>	<i>Inteligência Artificial.</i>
<i>I&D</i>	<i>Investigação e Desenvolvimento.</i>
<i>IMUs</i>	<i>Inertial Measurement Units.</i>
<i>IoT</i>	<i>Internet of Things.</i>
<i>KDD</i>	<i>Knowledge Discovery from Data.</i>
<i>LR</i>	<i>Linear Regression.</i>
<i>LSTM</i>	<i>Long short-term Memory.</i>
<i>MAE</i>	<i>Mean Absolute Error.</i>
<i>ML</i>	<i>Machine Learning.</i>
<i>MLP</i>	<i>Multilayer Perceptrons.</i>
<i>MoCap</i>	<i>Motion Capture.</i>
<i>MSE</i>	<i>Mean Squared Error.</i>
<i>R²</i>	<i>Coefficient of Determination.</i>
<i>RF</i>	<i>Random Forest.</i>
<i>RMSE</i>	<i>Root Mean Squared Error.</i>
<i>RNN</i>	<i>Recurrent Neural Network.</i>

<i>Seq2Seq</i>	<i>LSTM Sequence to Sequence.</i>
<i>STV</i>	<i>Setor Têxtil e Vestuário.</i>
<i>SVM</i>	<i>Support Vector Machines.</i>
<i>SVR</i>	<i>Support Vector Regression.</i>
<i>TSF</i>	<i>Time Series Forecasting.</i>
<i>WMSDs</i>	<i>Work-related Musculoskeletal Disorders.</i>

Introdução

1.1 Enquadramento e Motivação

O Setor Têxtil e Vestuário (STV) é dos setores nacionais mais antigos, mas também mais importantes, onde se verifica um enorme potencial para o seu desenvolvimento e um claro aumento do ritmo de trabalho. Dados apontam para que este seja dos setores “mais importantes da economia nacional, devido à criação de emprego e de riqueza, com um peso de 4% no PIB, em 2016” (DGAE, 2018). Um crescimento também notório no que diz respeito ao nível de produção, ao volume de negócios e ao nível de exportação. Surge então uma vontade enorme de aliar o crescimento deste setor com tecnologia de ponta, nomeadamente com a adoção de tecnologias inerentes à Indústria 4.0, pelo que se começa a incentivar as empresas a investirem em soluções tecnológicas que permitam não só aumentar a competitividade como impulsionar o desenvolvimento tecnológico neste setor.

Esta dissertação é realizada no contexto do projeto mobilizador STVgoDIGITAL: “Digitalização da cadeia de valor do Setor Têxtil e Vestuário”, em específico no domínio intitulado de “Trabalhador 4.0”, que está alinhado com um dos pilares estratégicos do *Cluster* Têxtil: Tecnologia e Moda - Indústria 4.0 e Digitalização. Este projeto tem como promotor líder a TMG – Tecidos para Vestuário e Decoração, S.A, sob a orientação técnica do CITEVE, cujos objetivos principais são a promoção e inovação do STV, através da adoção de tecnologias e aplicações; o aumento da competitividade no STV, objetivo também ele aliado à visão do Cluster Têxtil de tornar Portugal um dos países mais competitivos a nível Mundial; e, por fim, o objetivo central de melhorar a qualidade de vidas das trabalhadoras têxteis, em específico das costureiras da empresa têxtil PAFIL - *Clothing Industry*, utilizador final da solução a desenvolver.

Assim, pretende-se a conceção de um exoesqueleto em base têxtil, flexível, vestível e adaptável ao corpo e a qualquer tipo de movimento, assim como a criação de uma plataforma de tratamento de dados, essencial para monitorizar a atuação do exoesqueleto. Esta solução pretende colmatar um problema identificado neste tipo de indústrias em que o trabalho exige a realização de movimentos muito repetitivos e que, por vezes, levam a uma má postura dos trabalhadores, as *work-related musculoskeletal disorders* (WMSDs), em português, distúrbios músculo-esqueléticos, associados ao trabalho. É, por isso, essencial que a solução resulte num melhor desempenho e postura ergonómica das costureiras, diminuindo o risco de lesões e aumentando a produtividade das mesmas.

Num leque de diversas entidades e empresas, o CCG (Centro de Computação Gráfica), localizado no Campus de Azurém da Universidade do Minho, faz parte do desenvolvimento deste projeto. De entre

todas as tecnologias e domínios envolvidos, tais como *Internet Of Things (IoT)*, domínio de sensorização e atuação e *Machine Learning (ML)*, é neste último que incidirá o trabalho desenvolvido nesta dissertação.

São já longos os anos de investigação e desenvolvimento (*I&D*) em torno dos exoesqueletos, permitindo que, para além do seu uso centralizado na recuperação física, exista cada vez mais uma maior preocupação no aproveitamento desta tecnologia na melhoria da qualidade de trabalho e no auxílio à execução de tarefas profissionais. Com o contributo dos algoritmos e modelos de *Machine Learning*, que devem ser capazes de se adaptar rapidamente a qualquer tipo de movimento, espera-se alcançar uma solução prática e adequada que auxilie na capacidade motora das trabalhadoras.

Embora se verifique a impossibilidade de trabalhar com dados recolhidos na empresa para o desenvolvimento desta dissertação, o trabalho realizado enquadrar-se-á com o contexto do projeto de investigação, procurando cumprir os objetivos do mesmo, através da aplicação de algoritmos de *Machine Learning* para a previsão de movimentos com base em dados recolhidos por sensores inerciais.

1.2 Formulação do Problema

A existência e idealização de qualquer projeto é sempre fundamentada pela vontade de investir esforços para atacar um problema e procurar uma solução. Assim, a presente dissertação focar-se-á na aplicação de técnicas e modelos de *Machine Learning* que se mostrem adequados e capazes de se adaptarem aos dados provenientes de sensores inerciais, possibilitando a previsão dos movimentos. O *dataset* utilizado, embora num contexto diferente ao do projeto mobilizador, corresponde às necessidades do mesmo, uma vez que também este faz uso de módulos de sensores, designados de *IMUs (Inertial Measurement Units)*, responsáveis pela captura dos movimentos exercidos.

A utilização deste módulo de sensores acresce a complexidade do problema, uma vez que dela surge a necessidade de implementar técnicas de *sensor fusion* para transformar os dados *raw* provenientes dos sensores, em dados legíveis e passíveis de serem interpretados quer pelos algoritmos de *Machine Learning* quer pelas normas de avaliação de postura. Estas normas permitem tecer uma avaliação ergonómica dos movimentos, identificando aqueles que são prejudiciais e que podem vir a causar as *WMSDs*.

O foco incidir-se-á na utilização de dados inerciais recolhidos numa linha de montagem de uma indústria automóvel para a previsão do movimento de abdução/adução do braço. Este trabalho pretende contribuir para auxiliar na resolução do problema das lesões dos trabalhadores e sustentar a possibilidade desta aplicação no contexto industrial do projeto mobilizador, o setor têxtil.

1.3 Objetivos e Resultados Esperados

Se a existência de um problema dá azo à criação de um projeto, os objetivos nele definidos traçam o caminho que deve ser seguido para atingir uma solução com poder para resolver o problema. Naturalmente, esta dissertação reúne um conjunto de objetivos, também eles alusivos ao projeto STVgoDigital,

que garantem o cumprimento das necessidades identificadas. Complementando a necessidade inicial de estudar os conceitos e tecnologias necessárias, de uma maneira geral, o trabalho focar-se-á em dois grandes objetivos principais, um deles na área de *sensor fusion* para o processamento dos dados provenientes dos sensores, e o segundo na componente de *Machine Learning*. Estes objetivos, que são sumariados de seguida, permitirão atingir o objetivo principal de prever os movimentos incorretos através da utilização de dados provenientes de sensores inerciais.

1. Estudar e utilizar o processo de *Sensor Fusion* no processamento dos dados: estudo do conceito e identificação das diversas técnicas de *sensor fusion*, garantindo a compreensão e a identificação daquelas que podem ser utilizadas em contextos semelhantes ao do presente problema. Naturalmente, o grande objetivo será a implementação de uma técnica capaz de utilizar os dados provenientes dos sensores, transformando-os em valores de ângulos. Este passo será essencial como ponte de ligação entre os dados em bruto recolhidos pelos sensores e os algoritmos de *Machine Learning* a aplicar;
2. Estudar e utilizar técnicas de *Machine Learning*: estudo e aplicação de algoritmos de *Machine Learning* para a previsão dos movimentos incorretos, de forma a detetar aqueles que devem ser evitados e corrigidos. Para este passo, e como prova de conceito, será também importante aplicar um parâmetro da norma de avaliação de postura ISO 11226: *Ergonomics – Evaluation of static working postures*, que será também utilizada no contexto do projeto mobilizador;

1.4 Abordagem Metodológica

Da procura incessante de cumprir com os objetivos traçados, surge a necessidade de utilizar uma metodologia que permita orientar o trabalho a desenvolver. A metodologia *CRISP-DM (Cross Industry Process for Data Mining)* será a metodologia utilizada nesta dissertação, uma vez que a mesma permite diminuir a complexidade de projetos de *Data Mining*, tornando-os, por um lado, menos dispendiosos e repetíveis, mas, por outro, mais confiáveis, mais fáceis de gerir e rápidos (Wirth and Hipp, 2000).

Tradicionalmente, esta metodologia apresentada na Figura 1, é composta por seis fases diferentes sendo que cada uma delas é composta por uma série de tarefas de segundo nível, que podem ou não ser aplicadas. Nesta dissertação, a forma adaptada desta metodologia, apresentada na Figura 2, é utilizada para descrever o processo de implementação de *Machine Learning*, reduzindo a sua complexidade e garantindo o cumprimento dos requisitos.

Esta abordagem mais ajustada da tradicional metodologia *CRISP-DM* representa a real complexidade de projetos deste tipo. Idealmente, as transições entre as seis fases deveriam ser mínimas seguindo um processo o mais linear possível, no entanto, a realidade de projetos complexos de *Machine Learning* mostra que as transições entre as diferentes etapas ao longo de um projeto são uma constante e, acima de tudo, necessárias (Taifi, 2020). Assim, também no presente projeto se verifica a necessidade de recuar nas fases para que se cumpram os requisitos do mesmo e o código e modelo de *Machine Learning* implementados reflitam este processo. As seis fases da metodologia são apresentadas e descritas de seguida,

resumindo, desta forma, as iterações que serão necessárias no decorrer do desenvolvimento do trabalho.

1. **Business Understanding (Compreensão do Negócio):** Fase em que importa perceber as necessidades e requisitos do negócio e de que forma é possível convertê-las num problema de *Data Mining* capaz de alcançar essas mesmas necessidades e requisitos. Naturalmente, esta é uma fase em que muitas vezes é preciso recuar, mesmo quando o projeto se encontra já numa fase mais avançada, como por exemplo na fase de *deployment*, uma vez que alguns requisitos podem se alterar no decorrer do projeto. Neste contexto, o problema enquadra-se com a necessidade de implementação de um modelo de *Machine Learning* capaz de prever os movimentos realizados e, conseqüentemente, capaz de alertar para os movimentos incorretos de forma a evitar lesões músculo-esqueléticas.
2. **Data Understanding (Compreensão dos Dados):** Fase com foco na reunião dos dados e consequente compreensão e familiarização dos mesmos, incluindo a identificação dos problemas que terão de ser resolvidos. Em todas as fases pode se mostrar necessário recuar até aqui, seja pela alteração ou surgimento de novos dados. No presente problema, sabemos que os dados são provenientes de sensores inerciais triaxiais, em concreto dos sensores acelerómetro, magnetómetro e giroscópio, e que estes necessitam de um processamento.
3. **Data Preparation (Preparação dos Dados):** Fase importante do projeto em que se resolve os problemas do *dataset* encontrados na fase anterior. Normalmente, ocorre a seleção dos atributos e linhas a utilizar, a transformação e a limpeza dos dados necessários para que se possa, finalmente, aplicar técnicas e modelos de *Machine Learning*. Nas etapas de *modeling* e *deployment* pode ser necessário voltar a esta fase para, por exemplo, melhor adaptação dos dados. Uma das principais necessidades identificadas nos dados a trabalhar, é a aplicação de técnicas que permitam a adaptação dos dados em bruto provenientes dos sensores inerciais em valores de ângulos.
4. **Modeling (Modelação):** Fase de seleção de técnicas e modelos de *Machine Learning* e respetiva adaptação dos parâmetros de modo a atingir os melhores resultados possíveis. Também por isso se verifica as transições das fases de *evaluation* e *deployment*, pois são fases que permitem perceber se o modelo obtido na fase de modelação é, de facto, o mais adequado.
5. **Evaluation (Avaliação):** Fase de avaliação do desempenho dos modelos obtidos, através de determinadas métricas e técnicas. É uma fase extremamente crucial para perceber se os requisitos de negócio definidos na primeira fase estão de facto a ser cumpridos. Caso contrário, poderá ser necessário reconsiderar a fase inicial de *business understanding* ou de *data preparation* ou ainda recuar até à fase anterior de modelação.

6. **Deployment (Implementação):** A implementação não significa, necessariamente, o fim do projeto. Por isso mesmo, esta fase pode ter diferentes objetivos conforme os requisitos da organização, como por exemplo a criação de relatórios.

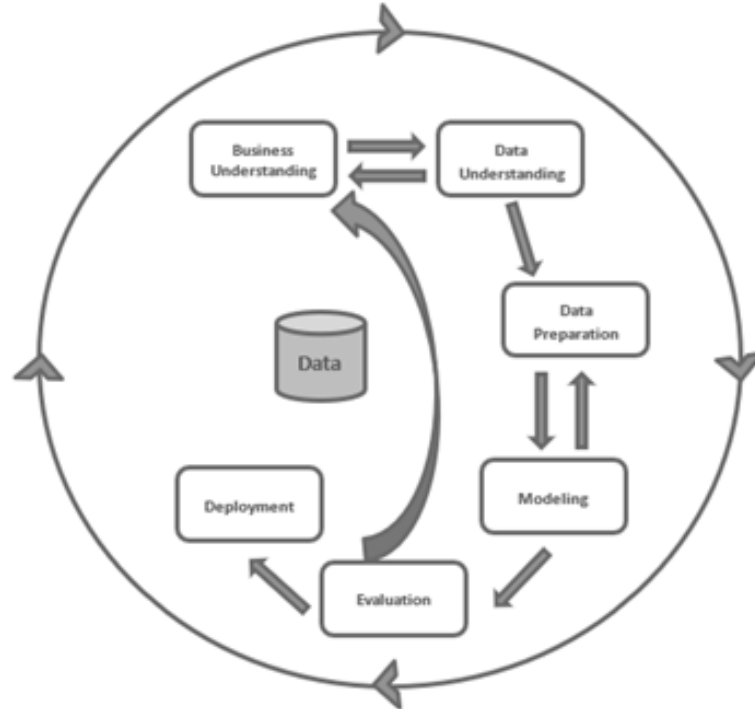


Figura 1: Metodologia CRISP-DM tradicional (adaptado de Wirth and Hipp, 2000).

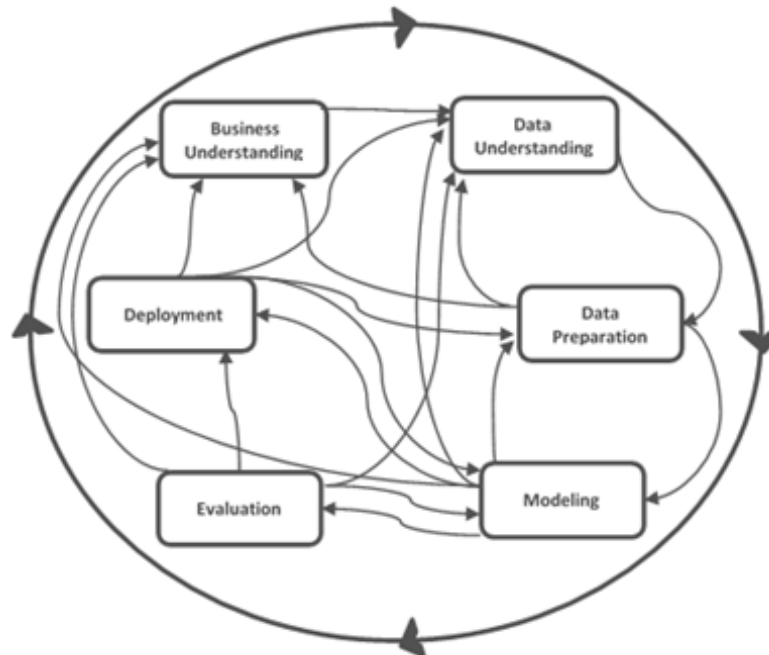


Figura 2: Metodologia CRISP-DM adaptada (adaptado de Taifi, 2020).

1.5 Organização do Documento

Este documento está dividido em quatro capítulos principais que, no seu conjunto, permitem atingir e demonstrar os objetivos e resultados esperados. No Capítulo 1 (Introdução), é efetuado um enquadramento e motivação do trabalho, assim como a formulação do problema em questão e objetivos e resultados esperados. Para além disso, é também apresentada a metodologia utilizada, demonstrando e apresentando todas as fases da metodologia *CRISP-DM*.

No Capítulo 2 é feita a revisão da literatura, onde são esclarecidos e aprofundados todos os conceitos relativos ao tema, quer teóricos quer práticos, nomeadamente sobre os conceitos de *Machine Learning* e *Deep Learning*, mas também sobre *Work-related musculoskeletal disorders*, *Inertial measurement units*, *sensor fusion* e as suas técnicas, e normas de avaliação de postura. Para além disso, são apresentados vários estudos onde são utilizadas técnicas de *Machine Learning* ou métodos de captura de movimento e/ou ferramentas de avaliação ergonómica de postura. Por fim, são apresentadas as ferramentas utilizadas para a realização do projeto.

No Capítulo 3 é apresentado e explorado o caso de estudo onde é realizada a previsão dos movimentos com base em dados inerciais. Esta demonstração tem como base a estrutura da metodologia *CRISP-DM*, permitindo uma demonstração mais completa e detalhada do trabalho desenvolvido. Aqui são apresentados os passos realizados assim como os resultados obtidos.

Por fim, o Capítulo 4 reúne as considerações finais relativas ao trabalho desenvolvido, aos contributos fornecidos pelo mesmo e à investigação e objetivos futuros.

Revisão da Literatura

2.1 Estratégia de Pesquisa Bibliográfica

A revisão da literatura é uma parte fulcral de qualquer projeto de investigação, que permite reunir informação de estudos anteriores e responder de forma pertinente e com valor científico às questões impostas pelo problema em questão (Snyder, 2019). De forma a cumprir com este pressuposto, é importante fazer uma pesquisa inteligente e escolher referências bibliográficas adequadas que possibilitem obter um conhecimento atual e rigoroso do tópico. Desde logo, entende-se a necessidade de escolher fontes de informação credíveis, científicas e precisas.

A grande maioria das pesquisas foram efetuadas através do motor de pesquisa *Google Scholar*, com a expectativa de encontrar livros e artigos científicos de relevância, que mostrassem ser uma mais valia para o desenvolvimento do problema e a sua compreensão. Não obstante, também foi efetuada uma pesquisa em *blogs* e *websites* que se mostraram úteis para a compreensão dos conceitos abordados. Para além da relevância do conteúdo, foi frequentemente tida em consideração o ano da publicação para que a informação aqui apresentada fosse a mais atualizada e próxima da realidade possível, pelo que a bibliografia escolhida foi publicada a partir do ano de 2017, à exceção daquelas que, apesar de serem mais antigas, se mostraram ser muito relevantes. Para além disso, outros aspetos foram tidos em consideração, como por exemplo o número de citações.

Com o tema da dissertação assente na “Utilização de Técnicas de *Machine Learning* na Previsão de Movimentos Incorretos dos Trabalhadores em Contexto Industrial”, é natural o foco da pesquisa nas técnicas de *Machine Learning*, procurando desde logo perceber quais os algoritmos que poderão ser utilizados na previsão dos movimentos. Para além disso, considerando o contexto industrial do problema e as tecnologias envolvidas na captura dos dados utilizados, a pesquisa abrangeu também conceitos relacionados com essas condicionantes, como por exemplo o conceito de *sensor fusion* e o conceito de *musculoskeletal disorders*.

Apesar do tema da presente dissertação ser o foco principal da revisão literária, é pertinente fazer uma ponte de ligação entre o tema e o projeto mobilizador que o sustenta, havendo também, por isso, espaço para um estudo relacionado diretamente com o projeto de investigação, nomeadamente com o desenvolvimento de exoesqueletos e do contexto têxtil em que está envolvido. Assim, após a seleção da literatura que pareceu mais pertinente, o conteúdo apresentado no presente documento é resultado dessa pesquisa e apoiado cientificamente pelas referências bibliográficas apresentadas.

2.2 Mapa Conceitual

De forma a organizar os conceitos apresentados neste capítulo, cuja abordagem é necessária na compreensão do contexto do trabalho, a Figura 3 sintetiza os tópicos de seguida explorados. O projeto está assente na utilização das tecnologias de *Machine Learning* para ajudar a solucionar o problema das *WMSDs* na indústria, utilizando para isso dados provenientes da captura humana através de sensores inerciais. Alinhado com a utilização dos algoritmos de *Machine Learning*, é também necessária a aplicação de técnicas de *sensor fusion* no processamento dos dados inerciais e a aplicação de uma norma de avaliação de postura, de forma a identificar os movimentos prejudiciais. A exploração destes conceitos permitirá obter um conhecimento mais aprofundado daquilo que é necessário para resolver o problema em mãos.

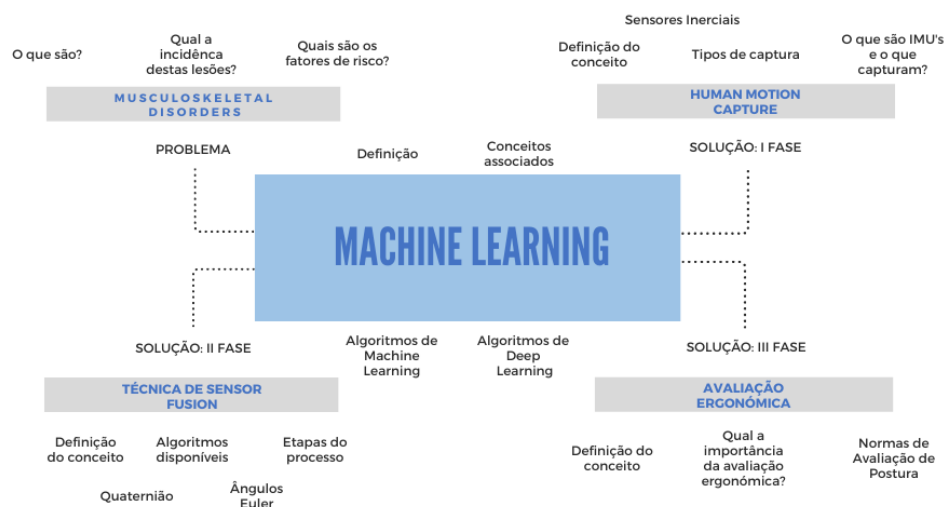


Figura 3: Mapa conceitual.

