CEFET/RJ
DIRETORIA DE PESQUISA E PÓS-GRADUAÇÃO

DETECÇÃO DE QUEDAS DE SOLDADOS COMBATENTES POR MEIO DE REDES NEURAIS ARTIFICIAIS

Leandro de Mattos Bento Soares

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Ciência da Computação, Centro Federal de Educação Tecnológica Celso Suckow da Fonseca CEFET/RJ, como parte dos requisitos necessários à obtenção do grau de mestre.

Orientadores:
Eduardo Bezerra, D.Sc.
Diego Brandão, D.Sc.

Rio de Janeiro,
Janeiro de 2025

Detecção de quedas de soldados combatentes por meio de redes neurais artificiais

Dissertação de Mestrado em Ciência da Computação, Centro Federal de Educação Tecnológica Celso Suckow da Fonseca, CEFET/RJ.

Leandro de Mattos Bento Soares

Aprovada por:

Eduardo Bezerra, D.Sc. (CEFET/RJ) - Presidente e orientador

Diego Brandão, D.Sc. (CEFET/RJ) - Orientador

Jorge Soares, D.Sc. (CEFET/RJ)

Pedro González, D.Sc. (UFRJ)

Pablo Rangel, D.Sc. (IPqM)

Cláudio Miceli de Farias, D.Sc. (UFRJ)

Rio de Janeiro,

Janeiro de 2025

Ficha catalográfica elaborada pela Biblioteca Central do CEFET/RJ

S676 Soares, Leandro de Mattos Bento
Detecção de quedas de soldados combatentes por meio de redes neurais artificiais / Leandro de Mattos Bento Soares. — 2024.
93f. : il. color. , enc.

Dissertação (Mestrado) Centro Federal de Educação Tecnológica Celso Suckow da Fonseca, 2024.
Bibliografia : f. 90-93
Orientador: Eduardo Bezerra
Coorientador: Diego Brandão

1. Redes neurais (Cmputação). 2. Detecção de queda. 3. Soldados - Brasil. 4. Brasil – Forças armadas. I. Bezerra, Eduardo. (Orient.). II. Brandão, Diego. (Coorient.). III. Título.

CDD 006.3

Elaborada pela bibliotecária Tania Mello – CRB/7 nº 5507/04

DEDICATÓRIA

Dedico este trabalho à minha amada esposa, Thayssa, por estar sempre ao meu lado, oferecendo suporte incondicional e me encorajando a persistir, mesmo nos momentos mais difíceis. Também dedico à minha filha, Ayla, que, mesmo antes de chegar a este mundo, foi uma fonte de força e inspiração para superar os desafios dessa maratona final. Amo vocês infinitamente.

AGRADECIMENTOS

A realização desta dissertação só foi possível graças ao apoio e dedicação de diversas pessoas e instituições, às quais sou profundamente grato.

Em primeiro lugar, agradeço ao CEFET/RJ e aos meus orientadores, Prof. Dr. Eduardo Bezerra e Prof. Dr. Diego Brandão, pela orientação exemplar, paciência e incentivo ao longo de toda a pesquisa. Suas contribuições foram indispensáveis para o desenvolvimento deste trabalho.

Ao Instituto de Pesquisas da Marinha (IPqM), expresso minha gratidão por ter proporcionado os recursos necessários para a coleta de dados que sustentaram esta pesquisa. Um agradecimento especial aos pesquisadores Pablo Rangel e José Gomes, cujos esforços e colaborações técnicas foram fundamentais para o sucesso do projeto.

Sou também imensamente grato à FINEP (Financiadora de Estudos e Projetos) e à CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) pelo financiamento e apoio institucional, que tornaram esta pesquisa viável.

A todos, meu mais sincero muito obrigado.

RESUMO

Detecção de quedas de soldados combatentes por meio de redes neurais artificiais

Leandro de Mattos Bento Soares

Orientadores:
Eduardo Bezerra, D.Sc.
Diego Brandão, D.Sc.

Resumo da Dissertação submetida ao Programa de Pós-graduação em Ciência da Computação do Centro Federal de Educação Tecnológica Celso Suckow da Fonseca CEFET/RJ como parte dos requisitos necessários à obtenção do grau de mestre.

Durante operações militares envolvendo soldados combatentes, estes enfrentam uma série de desafios físicos devido à natureza das atividades, incluindo lesões que podem resultar em choque hemorrágico hipovolêmico e, em casos extremos, levar ao óbito. Assim, é fundamental que os centros de controle dessas operações sejam rapidamente informados sobre a condição dos combatentes, permitindo a agilização do resgate e a preservação de suas vidas. Nesse contexto, a Marinha do Brasil desenvolveu o projeto "Combatente do Futuro", que tem como objetivo criar modelos capazes de detectar situações que demandem socorro imediato ao combatente, bem como outras ocorrências relevantes, como conflitos ou situações de risco, que possam ser identificadas automaticamente a partir das variações inerciais dos combatentes. Para a coleta dessas variáveis inerciais, foram escolhidos dispositivos específicos: um modelo de *smartwatch*, posicionado nos pulsos direito e esquerdo, e um modelo de *smartphone*, posicionado no peito do soldado. Dentro desse projeto, destaca-se a investigação de métodos para detecção de quedas de soldados, dividida em duas etapas principais. A primeira etapa consiste na implementação de um modelo de rede neural para detectar padrões de queda, enquadrando o problema como uma tarefa de classificação. Já a segunda etapa busca propor um classificador binário, com o objetivo de diferenciar quedas operacionais de quedas causadas por ferimentos. Esta dissertação concentra-se na detecção de quedas, pertencente à primeira etapa. Para a realização desta pesquisa, foram coletados dados de atividades de vida diária, atividades de quedas e atividades militares com porte e sem porte de arma de 15 militares da Marinha do Brasil, simulando as entradas para o modelo de rede neural a ser implementado e testado em situações de campo. A partir desse conjunto de dados, foram estabelecidos 12 cenários experimentais para cada posição dos sensores, explorados nos domínios do tempo e da frequência, com o objetivo de identificar o conjunto de variáveis inerciais, a posição dos sensores e o domínio que apresentassem o melhor desempenho na classificação de eventos de queda. O conjunto de configurações escolhido foi implementado na arquitetura de rede neural Rede Neurais Convolucionais Unidimensionais (CNN1D), cujos hiperparâmetros foram ajustados por meio de otimização *bayesiana*.

Palavras-chave:
detecção de queda, soldados, redes neurais, dispositivos vestíveis, aplicação militar, variáveis inerciais.

Rio de Janeiro,

Janeiro de 2025

ABSTRACT

Detecção de quedas de soldados combatentes por meio de redes neurais artificiais

Leandro de Mattos Bento Soares

Advisors:
Eduardo Bezerra, D.Sc.
Diego Brandão, D.Sc.

Abstract of dissertation submitted to Programa de Pós-graduação em Ciência da Computação - Centro Federal de Educação Tecnológica Celso Suckow da Fonseca CEFET/RJ as partial fulfillment of the requirements for the degree of master.