2.3 *Machine Learning*

Os avanços tecnológicos aliados à vontade das empresas de se tornarem cada vez mais dinâmicas e competitivas, torna o conceito de *Artificial Intelligence* cada vez mais presente nas mais diversas áreas e tarefas, evidenciando-se uma enorme procura na aplicação de tecnologias de ponta para simplificar os seus processos e atividades. Importa por isso perceber de que forma estas tecnologias se relacionam e de que forma as mesmas podem impactar positivamente as indústrias. As tecnologias inerentes à Inteligência Artificial (*IA*), nomeadamente as que são apresentadas de forma relacional na Figura 4, têm vindo a garantir o seu espaço quer nas áreas académicas, nomeadamente na investigação, quer na indústria com projetos estimulantes para as empresas.

O termo “inteligência” remete-nos não só para a capacidade de conseguir utilizar o conhecimento adquirido e as nossas competências para resolver problemas e responder a questões, como para aprender com essas mesmas situações (Shabbir and Anwer, 2018). Inteligência Artificial é, por isso, um ramo

da ciência que torna as máquinas aptas a desempenhar tarefas que dependeriam de inteligência se fossem realizadas pelo Homem. John McCarthy, conhecido como um dos principais responsáveis pelo nascimento da Inteligência Artificial, descreve este ramo como um comportamento capaz de simular o cérebro humano e, conseqüentemente, capaz de tomar decisões e resolver problemas (Fang et al., 2018).

Este é uma tecnologia que, pelas vantagens associadas e pela sua aplicabilidade, tem ganho cada vez mais força nas mais diversas áreas, verificando-se um forte crescimento no seu investimento. Em 2018, 36% dos negócios na União Europeia e nos Estados Unidos da América, estavam a investir de forma ativa em projetos de Inteligência Artificial (Fedak, 2018).

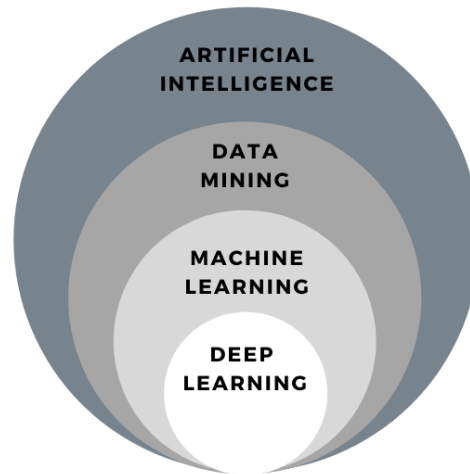


Figura 4: Enquadramento conceitual (adaptado de Taifi, 2020).

Os conceitos de *Data Mining* e *Machine Learning* são frequentemente associados a projetos de elevado grau de complexidade e aplicados nas mais diversas áreas de negócio. É, por isso, um desafio empregar técnicas e ferramentas adequadas para que destes projetos surja um aproveitamento vantajoso, nomeadamente no aumento da competitividade. Com o passar dos anos, o crescente desenvolvimento de *hardware* e ferramentas de armazenamento de dados trouxe consigo associado o desafio complexo de se conseguir lidar com uma quantidade tão volumosa de dados. Inevitavelmente, torna-se cada vez mais difícil para as empresas saberem como lidar e armazenar esses dados e, conseqüentemente, como tirar o melhor proveito dos mesmos.

Data Mining, também usualmente designado de *Knowledge discovery from data (KDD)*, é definido como, “*process of discovering or extracting interesting patterns, associations, changes, anomalies and significant structures from large amounts of data which is stored in multiple data sources such as file systems, databases, data warehouses or other information repositories*” (Han et al., 2011). De forma a descobrir esses padrões, podem ser utilizadas diversas técnicas e algoritmos, como é o exemplo dos algoritmos de *Machine Learning* (Gupta and Chandra, 2020). Estes conceitos são regularmente confundidos, uma vez que ambos acabam por ter a mesma finalidade, mas poder-se-á dizer que *Machine Learning* é uma parte integrante do *Data Mining*, nomeadamente na etapa de modelação, uma das seis etapas da metodologia *CRISP-DM* apresentada anteriormente.

O processo de *Data Mining* está também presente no setor têxtil, onde se verifica a sua forte aplicação em sistemas de apoio à decisão, sistemas de recomendação, análise de dados visuais, recuperação de informações ou ainda gestão de armazenamento de dados uma vez que, nesta área, são produzidas enormes quantidades de dados até nas mais simples atividades de produção (Yildirim et al., 2018a).

A definição de *Machine Learning* ou, em português, “uma máquina que aprende”, está diretamente relacionada, tal como o próprio nome indica, com a capacidade de tornar uma máquina capaz de aprender sem a intervenção do toque humano, estando esta apta a perceber e adaptar-se à complexidade do problema (Alzubi et al., 2018). O uso de técnicas e modelos de *Machine Learning*, é o que torna a máquina capacitada para extrair conhecimento relevante a partir dos dados (Murdoch et al., 2019). *Deep Learning (DL)* como um sub ramo de *Machine Learning*, diferencia-se pela sua aprendizagem mais eficaz e económica com maiores quantidade de dados, permitindo atingir um elevado nível de precisão em contextos mais exigentes, como por exemplo em problemas de reconhecimento facial (Kumar and Dargan, 2019).

Com a adoção de tecnologias da indústria 4.0 para a transformação digital e a criação de valor nas manufaturas, também neste tipo de indústrias se verifica uma forte aposta na utilização de algoritmos de *Deep Learning* para a resolução de problemas, uma vez que estas são indústrias que, à partida, produzem grandes quantidades de dados. Para além da sua eficácia em contextos de *big data*, o facto de transformar os dados em representações abstratas de alto nível, faz com que esta técnica de aprendizagem se adapte muito bem à variedade de dados (Wang et al., 2018).

Machine Learning tem vindo a ser desenvolvido desde 1950, com o primeiro contributo por parte de Alan Turing que acreditava ser possível programar um computador ao ponto de o tornar capaz de aprender ou descobrir novas regras, da mesma forma que os matemáticos conseguiam (Piccinini, 2003). Mais recentemente, em 2014, o Facebook inventou o *Deep Face*, um sistema de reconhecimento facial através de fotografias, criado com um algoritmo baseado no *Deep Neural Networks*. Também no ramo têxtil se verifica a aplicação de técnicas de *Machine Learning*, como por exemplo através de *Artificial Neural Network* para prever o consumo de determinada peça, através de *Support Vector Machines* para identificar o tipo de fibra a utilizar ou através de *Decision Trees* para prever o envelhecimento dos tecidos (Yildirim et al., 2018b).

Independentemente do algoritmo a ser treinado, existem seis passos genéricos de um modelo de *Machine Learning*, que devem ser seguidos para que, no fim, se possa avaliar o desempenho do modelo e, consequentemente, verificar se o mesmo responde às necessidades identificadas para dar resposta ao problema. Essas fases são apresentadas na Figura 5.

- **Collection and Preparation of Data (Levantamento e preparação dos dados):** Fase de reunião dos dados e do respetivo tratamento, para que os mesmos possam então ser utilizados na fase de modelação.
- **Feature Selection (Seleção de Recursos):** Fase de seleção dos atributos e variáveis relevantes para a aprendizagem e construção do modelo.

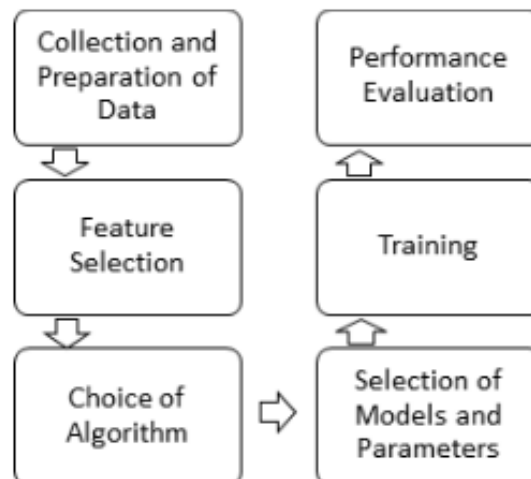


Figura 5: Fases genéricas de um modelo de *Machine Learning* (retirado de Alzubi et al., 2018).

- **Choice of Algorithm (Seleção do Algoritmo):** Fase de escolha de algoritmos que são aplicáveis ao problema e que melhor satisfaçam as necessidades.
- **Selection of Models and Parameters (Seleção do Modelo e Parâmetros):** Aplicação de uma série de critérios de modo a avaliar e, conseqüentemente, escolher o modelo e parâmetros mais adequados.
- **Training (Treino do Modelo):** Fase em que efetivamente se treina os dados, usando uma parte do *dataset* designado por *training set*.
- **Performance Evaluation (Avaliação do Desempenho):** É a fase que permite perceber, através de uma série de métricas, como por exemplo a *accuracy*, se de facto o modelo está a aprender e se pode efetivamente ser implementado.

Aprendizagem Supervisionada

A aprendizagem supervisionada é também conhecida como a aprendizagem através de exemplos ou até mesmo como aprendizagem “com professor”. Neste tipo de aprendizagem, o algoritmo recebe como *input* um conjunto de exemplos rotulados e o seu *output* correto. O algoritmo faz então uso dessa informação, comparando o *output* atual com o *output* correto, de forma a encontrar os erros, aperfeiçoando assim o modelo preditivo (Alzubi et al., 2018). Após esta fase de treino, segue a fase de teste para avaliar a precisão do algoritmo (Praveena and Jaiganesh, 2017). Durante do processo de teste, o modelo vai tentar prever um novo conjunto de amostras e, dependendo da natureza do *target* em questão, pode-se classificar este tipo de aprendizagem em dois tipos de problemas: problemas de classificação e problemas de regressão (Nasteski, 2017).

Num problema de classificação, o *output* é um valor discreto e é utilizado, por exemplo, para verificar se um email é ou não *spam*. Num problema de regressão, o *output* é um valor contínuo, ou seja, um valor quantitativo. Pode ser utilizado, por exemplo, para prever o valor de uma habitação (Castle, 2018). Embora com propósitos diferentes, ambos têm o objetivo principal de alcançar o melhor modelo possível, ou seja, minimizar a diferença entre o valor previsto e o valor atual. Para além disso, funcionam da mesma maneira: os dados são divididos em duas partes, uma para o treino e outra para o teste, onde neste último se verifica o que foi aprendido na fase de treino (Stellwall, 2019).

São vários os algoritmos utilizados neste tipo de aprendizagem, de seguida são mencionados e descritos alguns deles.

- **Decision Trees:** é um algoritmo bastante utilizado quer em problemas de classificação quer em problemas de regressão, sendo, por exemplo, bastante utilizado em contextos de previsão. O seu funcionamento consiste na divisão do *dataset* em pequenos *subdatasets*, em que a raiz da árvore de decisão representa o *dataset* inteiro e cada ramo representa uma decisão (Fernandes, 2019). A avaliação começa na raiz da árvore e termina na última folha, sendo que cada folha corresponde a uma decisão. Cada nó está rotulado com o atributo a ser testado e cada um dos ramos desse nó apresenta o respetivo valor do *output* (Nasteski, 2017).

Esta abordagem do problema, faz com que este algoritmo seja uma excelente opção para diminuir a complexidade, tornando o processo de decisão mais simples. Para além disso, é um algoritmo também útil na identificação de padrões de relação entre as variáveis (Damanik et al., 2019). Estes algoritmos são também muitas vezes utilizados em problemas de processamento de imagem (Charbuty and Mohsin Abdulazeez, 2021).

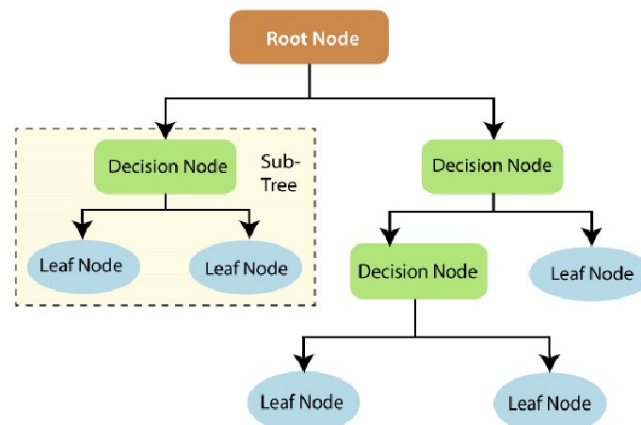


Figura 6: Arquitetura de uma *Decision Tree* (retirado de Charbuty and Mohsin Abdulazeez, 2021).

- **Random Forest (RF):** este algoritmo é também ele baseado num conjunto de árvores de decisão, podendo ser utilizado quer em problemas de classificação quer em problemas de regressão. Cada uma destas árvores é construída e treinada com recursos selecionados aleatoriamente, obtendo cada uma delas um resultado diferente. Como cada uma destas árvores treina recursos diferentes, cada uma delas vai ter também previsões diferentes. Assim sendo, a previsão final é obtida através

do cálculo da média de cada uma das previsões obtidas pelas diferentes árvores de decisão (Zhang et al., 2021). Como se pode ver na Figura 7, para cada um dos dois tipos de problemas, o processo para alcançar um resultado final é diferente. Num problema de classificação, cada árvore tem a sua classificação e vota para essa classe, enquanto que num problema de regressão, a previsão final é obtida calculando a média das previsões de cada uma das árvores.

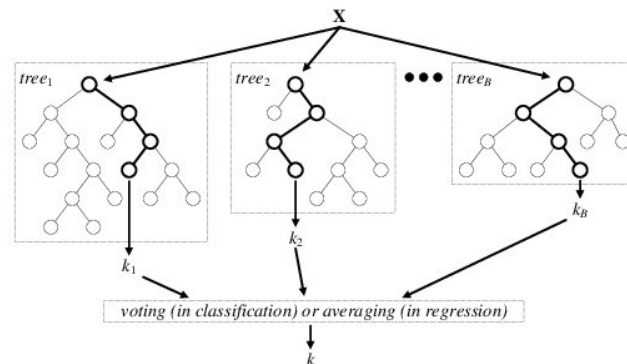


Figura 7: Arquitetura do algoritmo *Random Forest* (retirado de Verikas et al., 2016).

- **XGBoost:** é outro algoritmo baseado num conjunto de árvores de decisão, que utiliza a técnica *boosting* para melhorar os resultados de previsão, sendo, por isso, um método que obtém frequentemente um bom desempenho neste tipo de tarefas (Pesantez-Narvaez et al., 2019). Como é possível verificar na Figura 8, e ao contrário do *Random Forest*, cada árvore de decisão treina através dos resultados obtidos pelas árvores anteriores. Assim sendo, o resultado final deste algoritmo representa a soma de todos os resultados obtidos pelas árvores de decisão (Wang et al., 2019).

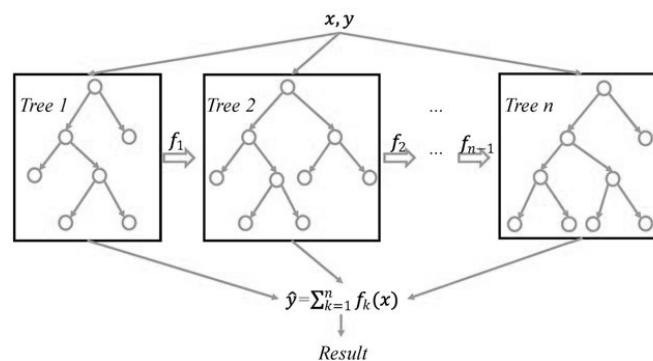


Figura 8: Arquitetura do algoritmo *XGBoost* (retirado de Wang et al., 2019).

- **LightGBM:** é um algoritmo baseado em árvores de decisão, com a particularidade de crescer verticalmente e, conseguir assim assegurar o seu desempenho com quantidades de dados mais elevadas (Sun et al., 2020). Na Figura 9, é perceptível que a divisão ocorre em termos de folha, fazendo com que se reduza as perdas normais de um crescimento em profundidade, como acontece noutros algoritmos em árvore.

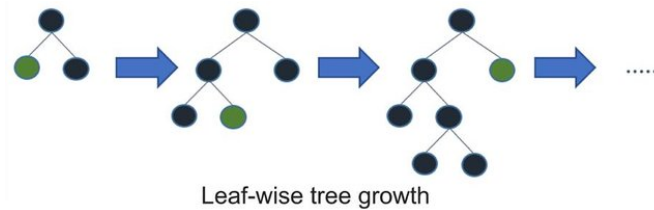


Figura 9: Arquitetura do algoritmo *LightGBM* (retirado de Ju et al., 2019).

- **Linear Regression (LR):** este é um dos algoritmos mais simples e fáceis de implementar, que avalia a relação entre as variáveis, assumindo que a mesma se trata de uma relação linear. Costuma ser adequado para *datasets* mais pequenos, mas podem ser menos adequados para *datasets* com muitas variáveis preditivas em consideração (Hope, 2020).
- **Logistic Regression :** é um algoritmo de classificação binária, em que os valores finais podem ser 0 ou 1 (Nelson, 2020). Faz uso de diversas variáveis, sejam elas numéricas ou categóricas, e ajusta-as numa função logística para prever a probabilidade de ocorrência de determinado evento (Nasteski, 2017).
- **Support Vector Machines (SVM):** é um algoritmo que pode ser utilizado para problemas quer de classificação quer de regressão. É caracterizado pelo mapeamento do *input* (das variáveis independentes) a um determinado *target* (a variável dependente) (Tinoco et al., 2014). Este algoritmo é uma boa opção para a identificação de padrões em problemas mais complexos, como por exemplo no reconhecimento facial (HUANG et al., 2018).
- **Support Vector Regression (SVR):** este algoritmo, baseado em *Support Vector Machines* mas em contexto de problemas de regressão, costuma ter uma boa *performance* ao lidar com dados dimensionais elevados (Li et al., 2020a). Analisa a relação entre as variáveis preditivas e uma variável contínua de valor real (Zhang and O'Donnell, 2020).
- **ARIMA:** o *Auto-Regressive Integrated Moving Average*, é um modelo linear que prevê os valores futuros baseado nos valores passados, com a forma $ARIMA(p,d,q)$. Cada uma destas componentes tem o seguinte significado: p (*Autoregressive*), o número de observações de latência utilizadas no treino; d (*integration*), o número de vezes que a diferenciação é aplicada e q (*moving average*) é o tamanho da janela de *moving average* (Aasim et al., 2019; Siami-Namini and Namin, 2018). Os dados devem ser estacionários, e a sua verificação pode ser feita através do teste *Augmented Dickey-Fuller* (ADF). Para além disso, os valores ideais para cada um dos três parâmetros anteriormente mencionados, pode ser conseguido através da função `auto_arima`.
- **Holt-Winters Exponential Smoothing:** este método de séries temporais utiliza uma combinação de suavização exponencial das observações anteriores para gerar a previsão seguinte (Djakaria and Saleh, 2021; Jiang et al., 2020). Ao contrário no modelo *ARIMA*, este modelo melhora o seu desempenho quando há padrões sazonais e de tendência bem definidos.

Aprendizagem Não-Supervisionada

A aprendizagem não-supervisionada é uma aprendizagem “sem professor”, quer isto dizer que, ao contrário da aprendizagem supervisionada, os dados a treinar não têm com eles associado um *output*. Por isso mesmo, é importante descobrir a estrutura por trás desses dados (Alzubi et al., 2018). Nesta aprendizagem, é necessário criar um modelo que seja capaz de extrair os padrões e determinar, por si mesmo, quais são os importantes. Este modelo procura as semelhanças nos dados e organiza-os por grupos/categorias (Nelson, 2020).

A esta aprendizagem, está associada a técnica de *Clustering*, utilizada para encontrar características semelhantes de modo a agrupar os dados de forma lógica. Esta técnica pode ser utilizada, por exemplo, para descobrir segmentos de clientes de determinado negócio (Stellwall, 2019). O algoritmo seguinte é o mais popularmente utilizado no que diz respeito a este tipo de aprendizagem:

- **K-Means Clustering:** o objetivo deste algoritmo é agrupar um conjunto de “n” observações num conjunto “k” de *clusters*, tendo como base as características dos dados observados (Alzubi et al., 2018). Assim, está pressuposto que o algoritmo é capaz de atribuir cada observação a um *cluster* (Trevino, 2016).

Aprendizagem Semi-Supervisionada

A aprendizagem semi-supervisionada, tal como o próprio nome sugere, é uma junção das duas aprendizagens anteriormente referidas. Reúne as características de ambos, que são adaptadas para situações em que alguns dados podem estar rotulados mas outros não (Alzubi et al., 2018). Este tipo de aprendizagem pode ser vantajosa essencialmente por dois aspetos: (1) na medida em que rotular todos os dados para a aprendizagem supervisionada pode ser demorado e caro e (2) incluir dados não rotulados pode ser vantajoso para o modelo e para a sua precisão final, uma vez que rotular todos os dados implica que aos mesmos esteja associado algum julgamento humano (Castle, 2018).

Aprendizagem Ativa

A aprendizagem ativa é um tipo de aprendizagem em que, na presença de dados não rotulados, surge a hipótese de consultar uma fonte de informação, que pode ser o chamado “professor”, para rotular esses mesmos dados com o respetivo *output* (Settles, 2009). Em muitos casos, para que os dados possam ser treinados, existem custos e muitas horas associadas à rotulação manual desses mesmos dados. Este tipo de aprendizagem é vantajosa uma vez que permite reduzir essas despesas. Outra vantagem da aprendizagem ativa está no facto de que, ao selecionar os dados corretamente, há uma clara diminuição do risco de não se cumprir os requisitos de dados (Schohn and Cohn, 2000). Este tipo de aprendizagem, que é uma forma de aprendizagem semi-supervisionada, pode ser utilizado, por

exemplo, no reconhecimento de voz e na extração de informação de documentos, uma vez que ambos possuem informação muito detalhada e muitas vezes ambígua (Settles, 2009).

Aprendizagem por Reforço

É comum associar este tipo de aprendizagem à aprendizagem não supervisionada, uma vez que em ambos não é fornecida informação para sustentar uma decisão. No entanto, na aprendizagem por reforço, não se procura encontrar a estrutura dos dados, mas sim experimentar, através da metodologia de tentativa-erro, até perceber quais são as ações que mais recompensam. Deste modo, a máquina irá conseguir atuar corretamente mesmo que a situação em mãos não esteja presente no conjunto de treino (Sutton and Barto, 2018).

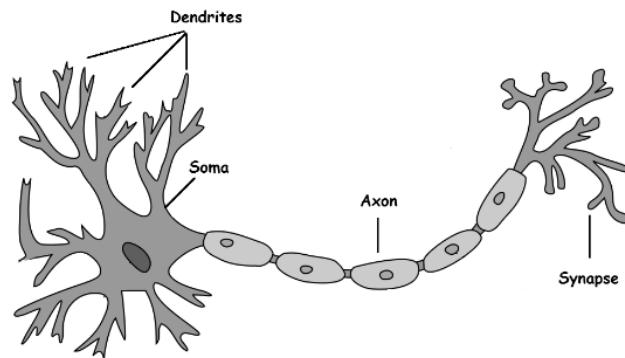
Pode-se ainda dizer que neste tipo de aprendizagem, é importante que a máquina aprenda a otimizar o seu comportamento para que possa evitar as consequências negativas e potencializar as recompensas, aplicando essa aprendizagem a uma sequência de ações (Dayan and Niv, 2008). Há então duas características associadas a este tipo de aprendizagem: a aprendizagem por tentativa-erro e a recompensa atrasada, sendo que esta última se deve ao facto de, por vezes, a recompensa da aprendizagem não ser imediata, mas que se verifica em situações seguintes (Sutton and Barto, 2018).

2.4 Deep Learning

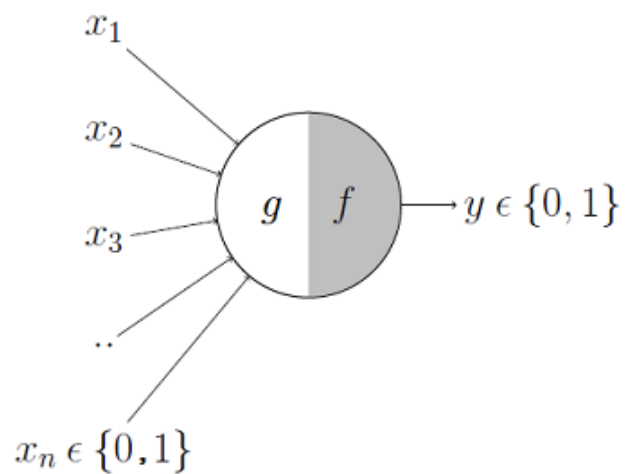
Sendo uma abordagem mais robusta de *Machine Learning*, este ramo apresenta maior poder computacional e melhor capacidade em lidar com maiores volumes de dados, tornando o conceito de *Deep Learning* muito vantajoso e cada vez mais explorado ao longo dos anos, embora esta seja uma tecnologia datada dos anos 40. Estes algoritmos acabam muitas vezes por serem "escolhidos", uma vez que, comparando aos algoritmos de *Machine Learning*, as máquinas são mais capacitadas na compreensão e manipulação dos dados, sem necessitarem tanto da intervenção humana (Esteva et al., 2019). Embora os algoritmos de *Machine Learning*, como por exemplo os que foram anteriormente aqui apresentados, sejam adequados para diversos problemas de contextos diferentes, os algoritmos de *Deep Learning* vieram resolver problemas mais complexos, como é o caso do reconhecimento facial, de objetos ou de voz (Goodfellow et al., 2016).

Os algoritmos de *Deep Learning* conseguem que as máquinas tenham uma capacidade de compreensão e manipulação dos dados mais autónoma do que os algoritmos de *Machine Learning*, garantido pela aprendizagem através de uma hierarquia de conceitos. Ou seja, com esta hierarquia de conceitos, o computador aprende conceitos mais complexos em relação a conceitos mais simples, exigindo assim menor intervenção humana na aprendizagem. Considerando que os primeiros algoritmos foram utilizados para compreender a aprendizagem humana de forma a reproduzi-la a nível computacional, *Deep Learning* é também usualmente associado ao conceito de *Artificial Neural Networks (ANN)*, considerada a primeira designação dada a este tipo de aprendizagem (Goodfellow et al., 2016). Em 1943, MuCul-

loch (neurocientista) e Walter Pitts (especialista em logística), propuseram o primeiro modelo neuronal artificial, apresentado e explicado na Figura 10b.



(a) Representação de um neurônio (retirado de Chandra, 2018).



(b) Representação do neurônio de McCulloch-Pitts (retirado de Chandra, 2018).

Figura 10: De forma abstrata, podemos explicar o neurônio sugerido por McCulloch-Pitts em (b), fazendo uma analogia com o neurônio biológico em (a). Tal como o neurônio biológico, também o neurônio de McCulloch-Pitts recebe sinais de outros neurônios, processa a informação e transmite essa informação a outros neurônios. Este neurônio recebe vários inputs binários (x_1, x_2, \dots), em g , processa-os e transmite-os, num único output binário, em f , para o neurônio seguinte. Este output pode ser 0 ou 1 e é determinado verificando se o valor da soma dos pesos de cada um dos inputs é superior ou inferior ao valor do threshold definido. Se o valor da soma for igual ou inferior ao valor do threshold o output é igual a 0, se o valor da soma for superior ao valor do threshold o output é igual a 1 (Neapolitan and Jiang, 2018).

Artificial Neural Networks (ANN), como é representada na Figura 11, é composta por múltiplas camadas (*layers*) de neurônios, em específico uma *input layer*, várias *hidden layers* e uma *output layer*, que estão interligadas entre si para que as decisões tomadas em cada uma das *layers* sejam transmitidas à *layer* seguinte. Como cada um dos neurônios chega a uma decisão que é transmitida para o neurônio seguinte, a primeira camada (a *input layer*) vai obter três decisões que são alcançadas com a medição dos pesos dos *inputs*. Na camada seguinte, as decisões já são realizadas de forma mais robusta, uma vez que

consideram as decisões anteriores realizadas na primeira camada. Esta complexidade acompanha o número de *layers* existentes, até atingir o *output* final (Neapolitan and Jiang, 2018).

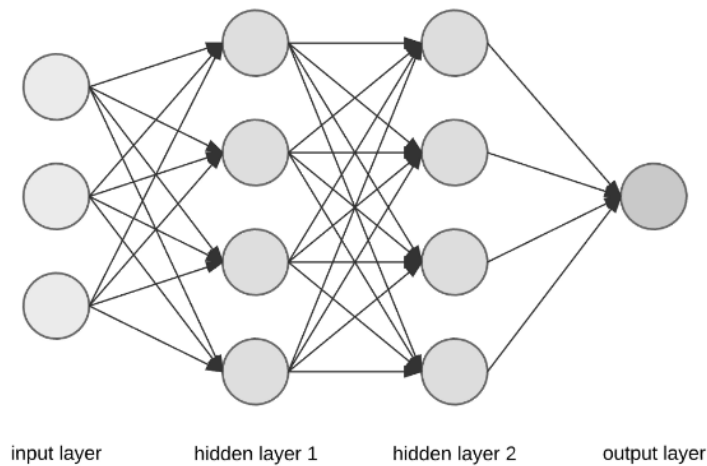


Figura 11: Ilustração de uma ANN (retirado de Dertat, 2017).

As *Recurrent Neural Networks (RNN)* são um tipo de *Artificial Neural Networks* especializado para a previsão de séries temporais, baseado numa aprendizagem sequencial. Como é possível verificar na Figura 12 a), as redes neurais recorrentes caracterizam-se por terem um método de *loop* no processamento da informação, garantindo que esta rede "lembra" não só a nova informação como a informação anterior. Por outro lado, as *Feed-Forward Neural Networks FNN*, apresentadas na Figura 12 b), não têm essa capacidade de manter a memória da informação processada anteriormente. Embora, a *RNN* tenha esse potencial, no que toca a memória de longo prazo ela já não é tão adequada. Assim sendo, as *Long short-term Memory* são mais apropriadas, uma vez que apresentam uma "*forget gate*" responsável por resolver a questão a longo prazo. (Eliasy and Przychodzen, 2020; Salman et al., 2018; Zhao et al., 2017).

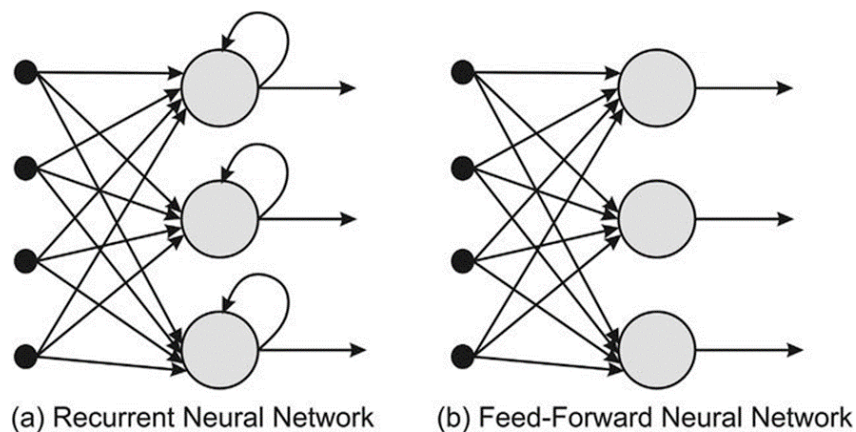


Figura 12: Comparação entre RNN e FNN (retirado de Eliasy and Przychodzen, 2020).

Considerando os diversos algoritmos de *Deep Learning*, importa neste trabalho focar a compreensão nos seguintes algoritmos, o *Multilayer Perceptron* e o *Long short-term Memory* que representam um exemplo de *FNN* e de *RNN*, respetivamente.

- **Multilayer Perceptrons (MLP):** também conhecido por *Deep Feedforward Neural Network*, é um algoritmo de *Deep Learning* que segue uma estrutura como a apresentada na Figura 11, com pelo menos três *layers*, sendo que este valor é variável conforme o número de *hidden layers*. O *MLP* é considerado um algoritmo de fácil implementação, com bom desempenho e que consegue simultaneamente, ter um tempo de treino mais baixo quando comparado com algoritmos mais complexos (Car et al., 2020). Este utiliza vários nós computacionais que são conectados por pesos e organizados em camadas, sendo popularmente usado em tarefas de classificação e regressão (Goodfellow et al., 2016; Matos et al., 2019; Truong et al., 2020).
- **Long short-term Memory (LSTM):** este algoritmo é um tipo especial de *Recurrent neural network (RNN)*, que resolve o problema da "memória de curto prazo" através de um mecanismo de portas responsáveis por controlar as informações (Goodfellow et al., 2016; Nguyen et al., 2021). Estas portas, como é possível verificar na Figura 13, são a "*forget gate*", responsável por definir que informação é relevante e que informação deve ser esquecida; a "*input gate*", ou também conhecida por "*memory gate*", responsável por decidir quais os dados que devem ser guardados pela célula e, por fim, a "*output gate*" que define o que é produzido por cada uma das células (Siami-Namini et al., 2018). Cada uma das três portas recebe $x(t)$ e $h(t - 1)$ como *input*, que representam um vector no instante atual t e o *output* da *memory cell* no instante anterior $t - 1$, respetivamente (Fischer and Krauss, 2018).

Na "*forget gate*", a decisão de excluir ou não a informação é feita através da função *sigmoid* (σ) que considera a informação relevante e mantém a mesma se o valor da porta for igual a 1 ou considera a informação irrelevante e rejeita-a se o valor da porta for igual a 0 (Yu et al., 2019). Na "*input gate*", a informação representada por $\tilde{c}(t)$, é processada pela função *tanh*, podendo ser adicionada à célula que é processada, e pela função *sigmoid* (σ) que, mais uma vez, decide se a informação é mantida ou não. Por último, na *output gate*, ambas as funções anteriores são utilizadas de forma a finalmente produzir o *output* para a célula seguinte (Fischer and Krauss, 2018).

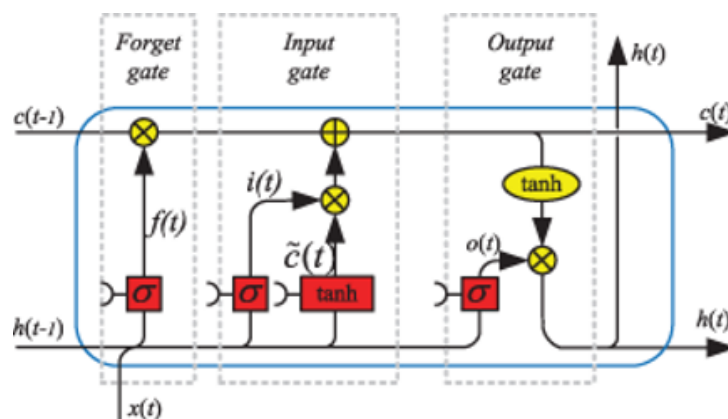


Figura 13: Ilustração de uma memory cell (retirado de Yu et al., 2019).

2.5 *Work-related Musculoskeletal Disorders (WMSDs)*

Nas manufaturas, como é o exemplo da indústria têxtil e automóvel, em que os seus produtos e serviços exigem dos seus trabalhadores movimentos muito repetitivos e até posições desapropriadas, é muito comum a associação destes movimentos às *WMSDs*. Estas são definidas pela Organização Mundial de Saúde (OMS) como, “*a disorder of the muscles, tendons, joints, intervertebral discs, peripheral nerves and vascular system, not directly resulting from an acute or instantaneous event but installing gradually and chronically*” (Anghel et al., 2007). As *Work-related Musculoskeletal Disorders (WMSDs)* apresentam um grande impacto na qualidade de vida dos trabalhadores e, conseqüentemente, na produtividade e qualidade dos produtos finais. Na tabela seguinte, são apresentados os diversos fatores de risco identificados como possíveis causas no surgimento das *WMSDs*.

Tabela 1: Fatores de risco associadas às *WMSDs* (retirado de Santos et al., 2020a).

Riscos individuais	Riscos relacionados com o trabalho	Riscos relacionados com as máquinas
Idade	Tarefas repetitivas	Peso das máquinas superior a 3kg
Maus hábitos de saúde	Tarefas que exigem que seja exercida alguma força	Vibração da mão e/ou corpo
Más práticas de trabalho	Posturas prejudiciais	Força exercida
Stress		Movimentos repetitivos
		Postura estática ou movimento do corpo

Em Portugal, é forte o impacto das manufaturas automóveis na economia nacional, onde em 2020 existiam cerca de 360 empresas e 61000 pessoas empregadas. Estes valores, fornecidos pela AFIA - Associação de Fabricantes para a Indústria Automóvel, em <https://afia.pt/estatisticas/>, realçam a crescente necessidade das empresas focarem a sua preocupação na saúde dos seus trabalhadores. Um estudo realizado numa indústria automóvel portuguesa em 2017 com um total de 270 participantes, mostra que aquando a realização do mesmo, 69% dos trabalhadores diziam sofrer destes distúrbios contra os 31% que diziam não apresentar quaisquer sintomas (Guerreiro et al., 2017).

Na Figura 14, verifica-se as zonas do corpo com maior incidência das dores sentidas pelos trabalhadores que apresentaram queixas, sendo os membros superiores a zona mais afetada, especialmente nos ombros, pulsos e mãos (Guerreiro et al., 2017). Com a adoção massiva de tecnologias da Indústria 4.0 por parte destas manufaturas, procura-se já colmatar as conseqüências das *WMSDs*, aumentando não só a competitividade como a criação de valor nestes setores.

Vários estudos realizados na indústria têxtil evidenciam a prevalência destas lesões nos seus trabalhadores, justificadas não só pelas suas atividades muito repetitivas e que requerem a movimentação constante dos braços como por posturas indevidas que podem acontecer com o decorrer do dia de trabalho (Kanniappan and Palani, 2020). Num estudo da avaliação do risco ergonómico dos trabalhadores

de uma confecção em Bangladesh, constatou-se que, num total de 232 trabalhadores, de entre os quais 186 mulheres e 46 homens, 24.7% das mulheres apresentaram dores no fundo das costas, 23.7% dores no pescoço e 21.7% dos homens apresentaram dores no pescoço e 13% queixaram-se de dores no joelho (Hossain et al., 2018).

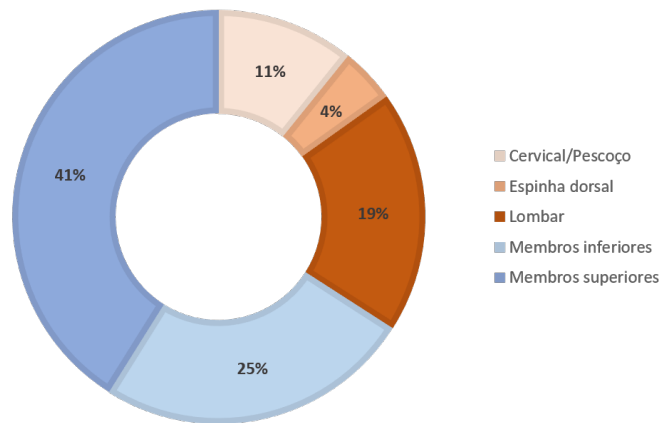


Figura 14: Incidência de WMSDs numa indústria automóvel portuguesa (adaptado de Guerreiro et al., 2017).

Num outro estudo com 100 participantes realizado numa indústria de couro, mostrou-se que nos últimos 7 dias, 86% das pessoas apresentavam dores no fundo das costas, 84% apresentavam dores nos joelhos e 74% apresentavam dores no pescoço. Ao avaliar as mesmas incidências mas num espaço temporal de 12 meses, os valores subiram para 88% nas dores no fundo das costas, 86% para dores nos joelhos e 76% para dores no pescoço (Kanniappan and Palani, 2020). Na Etiópia, num estudo com uma amostra alargada a 422 participantes, dos quais são considerados válidos 419 recolhas, foram reportadas dores no pescoço e nos ombros em 72.1% e 68.3%, respetivamente (Mekonnen et al., 2020).

Neste setor, em específico com os objetivos traçados no projeto STVgoDigital, procura-se resolver as WMSDs através de um exoesqueleto que irá dar suporte às atividades das costureiras, tornando-as menos suscetíveis a lesões. Este tipo de solução está já presente na indústria automóvel com a ambição de atingir uma melhoria ergonómica dos trabalhadores, auxiliando os mesmos essencialmente com a postura dos braços e o peso das ferramentas. Nas linhas de produção das fábricas da PSA Peugeot Citroen, foi criado e utilizado o exoesqueleto ABLE, com o objetivo de reduzir as dores e as WMSDs (Sylla et al., 2014). Também a Ford adotou o uso de um exoesqueleto vestível, o EksoVest, para aliviar a carga dos seus trabalhadores. Este exoesqueleto, que se caracteriza por ser bastante leve, dá apoio aos ombros, uma vez que as tarefas mais comuns na linha de produção envolvem a elevação dos ombros acima da cabeça (Crowe, 2018).

2.6 Human Motion Capture

Considerando as WMSDs como um problema atual das vidas dos trabalhadores destas indústrias, surge a necessidade de descobrir os padrões de movimentos que são suscetíveis de causar estes distúrbios. *Human Motion Capture (MoCap)* é uma técnica ideal na identificação destes padrões, uma vez que se caracteriza por ser um processo de captura do movimento humano, utilizando tecnologias com capacidade de recolher os movimentos em tempo real. Conforme a tecnologia utilizada para o seu funcionamento, existem diversos tipos de ferramentas que podem ser utilizadas, entre as quais os sistemas ópticos baseados em marcadores, como é o exemplo do Vicon, e os sistemas baseados na captura através de sensores inerciais, como é o exemplo dos IMUS Sensry Kallisto (Shi et al., 2014).

O sistema Vicon, representado na Figura 15, utiliza marcadores refletivos colocados em zonas específicas do corpo, e câmeras com luz LED colocada à volta da lente e que são estrategicamente colocadas na área de ensaio. À medida que o corpo se movimenta, a luz da câmera é refletida de novo na mesma, responsável pela criação de um sinal de vídeo. O *output* gerado por este sistema, em específico de cada uma das capturas, são as coordenadas x , y e z do objeto em estudo, podendo ser ainda geradas outras informações relevantes nestes contextos, como os ângulos Euler ou quaterniões, conceitos que serão aprofundados mais à frente (Pellatt et al., 2021).

Embora esta técnica apresente uma precisão bastante alta no que diz respeito à recriação dos movimentos efetuados, mostrando-se também, nomeadamente no projeto STVgoDigital, bastante útil para, numa primeira instância, analisar e avaliar ergonomicamente esses movimentos, a sua aplicação acaba, no entanto, por limitar a constante avaliação em tempo real do comportamento humano. Além disso, a captura e análise do movimento acaba por estar muito limitada às posições das câmeras e aos ângulos que as mesmas conseguem capturar (Santos et al., 2020a). Por isso mesmo, uma das vantagens dos sistemas de captura do movimento através de sensores inerciais, é a sua facilidade na aplicação e utilização dos mesmos para uma constante captura do movimento, sem que para isso seja necessária a montagem de equipamento mais robusto, como é o caso das câmeras.

Para além disso, os *IMUs* são uma opção bem mais barata e de tamanho mais reduzido, com menor probabilidade de degradação (Zhang et al., 2020). A sua praticidade, torna esta uma solução ideal para ser incorporada no exoesqueleto desenvolvido no projeto, garantido o constante registo dos movimentos efetuados pelas costureiras em contexto laboral.

Os *IMUs* caracterizam-se por conseguirem capturar informação mais detalhada, uma vez que incorporam, normalmente, três sensores acelerómetro, três giroscópio e três magnetómetro. Com os dados provenientes destes sensores, é possível obter, por exemplo, a velocidade e posição do objeto em movimento, essenciais para a tão desejada avaliação ergonómica (Santos et al., 2020a). Desta forma, é possível capturar o movimento humano em três dimensões e em tempo real, tornando este sistema pequeno e transportável, uma ótima alternativa a sistemas óticos. Os sensores implementados nestes dispositivos são descritos de seguida (Pereira, 2016):

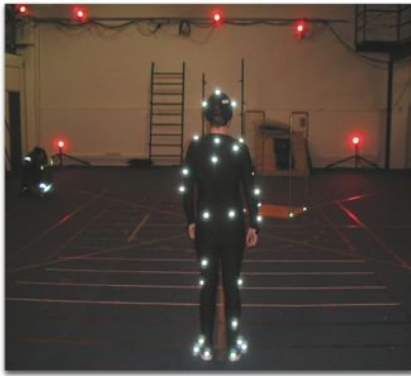


Figura 15: Sistema Vicon (retirada de Esteves, 2007).



Figura 16: Sensores inerciais (retirada de Pereira, 2016).