During military operations involving combat soldiers, they face a series of physical challenges due to the nature of the activities, including injuries that can result in hypovolemic hemorrhagic shock and, in extreme cases, lead to death. Thus, it is essential that the control centers of these operations be quickly informed about the condition of the combatants, allowing for the acceleration of rescue efforts and the preservation of their lives. In this context, the Brazilian Navy developed the "Future Combatant"project, which aims to create models capable of detecting situations requiring immediate assistance for the combatant, as well as other relevant occurrences, such as conflicts or risk situations, that can be automatically identified from the inertial variations of the combatants. For the collection of these inertial variables, specific devices were chosen: a smartwatch model, positioned on the right and left wrists, and a smartphone model, positioned on the soldier's chest. Within this project, the investigation of methods for detecting soldier falls stands out, divided into two main stages. The first stage involves implementing a neural network model to detect fall patterns, framing the problem as a classification task. The second stage aims to propose a binary classifier to differentiate operational falls from those caused by injuries. This dissertation focuses on fall detection, belonging to the first stage. For this research, data were collected on daily life activities, fall activities, and military activities with and without carrying weapons from 15 members of the Brazilian Navy, simulating inputs for the neural network model to be implemented and tested in field situations. From this dataset, 12 experimental scenarios were established for each sensor position, explored in the time and frequency domains, aiming to identify the set of inertial variables, sensor positions, and the domain that presented the best performance in classifying fall events. The chosen configuration set was implemented in the CNN1D neural network architecture, whose hyperparameters were adjusted through Bayesian optimization.

Key-words:
fall detection, soldiers, neural networks, wearable devices, military application, inertial variables

Rio de Janeiro,

Janeiro de 2025

Sumário

I Introdução **13**

I.1 Contextualização 13

I.2 Motivação 15

I.3 Problema e Suposição 16

I.4 Objetivos 17

I.5 Metodologia 18

I.6 Organização dos capítulos 18

II Fundamentação Teórica **19**

II.1 Redes Neurais Artificiais 19

II.1.1 Redes Neurais Convolucionais 21

II.2 Otimização Bayesiana de Hiperparâmetros 24

II.3 Transformada de Fourier 25

II.4 Métricas para Avaliação de Classificadores 26

III Trabalhos Relacionados **28**

III.1 Introdução 28

III.2 Análise dos Trabalhos Relacionados 29

III.3 Trabalhos Relevantes 32

IV Metodologia **43**

IV.1 Design do Sistema 43

IV.2 Conjunto de Dados *IPqM_Fall* 46

IV.2.1 Dispositivos Utilizados e Variáveis Coletadas 47

IV.2.2 Coleta de Dados 47

IV.2.3 Participantes Voluntários 49

IV.2.4 Atividades Coletadas 50

IV.2.5 Organização do Conjunto de Dados 53

IV.3 Escolha das Arquiteturas de Redes Neurais 54

IV.4 Pré-Processamento 56

IV.4.1 Extração de características 56
IV.4.2 Análise e Preparação dos Dados 56
IV.5 Cenários Experimentais e Abordagem de Rotulação 62
IV.6 Implementação das Redes Neurais 63

V Experimentos **69**
V.1 Introdução 69
V.2 Resultados - *Smartwatch* esquerdo 71
V.2.1 Resultados - binário_1 71
V.2.2 Resultados - binário_2 75
V.2.3 Comparação entre as abordagens de rotulação - lado esquerdo. 78
V.3 Resultados - *Smartwatch* direito 81
V.3.1 Resultados - binário_1 81
V.3.2 Resultados - binário_2 84
V.3.3 Comparação entre as abordagens de rotulação - lado direito. 87
V.4 Resultados - *Smartphone* peito 89
V.4.1 Resultados - binário_1 89
V.4.2 Resultados - binário_2 92
V.4.3 Comparação entre as abordagens de rotulação - Peito. 95
V.5 Comparação entre as três posições: esquerda, direita e peito. 96

VI Conclusão **99**
VI.1 Visão Geral dos Resultados 99
VI.2 Limitações e Trabalhos Futuros 100
VI.3 Reprodutibilidade 101

Referências **102**

Lista de Figuras

II.1 Neurônio Artificial 20

II.2 Operação de Convolução 22

II.3 Rede Neural Convolucional 2D 23

II.4 Rede Neural Convolucional 1D 24

III.1 Artigos por ano 29

III.2 Ocorrências por Tipo de Dispositivo 30

III.3 Ocorrências por Tipo de Sensor 31

III.4 Ocorrências por Tipo de Rede Neural 32

IV.1 Esboço do Sistema de Detecção de Baixas 44

IV.2 Funcionamento do algoritmo de detecção de quedas do projeto Combatente do Futuro 46

IV.3 Posição dos dispositivos utilizados no corpo. 48

IV.4 Uniformes OP1, AD e OP2, respectivamente. 50

IV.5 Fuzis Airsoft M4 4003MG e Colt M4 Carbine 53

IV.6 Atividades de queda registradas pelo *smartwatch* esquerdo do ID 3: Fall_1, Fall_2, Fall_3 e Fall_5, respectivamente. 57

IV.7 Atividades de queda registradas pelo *smartwatch* esquerdo do ID 3: ADL_1, ADL_2, ADL_3, ADL_4, ADL_5 e ADL_7, respectivamente. 58

IV.8 Atividades de queda registradas pelo *smartwatch* esquerdo de ID 3: ADl_12, ADL_12_withRifle, Fall_3 e Fall_3_withRifle, respectivamente. 59

IV.9 Atividades registradas pelo *smartwatch* esquerdo do ID 3: Fall_3 e OM_6, respectivamente. 61

V.1 Teste de *Kruskal-Wallis*, lado esquerdo, binário_1. 72

V.2 Teste de *Mann-Whitney* para os domínios da frequência e do tempo, lado esquerdo, binário_1. 74

V.3 Teste de *Kruskal-Wallis*, lado esquerdo, binário_2. 75

V.4 Teste de *Mann-Whitney* para os domínios da frequência e do tempo, lado esquerdo, binário_2. 77

V.5 Teste de *Mann-Whitney* para as abordagens de rotulação, lado esquerdo. 79
V.6 Teste de *Mann-Whitney* para os melhores cenários experimentais para ambas abordagens de rotulação, lado esquerdo, *binary_ class_ label_ 2.* 80
V.7 Teste de *Kruskal-Wallis*, lado direito, binário_1. 81
V.8 Teste de *Mann-Whitney* para os domínios do tempo e da frequência, lado direito, binário_1. 83
V.9 Teste de *Kruskal-Wallis*, lado direito, binário_2. 84
V.10 Teste de *Mann-Whitney* para os domínios do tempo e da frequência, lado direito, binário_2. 86
V.11 Teste de *Mann-Whitney* para as abordagens de rotulação, lado direito. 87
V.12 Teste de *Mann-Whitney* para os melhores cenários de ambas as abordagens de rotulação, lado direito. 88
V.13 Teste de *Kruskal-Wallis*, peito, binário_1. 89
V.14 Teste de *Mann-Whitney* para os domínios do tempo e da frequência, peito, binário_1. 91
V.15 Teste de *Kruskal-Wallis*, peito, binário_2 92
V.16 Teste de *Mann-Whitney* para os domínios do tempo e da frequência, peito, binário_2. 94
V.17 Teste de *Mann-Whitney* para as abordagens de rotulação, peito. 95
V.18 Teste de *Mann-Whitney* para os melhores cenários de ambas as abordagens de rotulação, peito. 96
V.19 Teste de *Kruskal Wallis* para os melhores cenários para cada posição. 97

Lista de Tabelas

III.1 Critérios de exclusão 28

III.2 Artigos selecionados 39

IV.1 Idade, patente, sexo e informações antropométricas dos militares voluntários. 49