- **Acelerómetro:** mede a aceleração linear em três direções ortogonais e também a provocada pelo movimento ou aceleração gravitacional da Terra ($g = 9.8 \text{ m/s}^2$). Quando o objeto está em queda livre a aceleração é de 0 m/s^2 quando está na direção da queda, e quando está apoiado na mesa a aceleração, no sentido ascendente, é igual à gravidade da Terra. Por isso mesmo, o sinal tem uma componente estática, provocada pela aceleração gravitacional da Terra, e uma componente dinâmica, provocada pelo movimento do objeto. As medições podem ser representadas em m/s^2 ou por unidades de força g .
- **Giroscópio:** este é um sensor tridimensional que mede a velocidade angular do objeto e que pode também ser utilizado para calcular a orientação relativa do objeto num instante anterior. As medições são representadas em radianos por segundo (rad/s).
- **Magnetómetro:** mede o vetor do campo magnético terrestre, o que permite dar informações mais precisas relativamente à orientação do objeto. Para determinar essa orientação é necessário, pelo menos, o vetor de gravidade, ou seja, o acelerómetro. As medições são representadas em unidades de microtesla (μT).

Também no contexto industrial, os *wearable devices* são cada vez mais uma opção para a captura em tempo real do movimento humano. Foi realizado um estudo com o objetivo de implementar este tipo de sistema diretamente numa linha de montagem industrial, de modo a capturar e analisar os dados do movimento humano. Verificou-se que, através deste sistema, foi possível reter e processar um elevado número de dados que permitiram analisar detalhadamente o movimento humano e avaliar as posturas de trabalho (Caputo et al., 2019a).

Embora as vantagens mencionadas, há, naturalmente, alguns obstáculos na utilização destes sensores, nomeadamente nas derivações das medições do giroscópio, que exigem uma solução que é conseguida com a utilização de técnicas que utilizam as medições dos restantes sensores de forma a colmatar esta condicionante. Assim, é necessário para além de um processamento inicial destes dados, desde logo a sincronização temporal dos dados recolhidos de todos os sensores e a redução de ruído, a utili-

zação de técnicas ideias para a fusão de todos os dados inerciais recolhidos por estes sistemas (Santos et al., 2020a).

2.6.1 Sensor Fusion Para Estimativa da Orientação

As técnicas de *sensor fusion* são frequentemente utilizadas para ultrapassar as limitações dos dados em bruto, através da combinação dos dados dos sensores triaxiais, de forma a obter uma estimativa da orientação, em forma de quaternião. De entre as diversas limitações da utilização dos sensores de forma individual, destaca-se a imprecisão das medições dos mesmos assim como a ambiguidade e incerteza que podem estar associadas a essas medições que, naturalmente, prejudicam a correta interpretação do movimento realizado (Santos et al., 2020b).

Existem diversas técnicas de *sensor fusion* que podem ser utilizadas para combinar informação de múltiplos e diferentes sensores de forma a obter informação mais consistente e precisa do que com os dados originais dos sensores (Meng et al., 2020). Estas técnicas podem fazer uso dos três sensores, como é apresentado na Figura 17, de apenas dois sensores ou até mesmo de um só sensor.

Este processo não termina na aplicação da técnica de *sensor fusion*, uma vez que é também necessário converter o quaternião, uma representação matemática das orientações e rotações espaciais em forma de matriz, em ângulos Euler, de forma a obter uma representação do movimento em ângulos. Os ângulos Euler - pitch, ϕ , roll, θ , e yaw, ψ - representam a orientação em torno de cada um dos eixos, dependendo do referencial utilizado.

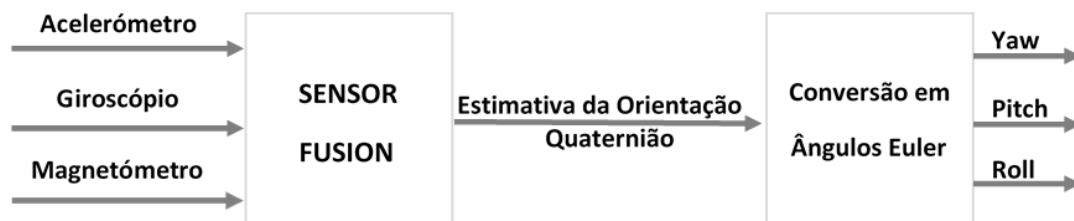


Figura 17: Ilustração do processo de *sensor fusion*.

Ainda que esta ferramenta permita ultrapassar obstáculos na utilização de sensores inerciais, esta apresenta desafios na sua aplicação uma vez que lida com dados diferentes, de sensores também eles diferentes, o que acrescenta uma natural dificuldade ao processo. Esta heterogeneidade dos dados é um dos inconvenientes identificados, uma vez que a técnica de *sensor fusion* utilizada deve ser capaz de ultrapassar a diversidade dos dados. Considerando a ambiguidade e a inconsistência destes dados, a sua complexidade acaba por ser um dos principais desafios, uma vez que esta informação pouco clara pode prejudicar a qualidade do resultado da aplicação da técnica de *sensor fusion* (Meng et al., 2020).

Na Tabela 2 são apresentados os algoritmos disponíveis no *AHRS (Attitude and Heading Reference System)*, uma ferramenta *open source* em Python para estimar a orientação de um corpo com base num sistema de referência. Destacado a cinzento, estão os algoritmos que podem utilizar os dados provenientes dos três sensores e que, por isso, merecem maior destaque no contexto do presente trabalho.

De entre os seis algoritmos sublinhados, o *Extended Kalman Filter (EKF)* é o mais frequentemente utilizado uma vez que necessita de baixa carga computacional, tornando-o barato, mais poderoso e robusto, embora seja simultaneamente o mais complexo de implementar. Esta desvantagem pode ser ultrapassada com a utilização de algoritmos mais simples e acessíveis, como o *Complementary filter*, incluindo o *Madgwick orientation filter* e o *Mahony orientation filter* que, comparando o desempenho entre os mesmos, mostram ser muitas das vezes alternativas fiáveis ao tão conhecido *EKF* (Ludwig, 2020).

Tabela 2: Algoritmos de *sensor fusion* disponíveis no AHRS (retirado do website AHRS¹).

Algoritmo	Giroscópio	Acelerómetro	Magnetómetro
Algebraic Quaternion Algorithm	Sim	Opcional	Opcional
Complementary Filter	Sim	Sim	Opcional
Davenport's q-Method	Não	Sim	Sim
Extended Kalman Filter (EKF)	Sim	Sim	Sim
Fast Accelerometer-Magnetometer Combination (FAMC)	Não	Sim	Sim
Fast Linear Attitude Estimator (FLAE)	Não	Sim	Sim
Fourati's Nonlinear Attitude Estimation	Sim	Sim	Sim
Factored Quaternion Algorithm Integration	Não	Sim	Opcional
Integration	Sim	Não	Não
Madgwick Orientation Filter	Sim	Sim	Opcional
Mahony Orientation Filter	Sim	Sim	Opcional
Optimal Linear Estimator of Quaternion (OLEQ)	Não	Sim	Sim
QUEST	Não	Sim	Sim
Recursive Optimal Linear Estimator of Quaternion (ROLEQ)	Não	Sim	Sim
Super-fast Attitude from Accelerometer and Magnetometer (SAAM)	Não	Sim	Sim
Attitude from Gravity (Tilt)	Não	Sim	Opcional
TRIAD	Não	Sim	Sim

Um dos principais entraves neste tipo de problema está no facto das medições do giroscópio apresentarem derivações que devem ser resolvidas. Assim sendo, uma das necessidades na escolha do algoritmo de *sensor fusion* ideal em situações em que são utilizados estes sensores, é a sua capacidade de utilizar os dados dos sensores acelerómetro e magnetómetro para corrigir essa derivação e, dessa forma, conseguir a normalização do quaternião que irá possibilitar obter a tão desejada estimativa da orientação.

¹ <https://ahrs.readthedocs.io/en/latest/>

Dos vários algoritmos mencionados, que são aptos à utilização das medições dos três sensores, o *Madgwick orientation filter* é um algoritmo capaz de resolver o problema anteriormente mencionado de uma forma mais simples que, por exemplo, o tão famoso *EKF* (Ludwig, 2020). Outra vantagem do *Madgwick*, quando comparado com o *EKF*, é o facto deste ter em consideração que os humanos só conseguem fazer movimentos com acelerações lineares durante alguns segundos. Assim sendo, nos contextos em que está em causa a utilização de sensores inerciais para a análise do movimento humano, é mais correto utilizar este algoritmo (Haslwanter, 2018).

O *Madgwick orientation filter*, cujo funcionamento é esquematizado na Figura 18, depende essencialmente de dois processos principais: o "*quaternion propagation*", responsável por alinhar as medições do giroscópio e o "*gradient descent algorithm*", responsável por fazer a junção das medições dos sensores acelerómetro e magnetómetro. Durante o "*quaternion propagation*", são utilizados as derivações das medições do giroscópio, através da aplicação de um parâmetro de forma a minimizar os seus efeitos. Desta forma o "*quaternion propagation*" inicializa-se com a orientação estimada no passo anterior. As medições do acelerómetro e magnetómetro são fundidas através de um parâmetro de ajuste, o "*gradient descent algorithm*", para que o seu *output* corrija a estimativa de orientação que foi obtida apenas com as medições do giroscópio (Ludwig, 2020).

Este processo evidencia uma outra vantagem relacionada com este algoritmo, focada na utilização de quatérniões para representar a estimativa da orientação. Como os dados recolhidos pelo acelerómetro e magnetómetro são aplicados num "*gradient descent algorithm*", este passo permite que seja possível computar os erros das medições do giroscópio enquanto quatérniões (Haslwanter, 2018). Obtendo, desta forma, uma estimativa da orientação em forma de quatérnião, considerando as medições obtidas por cada um dos três sensores.

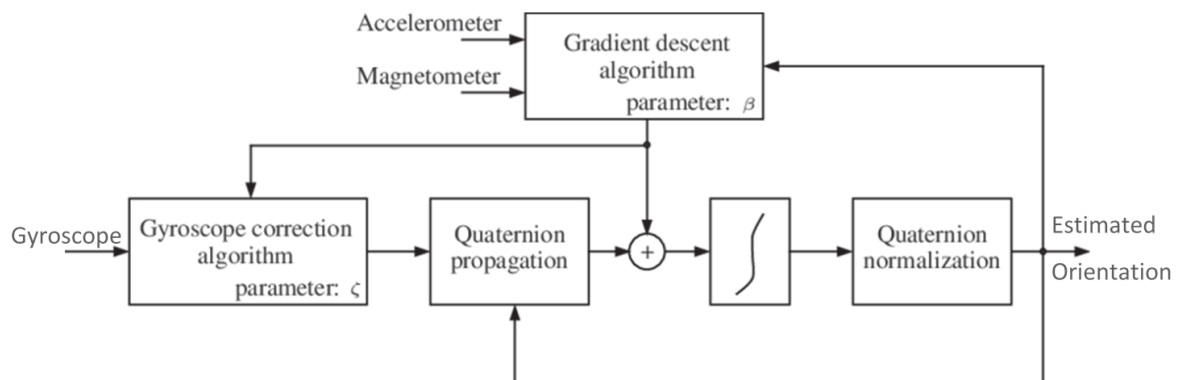


Figura 18: Funcionamento do *Madgwick orientation filter* (retirado de Cirillo et al., 2016).

2.6.2 Ergonomic Risk Assessment Tools

A necessidade de avaliar ergonomicamente o risco das posturas realizadas, implica a utilização de ferramentas adequadas para detetar as posturas e/ou os fatores de risco associados às *WMSDs*. Existem vários instrumentos que podem ser utilizados, como por exemplo os tradicionais *RULA* (*Rapid Upper*

Limb Assessment) e PERA (*Postural Ergonomic Risk Assessment*) ou como as normas utilizadas no contexto do projeto mobilizador, a norma ISO 11226: *Ergonomics - Evaluation of Static Working Postures*, a norma EN 1005-4: *Safety of Machinery - Human Physical Performance. Evaluation of Working Postures and Movements in Relation to Machinery* e a norma EN 1005-5: *Safety of Machinery. Human Physical Performance - Part 5: Risk Assessment for Repetitive Handling at High Frequency*.

A técnica RULA, desenvolvida por Lynn McAtamney e Nigel Corlett em 1993, é uma técnica bastante simples que permite identificar os distúrbios dos membros superiores através da análise da força, postura e movimentos exercidos. No entanto, não considera fatores também eles relevantes como o tempo das operações, fatores individuais (como por exemplo a idade), fatores ambientais do local de trabalho e fatores psicossociais (McAtamney and Nigel Corlett, 1993). A sua avaliação é feita através de uma pontuação que é obtida conforme os pontos associados a cada movimento, sendo que essa pontuação estará entre 1 e 7. sendo que, quanto maior for o valor obtido, maior é o risco associado. Os resultados podem ser interpretados da seguinte forma (Davudian-Talab et al., 2017):

- Nível 1 (pontuação de 1-2): postura considerada aceitável, desde que não seja exercida durante muito tempo;
- Nível 2 (pontuação de 3-4): postura que deve ser avaliada e que deve necessitar de alterações;
- Nível 3 (pontuação de 5-6): postura que deve ser avaliada e cujas mudanças são urgentes;
- Nível 4 (pontuação de 7+): postura que impõe mudanças imediatas.

Por sua vez, a técnica PERA, que apresenta duas abordagens - a de Sperling em 1993 e a de Kadefors em 1994 - permite avaliar os riscos da postura num ciclo curto de trabalho. A grande vantagem desta técnica, acaba por ser a capacidade da mesma analisar a totalidade das tarefas desse ciclo, identificando, por isso, todos os riscos a que um trabalhador está sujeito. Ambas as abordagens avaliam os mesmos parâmetros: a força (F), a precisão (P) e o tempo (T) do movimento, distinguindo-se na sua aplicação e no método de classificação. Sperling foca a sua aplicação na análise do trabalho com ferramentas manuais e desempenha uma classificação individual de cada um dos parâmetros, enquanto Kadefors analisa o trabalho manual e desempenha uma classificação multiplicativa de todos os parâmetros ($F \cdot P \cdot T$) (Chander and Cavatorta, 2017).

Relativamente às normas que serão utilizadas no projeto para a avaliação ergonómica das posturas realizadas pelas costureiras, a norma ISO 11226, permite avaliar as posturas de trabalho estáticas. Numa primeira fase, é tida em consideração os ângulos do corpo, sendo que dessa avaliação, pode resultar uma das seguintes opções: postura aceitável, postura não recomendada e ainda a necessidade de avaliar tendo em consideração um segundo fator. Este segundo fator contempla o tempo em que a pessoa fica nessa mesma postura (Delleman and Dul, 2007).

Através desta norma, é possível avaliar a postura dos membros superiores, nomeadamente o tronco, a postura da cabeça, a postura do ombro e braço, a postura do antebraço e da mão e a postura da extremidade inferior. Já a norma EN 1005-4, permite fazer uma avaliação mais completa, uma vez

que considera a postura e os movimentos executados em relação à máquina de trabalho. Tem em consideração dois fatores, a repetição dos movimentos e a percentagem do tempo de ciclo em cada postura.

Por fim, a norma EN 1005-5 faz uma avaliação mais completa do risco associado às tarefas repetitivas e em alta frequência, considerando diversos fatores na sua avaliação. Desde logo a repetição do movimento, a força exercida, a postura, movimento e duração a que o trabalhador se sujeita, a recuperação insuficiente, as características do objeto, as condições do ambiente e ainda fatores individuais e organizacionais ou vibração e força de impacto. Se se verificar que o operador se encontra em risco perante estes fatores, é necessário proceder a uma avaliação mais detalhada do mesmo, de forma a identificar se este exige uma mudança no imediato.

2.7 Previsão de Movimentos com Base em Dados Inerciais

Embora os métodos manuais ainda sejam os mais comumente utilizados, existem cada vez mais estudos em que são utilizados os *IMUs* para a captura de movimento humano. Esta tecnologia pode ser muito útil em ambiente industrial, nomeadamente na avaliação ergonómica dos movimentos dos trabalhadores. De forma a perceber que trabalhos existem e de que forma estes contribuem para o objetivo da presente dissertação, ou seja, na previsão de movimentos com base em dados inerciais, importa fazer uma revisão da literatura.

Por isso mesmo, segue-se uma síntese de trabalhos realizados em que são utilizados os *IMUs* para a avaliação ergonómica da parte superior do corpo ou estudos em que são utilizadas técnicas de *Machine Learning* para a interpretação de dados inerciais. Para além disso, são também apresentados trabalhos em que são utilizados algoritmos de *Machine Learning* na indústria automóvel.

A literatura presente na Tabela 3, apresenta para cada estudo, o seu autor e objetivo do mesmo. Para além disso, conforme os tópicos nele abordados, apresenta os algoritmos de *Machine Learning* utilizados (se se verificar a sua utilização) e os métodos utilizados, nomeadamente sistemas de captura de movimento ou normas de avaliação de postura.

Tabela 3: Revisão da Literatura.

Autor	Objetivo	Algoritmos Utilizados	Métodos Utilizados
(Peres et al., 2019)	Prever defeitos dimensionais numa linha de montagem automóvel.	<i>Random Forest</i> <i>Support Vector Classification</i> <i>XGBoost</i> <i>K-Nearest Neighbours</i> <i>Logistic Regression</i> <i>Naive Bayes</i>	-
(Syafudin et al., 2018)	Melhorar a tomada de decisão e ajudar a prevenir perdas inesperadas causadas por falhas durante o processo de fabricação numa indústria automóvel.	<i>Random Forest</i>	Sensores IoT
(Anghel, Daniel-Constantin et al., 2019)	Aumentar a produtividade e proteger os trabalhadores da indústria automóvel de problemas de saúde.	<i>Artificial Neural Networks</i>	RULA
(Santos et al., 2020b)	Avaliar a exposição automática ao risco do operador na indústria de manufatura.	-	IMUs adRULA
(Caputo et al., 2019b)	Analisar e avaliar a postura dos trabalhadores de uma indústria automóvel.	-	IMUs EAWS
(Hua et al., 2020)	Classificar diferentes movimentos.	<i>Random Forest</i>	IMUs

Autor	Objetivo	Algoritmos Utilizados	Métodos Utilizados
(Iranzo et al., 2020)	Utilizar um exoesqueleto para a parte superior do corpo dos trabalhadores de uma indústria automóvel, para provar que estes conseguem reduzir a tensão muscular.	-	EMG
(Mundt et al., 2021)	Utilizar diferentes abordagens de <i>Deep Learning</i> para prever ângulos articulares com base em dados capturados por <i>IMUs</i> .	Multilayer Perceptron Long short-term Memory Convolutional Neural Networks	<i>IMUs</i>
(Su et al., 2020)	Utilizar uma arquitetura DCNN para distinguir cinco fases de um ciclo de marcha, com base em dados capturados por <i>IMUs</i> .	Deep Convolutional Neural Networks	<i>IMUs</i>
(Ribeiro et al., 2020)	Melhorar a captura do movimento dos <i>IMUs</i> em vários segmentos do corpo ao realizar diferentes movimentos, utilizando para isso algoritmos de classificação.	Logistic Regression Support Vector Machine Random Forest A simple LSTM A Six-Layer LSTM	<i>IMUs</i>

2.8 Ferramentas Tecnológicas

De forma a contextualizar o trabalho desenvolvido nesta dissertação, surge a necessidade de identificar os materiais tecnológicos que poderão ser utilizados para cumprir as necessidades identificadas. Desde logo, o código será praticamente todo desenvolvido em Python, linguagem de programação de alto nível e orientada a objetos, criada por Guido van Rossum em 1991, e que tem vindo a ser cada vez mais utilizada nos últimos anos, nomeadamente em *data science*, encontrando-se no top 10 de linguagens de programação mais populares.

Python permite acompanhar todos os passos da metodologia *CRISP-DM*, sendo possível fazer quer a preparação dos dados quer a modelação, uma vez que existe um vasto leque de bibliotecas disponíveis para a utilização desta linguagem em contexto de *Machine Learning*. Para além disso, a utilização desta linguagem acaba por ser ainda mais vantajosa no âmbito deste trabalho, uma vez que pode também ser utilizada na fase de pré-processamento dos dados inerciais, através da implementação das técnicas de *sensor fusion* disponíveis na ferramenta *AHRS*.

As bibliotecas e ferramentas que serão utilizadas no decorrer do projeto, quer na fase que antecede a aplicação de *Machine Learning*, nomeadamente na fase de preparação dos dados com o processamento dos dados inerciais, quer nas fases posteriores, em específico a modelação e avaliação com a utilização de técnicas e parâmetros específicos, são identificadas e especificadas na Tabela 4.

Tabela 4: Ferramentas tecnológicas.

Ferramenta	Descrição
AHRS	Ferramenta utilizada no processamento dos dados capturados pelos sensores inerciais, fornece um leque de funções, técnicas e algoritmos, devidamente documentados, de forma a simplificar a obtenção da estimativa da orientação para todos aqueles que precisam de utilizar as técnicas de <i>sensor fusion</i> nos seus projetos. A sua última versão, e a que será utilizada no trabalho, é a versão 0.3 e utiliza a versão 3.6 ou superior do Python (neste caso será utilizada a versão Python 3.8.8), sendo ainda a sua utilização dependente da biblioteca NumPy.
Anaconda	É uma plataforma especificada para <i>data science</i> , com diversas ferramentas <i>open source</i> disponíveis para a implementação de várias tecnologias como <i>Machine Learning</i> ou o processamento de imagens. Disponibiliza diversas bibliotecas, que serão utilizadas e identificadas de seguida. Fornece ainda vários IDEs para a implementação de código Python, nomeadamente o <i>Jupyter Notebook</i> , cuja versão 6.3.0 será utilizada no desenvolvimento prático.
Pandas	É uma biblioteca utilizada para a manipulação e análise dos dados em Python, de forma acessível, rápida e de fácil utilização. Será utilizada na manipulação dos dados dos sensores inerciais.
Scikit-learn	Esta biblioteca permite a implementação de diversos algoritmos de <i>Machine Learning</i> , como por exemplo algoritmos de classificação, regressão ou de séries temporais. No contexto do atual trabalho, a biblioteca será utilizada para implementar algoritmos de regressão. Para além disso, permite também utilizar métricas para a avaliação do desempenho dos modelos.
NumPy	A biblioteca numpy apresenta uma vasta coleção de funções matemáticas, que facilitam a execução de operações em grandes quantidades de dados, suportando essencialmente arrays e matrizes multi-dimensionais.
Matplotlib	Matplotlib é uma biblioteca que permite criar gráficos de visualização de dados e resultados, sendo uma extensão da biblioteca NumPy.
Plotly	Esta biblioteca <i>open-source</i> permite a criação de gráficos de visualização mais iterativos e que podem ser visualizados de diversas formas.
TensorFlow	TensorFlow é uma biblioteca <i>open source</i> que, para além de <i>Machine Learning</i> , é adequada para a criação e treino de redes neuronais, sendo por isso uma ferramenta de fácil utilização para a implementação de algoritmos de <i>Deep Learning</i> .
Keras	O Keras é uma API de <i>Deep Learning</i> , que corre "em cima" do Tensorflow na qual a sua estrutura base são <i>layers</i> e modelos.
rpy2	Este pacote permite fazer uma ponte de ligação entre as linguagens Python e R. Com o mesmo, é possível aceder e utilizar funções R em Python.