IV.2 Atividades da Vida Diária. 51

IV.3 Atividades de Operações Militares. 52

IV.4 Atividades de Queda. 53

IV.5 Espaço de Busca CNN1D 65

IV.6 Arquiteturas CNN1D otimizadas para os dados do *smartwatch* esquerdo, rótulos: binário_1. 65

IV.7 Arquiteturas CNN1D otimizadas para os dados do *smartwatch* esquerdo, rótulos: binário_2. 66

IV.8 Arquiteturas CNN1D otimizadas para os dados do *smartwatch* direito, rótulos: binário_1. 66

IV.9 Arquiteturas CNN1D otimizadas para os dados do *smartwatch* direito, rótulos: binário_2. 67

IV.10 Arquiteturas CNN1D otimizadas para os dados do *smartphone* peito, rótulos: binário_1. 67

IV.11 Arquiteturas CNN1D otimizadas para os dados do *smartphone* peito, rótulos: binário_2 68

V.1 Resultados do desempenho dos Treinamento para cada cenário, lado esquerdo, binário_1. 73

V.2 Resultados do desempenho dos Treinamento para cada cenário, lado esquerdo, binário_2. 76

V.3 Resultados do desempenho dos Treinamento para cada cenário, lado direito, binário_1. 82

V.4 Resultados do desempenho dos Treinamento para cada cenário, lado direito, binário_2. 85

V.5 Resultados do desempenho dos Treinamento para cada cenário, peito, binário_1. 90

V.6 Resultados do desempenho dos Treinamento para cada cenário, lado peito, binário_2. 93

Lista de Abreviações

AVD	Atividades Da Vida Diária	37, 38, 50, 57, 58
C2	Sistema De Comando E Controle	13, 14, 16
CNN	Rede Neurais Convolucionais	20, 21, 22, 23, 31, 35, 54, 55
CNN1D	Rede Neurais Convolucionais Unidimensionais	23, 24, 31, 43, 63, 64, 65, 66, 67, 68, 70, 72, 76, 82, 85, 90, 92, 98, 100
CNN2D	Rede Neurais Convolucionais Bidimensionais	23, 24, 31
ES	Especificidade	27, 73
FN	Falso Negativos	26, 73
FP	Falso Positivos	26, 73
GLO	Garantia Da Lei E Da Ordem	13, 14
GRU	Unidade Recorrente Com Portas	35
IPQM	Instituto De Pesquisas Da Marinha	14, 17, 49
LSTM	Redes Recorrentes De Memória De Longo E Curto Prazo	35
MB	Marinha Do Brasil	13, 14
MCC	Coeficiente De Correlação De Matthews	26, 64, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 86, 87, 88, 89, 91, 92, 93, 94, 95, 96, 97, 98
MINUSTAH	Missão Das Nações Unidas Para A Estabilização Do Haiti	14
MLP	Rede Neural Perceptron De Multicamadas	31, 34, 35, 37, 54, 55
OM	Operações Militares	51
PR	Precisão	27, 73
RNA	Redes Neural Artificial	20
RNAS	Redes Neurais Artificiais	19, 20
RNN	Rede Neural Recorrente	31, 37, 54, 55, 58
SE	Sensibilidade	26, 73
TN	Verdadeiros Negativos	26, 73
TP	Verdadeiros Positivos	26, 73

Capítulo I Introdução

I.1 Contextualização

Operações militares que envolvem o engajamento em campo por parte de combatentes são uma necessidade para todos os países. Elas podem variar desde combates em tempos de guerra, como a retomada de instalações militares, resgate de reféns, neutralização de alvos, defesa de cidades sitiadas, entre outras, até operações em tempos de paz, como as de Garantia da Lei e da Ordem (GLO). De acordo com o Manual de GLO [Ministério da Defesa, 2014], as operações de GLO são conduzidas pelas Forças Armadas em situações graves de perturbação da ordem, quando as forças tradicionais de segurança pública se esgotam. O Presidente da República as determina de forma emergencial em uma área pré-estabelecida e por tempo limitado. Tais operações têm como objetivo a preservação da ordem pública, da incolumidade das pessoas e do patrimônio. Também pode-se citar as incursões das polícias militares áreas dominadas pelo tráfico de entorpecentes dentro de favelas em perímetros urbanos, que assumem características de guerrilha. A Polícia Militar do Rio de Janeiro contabiliza 477 mortos em ação desde 2003, sendo esta uma estatística geral que engloba os mortos nas incursões em favelas [Instituto de Segurança Pública do RJ, 2023]. Durante tais operações, os combatentes estão expostos ao risco de ferimentos derivados do confronto com as tropas inimigas, sendo o mais comum deles os ferimentos causados por armas de fogo, ocorrências essas conhecidas como baixas. Tais lesões podem levar o combatente ao óbito caso não seja estabelecida a comunicação imediata do sinistro junto ao Sistema de Comando e Controle (C2) e não ocorra presteza na operação de resgate. O atraso na comunicação do sinistro também pode ocasionar lentidão no envio de tropas para reforço, que têm como objetivo garantir a vida dos feridos e o sucesso das operações.

O Ministério da Defesa contabilizou, no estudo Histórico de operações de GLO [Ministério da Defesa, 2023], cento e quarenta e cinco operações, classificadas nas seguintes categorias: Violência Urbana, Greve da Polícia Militar, Segurança de Eventos, Garantia de Votação/Apuração e Outros. No âmbito da Marinha do Brasil (MB), dentre as 145 operações de GLO, a MB participou de 46 operações, destacando-se a descritas a seguir.

1. Operação Choque de Paz. Foi realizada em 2011 e teve como objetivo a retomada das favelas da Rocinha e do Vidigal, duas importantes favelas do Rio de Janeiro.

2. Operações Arcanjo. As operações Arcanjo foram realizadas no período de 2010 a 2012. Configuravam uma resposta imediata do Governo Federal em reforço ao Governo Estadual diante das retaliações violentas do crime organizado. Elas tiveram como objetivo a retomada do Complexo do Alemão e da Penha, composto por várias favelas no Rio de Janeiro.

3. Operação São Francisco. Ocorreu em 2014, quando as forças de segurança do estado realizaram uma ação integrada para combater a criminalidade e o tráfico de drogas em comunidades da zona norte do Rio de Janeiro, incluindo favelas como o Complexo da Maré.

Além das operações de GLO, a MB teve uma importante participação na Missão das Nações Unidas para a Estabilização do Haiti (MINUSTAH). Tal operação ocorreu de 2004 a 2017 e teve como objetivo estabilizar o Haiti após o conflito civil. A Marinha contribuiu com forças navais e tropas de fuzileiros navais para apoiar a segurança e a reconstrução do Haiti.

Tendo em vista a participação ativa de seus combatentes em operações militares que podem resultar em conflito armado, assim como as operações referentes às outras Forças Armadas e Polícias Militares, a Marinha do Brasil, por meio do Instituto de Pesquisas da Marinha (IPqM), iniciou o projeto Combatente do Futuro. Uma das metas do projeto é o estudo de parâmetros biométricos que possam ser coletados dos combatentes em ação e que sejam relevantes na inferência automática de intercorrências úteis em procedimentos de apoio e/ou salvaguarda dos combatentes.

O objetivo é criar modelos capazes de detectar tanto situações em que seja necessário socorrer o combatente quanto intercorrências de interesse, como conflitos ou situações de perigo, que possam ser automaticamente identificadas a partir das variações biométricas dos combatentes. Esses modelos serão integrados aos sistemas de comando e controle, que é o órgão responsável pela gestão de todas as forças em operação envolvendo os militares. Cada uma das três Forças Armadas possui um software para apoiar seu sistema de comando e controle [CASNAV, 2023]. Dentre as diversas responsabilidades do C2, uma delas é minimizar as baixas em operações tanto para militares quanto para civis.

Com o objetivo de alcançar essa meta, foram escolhidos dispositivos vestíveis portáteis para a coleta de dados biométricos, como *smartwatches* e *smartphones*. Esses equipamentos foram selecionados devido à natureza das operações militares, que exige que os dispositivos usados para a detecção de dados biométricos sejam leves, fáceis de usar e de vestir, permitindo a obtenção de informações de forma eficiente. Através dos dados fornecidos por esses dispositivos, busca-se inferir informações sobre a saúde e o desempenho dos combatentes, bem como identificar situações operacionais relevantes relacionadas aos dados biométricos. Nesse contexto, o projeto adquiriu relógios de pulso da marca Samsung, especificamente o modelo Galaxy Watch 4, e o aparelho celular LG Velvet.

Em termos de sensores, esses dispositivos vestíveis possuem uma variedade de recursos, incluindo acelerômetro, barômetro, sensor geomagnético, giroscópio, sensor de luz, foto pletismógrafo (sensor óptico de frequência cardíaca), sensor elétrico de coração e sensor de análise biométrica por impedância. O fabricante disponibiliza uma interface de programação que permite que aplicativos, desenvolvidos por qualquer usuário, acessem alguns dos parâmetros biométricos medidos. No entanto, certas informações são exclusivas das aplicações nativas desenvolvidas pelo fabricante ou fornecedores de aplicativos associados a ele.

Por meio dessa interface de aplicação é possível obter medidas biométricas como número de passos, frequência cardíaca, aceleração nos três eixos e o giroscópio nos três eixos. Esses dados fornecem esclarecimentos valiosos para a avaliação do estado físico e emocional dos combatentes, bem como para o monitoramento de situações críticas durante as operações militares.

É possível que a utilização da inteligência artificial e do aprendizado de máquina no contexto discutido possa proporcionar uma maior eficiência na detecção de situações de risco e na tomada de decisões. Sendo assim, utilizada como uma ferramenta de auxílio na redução das baixas de militares em combate. Dessa maneira, os dados biométricos dos combatentes poderiam ser analisados em tempo real, permitindo uma resposta rápida e precisa diante de situações críticas.

Essa abordagem, que une a realidade das operações táticas com o potencial da inteligência artificial, representa um avanço significativo no campo das operações militares. A capacidade de prever e reagir prontamente a eventos perigosos pode contribuir para a preservação da vida dos combatentes, bem como para o sucesso das operações em geral.

A pesquisa abordada nesta dissertação está inserida no contexto da inteligência artificial, mais especificamente no campo do aprendizado de máquina, e faz parte do escopo do projeto Combatente do Futuro. O objetivo da pesquisa proposta é coletar dados oriundos dos sensores de acelerômetro e giroscópio de militares da Marinha do Brasil e utilizar esses dados para treinar modelos de redes neurais artificiais em diferentes cenários para a detecção de quedas. Posteriormente, pretende-se comparar a eficácia dos diferentes cenários de treinamento, selecionando aquele que obtiver melhor desempenho preditivo para ser implementado no sistema de detecção de quedas do projeto Combatente do Futuro.

I.2 Motivação

A vida é um dos bens mais valiosos que possuímos, e sua preservação deve ser sempre uma prioridade. Em nenhum lugar tal pensamento é mais evidente do que no contexto dos combatentes envolvidos em conflitos armados. Para esses homens e mulheres, “tempo” é sinônimo de “vida”. O manual de Apoio de Saúde em Operações Conjuntas [Ministério da Defesa, 2017] estabelece que os primeiros dez minutos após a lesão sofrida pelo combatente são os mais vitais e correspondem

às ações que devem ser tomadas imediatamente para estabilizar uma baixa, a fim de evitar o agravamento do seu estado de saúde. Em outras palavras, os procedimentos realizados nos primeiros 10 minutos, na maioria dos casos, salvam uma vida. Essa diretriz recebeu o nome de "10 Minutos de Platina" ("Platinum 10").

Outro conceito relevante é chamado de "Hora de Ouro", que se refere ao tempo crucial em que uma vítima tem maiores chances de sobrevida após a ocorrência de um incidente. Dessa forma, o planejamento da localização das Instalações de Saúde é condicionado especialmente ao fator tempo e à velocidade dos meios disponíveis na Cadeia de Evacuação [Ministério da Defesa, 2017].

Diante do exposto, é crucial para a sobrevivência de um combatente ferido que a informação sobre seu abatimento chegue ao C2 de forma rápida e precisa, permitindo a mobilização do resgate no menor tempo possível. Isso garante que os 10 Minutos de Platina sejam respeitados, aumentando as chances de sobrevivência. Nesse sentido, a presente pesquisa é justificável, pois visa desenvolver um método de aprendizado de máquina para a detecção imediata da queda de soldados em combate. A pesquisa também servirá de base para estudos complementares que utilizarão seus resultados na identificação de ferimentos graves, além de assegurar o envio ágil das informações ao sistema de comando e controle. Isso é vital, pois, em diversas operações, o comunicado sobre a baixa de um soldado pode ser atrasado devido aos entraves do combate, tornando-se tarde demais para o resgate e a salvaguarda da vida.

Em adição ao exposto, historicamente as pesquisas de âmbito militar tendem a alavancar os avanços de pesquisas no meio civil. O algoritmo de reconhecimento de quedas desenvolvido poderá ser usado no auxílio aos idosos em um sistema de assistência emergencial, facilitando e agilizando o socorro médico. A aplicação desse método de aprendizado de máquina pode trazer benefícios significativos além do campo militar, mas também na melhoria da qualidade de vida da população civil.

I.3 Problema e Suposição

A problemática da pesquisa dessa dissertação deriva-se das seguintes perguntas de pesquisa:

- Quais componentes da força de aceleração, provenientes dos sensores de acelerômetro e giroscópio, devem ser selecionados para uma detecção eficiente de quedas?
- Um modelo de rede neural artificial é capaz de aprender os padrões inerentes à queda do combatente e obter uma alta taxa de acerto viabilizando o uso do algoritmo em cenário real?

Um modelo de rede neural, supostamente, será capaz de aprender o padrão de queda e detectar o momento em que o corpo do combatente se choca com o chão.

I.4 Objetivos

O presente trabalho tem como objetivo geral a detecção da queda de um combatente em ação. Objetivo que está alinhado com a primeira etapa do projeto Combatente do Futuro que possui como objetivo final detectar a baixa de um combatente, que no âmbito do projeto é classificada como quedas que resultam em choque hipovolêmico por hemorragia. O propósito é o de acionar alertas para os operadores do sistema de C2 e desencadear ações de socorro ou envio de apoio de campo. O Sistema de detecção de baixas a ser desenvolvido pelo IPqM para o projeto está dividido em dois algoritmos. O primeiro corresponde a um modelo de Rede Neural capaz de detectar padrões de queda com eficiência, alta acurácia e alta precisão por meio de classificação. O segundo corresponde a um algoritmo classificador binário, com o objetivo de distinguir quedas operacionais de quedas causadas por ferimentos, onde estudos preliminares consideram a implementação de uma máquina de estados finitos.