Caso de Estudo: Previsão de Movimentos Prejudiciais aos Trabalhadores com Base em Dados Inerciais

As empresas de manufatura, como por exemplo a indústria têxtil e a indústria automóvel, têm vindo a promover a transição para a Indústria 4.0 procurando simultaneamente o desenvolvimento económico-social e o aumento da produtividade e eficiência da cada empresa. Estas indústrias dependem fortemente do trabalho manual dos seus trabalhadores, estando a sua produtividade e qualidade final do produto diretamente ligadas a esta condicionante. Como tal, pretende-se a aplicação de tecnologias inerentes à indústria 4.0 de forma a assegurar, simplificar e facilitar o processo de produção, tornando a saúde do trabalhador uma prioridade.

Neste contexto, o projeto mobilizador STVgoDigital tem como principal objetivo o desenvolvimento de um produto centralizado no papel das trabalhadoras do STV, que assegure o aumento da sua segurança e conforto e, como resultado disso, melhore o seu desempenho e capacidade para se adaptar a ambientes flexíveis e dinâmicos. Considerando o pressuposto de que uma empresa só consegue ser competitiva se aliar o desenvolvimento tecnológico ao bem-estar e motivação dos seus trabalhadores e que, os custos associados à saúde dos mesmos afetam diretamente a economia da empresa, é urgente a aplicação de uma solução que preserve a saúde destes.

Nesta indústria, as costureiras utilizam diversos tipos de máquinas e exercem movimentos muito repetitivos e complexos ao longo de um dia de trabalho, estando justificada a premissa de que esta profissão apresenta vários riscos para o desenvolvimento de *WMSDs*. Assim, pretende-se a implementação de um exoesqueleto robotizado para proteção individual dos operadores. Este exoesqueleto terá, na sua integração, uma componente de sensorização, uma componente de atuação e um sistema de algoritmos de aprendizagem e atuação, sendo que esta última carece da realização antecipada das restantes componentes.

Ainda que se tenha verificado a impossibilidade de realizar este trabalho no contexto do projeto pela razão anteriormente referida, este caso de estudo enquadra-se nas necessidades e objetivos desta solução, garantindo que o sistema de aprendizagem é capaz de utilizar os dados da componente de sensorização para prever, com antecedência, os movimentos a serem realizados. Embora num contexto industrial diferente, nomeadamente numa indústria automóvel, os dados utilizados para a previsão dos movimentos são também eles provenientes de sensores inerciais. Este contributo permitirá validar a utilização de algoritmos de *Machine Learning* para a previsão de movimentos, utilizando para isso dados recolhidos

por sensores inerciais, elemento importante do produto a desenvolver no projeto mobilizador, para que o exoesqueleto possa atuar em circunstâncias necessárias e em tempo útil.

Esta demonstração compreenderá o trabalho prático desenvolvido em todas as fases da metodologia *CRISP-DM*, à exceção da última fase da mesma, uma vez que se verifica a impossibilidade de implementar o trabalho desenvolvido. Para melhor compreensão das necessidades do presente trabalho prático, o mesmo é apresentado seguindo a estrutura das seguintes fases da metodologia *CRISP-DM*: compreensão do negócio, compreensão dos dados, preparação dos dados, modelação, avaliação e implementação.

No trabalho correspondente às quatro últimas fases da metodologia implementadas, enquadram-se a aplicação de uma técnica de *sensor fusion* para a estimativa da orientação na compreensão e preparação dos dados; a aplicação dos algoritmos de *Machine Learning* na fase de modelação e, por fim, segue-se naturalmente a fase de avaliação dos mesmos. Não se enquadrando necessariamente em nenhuma das etapas da metodologia, ressalva-se ainda a aplicação do parâmetro de uma norma de avaliação de postura ISO.

3.1 Compreensão do Negócio

Nos setores industriais, como por exemplo as indústrias têxtil e automóvel, em que a sua produção é fortemente dependente da colaboração entre trabalhador e máquina, é propício o aparecimento de lesões ou distúrbios músculo-esqueléticos. Uma das principais causas para as *WMSDs* estarem identificadas como sendo bastante impactantes na vida destes trabalhadores, é a repetitividade dos movimentos na execução das suas tarefas.

Embora as empresas tenham conhecimento da importância de assegurar boas condições de trabalho, são ainda poucas as que investem para solucionar este problema ou, no limite, reduzir o seu impacto. É, por isso, importante focar em dar aos seus colaboradores melhores condições de trabalho, nomeadamente na indústria automóvel que, tal como a têxtil, apresenta um forte posicionamento a nível nacional.

O desenvolvimento de uma solução como um exoesqueleto inteligente, adaptável a diversos movimentos e com atuação em tempo real, exige a integração complexa de 3 componentes principais: uma componente de sensorização, uma componente de aprendizagem e uma componente de atuação.

A componente de sensorização inclui a incorporação de vários sensores inerciais no exoesqueleto, também designados por *IMUs*, com o objetivo de capturar os movimentos exercidos pela pessoa. Os dados por estes recolhidos, correspondem aos valores dos sensores triaxiais acelerómetro, giroscópio e magnetómetro, ou seja, aos valores da aceleração, velocidade angular e campo magnético, respetivamente. A sua colocação deve ser realizada em zonas específicas do corpo, identificadas como zonas problemáticas, de maior tensão muscular e, por isso, sujeitas a um nível maior de risco.

A segunda parte correspondente à componente de aprendizagem, representa um papel importante no desenvolvimento de um exoesqueleto inteligente, adaptado a todos os tipos de movimento e com atuação em tempo real. Com a integração desta componente, a aplicação de algoritmos de *Machine Learning* para a previsão de movimentos utilizando os dados recolhidos pelos sensores inerciais, permitirá perceber, com antecedência, quando um movimento considerado perigoso estiver prestes a ser realizado. Desta

forma, os atuadores poderão receber a informação antecipada de que deverão ser ativados. Nesta fase, está ainda identificada a importância de processar os dados recolhidos pelos sensores inerciais de forma a que estes possam ter valor para os algoritmos a utilizar.

Por fim, a componente de atuação é responsável pela atividade do exoesqueleto, garantindo que o mesmo atua em benefício do trabalhador, sem por isso interferirem na realização da tarefa em causa. Para além disso, o mesmo só deve atuar em momentos que constituem perigo para o utilizador.

No presente trabalho, as 3 componentes anteriormente identificadas estão interligadas mas o foco está, naturalmente, na componente de aprendizagem. É apresentada uma investigação de algoritmos de previsão de séries temporais para a previsão dos movimentos, com a sua aplicação assente em dados provenientes de sensores inerciais. É, também por isso, apresentada a aplicação de uma técnica de sensor fusion para o processamento dos dados recolhidos pelos *IMUs*.

Por outro lado, embora não exista um contributo mais prático para a componente de atuação, a previsão antecipada dos movimentos permite, desde logo, garantir que os atuadores recebem um "aviso" com algum tempo útil para que a sua atuação ocorra em situações de risco e, assim, dar o devido suporte aos trabalhadores.

3.2 Compreensão dos Dados

De forma a demonstrar a aplicabilidade da componente de algoritmia em dados sensoriais para a previsão de movimentos incorretos, o presente estudo foca na análise do movimento de abdução/adução do braço, exemplificado anteriormente na Figura 16. A escolha deste movimento para a sua análise, permite aplicar os diversos conceitos anteriormente referidos e, desta forma, obter uma prova de conceito funcional.

Para isso, foram utilizados dados de um *dataset open-source*, que representam dados recolhidos por sistemas de *MoCap*, nomeadamente por *IMUs* colocados no corpo de diversos participantes. O *dataset* utilizado designa-se por "*Upper-body movements: precise tracking of human motion using inertial sensors*" e pode ser encontrado na plataforma Zenodo (Santos et al., 2020a). Os dados representam séries temporais e foram recolhidos em contexto simulado e laboratorial, com o objetivo de analisar e avaliar ergonomicamente os movimentos realizados numa indústria automóvel.

Embora seja estudado apenas um movimento, o *dataset* contém seis *sets* diferentes de movimentos, todos eles apresentados na Tabela 5, para um total de 11 participantes, sendo eles 5 mulheres e 6 homens. Estes conjuntos de movimentos foram obtidos através da colocação de 4 *IMUs* na parte superior do corpo, cada um deles com um sensor acelerómetro, magnetómetro e giroscópio incorporados.

Os *IMUs*, como se pode verificar na Figura 19, foram dispostos da seguinte forma: 1 *IMU* colocado no braço direito, 1 *IMU* colocado no antebraço direito, 1 *IMU* colocado no pulso direito e, por fim, 1 *IMU* colocado no peito. Neste caso, os dados que correspondem ao movimento em estudo foram recolhidos pelo *IMU* colocado no braço direito, sinalizado na figura a vermelho. Para além disso, de forma a obter a representação do movimento de abdução/adução do braço, foi utilizado o sistema de referência ENU

(East, North, Up). Este é um sistema cartesiano, em que os 3 eixos estão alinhados com o Este, Norte e com as direções verticais (Héry et al., 2017).

Tabela 5: Sumário dos seis sets de movimentos realizados.

Descrição	Articulação do Pulso	Articulação do Cotovelo	Articulação do Ombro	Tronco
Set 1: flexão/extensão do antebraço; abdução/adução do braço; posição anatômica	-	avaliação estática	avaliação estática	-
Set 2: flexão/extensão do pulso; desvio radial/ulnar do pulso; posição anatômica	avaliação estática	avaliação estática	-	-
Set 3: radial/flexão/extensão; flexão lateral do tronco; posição anatômica	-	-	-	avaliação estática
Set 4: flexão/extensão do braço; flexão/extensão do tronco; posição anatômica	-	-	avaliação estática	-
Set 5: flexão/extensão do braço; posição anatômica; posição anatômica	-	.	avaliação dinâmica	avaliação dinâmica
Set 6: flexão/extensão do tronco; flexão/extensão do braço; posição anatômica	-	-	-	avaliação dinâmica
	-	-	avaliação dinâmica	-

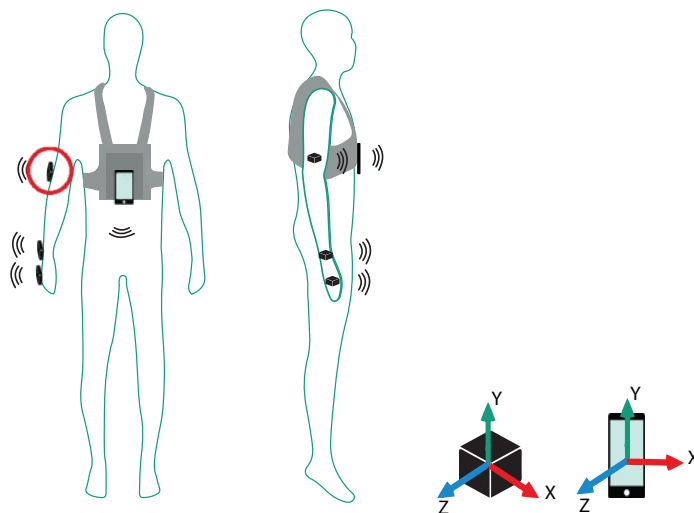


Figura 19: Colocação dos IMUs em cada um dos participantes e sistema de referência (retirado de Santos et al., 2020b).

Existem dois tipos de avaliação para os diferentes movimentos, a avaliação estática, quando a captura é feita com a pessoa em pé, ou dinâmica, quando o movimento é capturado com a pessoa a andar

(Santos et al., 2020b). O movimento de abdução/adução do braço representa a avaliação estática da articulação do ombro.

Assim, considerando o movimento em estudo e a sua forma de recolha, foram utilizados os dados recolhidos pelo acelerómetro, magnetómetro e giroscópio nos eixos do x , y e z , que corresponde aos dados da aceleração, campo magnético e velocidade angular, respetivamente. Naturalmente, utilizamos também o tempo de recolha do movimento, ou seja, o tempo em que foram capturados os dados pelos sensores. Na Tabela 6, são apresentados estes dados de forma mais clara.

Tabela 6: Síntese dos dados utilizados.

	Colunas	Descrição	Unidade
Acelerómetro	ax	Valor da aceleração no eixo do x	m/s^2
	ay	Valor da aceleração no eixo do y	m/s^2
	az	Valor da aceleração no eixo do z	m/s^2
Magnetómetro	mx	Valor do campo magnético no eixo do x	μT
	my	Valor do campo magnético no eixo do y	μT
	mz	Valor do campo magnético no eixo do z	μT
Giroscópio	gx	Valor da velocidade angular no eixo do x	rad/s
	gy	Valor da velocidade angular no eixo do y	rad/s
	gz	Valor da velocidade angular no eixo do z	rad/s
	<i>Time</i>	Tempo de recolha dos sensores	s

3.3 Preparação dos Dados

A preparação dos dados, etapa bastante importante e que antecede a modelação, é aquela que reúne todas as alterações necessárias aos dados. Algumas das atividades comuns neste pré-processamento passam pela seleção dos atributos relevantes ao problema ou pela transformação e limpeza dos dados. Considerando os dados, nomeadamente as variáveis descritas na Tabela 6, numa primeira instância não se verifica a necessidade de selecionar variáveis, uma vez que todas elas são importantes para obter a representação do movimento.

Para além disso, numa primeira análise foi possível constatar que estes dados, sendo recolhidos por sensores, não apresentam erros a serem corrigidos nem valores em falta, pelo que também não se verificou nenhuma necessidade de aplicar técnicas de transformação e limpeza aos mesmos. Adicionalmente, os valores dos atributos são já todos do tipo numérico, não havendo utilidade na aplicabilidade de técnicas comumente utilizadas em problemas de *Machine Learning*, como o *One Hot Encoding*, uma vez que se trata de um problema de previsão de séries temporais.

Embora não sejam imperativas as alterações anteriormente mencionadas, existe a necessidade de uma etapa de *feature engineering*, de forma a transformar os dados em bruto em variáveis que melhor representam o problema em mãos. Neste caso, este processo passa por transformar os dados provenientes dos sensores em valores de ângulos. Este processamento foi realizado através da ferramenta *AHRS*, que permite, de uma forma mais simplificada e clara, obter uma estimativa da orientação.

Assim, o processo de *feature engineering* reúne duas atividades essenciais: (1) a aplicação do *Madgwick orientation filter* como técnica de *sensor fusion* e (2) a conversão do quaternião obtido pela fase anterior, em ângulos Euler. Como é possível verificar na Figura 20, a etapa (1) utiliza os dados das 9 variáveis, que correspondem aos dados triaxiais de cada um dos sensores, para obter uma estimativa da orientação; e a etapa (2) transforma esta estimativa em forma de quaternião numa representação em ângulos do movimento.

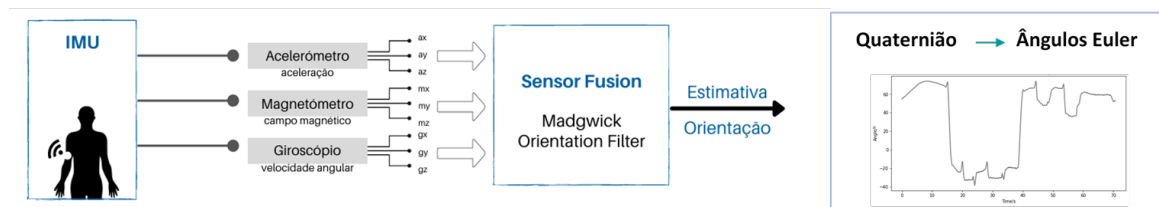


Figura 20: *Workflow* do processamento dos dados.

Embora tenha havido a necessidade de preparar os dados para os 11 participantes de forma a realizar todas as experiências nas etapas seguintes demonstradas, é através dos dados inerciais do participante A que será exemplificado este processamento. Na Figura 21, são apresentados os dados recolhidos pelos sensores do IMU colocado no braço direito do participante A durante a realização do movimento de abdução/adução do ombro.

Como se pode constatar, são fornecidos dados da aceleração, giroscópio e magnetómetro nos 3 eixos ao longo do tempo de recolha. Estes dados, bastantes confusos e de difícil entendimento, não nos permitem retirar nenhuma conclusão dos mesmos no que ao tipo de movimento diz respeito. Por isso mesmo, a aplicação da função do filtro *Madgwick orientation filter* do *AHRS*, permite "pegar" nestes dados e obter um quaternião, representativo da orientação e que, posteriormente, será utilizado para converter o mesmo em ângulos Euler.

A função `ahrs.filters.madgwick.Madgwick`, permite utilizar este filtro com apenas um linha de código Python, como demonstrado em 3.1. Para que o mesmo funcione, é necessário que os seus parâmetros, neste caso os dados do acelerómetro (`acc_data`), do magnetómetro (`mag_data`) e do giroscópio (`gyr_data`), sejam em forma de *array*.

```
acc_data = slarm [['ax', 'ay', 'az']].to_numpy()
mag_data = slarm [['mx', 'my', 'mz']].to_numpy()
gyr_data = slarm [['gx', 'gy', 'gz']].to_numpy()
```

```

angulos = ahrs.filters.madgwick.Madgwick(acc=acc_data, gyr=
    gyr_data, mag=mag_data)

Quaternioes = pd.DataFrame(angulos.Q, columns = ["qw", "qx", "
    qy", "qz"])

```

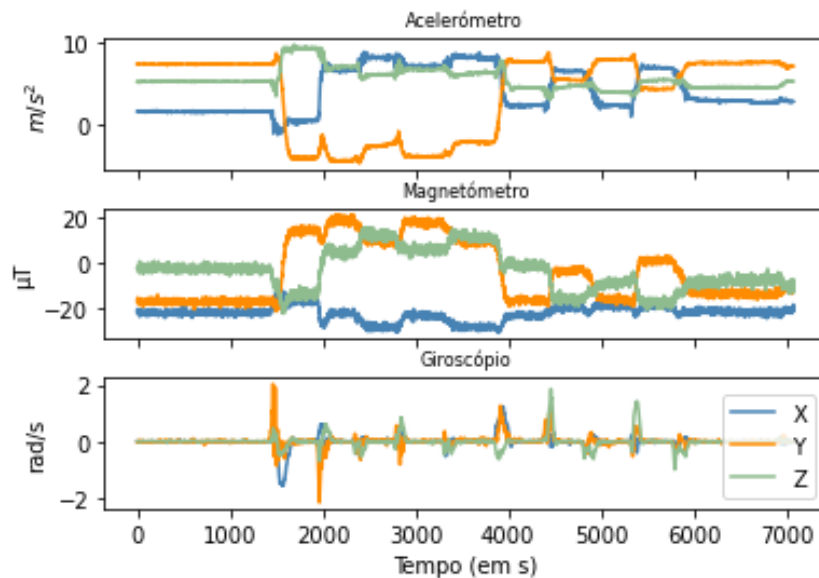
Listing 3.1: Código *Madgwick orientation filter*.

Figura 21: Exemplo dos dados recolhidos pelo IMU 3 colocado no braço direito.

Obtido o valor do quaternião, procede-se à aplicação da função ***quaternion_to_euler_angle_vectorized1*** para obter os valores dos ângulos Euler (*roll*, ϕ , *pitch*, θ , e *yaw*, ψ) que, neste caso, correspondem aos valores do X , Y , Z , respetivamente. O ϕ representa a rotação em torno do eixo do x , o θ representa a rotação em torno do eixo do y e o ψ representa a rotação em torno do eixo do z . Considerando o sistema de referência ENU, como demonstrado anteriormente na Figura 19, o valor do θ , ou seja, do X , corresponde ao ângulo que o braço faz com o tronco no movimento de abdução/adução do braço.

```

def quaternion_to_euler_angle_vectorized1(qw, qx, qy, qz):
    ysqr = qy * qy

    t0 = +2.0 * (qw * qx + qy * qz)
    t1 = +1.0 - 2.0 * (qx * qx + ysqr)
    X = np.degrees(np.arctan2(t0, t1))

    t2 = +2.0 * (qw * qy - qz * qx)
    t2 = np.where(t2 > +1.0, +1.0, t2)

    t2 = np.where(t2 < -1.0, -1.0, t2)
    Y = np.degrees(np.arcsin(t2))

```

```

t3 = +2.0 * (qw * qz + qx * qy)
t4 = +1.0 - 2.0 * (ysqr + qz * qz)
Z = np.degrees(np.arctan2(t3, t4))

return X, Y, Z

```

Listing 3.2: Transformação do quaternião em ângulos Euler.

Concluído o pré-processamento dos dados e obtida a representação do movimento em ângulos, procede-se agora à seleção das variáveis necessárias para a modelação. Assim sendo, são seleccionadas as variáveis **"Time"** e **"Angle"**, anteriormente designada de "X". A Figura 22 representa a reconstrução do movimento de abdução/adução do participante A, com a linha a tracejado a representar o limite de 60° definido pela norma ISO 11226.

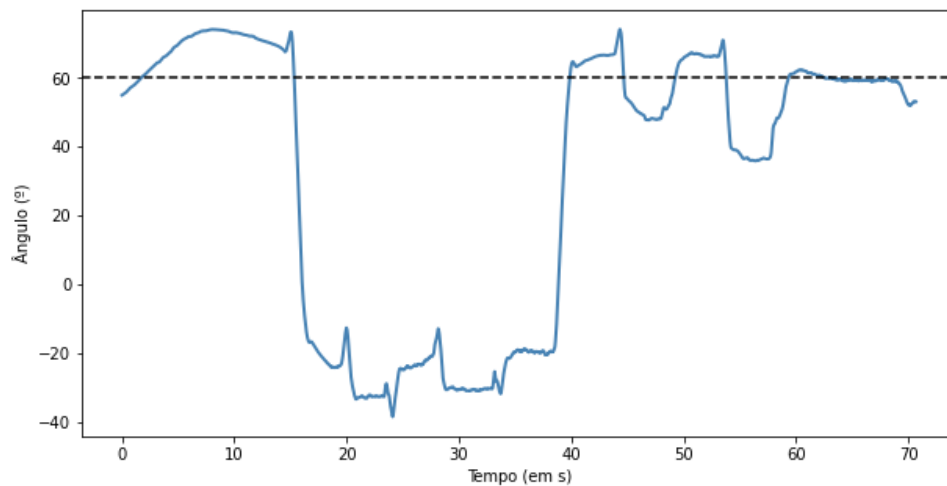


Figura 22: Exemplo da reconstrução angular do movimento de abdução/adução do braço para o participante A depois de utilizar o Madgwick orientation filter.

3.4 Modelação

Com os dados devidamente processados e prontos a serem utilizados, segue-se a fase de modelação onde são realizadas as experiências de *Machine Learning* e *Deep Learning*. Nesta etapa, é assumida uma perspectiva de *Time Series Forecasting (TSF)*, que permite prever o movimento futuro baseado em padrões retirados anteriormente desse mesmo movimento. Neste caso, é utilizada a abordagem específica de previsões *multi-step ahead*, que permitirá prever os movimentos com antecedência e, consequentemente, permitir um futuro contributo para a atuação do exoesqueleto (Cortez and Donate, 2014).