O primeiro algoritmo é o foco deste trabalho, uma vez que para iniciar as pesquisas para o segundo algoritmo será necessário a detecção eficiente das quedas e um mapeamento dos estados possíveis.

Para que seja possível alcançar o objetivo geral, este trabalho possui os seguintes objetivos específicos:

- Realizar a coleta das variáveis inerciais de aceleração e giroscópio de militares da Marinha, com o objetivo de coletar dados não apenas de atividades diárias e quedas rotineiramente registradas em conjuntos de dados de detecção de atividades, mas também de atividades militares realizadas com o porte de armamento de assalto (fuzil).

- Extrair e preprocessar os dados coletados para criar conjuntos de dados de validação, teste e treinamento, que serão utilizados para avaliar e comparar os diferentes cenários de treinamento.

- Estudar as componentes da aceleração linear (proveniente do acelerômetro) e da aceleração angular (proveniente do giroscópio) e testar, em cenários de treinamentos distintos, qual combinação de componentes é mais eficiente para a detecção de quedas.

- Identificar a posição ideal para vestir o dispositivo, considerando as opções de peito, pulso esquerdo e pulso direito, a fim de maximizar a eficiência na detecção de quedas.

- Realizar a otimização de hiperparâmetros e implementar o conjunto de componentes mais eficiente no modelo de rede neural escolhido para a detecção de quedas.

I.5 Metodologia

A metodologia empregada neste estudo adota uma abordagem mista, combinando pesquisa descritiva e pesquisa exploratória.

A pesquisa descritiva visa detalhar e caracterizar fenômenos específicos, além de identificar e analisar as relações entre variáveis [Mascarenhas, 2018]. Neste contexto, apresentaremos o conjunto de dados que denominamos de *IPqM-Fall*. Este conjunto faz parte da pesquisa descritiva, pois fornece uma visão detalhada das atividades realizadas por militares, englobando tanto atividades diárias quanto operações militares que envolvem o uso de armamentos. O *IPqM-Fall* é projetado para emular os dados que serão obtidos futuramente por dispositivos vestíveis em cenários de operações militares reais e será utilizado para treinar algoritmos de redes neurais.

O *IPqM-Fall* permite a análise de variáveis inerciais, como aceleração linear e angular, capturadas por dispositivos vestíveis. Os dados foram coletados usando *smartwatches* Samsung Galaxy Watch 4 e *smartphones* LG Velvet, a partir de 15 militares com diferentes níveis hierárquicos. O conjunto abrange atividades como caminhar, correr, pular, posição de tiro, rastejar, engajamento e simulações de quedas em várias direções. Os dados estão organizados em diretórios específicos para cada participante, facilitando o desenvolvimento e treinamento de redes neurais e modelos de aprendizado de máquina para aplicações militares e civis.

Por outro lado, a pesquisa exploratória visa investigar temas pouco conhecidos ou pouco explorados, criando familiaridade com o problema e possibilitando a formulação de hipóteses [Mascarenhas, 2018]. Neste estudo, a pesquisa exploratória será empregada para avaliar se as quedas frequentemente analisadas pela comunidade científica podem ser confundidas com atividades militares no contexto do treinamento de algoritmos de aprendizado de máquina. Além disso, será utilizada para explorar diferentes cenários de treinamento com variados componentes de aceleração, permitindo identificar quais características melhor descrevem a atividade de queda.

I.6 Organização dos capítulos

Esta dissertação está dividida em seis capítulos, iniciando com esta introdução. O Capítulo II aborda a fundamentação teórica, onde serão discutidos todos os assuntos necessários para o melhor entendimento da pesquisa proposta. O Capítulo III aborda os trabalhos relacionados. O Capítulo IV é o cerne da pesquisa, apresentando a metodologia. No Capítulo V são apresentados os resultados e a discussão acerca das técnicas implementadas. Por fim, no Capítulo VI são realizadas as considerações finais e mencionadas as contribuições para trabalhos futuros.

Capítulo II Fundamentação Teórica

Este capítulo traz a fundamentação teórica, onde será explanado os principais assuntos técnicos que possibilitam a compreensão da pesquisa proposta na presente dissertação. Tais temas estão particionados em quatro seções, onde a Seção II.1 aborda o assunto Redes Neurais Artificiais. Na Seção II.2 é apresentado os conceitos acerca da Otimização Bayesiana de Hiperparâmetros, assim como a Seção II.3 apresenta a Transformada de Fourier. Por fim, na Seção II.4 é introduzido o conceito das Métricas para a avaliação de um modelo de aprendizado de máquina.

II.1 Redes Neurais Artificiais

O uso de Redes Neurais Artificiais (RNAs) por parte da comunidade científica aumentou de forma exponencial ao longo das últimas décadas. Embora tenha alcançado, para muitos, o estado de novidade, os conceitos que estruturaram as RNAs foram propostos entre as décadas de quarenta e setenta. Contudo, a ampla utilização citada só foi possibilitada pelos avanços tecnológicos na área da computação que ocorreram a partir de 1999, permitindo a aplicação prática dos conceitos teorizados nos trabalhos basilares, como:

1. McCulloch and Pitts [1943] que argumentaram que o funcionamento do sistema nervoso pode ser descrito usando operações e princípios lógicos. Eles propuseram um cálculo formal baseado em lógica simbólica para representar o comportamento dos neurônios e das redes neurais. Esse cálculo permite descrever a atividade neural em termos de regras e funções lógicas.

2. Rosenblatt [1958] que introduziu o conceito de perceptron, um modelo simplificado de um neurônio biológico que recebe sinais de entrada, pondera-os e produz uma saída com base em uma função de ativação. O perceptron é apresentado como uma ferramenta para reconhecimento e classificação de padrões, onde seus pesos podem ser calibrados por meio de um processo de ajuste iterativo com base na comparação entre a saída do perceptron e o resultado desejado. Tal processo foi denominado por F. Rosenblatt de "aprendizado". O perceptron limita-se à resolução de problemas linearmente separáveis.

3. Werbos [1974] apresentou o conceito do algoritmo de retropropagação de erros (*backpropagation*), que permite o treinamento eficiente de redes neurais de múltiplas camadas. O algoritmo

utiliza a noção de gradientes de erro para ajustar os pesos das conexões entre os neurônios. A retropropagação de erros permitiu a representação de relações não lineares complexas entre as variáveis de entrada e saída.

As ideias propostas pelos trabalhos citados tiveram um impacto significativo no campo do Aprendizado de Máquina e abriram caminho para o desenvolvimento das redes neurais artificiais e, consequentemente, da inteligência artificial. Uma Redes Neural Artificial (RNA) é um modelo de aprendizado de máquina inspirado no funcionamento do cérebro humano. Ela é calibrada por meio de conjuntos de dados, que representam o problema em consideração. O algoritmo de aprendizagem da rede neural generaliza esses dados, permitindo que a rede aprenda a reconhecer padrões e realizar tarefas específicas. O conhecimento adquirido é armazenado nos parâmetros adaptáveis da rede, conhecidos como pesos. Esses pesos são ajustados iterativamente durante o treinamento, de modo que a rede se torne mais habilidosa na resolução do problema em questão [Rauber, 2005].