Este trabalho integra a implementação de seis algoritmos diferentes no total: três algoritmos tradicionais de regressão, em específico o *Linear Regression (LR)*, o *Random Forest (RF)* e o *Support Vector*

Regression (SVR) e de 3 arquiteturas *Long short-term memory (LSTM)*, nomeadamente uma LSTM normal, um *Stacked LSTM* e um *LSTM Sequence to Sequence (Seq2Seq)*.

Estando os algoritmos tradicionais e a LSTM normal descritos na Secção 2.3, resta abordar os restantes algoritmos utilizados. Um *Stacked LSTM* é uma arquitetura com múltiplas *hidden layers* e múltiplas *memory cells* que permite aumentar a complexidade, a "profundidade" e o nível de abstração do modelo, melhorando consequentemente os resultados das TSF (Li et al., 2020b). Esta arquitetura pode ser considerada uma *Deep Neural Network (DNN)*, na qual cada uma das camadas processa determinada informação e passa-a para a camada seguinte, sendo que o processamento na última camada gera finalmente o *output* (Hermans and Schrauwen, 2013).

O modelo *Stacked LSTM* é muito semelhante ao modelo normal LSTM, distinguindo-se apenas por possuir uma ou mais *hidden layers* adicionais. Neste trabalho, o modelo *Stacked* possui duas *hidden layers* no total, sendo que a *layer* extra é representada a azul na Figura 23, que se sobrepõe à arquitetura normal do LSTM, representada a cinzento.

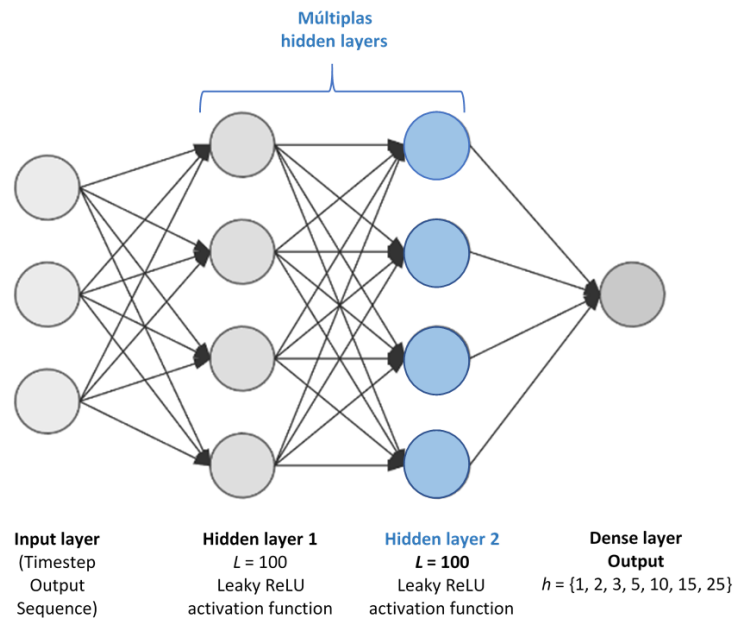


Figura 23: Arquitetura do Stacked LSTM (adaptado de Dertat, 2017).

O terceiro modelo LSTM, assume uma arquitetura *sequence to sequence*, ou também designada por *encoder-decoder*, muito utilizada inicialmente para realizar traduções. Este modelo é baseado em duas redes neuronais (RNN), uma designada de *encoder*, que lê a sequência de *input* e outra designada de *decoder*, responsável por gerar a sequência de *output*. Estas redes são treinadas paralelamente, sendo que o *output* da camada *encoder* é fornecido como *input* à camada *decoder* (Goodfellow et al., 2016).

A Figura 24, representa a arquitetura deste modelo, que é constituída por dois modelos LSTM, um primeiro modelo *encoder* para ler e codificar a sequência de *input* e um segundo modelo LSTM para

descodificar e prever um *output* (Xu et al., 2021). Cada uma destas camadas *LSTM* possui 100 células ($L = 100$), uma camada *Repeat Vector Layer* com H nodes, para repetir o *input* até h vezes, e uma camada *Time Distributed Layer* com uma *dense layer* responsável por processar o *output* e gerar os valores de saída sequenciais h .

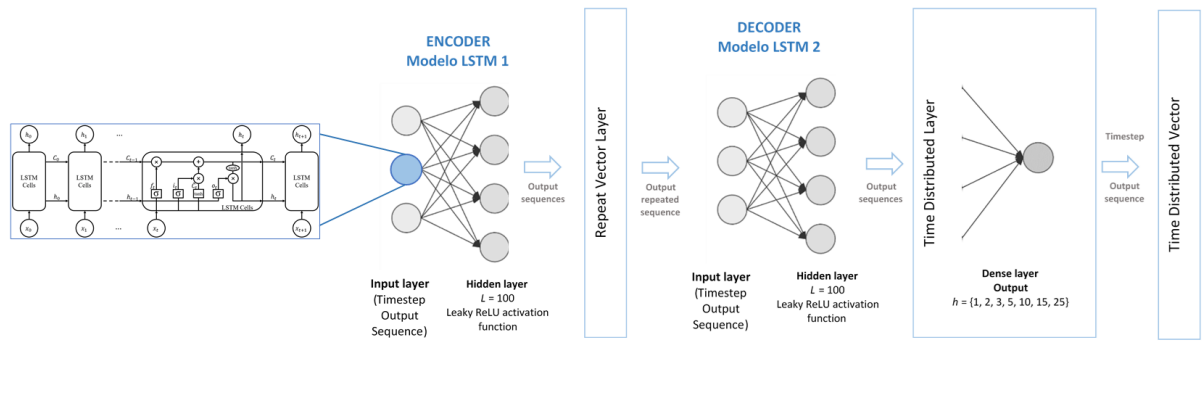


Figura 24: Arquitetura do LSTM Seq2Seq (adaptado de Dertat, 2017; Xiang et al., 2020).

Com o objetivo de obter um modelo com uma boa capacidade preditiva, a primeira fase de modelação foi realizada considerando apenas os dados recolhidos pelo IMU colocado no braço direito do participante A. Este contém um total de 7.079 observações, dos quais 4.948 foram utilizadas para treino e os restantes 2.131 valores para teste do modelo. Nesta primeira fase, foram treinados todos os modelos referidos anteriormente, como se pode constatar na Figura 25, baseada na forma de visualização proposta por Mundt et al., 2021.

Para a modelação de cada um dos modelos referidos, foi utilizada a linguagem Python e os *packages* *sklearn* para os modelos clássicos de regressão e o *Tensorflow* para os modelos *LSTM*. O modelo *Support Vector Regressor* foi treinado com os parâmetros base, enquanto que o *Random Forest Regressor* e o *Linear Regression* foram treinados com o parâmetros base mas com o parâmetro $n_jobs = -1$.

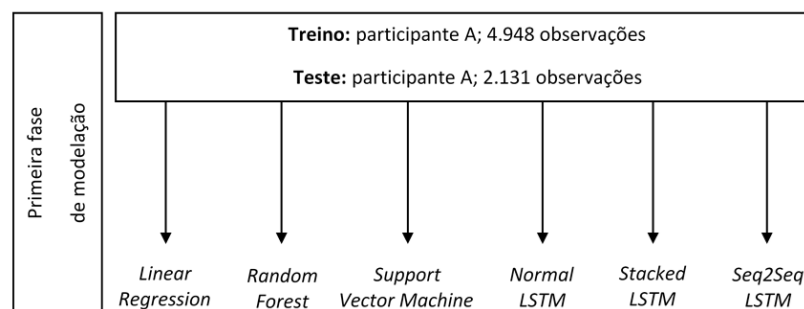


Figura 25: Primeira fase de modelação.

Em todos os modelos *LSTM*, o *input* fornecido é um *array* bidimensional, que representa o *timestep* e a *output sequence* e foi obtido através da função *split_sequence*, apresentada no Anexo A.3. Neste

caso, é utilizado o *timestep* anterior ($t - 1$) para prever os próximos 25 centésimos de segundo ($h = 25$). Cada uma das três arquiteturas é composta por *layers* com 100 células ($L = 100$) e assumem a função de ativação ReLU.

Na *LSTM Seq2Seq*, o modelo *decoder* tem o parâmetro *return_sequences = True*, permitindo que o *output* final seja também ele um *array* bidimensional, com a mesma estrutura do *input*. Para treinar estes modelos, foi utilizado o *Adam optimizer* com o *Mean Squared Error* como *loss function*.

Na segunda fase de modelação, foi utilizado o melhor modelo preditivo, o *LSTM Seq2Seq*, para treinar e testar com uma maior quantidade de dados. Nesta fase de modelação, foram utilizados dois procedimentos de avaliação da qualidade de antecipação de movimentos problemáticos e que serão explicados na Secção 3.5. A Figura 26 exemplifica um destes procedimentos de avaliação

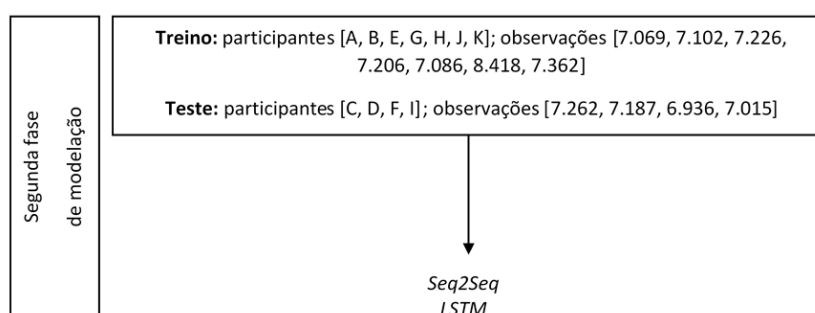


Figura 26: Segunda fase de modelação, através de um procedimento de avaliação que utiliza dados de múltiplas pessoas para treino e teste.

3.5 Avaliação

Esta fase, uma das mais importantes da metodologia *CRISP-DM*, reúne a avaliação dos modelos assim como os resultados obtidos pelos mesmos. Com esta análise, é possível verificar qual o melhor modelo para o problema, garantindo que é validado aquele com melhor capacidade preditiva. Neste trabalho, foram utilizadas três abordagens para avaliar os modelos utilizados na fase anterior.

Na primeira abordagem, na qual são avaliados os seis modelos da primeira fase de modelação, foram utilizadas quatro populares métricas de avaliação de desempenho de *TSF*, em específico o *coefficient of determination* (R^2), o *Mean Absolute Error* (*MAE*), o *Mean Squared Error* (*MSE*) e o *Root MSE* (*RMSE*).

Estas métricas, são apresentadas de seguida através das suas respetivas equações, em que o \mathcal{J} corresponde ao período de teste com um valor fixo de h . Os modelos com melhores previsões são os que apresentam valores de R^2 mais altos, enquanto que as restantes métricas (*MAE*, *MSE* e *RMSE*) devem apresentar valores mais baixos.

$$R^2 = 1 - \frac{\sum_{i \in \mathcal{J}} (y_i - \hat{y}_i)^2}{\sum_{i \in \mathcal{J}} (y_i - \bar{y}_i)^2} \quad (1)$$

$$MAE = \frac{1}{n} \sum_{i \in \mathcal{J}} |y_i - \hat{y}_i| \quad (2)$$

$$MSE = \frac{1}{n} \sum_{i \in \mathcal{J}} (y_i - \hat{y}_i)^2 \quad (3)$$

$$RMSE = \sqrt{MSE} \quad (4)$$

A Tabela 7, apresenta os resultados obtidos para cada um dos modelos treinados com os dados do participante A, para a previsão antecipada de 1 centésimo de segundo ($h = 1$). Com esta primeira fase de avaliação, foi possível verificar que os modelos *LSTM* são os que obtêm melhor desempenho, quando comparados com os modelos tradicionais de regressão (*LR*, *RF* e *SVR*).

Analisando os resultados gerais das métricas, e embora as três arquiteturas *LSTM* tenham obtido o mesmo valor de R^2 , o modelo *LSTM Seq2Seq* é o que tem resultados mais baixos de *MAE*, *MSE* e *RMSE*, podendo, por isso, ser considerado o melhor modelo preditivo. Com estes primeiros resultados, este modelo foi selecionado como o melhor algoritmo de *TSF* e, por isso, o escolhido para dar continuidade às experiências.

Tabela 7: Resultados para cada modelo (primeira fase de modelação, $h = 1$, os melhores valores a **negrito**).

Model	R²	MAE	MSE	RMSE
MLR	0.66	15.62	304.12	15.65
RF	0.09	10.68	210.06	10.70
SVR	0.60	15.14	336.52	15.16
One layer LSTM	0.88	0.40	0.19	0.40
Stacked LSTM	0.88	0.38	0.16	0.38
Seq2Seq LSTM	0.88	0.11	0.02	0.12

Na segunda fase de avaliação, pretendeu-se avaliar a capacidade de antecipação de movimentos problemáticos do modelo preditivo de séries temporais anteriormente selecionado (*LSTM Seq2Seq*, conforme resultados da primeira fase de avaliação). Para tal, e para cada participante, utilizou-se a norma ISO 11226: “*Ergonomics – Evaluation of static working postures tool*”, permitindo identificar diversos pontos iniciais de risco (t_r) na segunda metade da série temporal. Esta norma, define um valor de risco positivo que não deve ser ultrapassado (ângulo de $R_u = 60$), pelo que se assume uma variação de ângulos permitidos de $[-\infty, 60]$.

Na Figura 27 e na Tabela 8, é possível constatar, a título de exemplo, que o movimento do participante A apresenta três pontos de análise no total: $t_{r_1} = 39.87$; $t_{r_2} = 49.51$ e $t_{r_3} = 59.34$. Nesta análise, o objetivo é perceber a capacidade preditiva que este modelo tem para prever, com uma antecedência que varia entre 1 centésimo ($h = 1$) até 25 centésimos ($h = 25$) de segundo, estes movimentos de risco em específico (os tais t_r).

Como era esperado, para este exemplo de participante A, os resultados pioram com o aumento da janela temporal preditiva ($h = 25$), especialmente os valores de MAE , MSE e $RMSE$. Isto é possível verificar na Figura 28, que mostra as previsões obtidas para os dois instantes para os três pontos de análise. Para a previsão com 1 centésimo de antecedência ($h = 1$), há um forte alinhamento entre os valores reais e os valores previstos, enquanto para um horizonte de 25 centésimos de antecedência ($h = 25$), os erros de previsão são maiores para a maior parte do período de tempo previsto.

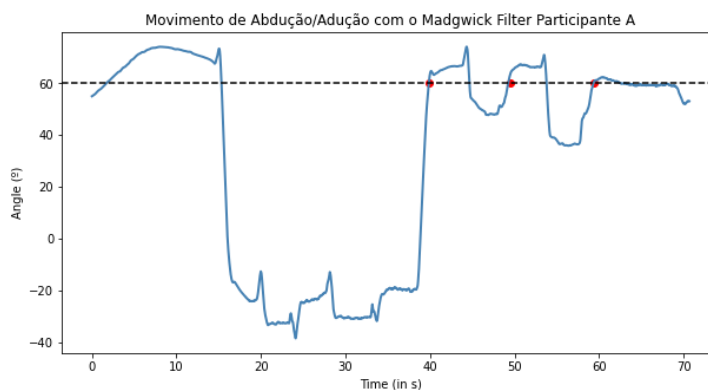


Figura 27: Exemplo do movimento do participante A; a linha tracejada horizontal demarca o nível de risco de WMSDs imposto pela norma ISO 11226; o círculo vermelho representa os pontos de análise que antecedem um movimento que ultrapassa o risco de WMSDs.

Tabela 8: Resultados das previsões obtidas pelo LSTM Seq2Seq para a pessoa A com três pontos de risco (t_r).

h	t_r	R^2	MAE	MSE	RMSE
1	$t_{r_1} = 39.87$	0.99	0.50	0.26	0.51
	$t_{r_2} = 49.51$	0.99	0.35	0.12	0.35
	$t_{r_3} = 59.34$	0.99	0.69	0.48	0.69
25	$t_{r_1} = 39.87$	0.98	8.08	70.73	8.41
	$t_{r_2} = 49.51$	0.98	2.53	6.69	2.59
	$t_{r_3} = 59.34$	0.89	1.65	2.99	1.73

Para obter uma maior robustez da avaliação da capacidade de antecipação de movimentos problemáticos pelo modelo preditivo selecionado, foram aplicados dois procedimentos de avaliação e que consideram dados de um total de 11 participantes.

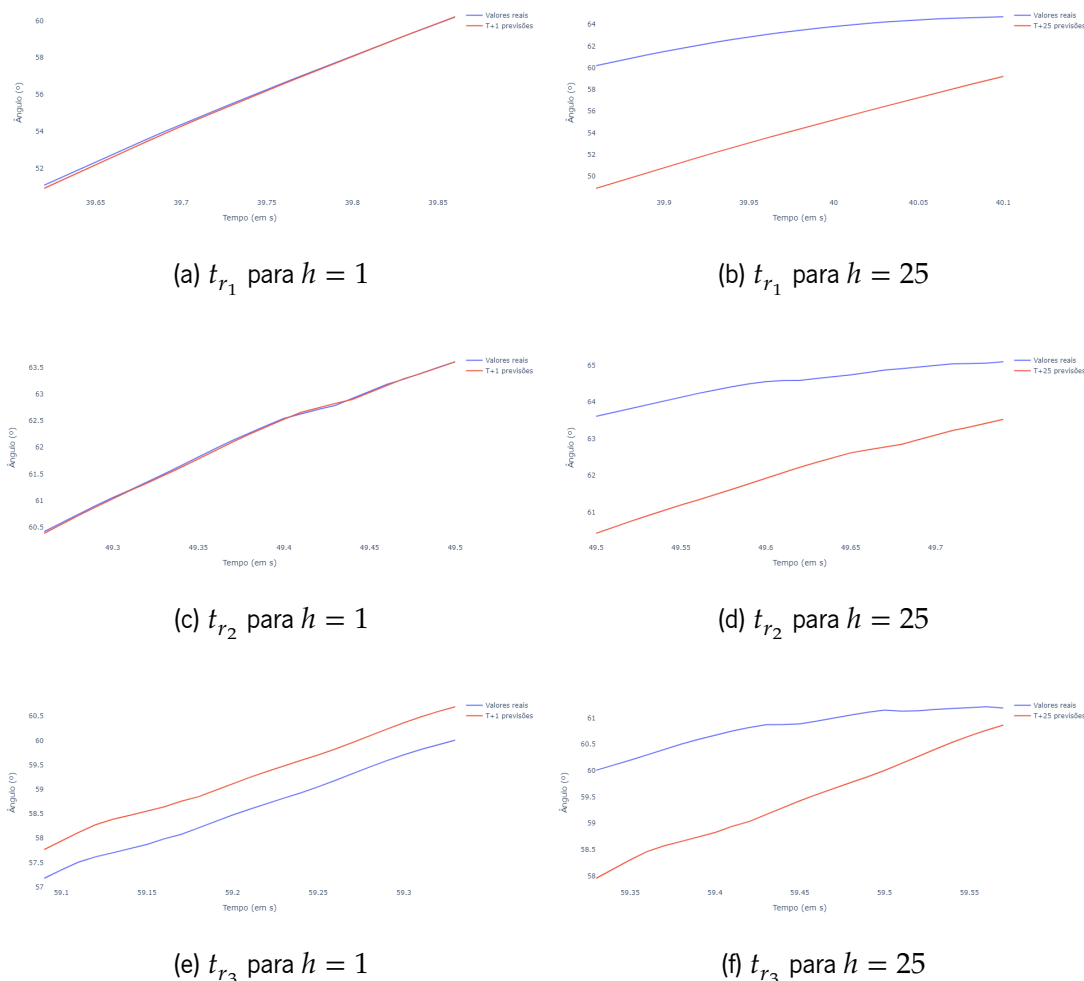


Figura 28: Comparação dos resultados obtidos para os três pontos críticos do participante A para $h = 1$ e $h = 25$.

O número total de períodos testados (t_{r_n}) para cada série é apresentado na Tabela 9. Para tempos de antecipação diferentes ($h \in \{1, 2, \dots, H\}$), foram executados dois procedimentos de avaliação de teste e treino:

- **dados de uma única pessoa**– neste procedimento, é aplicado um esquema *growing window* (Tashman, 2000; Lopes et al., 2011) de forma a que exista t_{r_n} atualizações do treino e teste do modelo *LSTM Seq2Seq*. Na segunda metade de cada série temporal, são definidos diversos pontos de risco (t_r), correspondendo cada um deles ao início de um movimento problemático de acordo com a norma ISO 11226. Para o primeiro valor de t_r (t_{r_1}), as observações anteriores a $t_{r_1} - H$ são utilizadas para treinar o modelo de *TSF*. De seguida, para $\max(h = H \text{ e } T = t_{r_1} - H)$, são calculadas previsões antecipadas até h , sendo um movimento de risco detetado corretamente se: $w(T, \max(h)) = 1$ se $\hat{y}_{T+h} + \tau > R_u$, senão $w(T, \max(h)) = 0$. A constante T permite definir uma folga, sendo que quanto maior o valor, maior será a sensibilidade da deteção de movimentos problemáticos. Contudo, um aumento do valor de folga também corresponderá a uma maior geração de falsos positivos. Assim, neste trabalho será assumido um valor pequeno

de folga, conforme explicado na apresentação dos resultados. Em seguida, executa-se um ciclo que atualiza o tempo atual T e diminui o horizonte máximo H , ou seja, $T = t_{r_1} - H + 1$ e $\max(h) = H - 1$. Este processo é realizado sucessivamente até obter uma previsão antecipada de somente 1 centésimo de segundo ($T = t_{r_1} - 1$ e $\max(h) = 1$). Este processo é repetido e o modelo retreinado, até que todos os pontos de risco t_{r_n} sejam analisados. Para um valor fixo de $\max(h)$, o *score* geral por pessoa (em %) é calculado como a soma de todos os $w(T, \max(h))$ a dividir pelo número total de pontos de risco (t_{r_n}). Este procedimento é aplicado separadamente ao total dos 11 participantes.

- **dados de múltiplas pessoas**– neste segundo procedimento, o modelo de previsão *LSTM Seq2Seq* é treinado com os dados de múltiplas pessoas, nomeadamente com o conjunto de participantes {A,B,E,G,H,J,K} que corresponde à série com três pontos críticos ($t_r = 3$). O teste é realizado com os dados dos restantes participantes ({C,D,F,I}), que têm apenas um ou dois pontos de risco *WMSD* (t_r). Esta avaliação permite calcular pontuações de risco de *WMSDs* individuais e gerais.

A Tabela 9 demonstra os resultados para o procedimento de avaliação com dados de uma única pessoa, no qual para cada um dos participantes está representada no numerador o total de antecipações corretas $w()$, e no denominador o número de pontos de risco considerados (t_{r_n}). Nestas experiências, optou-se por usar um valor de folga reduzido ($\tau = 1$ grau).

A última coluna da tabela representa uma avaliação horizontal para cada h , verificando qual é a taxa de acerto para cada um deles. Fica claro que, quanto maior for a antecipação, pior é a capacidade do modelo para prever o risco de lesão, sendo que o modelo tem o seu máximo desempenho para $h = 1$, com um *score* médio de 86%. Aumentado o $h = [2,3,5,10,15,25]$, o desempenho do modelo sofre um decréscimo considerável, sendo que o valor mais baixo é atingido para o $h = 15$ com um *score* médio de 43%, valor que se mantém para o $\max(h) = 25$.

Em termos verticais, numa análise individual para cada um dos participantes, a última linha representa o *score* geral obtido para cada pessoa, que representa a pontuação de antecipação de risco considerando todos os tempos de antecipação. Analisando estes valores, a pessoa A obtém uma deteção de risco perfeita com um *score* de 100%. Os resultados conseguidos para as pessoas K e F apresentam também resultados bastante bons, de 76% e 71% respetivamente. Por outro lado, a pessoa G obtém o pior valor com um *score* bastante baixo de 10%. Embora este resultado possa estar relacionado com o facto do movimento deste participante ser um movimento do ombro mais abrupto, foi feita uma segunda análise, para este movimento específico, com uma tolerância superior, com um valor de folga de $\tau = 2$ graus. Com esta tolerância, os resultados melhoraram significativamente com um *score* geral agora de 57%.

Para manter a consistência com os resultados da avaliação com os dados de uma única pessoa, o valor inicial de $\tau = 1$ grau será mantido para as restantes experiências garantindo que um maior valor de τ não gera um número acrescido de falsos positivos.

Para melhor compreensão de cada um dos movimentos efetuados por cada um dos participantes, no Anexo 11, estão apresentados os valores dos pontos críticos para cada um destes participantes, e a sua visualização pode ser consultada no Anexo A.1.

Tabela 9: Scores para a antecipação de perigo do ombro, considerando os dados de treino do participante A ($\tau = 1$).

max(h)	Pessoa											Score Médio Antecipação (%)
	A	B	C	D	E	F	G ($\tau = 2$)	H	I	J	K	
1	3/3	2/3	1/2	1/1	2/3	2/2	2/3 (3/3)	3/3	2/2	3/3	3/3	86
2	3/3	2/3	1/2	0/1	1/3	2/2	0/3 (3/3)	1/3	1/2	2/3	2/3	54
3	3/3	2/3	1/2	0/1	1/3	2/2	0/3 (2/3)	1/3	0/2	1/3	3/3	50
5	3/3	2/3	1/2	1/1	1/3	1/2	0/3 (1/3)	1/3	0/2	1/3	2/3	46
10	3/3	1/3	1/2	0/1	1/3	1/2	0/3 (1/3)	1/3	1/2	2/3	2/3	46
15	3/3	1/3	1/2	0/1	1/3	1/2	0/3 (1/3)	1/3	1/2	1/3	2/3	43
25	3/3	1/3	1/2	0/1	1/3	1/2	0/3 (1/3)	1/3	1/2	1/3	2/3	43
Score Geral por pessoa (%)	100	52	50	29	38	71	10 (57)	43	43	52	76	

Por fim, apresentam-se os resultados do procedimento que utiliza dados totais de múltiplas pessoas (7) para treino do modelo preditivo e teste de detecção de risco WMSD em dados de outras pessoas (4). Os resultados desta modelação são apresentados na Tabela 10, na qual é possível constatar resultados perfeitos de previsão. Para todos os participantes testados, o modelo de *TSF*, consegue identificar todos os pontos de risco (t_r) para todos os tempos de antecipação ($\max(h)$). Com isto, é possível concluir que o modelo tem melhor capacidade preditiva quando é treinado com movimentos recolhidos por diferentes participantes do que quando é treinado apenas com o movimento de um único participante.

Esta modelação, permite concluir que é possível prever os movimentos de risco e propícios a *WMSDs*, com pelo menos 25 centésimos de segundo, independentemente do movimento a ser avaliado. Estes resultados são bastante promissores no contexto do atual projeto, sugerindo que é possível prever os movimentos com tempo de antecedência suficiente para gerar um "gatilho" aos atuadores.

Tabela 10: Scores para a antecipação de perigo do ombro, considerando os dados de treino de vários participantes ($\tau = 1$).

max(h)	Pessoa				Score Médio Antecipação (%)
	C	D	F	I	
1	2/2	1/1	2/2	2/2	100
2	2/2	1/1	2/2	2/2	100
3	2/2	1/1	2/2	2/2	100
5	2/2	1/1	2/2	2/2	100
10	2/2	1/1	2/2	2/2	100
15	2/2	1/1	2/2	2/2	100
25	2/2	1/1	2/2	2/2	100
Score Geral por pessoa (%)	100	100	100	100	

3.6 Implementação

O trabalho desta dissertação não engloba a componente final de implementação, uma vez que se demonstrou impossível a realização das experiências de *Machine Learning* no contexto do projeto mobilizador STVgoDigital. Posto isto, o trabalho foi desenvolvido localmente, não com o objetivo final da sua implementação, mas sim com o principal foco na apresentação de contributos válidos e úteis para a componente de algoritmia do projeto.

No contexto do projeto STVgoDigital, a implementação do trabalho aqui desenvolvido corresponderia à sua integração na arquitetura do exoesqueleto inteligente em base têxtil. Esta implementação deverá ser feita num servidor e integrada numa plataforma de tratamento de dados, responsável por incorporar o modelo de *Machine Learning* para a aprendizagem e atuação associados aos movimentos das costureiras do setor têxtil, sendo capaz de fornecer informação ao exoesqueleto para auxiliar os movimentos do utilizador.

As previsões antecipadas deverão assistir a atuação dos atuadores incorporados no exoesqueleto, permitindo que estes atuem em situações críticas de movimentos realizados pelas costureiras de forma a alertar sobre e eventualmente bloquear situações de risco, prevenindo lesões músculo-esqueléticas.

Conclusões

O último capítulo, que marca o desfecho dos contributos do presente documento, apresenta uma visão daquilo que foi o trabalho desenvolvido, com especial destaque para as conclusões retiradas do mesmo. De forma a sintetizar os principais aspetos conseguidos com esta dissertação, as conclusões encontram-se divididas em três partes. A primeira parte, em que se descreve e sumariza todo o trabalho desenvolvido e demonstrado ao longo desta dissertação. A segunda parte, na qual são apresentados os resultados e, conseqüentemente, os contributos alcançados, verificando assim de que forma estes cumprem com os objetivos traçados. E, por fim, a terceira parte onde se identifica a possibilidade de dar continuidade ao projeto através de trabalhos futuros.

4.1 Síntese do Trabalho Realizado

Contextualizada com o projeto de *I&D STVgoDigital: "Digitalização da cadeia de valor do setor têxtil e vestuário"*, e focada em cumprir com os objetivos traçados para o mesmo, nomeadamente com a implementação de algoritmos de *Machine Learning* para a previsão de movimentos prejudiciais aos trabalhadores, esta dissertação engloba um enquadramento teórico dos conceitos abordados e, naturalmente, o trabalho prático desenvolvido.

Inicialmente, considerando a impossibilidade de utilizar dados recolhidos no âmbito do projeto, foram definidas as atividades a realizar e os objetivos a serem cumpridos com o presente trabalho de forma a garantir que o mesmo, embora num contexto um pouco distinto, pudesse colmatar os problemas e obter uma solução para o projeto. Definido o novo contexto e os dados a serem utilizados, foram identificadas as necessidades exigidas para a sua execução, em específico a aplicação de uma técnica de *sensor fusion* para processar os dados recolhidos pelos sensores, a aplicação de algoritmos de *Machine Learning* para prever antecipadamente os movimentos e a utilização de um parâmetro de uma norma ISO para identificar aqueles que poderão ser prejudiciais ao trabalhador.

Assim, numa primeira instância, foi feito um enquadramento da abordagem metodológica *CRISP-DM* utilizada neste trabalho, que permitiu organizá-lo em diferentes fases e diminuir a complexidade do problema. De seguida, foi feita uma revisão da literatura, onde são apresentados, descritos e explicados os diversos conceitos associados às atividades a realizar, destacando também um enquadramento conceitual através da explicação dos conceitos de *Artificial Intelligence*, *Data Mining*, *Machine Learning* e

Deep Learning, com especial foco nestes dois últimos, com a adicional descrição de algumas das suas técnicas.

Ainda no estado da arte, foram apresentados os riscos associados às *Work-related Musculoskeletal Disorders* e o seu impacto nos trabalhadores da indústria automóvel e foi descrita a técnica de *Human Motion Capture*, especificando as características dos *Inertial Measurement Units*. Foi ainda explicado o conceito de *sensor fusion* e enumeradas algumas das suas técnicas, assim como a descrição de várias ferramentas de avaliação ergonómica.

De forma a introduzir a parte prática, foram apresentadas as ferramentas tecnológicas utilizadas para a realização do caso de estudo: "Previsão de movimentos prejudiciais aos trabalhadores com base em dados inerciais". Aqui, o objetivo passou por demonstrar o trabalho realizado, organizado pelas 6 fases da metodologia *CRISP-DM*. Foi descrito o processo de preparação dos dados, onde se inclui a aplicação da técnica de *sensor fusion Madgwick orientation filter* e todas as experiências de *Machine Learning* realizadas, nas quais se destaca o algoritmo *LSTM Sequence to Sequence* como o melhor para a previsão antecipada dos movimentos.

4.2 Contributos

Concluído o trabalho desenvolvido nesta dissertação, é possível fazer um ponto de situação relativamente aos objetivos inicialmente traçados e os resultados realmente conseguidos. É seguro dizer que a grande maioria das expectativas iniciais foram alcançadas e que o presente trabalho cumpre com o objetivo principal de encontrar um modelo capaz de utilizar os dados provenientes de sensores inerciais e prever antecipadamente os movimentos realizados.

Embora uma das expectativas fosse a realização desta dissertação com dados do projeto mobilizador, recolhidos em ambiente têxtil através da colocação de sensores inerciais nas costureiras, este foi o único requisito que não foi cumprido devido ao atraso nas atividades do STVgoDigital. Em alternativa recorreram-se a conjuntos de dados públicos, permitindo que o trabalho desenvolvido nesta dissertação conseguisse cumprir com os pressupostos estipulados para a componente de *Machine Learning* a realizar pelo CCG, mas no setor automóvel.

Para além do estudo teórico dos demais tópicos, o principal contributo é a elaboração de um *pipeline* completo de *Machine Learning* para a prevenção das *WMSDs*. Este *pipeline*, que assume o uso de *Inertial Measurement Units* para a recolha automática dos dados de um movimento, mostrou ter um forte potencial para desencadear avisos prévios de risco de movimento prejudicial.

Este trabalho, permitiu dar uma solução para um dos principais problemas identificados com a utilização de *IMUs* como sistema de captura de movimento, assegurando a transformação dos dados em bruto recolhidos pelos sensores inerciais em valores angulares de movimento. Não só esta transformação foi feita com sucesso, como foi identificada uma ferramenta adequada e bastante prática para aplicar técnicas de *sensor fusion*. Sendo este um passo adicional à componente de *Machine Learning*, representando a preparação e processamento dos dados que alimentam os algoritmos, este é um contributo

muito importante para a sua futura aplicação no projeto e que permitiu ganhar alguma maturidade com esta técnica.

Com a comparação de diversos algoritmos de *Machine Learning* e também de *Deep learning*, foi possível constatar, através de diferentes métodos de avaliação, que os métodos tradicionais de regressão estudados não são adequados para o tipo de problema em questão e que, na sua generalidade, as arquiteturas profundas de *LSTM* implementadas são eficazes na previsão de movimentos. De facto, a arquitetura *LSTM Seq2Seq*, foi aquela que, para diferentes janelas temporais, forneceu os melhores resultados de previsão.

Um outro contributo prático, está na conclusão de que este modelo consegue resultados perfeitos de previsão para todas as janelas de antecipação estudadas, quando o mesmo é treinado com os dados de movimentos de abdução/adução de diferentes participantes do que quando o modelo é treinado com os dados de movimentos realizados por uma só pessoa.

A nível de contributos científicos, para além da sugestão da técnica *Madgwick orientation filter* como método de *sensor fusion* para a obtenção da representação dos movimentos realizados e de uma arquitetura *LSTM Seq2Seq* com resultados muito promissores para a previsão antecipada de movimentos, foi também submetido um artigo de investigação na conferência *37th ACM SIGAPP SAC 2022: Special Track on Health Informatics and Bioinformatics*, resultante do trabalho desenvolvido.

4.3 Trabalho Futuro

O desenvolvimento desta dissertação, e os resultados aqui conseguidos, abrem caminhos futuros para a sua aplicação no contexto têxtil do projeto mobilizador e sustenta a capacidade de se conseguir integrar uma componente de *Machine Learning* numa solução para a prevenção das *WMSDs* das trabalhadoras deste ramo. Assim, existem alguns aspetos a considerar para um trabalho futuro, aspetos estes que deverão ser aplicados no projeto *STVgoDigital*.

Desde logo, há a ambição de expandir o trabalho desenvolvido para a análise e avaliação de mais movimentos para além do de abdução/adução do braço, garantindo uma maior abrangência da avaliação ergonómica de movimentos. Esta adaptação poderia ser para qualquer parte do corpo mas, considerando as zonas críticas identificadas numa primeira avaliação das atividades das costureiras, deverá abranger movimentos realizados com os membros superiores, como por exemplo o movimento de flexão/extensão do braço. Por outro lado, um dos passos importantes em trabalho posterior, e também necessário com a abrangência a diferentes movimentos, é a aplicação de mais parâmetros de normas de avaliação de postura que tenham em consideração não só a posição estática, mas também a força exercida e a frequência com que o movimento é realizado.

Estando a componente de *Machine Learning* integrada na arquitetura do exoesqueleto a ser desenvolvido no projeto de *I&D*, este estudo deverá, num futuro, procurar gerar alarmes para os movimentos perigosos e fornece-los aos atuadores, para que estes possam atuar de forma a que o exoesqueleto dê suporte ao trabalhador, prevenindo-o de movimentos arriscados e, conseqüentemente, de *WMSDs*.

Por último, fazendo uma retrospectiva daquilo que foi conseguido e daquilo que ainda ficou por realizar, espera-se também cumprir com a implementação final de uma solução completa, capaz de agregar as diferentes componentes envolvidas no projeto e que cumpra os requisitos estipulados para a mesma.

Referências Bibliográficas

- Aasim, Singh, S., and Mohapatra, A. (2019). Repeated wavelet transform based arima model for very short-term wind speed forecasting. *Renewable Energy*, 136:758–768.
- Alzubi, J., Nayyar, A., and Kumar, A. (2018). Machine learning from theory to algorithms: an overview. In *Journal of physics: conference series*, volume 1142, page 012012. IOP Publishing.
- Anghel, M., Argeanu, V., Talpo, C., and Lungeanu, D. (2007). Musculoskeletal disorders (msds) consequences of prolonged static postures. *J Exper Med Surg Res*, 4:167–172.
- Anghel, Daniel-Constantin, Nitu, Eduard-Laurentiu, Rizea, Alin-Daniel, Gavriluta, Alin, Gavriluta, Ana, and Belu, Nadia (2019). Ergonomics study on an assembly line used in the automotive industry. *MATEC Web Conf.*, 290:12001.
- Caputo, F., Greco, A., D'Amato, E., Notaro, I., and Spada, S. (2019a). *IMU-Based Motion Capture Wearable System for Ergonomic Assessment in Industrial Environment*, pages 215–225.
- Caputo, F., Greco, A., D'Amato, E., Notaro, I., and Spada, S. (2019b). *IMU-Based Motion Capture Wearable System for Ergonomic Assessment in Industrial Environment*, pages 215–225.
- Car, Z., Segota, S. B., Andelic, N., Lorencin, I., and Mrzljak, V. (2020). Modeling the spread of COVID-19 infection using a multilayer perceptron. *Comput. Math. Methods Medicine*, 2020:5714714:1–5714714:10.
- Castle, N. (2018). Regression vs. classification algorithms. <https://blogs.oracle.com/ai-and-datascience/post/regression-vs-classification-algorithms>.
- Chander, D. S. and Cavatorta, M. P. (2017). An observational method for postural ergonomic risk assessment (pera). *International Journal of Industrial Ergonomics*, 57:32–41.
- Chandra, A. L. (2018). Mcculloch-pitts neuron — mankind's first mathematical model of a biological neuron. <https://towardsdatascience.com/mcculloch-pitts-model-5fdf65ac5dd1>.
- Charbuty, B. and Mohsin Abdulazeez, A. (2021). Classification based on decision tree algorithm for machine learning. *Journal of Applied Science and Technology Trends*, 2:20–28.
- Cirillo, P., Cirillo, A., De Maria, G., Natale, C., and Pirozzi, S. (2016). *A comparison of multisensor attitude estimation algorithms*, pages 529–539. CRC Press.

- Cortez, P. and Donate, J. (2014). Global and decomposition evolutionary support vector machine approaches for time series forecasting. *Neural Computing and Applications*, 25:1053–1062.
- Crowe, S. (2018). Ford adding eksovest exoskeletons to 15 automotive plants. <https://www.therobotreport.com/ford-eksovest-exoskeletons-automotive/>.
- Damanik, I., Windarto, A., Wanto, A., Poningsih, Andani, S., and Saputra, W. (2019). Decision tree optimization in c4.5 algorithm using genetic algorithm. *Journal of Physics: Conference Series*, 1255:012012.
- Davudian-Talab, A., Azari, G., Badfar, G., Shafeei, A., and Derakhshan, Z. (2017). Evaluation and correlation of the rapid upper limb assessment and rapid office strain assessment methods for predicting the risk of musculoskeletal disorders. *Internal Medicine and Medical Investigation Journal*, 2:155.
- Dayan, P. and Niv, Y. (2008). Reinforcement learning: the good, the bad and the ugly. *Current opinion in neurobiology*, 18(2):185–196.
- Delleman, N. J. and Dul, J. (2007). International standards on working postures and movements ISO 11226 and EN 1005-4. *Ergonomics*, 50(11):1809–1819.
- Dertat, A. (2017). Applied deep learning - part 1: Artificial neural networks. <https://towardsdatascience.com/applied-deep-learning-part-1-artificial-neural-networks-d7834f67a4f6>.
- DGAE, D.-G. d. A. E. (2018). Indústria Têxtil e Vestuário. *Sinopse 2018*, page 26.
- Djakaria, I. and Saleh, S. (2021). Covid-19 forecast using holt-winters exponential smoothing. *Journal of Physics: Conference Series*, 1882:012033.
- Eliasy, A. and Przychodzen, J. (2020). The role of ai in capital structure to enhance corporate funding strategies. *Array*, 6:100017.
- Esteva, A., Robicquet, A., Ramsundar, B., Kuleshov, V., DePristo, M., Chou, K., Cui, C., Corrado, G., Thrun, S., and Dean, J. (2019). A guide to deep learning in healthcare. *Nature medicine*, 25(1):24–29.
- Esteves, C. (2007). *Motion Planning : from Digital Actors to Humanoid Robots*. PhD thesis, Institut National Polytechnique de Toulouse, Toulouse, France.
- Fang, J., Su, H., and Xiao, Y. (2018). Will artificial intelligence surpass human intelligence? *SSRN Electronic Journal*.
- Fedak, V. (2018). Review: Artificial intelligence in 2018. <https://towardsdatascience.com/review-artificial-intelligence-in-2018-403ca386980f>.
- Fernandes, G. (2019). Modelos de predição | decision tree. <https://medium.com/turing-talks/turing-talks-17-modelos-de-predi>

- Fischer, T. and Krauss, C. (2018). Deep learning with long short-term memory networks for financial market predictions. *European Journal of Operational Research*, 270(2):654–669.
- Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press. <http://www.deeplearningbook.org>.
- Guerreiro, M., Serranheira, F., Cruz, E., and Sousa-Uva, A. (2017). Neck and upper limb musculoskeletal symptoms in assembly line workers of an automotive industry in portugal. pages 193–198.
- Gupta, M. and Chandra, P. (2020). A comprehensive survey of data mining. *International Journal of Information Technology*, pages 1–15.
- Han, J., Kamber, M., and Pei, J. (2011). *Data Mining: Concepts and Techniques, 3rd edition*. Morgan Kaufmann.
- Haslwanter, T. (2018). *3D Kinematics*. Springer.
- Hermans, M. and Schrauwen, B. (2013). Training and analysing deep recurrent neural networks. In Burges, C. J. C., Bottou, L., Welling, M., Ghahramani, Z., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc.
- Hope, T. M. (2020). Chapter 4 - linear regression. In Mechelli, A. and Vieira, S., editors, *Machine Learning*, pages 67–81. Academic Press.
- Hossain, M. D., Aftab, A., Al Imam, M. H., Mahmud, I., Chowdhury, I. A., Kabir, R. I., and Sarker, M. (2018). Prevalence of work related musculoskeletal disorders (wmsds) and ergonomic risk assessment among readymade garment workers of bangladesh: A cross sectional study. *PLOS ONE*, 13(7):1–18.
- Hua, A., Chaudhari, P., Johnson, N., Quinton, J., Schatz, B., Buchner, D., and Hernandez, M. E. (2020). Evaluation of machine learning models for classifying upper extremity exercises using inertial measurement unit-based kinematic data. *IEEE Journal of Biomedical and Health Informatics*, 24(9):2452–2460.
- HUANG, S., CAI, N., PACHECO, P. P., NARRANDES, S., WANG, Y., and XU, W. (2018). Applications of support vector machine (svm) learning in cancer genomics. *Cancer Genomics & Proteomics*, 15(1):41–51.
- Héry, E., Masi, S., Xu, P., and Bonnifait, P. (2017). Map-based curvilinear coordinates for autonomous vehicles. In *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, pages 1–7.
- Iranzo, S., Piedrabuena, A., Iordanov, D., Martinez-Iranzo, U., and Belda-Lois, J.-M. (2020). Ergonomics assessment of passive upper-limb exoskeletons in an automotive assembly plant. *Applied Ergonomics*, 87:103120.