Em sua arquitetura, a RNA é composta por camadas de neurônios artificiais interconectados. O neurônio artificial é uma estrutura lógico-matemática que emula os padrões de comportamento de um neurônio biológico. O neurônio é responsável em executar uma combinação linear das entradas ponderadas pelos pesos como visto na Figura II.1. Mais especificamente, as entradas x_j são ponderadas pelos seus pesos associados w_{kj} e conectadas ao neurônio que realiza a soma ponderada das entradas adicionando o parâmetro de bias que por fim as compara como um valor de limiar; Se a soma ponderada ultrapassar ou igualar o limiar, o neurônio é ativado [Furtado, 2019].

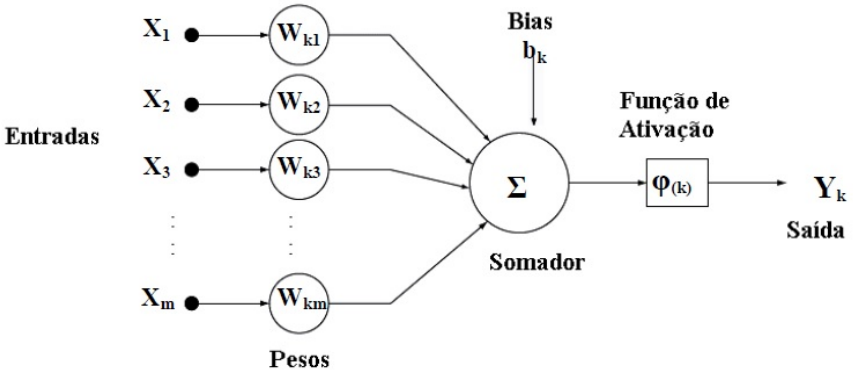


Figura II.1: Neurônio Artificial
Fonte: Aráujo et al. [2012]

O campo das Redes Neurais Artificiais RNAs é vasto e abrange uma ampla variedade de modelos e arquiteturas, tais como Redes Neurais Recorrentes, Redes Neurais Convolucionais, Redes Neurais Generativas Adversárias, Redes Perceptron de Múltiplas Camadas, entre outros. Neste contexto, daremos enfoque nesta dissertação na rede Rede Neurais Convolucionais (CNN), conforme o exposto na seção IV.3.

II.1.1 Redes Neurais Convolucionais

As redes convolucionais são um tipo especializado de arquitetura de redes neurais profundas projetadas para processamento de dados com estrutura de grade, como imagens. Foram introduzidas por LeCun et al. [1998], onde apresentaram a arquitetura LeNet–5 que serviu de protótipo para o avanço das arquiteturas das CNN e possibilitou avanços nos campos de classificação de imagens e visão computacional, como em: Reconhecimento Facial, veículos autônomos, tratamentos médicos inteligentes, supermercados self-service, entre outros. As CNN foram desenvolvidas para aproveitar as propriedades locais e espaciais dos dados de entrada e são capazes de extrair automaticamente características dos dados por meio de uma operação matemática denominada de convolução, o que elimina a necessidade de extração manual [Li et al., 2021].

Pode-se contextualizar Redes Convolucionais como redes neurais que usam a operação de convolução no lugar da multiplicação de matrizes em pelo menos uma de suas camadas [Goodfellow et al., 2016]. A convolução é um procedimento em que os neurônios nas camadas convolucionais aplicam filtros de convolução à imagem de entrada ou às saídas da camada anterior. Esses resultados são considerados como características aprendidas e armazenadas em um mapa de características na saída da camada. A convolução discreta é definida como um operador linear que combina duas funções de entrada para produzir uma terceira função. Essa combinação ocorre ao longo de uma região determinada pela sobreposição das funções de entrada e pelo deslocamento existente entre elas. Nas camadas convolucionais, os filtros de convolução são aplicados a uma pequena região dos dados de entrada para compor uma característica específica do componente, como, por exemplo, a cor em uma imagem RGB. Em seguida, esse processo é seguido por uma função de ativação que introduz um comportamento não linear à operação. Os mapas de características resultantes representam as características extraídas dos dados iniciais e se tornam entradas para a próxima camada. Em Redes Convolucionais profundas, os neurônios tendem a extrair características de mais alto nível que convergem para a classificação final. A Figura II.2 demonstra a operação de convolução matematicamente [Lopez and Mattos, 2021].

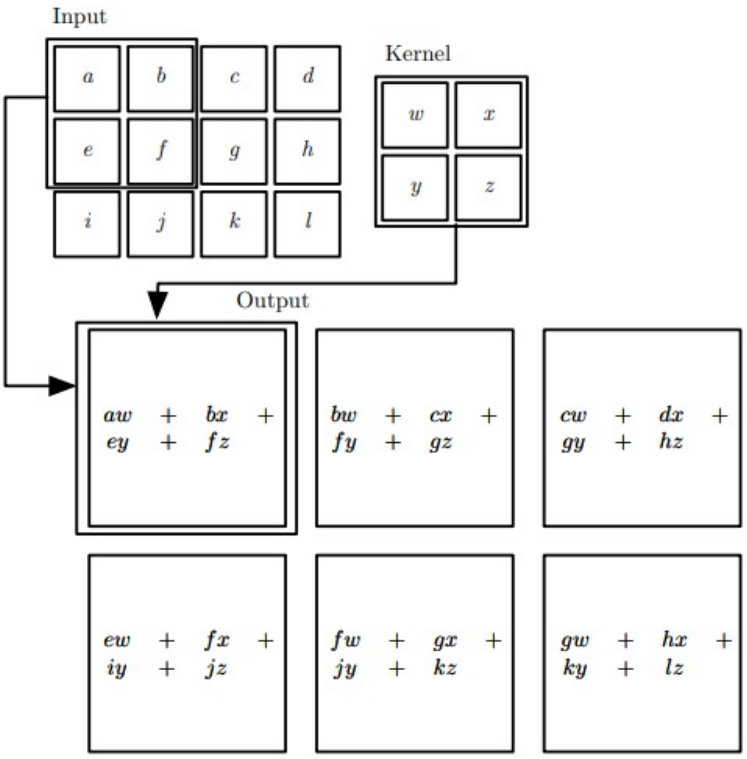


Figura II.2: Operação de Convolução
Fonte: [Goodfellow et al., 2016]

Na Equação II.1, a operação de convolução é representada por um asterisco. O $s(t)$ representa o mapa de características resultante da convolução entre a função de entrada x , que pode ser uma imagem ou um mapa de características de uma camada anterior da CNN e o *kernel* w, também conhecido como filtro, que se trata de um conjunto de pesos que define a operação de convolução. A convolução ocorre ao longo da dimensão temporal ou espacial dos dados de entrada representados por t [Goodfellow et al., 2016]:

$$s(t) = (x * w)(t) \tag{II.1}$$

Nas aplicações de aprendizado de máquina, a entrada geralmente é uma matriz multidimensional de dados, e o *kernel* é geralmente uma matriz multidimensional de parâmetros que são adaptados pelo algoritmo de aprendizado. Frequentemente usamos convoluções em mais de um eixo ao mesmo tempo. Por exemplo, se usarmos uma imagem bidimensional I como nossa entrada, também usaremos um *kernel* bidimensional K, tal operação representada na Equação II.2 [Goodfellow et al., 2016]:

$$S(i,j) = (I * K)(i,j) = \sum_m \sum_n I(m,n)K(i-m, j-n) \tag{II.2}$$

Uma CNN possui em sua arquitetura além da camada de convolução, as camadas de agrupa-

mento e densa (Figura II.3). A camada de agrupamento, também chamada de *pooling* substitui a saída da rede em uma determinada localização por uma estatística resumida das saídas próximas. Por exemplo, a operação de *max pooling* que retorna a saída máxima dentro de uma vizinhança retangular. Outras funções de *pooling* incluem a média de uma vizinhança retangular, a norma L2 de uma vizinhança retangular ou uma média ponderada com base na distância do pixel central. Em todos os casos, o *pooling* reduz a dimensionalidade dos parâmetros dos mapas de características na entrada da próxima camada. [Goodfellow et al., 2016].