- Jiang, W., Wu, X., Gong, Y., Yu, W., and Zhong, X. (2020). Holt–winters smoothing enhanced by fruit fly optimization algorithm to forecast monthly electricity consumption. *Energy*, 193:116779.
- Ju, Y., Sun, G., Chen, Q., Zhang, M., Zhu, H., and Rehman, M. U. (2019). A model combining convolutional neural network and lightgbm algorithm for ultra-short-term wind power forecasting. *IEEE Access*, 7:28309–28318.
- Kanniappan, V. and Palani, V. (2020). Prevalence of musculoskeletal disorders among sewing machine workers in a leather industry. *Journal of Lifestyle Medicine*, 10:121–125.
- Kumar, M. and Dargan, S. (2019). A survey of deep learning and its applications: A new paradigm to machine learning. *Archives of Computational Methods in Engineering*.
- Li, X., Yuan, C., and Wang, Z. (2020a). State of health estimation for li-ion battery via partial incremental capacity analysis based on support vector regression. *Energy*, 203:117852.
- Li, Y., Bao, T., Gong, J., Shu, X., and Zhang, K. (2020b). The prediction of dam displacement time series using stl, extra-trees, and stacked LSTM neural network. *IEEE Access*, 8:94440–94452.
- Lopes, C., Cortez, P., de Sousa, P. N. M., Rocha, M., and Rio, M. (2011). Symbiotic filtering for spam email detection. *Expert Syst. Appl.*, 38(8):9365–9372.
- Ludwig, S. (2020). Investigation of orientation estimation of multiple imus. *Unmanned Systems*, 09:1–9.
- Matos, L. M., Cortez, P., Mendes, R., and Moreau, A. (2019). Using deep learning for mobile marketing user conversion prediction. In *International Joint Conference on Neural Networks, IJCNN 2019 Budapest, Hungary, July 14-19, 2019*, pages 1–8. IEEE.
- McAtamney, L. and Nigel Corlett, E. (1993). Rula: a survey method for the investigation of work-related upper limb disorders. *Applied Ergonomics*, 24(2):91–99.
- Mekonnen, T., Yenealem, D., and Geberu, D. (2020). Physical environmental and occupational factors inducing work-related neck and shoulder pains among self-employed tailors of informal sectors in ethiopia, 2019: results from a community based cross-sectional study. *BMC Public Health*, 20.
- Meng, T., Jing, X., Yan, Z., and Pedrycz, W. (2020). A survey on machine learning for data fusion. *Information Fusion*, 57:115–129.
- Mundt, M., Johnson, W. R., Potthast, W., Markert, B., Mian, A., and Alderson, J. (2021). A comparison of three neural network approaches for estimating joint angles and moments from inertial measurement units. *Sensors*, 21(13).
- Murdoch, W. J., Singh, C., Kumbier, K., Abbasi-Asl, R., and Yu, B. (2019). Definitions, methods, and applications in interpretable machine learning. *Proceedings of the National Academy of Sciences*, 116(44):22071–22080.

- Nasteski, V. (2017). An overview of the supervised machine learning methods. *Horizons*, 4:51–62.
- Neapolitan, R. E. and Jiang, X. (2018). Neural networks and deep learning. In *Artificial intelligence: With an introduction to machine learning*, page 23. CRC Press.
- Nelson, D. (2020). Supervised vs unsupervised learning. <https://www.unite.ai/supervised-vs-unsupervised-learning/>.
- Nguyen, H., Tran, K. P., Thomassey, S., and Hamad, M. (2021). Forecasting and anomaly detection approaches using lstm and lstm autoencoder techniques with the applications in supply chain management. *International Journal of Information Management*, 57:102282.
- Pellatt, L., Dewar, A., Philippides, A., and Roggen, D. (2021). *Mapping Vicon Motion Tracking to 6-Axis IMU Data for Wearable Activity Recognition*, pages 3–20.
- Pereira, A. (2016). 3darm inertial sensor-based 3d upper limb motion tracking and trajectories reconstruction. Master's thesis, Faculdade de Engenharia Universidade do Porto, Porto, Portugal.
- Peres, R. S., Barata, J., Leitao, P., and Garcia, G. (2019). Multistage quality control using machine learning in the automotive industry. *IEEE Access*, 7:79908–79916.
- Pesantez-Narvaez, J., Guillen, M., and Alcañiz, M. (2019). Predicting motor insurance claims using telematics data—xgboost versus logistic regression. *Risks*, 7(2).
- Piccinini, G. (2003). Alan turing and the mathematical objection. *Minds and Machines*, 13:23–48.
- Praveena, M. and Jaiganesh, V. (2017). A literature review on supervised machine learning algorithms and boosting process. *International Journal of Computer Applications*, 169:32–35.
- Ribeiro, P. M. S., Matos, A. C., Santos, P. H., and Cardoso, J. S. (2020). Machine learning improvements to human motion tracking with imus. *Sensors*, 20(21).
- Salman, A. G., Heryadi, Y., Abdurahman, E., and Suparta, W. (2018). Single layer multi-layer long short-term memory (lstm) model with intermediate variables for weather forecasting. *Procedia Computer Science*, 135:89–98. The 3rd International Conference on Computer Science and Computational Intelligence (ICCSCI 2018) : Empowering Smart Technology in Digital Era for a Better Life.
- Santos, S., Folgado, D., and Gamboa, H. (2020a). Upper-body movements: precise tracking of human motion using inertial sensors. <https://doi.org/10.5281/zenodo.4298987>.
- Santos, S., Folgado, D., Rodrigues, J., Mollaei, N., Fujão, C., and Gamboa, H. (2020b). Explaining the ergonomic assessment of human movement in industrial contexts. In Vilda, P., Fred, A., and Gamboa, H., editors, *BIOSIGNALS 2020 - 13th International Conference on Bio-Inspired Systems and Signal Processing, Proceedings; Part of 13th International Joint Conference on Biomedical Engineering Systems and Technologies, BIOSTEC 2020*, pages 79–88. SciTePress. 13th International Conference

- on Bio-Inspired Systems and Signal Processing, BIOSIGNALS 2020 - Part of 13th International Joint Conference on Biomedical Engineering Systems and Technologies, BIOSTEC 2020 ; Conference date: 24-02-2020 Through 26-02-2020.
- Schohn, G. and Cohn, D. (2000). Less is more: Active learning with support vector machines. In *ICML*, volume 2, page 6. Citeseer.
- Settles, B. (2009). Active learning literature survey. Technical report, University of Wisconsin-Madison Department of Computer Sciences, Madison, WI, USA.
- Shabbir, J. and Anwer, T. (2018). Artificial intelligence and its role in near future. *CoRR*, abs/1804.01396.
- Shi, G., Wang, Y., and Li, S. (2014). Human motion capture system and its sensor analysis. *Sensors and Transducers*, 172:206–212.
- Siame-Namini, S., Tavakoli, N., and Siame Namin, A. (2018). A comparison of arima and lstm in forecasting time series. In *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 1394–1401.
- Siame-Namini, S. and Namin, A. S. (2018). Forecasting economics and financial time series: Arima vs. lstm. *ArXiv*, abs/1803.06386.
- Snyder, H. (2019). Literature review as a research methodology: An overview and guidelines. *Journal of Business Research*, 104:333–339.
- Stellwall, M. (2019). Overview of traditional machine learning techniques. <https://blogs.oracle.com/ai-and-datascience/post/overview-of-traditional-machine-learning-techniques>.
- Su, B., Smith, C., and Gutierrez Farewik, E. (2020). Gait phase recognition using deep convolutional neural network with inertial measurement units. *Biosensors*, 10(9).
- Sun, X., Liu, M., and Sima, Z. (2020). A novel cryptocurrency price trend forecasting model based on lightgbm. *Finance Research Letters*, 32:101084.
- Sutton, R. S. and Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.
- Syafrudin, M., Alfian, G., Fitriyani, N. L., and Rhee, J. (2018). Performance analysis of iot-based sensor, big data processing, and machine learning model for real-time monitoring system in automotive manufacturing. *Sensors*, 18(9).
- Sylla, N., Bonnet, V., Colledani, F., and Fraise, P. (2014). Ergonomic contribution of able exoskeleton in automotive industry. *International Journal of Industrial Ergonomics*, 44(4):475–481.
- Taifi, M. (2020). Clean machine learning code. <https://leanpub.com/cleanmachinelearningcode>.

- Tashman, L. (2000). Out-of-sample tests of forecasting accuracy: an analysis and review. *International Forecasting Journal*, 16(4):437–450.
- Tinoco, J., Gomes Correia, A., and Cortez, P. (2014). Support vector machines applied to uniaxial compressive strength prediction of jet grouting columns. *Computers and Geotechnics*, 55:132–140.
- Trevino, A. (2016). Introduction to k-means clustering. <https://blogs.oracle.com/ai-and-datascience/post/introduction-to-k-means-clustering>.
- Truong, T. T., Dinh-Cong, D., Lee, J., and Nguyen-Thoi, T. (2020). An effective deep feedforward neural networks (dfnn) method for damage identification of truss structures using noisy incomplete modal data. *Journal of Building Engineering*, 30:101244.
- Verikas, A., Vaiciukynas, E., Gelzinis, A., Parker, J., and Olsson, M. C. (2016). Electromyographic patterns during golf swing: Activation sequence profiling and prediction of shot effectiveness. *Sensors*, 16:592.
- Wang, J., Ma, Y., Zhang, L., Gao, R. X., and Wu, D. (2018). Deep learning for smart manufacturing: Methods and applications. *Journal of Manufacturing Systems*, 48:144–156. Special Issue on Smart Manufacturing.
- Wang, Y., Pan, Z., Zheng, J., Qian, L., and Mingtao, L. (2019). A hybrid ensemble method for pulsar candidate classification. *Astrophysics and Space Science*, 364.
- Wirth, R. and Hipp, J. (2000). Crisp-dm: Towards a standard process model for data mining. *Proceedings of the 4th International Conference on the Practical Applications of Knowledge Discovery and Data Mining*.
- Xiang, Z., Yan, J., and Demir, I. (2020). A rainfall-runoff model with lstm-based sequence-to-sequence learning. *Water resources research*, 56(1):e2019WR025326.
- Xu, J., Wang, K., Lin, C., Xiao, L., Huang, X., and Zhang, Y. (2021). FM-GRU: A Time Series Prediction Method for Water Quality Based on seq2seq Framework. *Water*, 13(8):1031.
- Yildirim, P., Birant, D., and Alpyildiz, T. (2018a). Data mining and machine learning in textile industry. *WIREs Data Mining and Knowledge Discovery*, 8(1):e1228.
- Yildirim, P., Birant, D., and Alpyildiz, T. (2018b). Data mining and machine learning in textile industry. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 8.
- Yu, Y., Si, X., Hu, C., and Zhang, J. (2019). A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures. *Neural Computation*, 31(7):1235–1270.
- Zhang, F. and O'Donnell, L. J. (2020). Chapter 7 - support vector regression. In Mechelli, A. and Vieira, S., editors, *Machine Learning*, pages 123–140. Academic Press.

- Zhang, J., Li, P., Zhu, T., Zhang, W.-A., and Liu, S. (2020). Human motion capture based on kinect and imus and its application to human-robot collaboration. In *2020 5th International Conference on Advanced Robotics and Mechatronics (ICARM)*, pages 392–397.
- Zhang, W., Wu, C., Zhong, H., Li, Y., and Wang, L. (2021). Prediction of undrained shear strength using extreme gradient boosting and random forest based on bayesian optimization. *Geoscience Frontiers*, 12(1):469–477.
- Zhao, R., Yan, R., Wang, J., and Mao, K. (2017). Learning to monitor machine health with convolutional bi-directional lstm networks. *Sensors*, 17:273.



Anexos

a.1 Reconstrução Angular dos Movimentos após o *Madgwick orientation filter*.

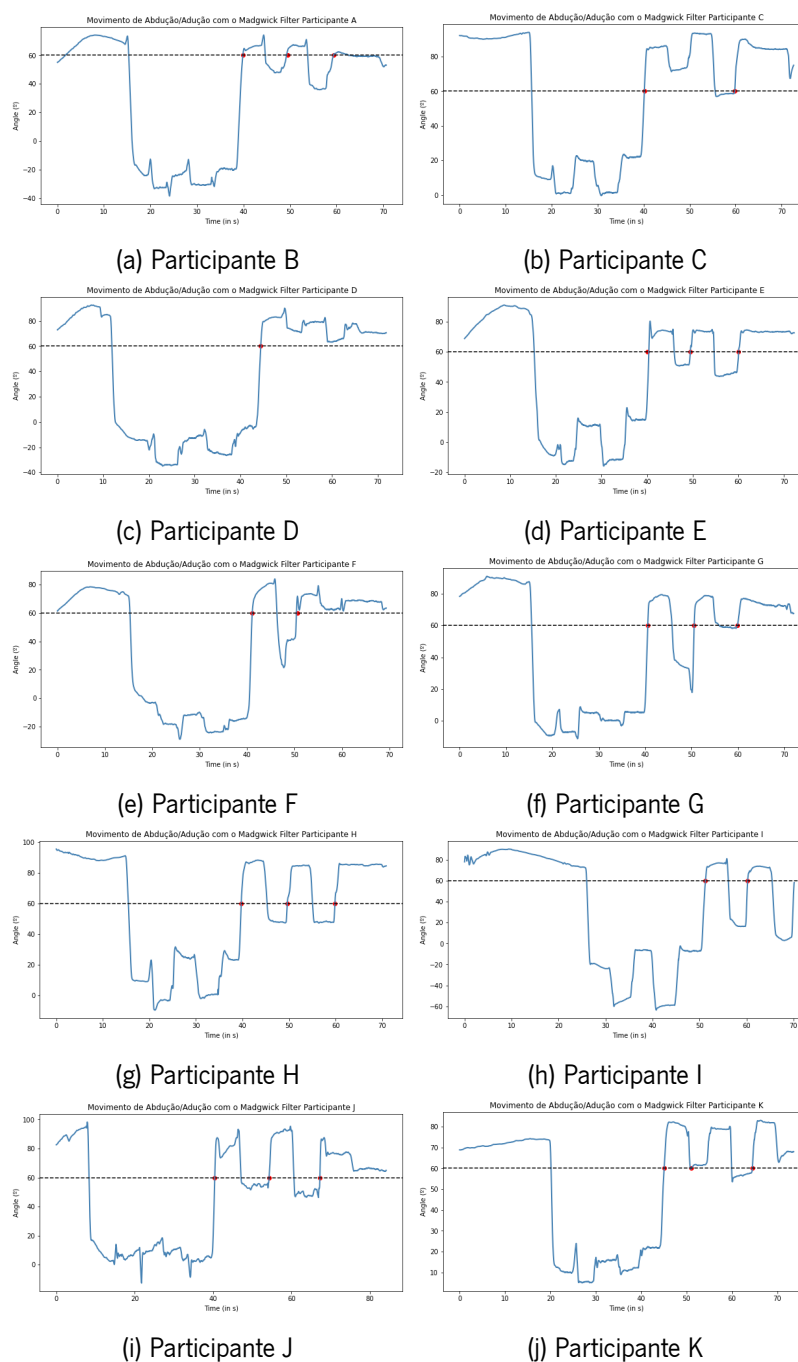


Figura 29: Reconstrução angular do movimento de abdução/adição de todos os participantes depois de utilizar o *Madgwick orientation filter*. A vermelho estão assinalados os pontos de análise.

a.2 Código Python Utilizado para a Implementação dos Modelos

```
def split_sequence(sequence, n_steps_in, n_steps_out):
    X, y = list(), list()
    for i in range(len(sequence)):
        end_ix = i + n_steps_in
        out_end_ix = end_ix + n_steps_out
        if out_end_ix > len(sequence):
            break
        seq_x, seq_y = sequence[i:end_ix], sequence[end_ix:out_end_ix]
        X.append(seq_x)
        y.append(seq_y)
    return array(X), array(y)
```

Listing A.1: Funções auxiliar para transformar os dados num array bidimensional.

```
CaseSeries = r(""" convertSeries = function(tr, lags){
  library("rminer")
  lag = 1:lags
  obj = CasesSeries(tr, lag)
  return (obj)
}""")
```

Listing A.2: Função auxiliar do *r* para converter as séries temporais num problema de supervised learning

.

```
metrics = r(""" r2 = function(ytrue, y_pred){
  obj = rminer::mmetric(unlist(ytrue), unlist(y_pred), metric=c("R2", "MAE", "MSE", "RMSE"))
  return (obj)
}""")
```

Listing A.3: Função auxiliar do *r* para utilizar as métricas de avaliação de desempenho.

a.3 Parâmetros dos Modelos Utilizados

```
TR2 = pd.DataFrame(CaseSeries(tr.X.values, n_steps_out))
TR2
TRy = TR2.pop("y")

TS2 = pd.DataFrame(CaseSeries(ts.X.values, n_steps_out))
```

TS2

```

rf = RandomForestRegressor(n_jobs=-1, verbose = 0)
rf.fit(TR2, TRy)

lr = LinearRegression(n_jobs=-1)
rf.fit(TR2, TRy)

svr = SVR()
rf.fit(TR2, TRy)

modelNormal = Sequential()
modelNormal.add(LSTM(100, activation=tf.keras.layers.LeakyReLU(),
input_shape=(n_steps_in, n_features)))
modelNormal.add(Dense(n_steps_out))
modelNormal.compile(optimizer='adam', loss='mse')
# fit model
modelNormal.fit(Xtrain, ytrain, epochs=50, verbose=2,
                shuffle = False)

modelStacked = Sequential()
modelStacked.add(LSTM(100, activation=tf.keras.layers.LeakyReLU(),
return_sequences=True, input_shape=(n_steps_in, n_features)))
modelStacked.add(LSTM(100, activation='relu'))
modelStacked.add(Dense(n_steps_out))
modelStacked.compile(optimizer='adam', loss='mse')
# fit model
modelStacked.fit(Xtrain, ytrain, epochs=50, verbose=2,
                shuffle=False)

ytrain2 = ytrain.copy()
ytrain2 = ytrain2.reshape((ytrain2.shape[0], ytrain2.shape[1],
                n_features))

modelSeq2Seq = Sequential()
modelSeq2Seq.add(LSTM(100, activation=tf.keras.layers.LeakyReLU(),
input_shape=(n_steps_in, n_features)))
modelSeq2Seq.add(RepeatVector(n_steps_out))
modelSeq2Seq.add(LSTM(100, activation=tf.keras.layers.LeakyReLU(),
return_sequences=True))
modelSeq2Seq.add(TimeDistributed(Dense(1)))
modelSeq2Seq.compile(optimizer='adam', loss='mse')
# fit model
modelSeq2Seq.fit(Xtrain, ytrain, epochs=50, verbose=2,

```



```
shuffle=False)
```

Listing A.4: Parâmetros dos modelos utilizados.

a.4 Pontos de Análise Para Cada Um dos Participantes

Tabela 11: Pontos que antecedem um momento que ultrapassa o risco de WMSDs para cada um dos participantes.

	A	B	C	F	E	F	G	H	I	J	K
tr1	39.87	34.27	40.15	44.44	40.04	41.09	40.58	39.72	51.27	40.44	45.05
tr2	49.51	43.76	59.85	-	49.50	50.67	50.59	49.61	60.25	54.30	51.09
tr3	59.34	53.72	-	-	60.06	-	59.85	59.85	-	67.26	64.51

a.5 Avaliação das Experiências

```
# choose a number of time steps
n_steps_in, n_tests, n_steps_out = 1, 25, 25
n_steps_incent, n_testscent, n_steps_outcent = n_steps_in/100,
n_tests/100, n_steps_out/100

Abovesixty = np.where(closeValbtc.X.values >= 60)

ps = closeValbtc.iloc[Abovesixty]
ps

#exemplo dos pontos de análise do participante A
PointCutters = [39.87, 49.51, 59.34]

splitpoint = np.round((PointCutters[2] - n_testscent), 2) #seconds
before first test

splitpointTs = PointCutters[2] + n_steps_outcent #test with the next
movement

rowSplit1 = np.where(closeValbtc.index.values == splitpoint)[0][0]
rowSplit2 = np.where(closeValbtc.index.values == splitpointTs)[0][0]

tr = closeValbtc.iloc[:rowSplit1]
ts1 = closeValbtc.iloc[rowSplit1:rowSplit2]
```

Listing A.5: Código Python para a avaliação dos pontos críticos.

```

def evaluate_forecasts(test, forecasts, n_seq, mode = "normal", slack =
0.5):
    verticalEvaluation = pd.DataFrame(columns=["T", "R2", "MAE", "MSE", "
        RMSE", "DatasetActual", "DatasetPred", "Score"])
    vertical = []
    for i in range(n_seq):

        actual = [row[i][0] for row in test]
        #print("Sample of ACTUAL SEQUENCE:", actual, "with total size of:",
            len(actual))
        if mode == "normal":
            predicted = [forecast[i]+slack for forecast in forecasts]
        else:
            predicted = [forecast[i][0]+slack for forecast in forecasts]
        #print("Sample of PREDICTED SIGNAL SEQUENCE:", predicted, "with
            total size of:", len(predicted))
        met = metrics(actual, predicted)
        r2 = met[0]
        MAE = met[1]
        MSE = met[2]
        RMSE = met[3]
        vertical2 = [[i+1, r2, MAE, MSE, RMSE, actual, predicted, len(np.where
            (np.array(predicted) >=60)[0])]
        vertical = pd.DataFrame(vertical2, columns=["T", "R2", "MAE", "MSE",
            "RMSE", "DatasetActual", "DatasetPred", "Score"])
        #add to original dataset
        verticalEvaluation = verticalEvaluation.append(vertical)
        print('t+%d - R^2: %f' % ((i+1), r2), '- MAE: %f' % (MAE), '- MSE:
            %f' % (MSE), '- RMSE: %f' % (RMSE), 'Number of risk predictions:
            %f' % len(np.where(np.array(predicted) >=60)[0]))
    return verticalEvaluation

```

Listing A.6: Função de avaliação.

```

horizontalFinalseq2seq = pd.DataFrame(columns=["T", "Instant", "R2", "MAE", "
    MSE", "RMSE", "DatasetActual", "DatasetPred"])
slack = 0.5
for i in range(0, n_steps_out):
    actual = [row[i][0] for row in ytest]
    predicted = [forecast[i][0]+slack for forecast in forecasts]
    contador = 0
    fimci = contador + n_steps_out
    for vals in range(0, len(actual)):

```

```

actual2 = actual[contador:fimci]

predicted2 = predicted[contador:fimci]

met = metrics(actual2, predicted2)
r2 = met[0]
MAE = met[1]
MSE = met[2]
RMSE = met[3]

print('t+%d - Instant+%d - R^2: %f' % (i+1,vals+1, r2), '- MAE: %f'
      % (MAE), '- MSE: %f' % (MSE), '- RMSE: %f' % (RMSE))
#break
horizontal2 = [[i+1, vals+1, r2, MAE,MSE,RMSE, actual2, predicted2
               ]]
horizontal = pd.DataFrame(horizontal2, columns=["T","Instant","R2",
        "MAE", "MSE", "RMSE", "DatasetActual", "DatasetPred"])
horizontalFinalseq2seq = horizontalFinalseq2seq.append(horizontal)
contador +=n_steps_in
fimci = contador + n_steps_out

```

Listing A.7: Função para obter as previsões.