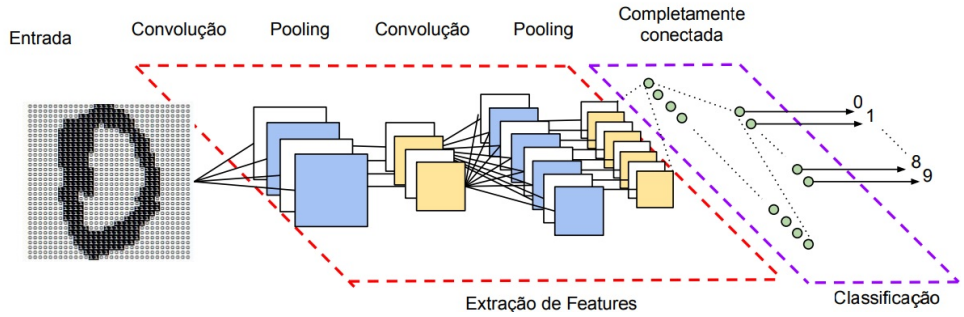


Figura II.3: Rede Neural Convolucional 2D
Fonte: [Vargas et al., 2016]

A saída da última camada, que pode ser convolucional ou de *pooling*, é passada por uma ou mais camadas densas. Essas camadas densas atuam como um classificador, aprendendo a relação entre as características extraídas e as classes ou categorias de interesse. Desta maneira, há dois estágios em uma arquitetura de rede convolucional, o primeiro trata-se do extrator de características que é composto pelas camadas de convolução e *pooling*, e o segundo estágio que refere-se a classificação que é responsabilidade das camadas densas.

Durante o treinamento de uma CNN, os pesos das camadas convolucionais, de *pooling* e densas são ajustados usando algoritmos de otimização. Esse processo é realizado por meio do cálculo do gradiente em relação aos pesos da rede e da atualização iterativa desses pesos com base no gradiente calculado e em uma taxa de aprendizado.

As redes convolucionais tratadas até o momento são arquiteturas poderosas para processamento de imagens e tarefas de visão computacional que operam em 2 dimensões. Contudo, recentemente foram criadas a redes convolucionais unidimensionais que demonstraram um desempenho superior em análise de séries temporais e em aplicações que possuem dados rotulados limitados e alta variação de sinal adquirida de diferentes fontes, como : classificação de dados biomédicos personalizados e diagnóstico precoce, monitoramento de saúde estrutural, detecção e identificação de anomalias em eletrônica de potência entre outros [Kiranyaz et al., 2021].

Assim como nas Rede Neurais Convolucionais Bidimensionais (CNN2D) convencionais, a camada inicial das CNN1D é uma camada que atua no recebimento dos dados de entrada, neste caso o sinal

1D bruto. E a camada de saída é uma camada densa com o número de neurônios equivalente ao número de classes. Tanto as CNN1D (Figura II.4) como as CNN2D compartilham hiperparâmetros e arquiteturas semelhantes divergindo no formato dos dados de entrada, ou seja, os mapas de características, *kernel*, camadas de convolução, de agrupamento e densas também estão presentes nas redes convolucionais unidimensionais [Kiranyaz et al., 2021].

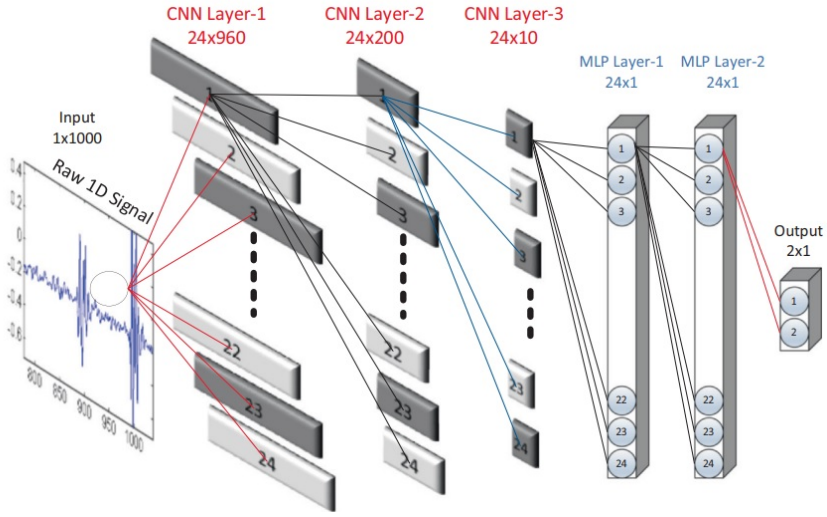


Figura II.4: Rede Neural Convolucional 1D
Fonte: Kiranyaz et al. [2021]

Em comparação, a complexidade computacional das convoluções 1D e 2D difere significativamente. A convolução 1D tem uma complexidade de aproximadamente O(NK), enquanto a convolução 2D tem uma complexidade de aproximadamente $O(N^2K^2)$. Portanto, as CNN1D tem uma complexidade computacional muito menor do que as CNN2D. Nas aplicações recentes, a maioria das CNN1D possui configurações compactas com menos de 10 mil parâmetros, enquanto as CNN2D geralmente possuem mais de 1 milhão de parâmetros. Dessa maneira as CNN1D são arquiteturas menos complexas e mais fáceis de treinar e implementar [Kiranyaz et al., 2021].

Uma das vantagens da utilização de redes convolucionais é a capacidade de aprender representações hierárquicas dos dados. À medida que as informações passam pelas camadas convolucionais e de *pooling*, as características mais básicas dos dados de entrada são combinadas para formar características mais complexas, permitindo que as redes convolucionais capturem padrões em diferentes níveis de abstração.

II.2 Otimização Bayesiana de Hiperparâmetros

Uma rede neural apresenta uma gama de hiperparâmetros em sua arquitetura, os quais quando são imputados de maneira incorreta podem ocasionar um desempenho baixo da classificação ou previsão. É possível até mesmo que retorne um bom desempenho, contudo, com uma rede super

parametrizada que demanda alto custo computacional. Por tal motivo, o uso de técnicas de otimização de hiperparâmetros se faz necessário para que se alcance a arquitetura mais enxuta que retorna o melhor resultado para o problema em questão. Na pesquisa proposta faz-se uso da otimização Bayesiana para calibrar os hiperparâmetros das arquiteturas das redes neurais avaliadas.

A otimização Bayesiana assume que a função desconhecida foi amostrada de um processo gaussiano e mantém uma distribuição posterior para essa função à medida que os resultados das experiências com diferentes hiperparâmetros do algoritmo de aprendizado são observados. Para escolher os hiperparâmetros da próxima iteração faz-se uso de uma função de aquisição, que pode otimizar a melhoria esperada em relação ao melhor resultado atual ou o limite de confiança superior do processo gaussiano [Snoek et al., 2012].

O objetivo da otimização Bayesiana é encontrar o mínimo de uma função $f(x)$ em um conjunto limitado X, que é considerado um subconjunto de $\mathbb{R}^D$. Um modelo probabilístico para $f(x)$, denominado de modelo substituto é construído, e posteriormente explorado visando a tomada de decisões sobre onde avaliar a função em X, enquanto integra a incerteza, utilizando todas as informações disponíveis de avaliações anteriores de $f(x)$ e não apenas se valendo de aproximações locais de gradientes e hessianas. Isso resulta em um procedimento que pode encontrar o mínimo de funções não-convexas difíceis com relativamente poucas avaliações, ao custo de realizar mais cálculos para determinar o próximo ponto a ser testado [Snoek et al., 2012].

A função de aquisição mencionada é usada para selecionar de forma inteligente os pontos no espaço de hiperparâmetros que têm maior probabilidade de levar a uma melhora significativa no desempenho do modelo. Existem diferentes estratégias de aquisição que podem ser usadas, sendo duas entre elas: a aquisição por Probabilidade de Melhoria e aquisição por melhora esperada. Na aquisição por Probabilidade de Melhoria busca-se maximizar a probabilidade de melhoria em relação ao melhor valor atual.

Já na aquisição por melhora esperada, leva-se em consideração tanto a incerteza estimada quanto o valor esperado das métricas de desempenho. Essa abordagem busca maximizar a diferença entre o desempenho atual do modelo e o melhor desempenho esperado com base nas previsões do modelo substituto.

II.3 Transformada de Fourier

A Transformada de Fourier é uma operação matemática utilizada para decompor uma função em suas componentes de frequência. Ela permite a análise de sinais e sistemas no domínio da frequência, em contraposição ao domínio do tempo. A transformada converte uma função do domínio do tempo em uma representação no domínio da frequência, revelando as diferentes frequências presentes no sinal.

A Transformada de Fourier é definida pela Equação II.3 [Conolly and Good, 1977]. Nessa equação, $f(t)$ é a função de tempo contínua, $F(\omega)$ é a função de frequência contínua e i é a unidade imaginária. A integral representa a contribuição de cada frequência ω para a função no domínio do tempo.

$$F(\omega) = \int_{-\infty}^{\infty} f(t)e^{-i\omega t}dt \tag{II.3}$$

No campo de aprendizado de máquina, a Transformada de Fourier possui diversas aplicações, principalmente para pré-processamento dos dados. Por exemplo, Chen and Lee [2020] utilizam a transformada de Fourier para converter os sinais de vibração em imagens no domínio da frequência, e posteriormente aplicam uma Rede Neural Convolucional para classificar os sinais. Já Qi et al. [2021] utilizam a transformada de Fourier em conjunto com uma rede convolucional para diagnosticar tecidos pulmonares utilizando espectros Raman, que são são gráficos que representam a intensidade da luz espalhada por uma amostra, em função da energia ou comprimento de onda da luz espalhada.

II.4 Métricas para Avaliação de Classificadores

A avaliação de um modelo de aprendizado de máquina para classificação é uma etapa fundamental para medir sua eficácia e identificar o quão bem ele está generalizando os padrões dos dados de treinamento para novos dados. Existem diversas métricas que podem ser utilizadas para esse fim, e a escolha das métricas corretas depende do tipo de problema (classificação ou regressão), do tipo de conjunto de dados (balanceado ou desbalanceado) e dos objetivos específicos do projeto. Abaixo são citadas, da mais importante para a menos importante, as métricas que são mencionadas nesta dissertação:

1. O Coeficiente de Correlação de Matthews (MCC) (Equação II.4): É uma medida estatística utilizada para avaliar a qualidade de classificadores binários. Ele varia entre -1 e +1, sendo que +1 indica uma previsão perfeita, 0 indica uma previsão aleatória e -1 indica uma previsão totalmente incorreta. O MCC leva em consideração os quatro elementos da matriz de confusão de um classificador binário: Verdadeiros Positivos (TP), Falso Positivos (FP), Verdadeiros Negativos (TN) e Falso Negativos (FN). O MCC é considerado uma métrica útil para avaliar classificadores em problemas desbalanceados, pois, mede a correlação entre as classificações binárias observadas e previstas [Junior et al., 2022].

$$\text{MCC} = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP) \times (TP + FN) \times (TN + FP) \times (TN + FN)}} \tag{II.4}$$

2. Sensibilidade (SE): Também conhecida como *recall* ou Taxa de Verdadeiro Positivo, é uma

métrica que avalia o desempenho de um modelo, especialmente em tarefas de classificação. Ela indica a proporção de instâncias positivas corretamente identificadas pelo modelo, ou seja, quantos dos casos realmente positivos foram classificados corretamente, expressa pela Equação II.5 [Prati et al., 2008].

$$\text{Sensibilidade} = \frac{TP}{TP + FN} \tag{II.5}$$

3. Precisão (PR) (Equação II.6): mede a proporção de verdadeiros positivos em relação ao total de padrões previstos como positivos. Em outras palavras, a precisão indica quantos dos exemplos classificados como positivos são realmente positivos [Hossin and Sulaiman, 2015]

$$\text{Precisão} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{II.6}$$

4. Especificidade (ES) (Equação II.7): é uma métrica usada para medir a fração de padrões negativos que são corretamente classificados. Ela avalia a capacidade de um modelo de identificar corretamente as instâncias negativas, ou seja, quantos dos exemplos verdadeiramente negativos são reconhecidos como tal [Hossin and Sulaiman, 2015].

$$\text{Especificidade} = \frac{\text{TN}}{\text{TN} + \text{FP}} \tag{II.7}$$

Capítulo III Trabalhos Relacionados

III.1 Introdução

Neste capítulo, realizamos uma revisão abrangente da literatura que aborda a aplicação de redes neurais artificiais na detecção de quedas por meio de dispositivos inteligentes, como *smartwatches* e *smartphones*. Tais temas estão particionados em três seções, onde a Seção III.1 resume a forma com que a pesquisa dos trabalhos foi realizada e a quantidade de trabalhos encontrados, assim como o critério de exclusão dos mesmos. Na Seção III.2 é apresentado uma análise estatística dos trabalhos. Por fim, na Seção III.3 é relacionado os trabalhos mais relevantes para a pesquisa. A ampliação do escopo desta revisão se deve à escassez de artigos relacionados à classificação de quedas em contextos militares específicos. Para realizar essa revisão, uma busca na base de dados Scopus foi realizada em 31 de Agosto de 2023, utilizando os seguintes termos de pesquisa: *"Fall Detection" and "Neural Networks" and "Smartwatch" or "Smartphone"*. A busca abrangeu o período de 2015 a 2023, englobando, assim, os últimos oito anos de desenvolvimento na área e visando abordar as metodologias mais atuais.

Essa pesquisa resultou na identificação de 43 artigos relevantes (Tabela III.2), que inicialmente foram submetidos a critérios de exclusão (Tabela III.1) para refinar ainda mais a seleção. Após a aplicação desses critérios, 32 artigos foram considerados como parte desta revisão.

Tabela III.1: Critérios de exclusão

Critério	Descrição	Qtd. de Artigos Excluídos
C1	Trabalho não se aplica a detecção de queda.	2
C2	Trabalho utiliza outro dispositivo que não seja *Smartphone* ou *Smartwatch*.	7
C3	Impossibilidade de acesso ao texto do trabalho.	2

Embora a maioria desses artigos se concentre na detecção de quedas em idosos, as metodologias e abordagens utilizadas podem ser relacionadas diretamente à pesquisa em questão nesta dissertação. A escolha de limitar a busca a *smartphones* e *smartwatches* se justifica pela sua estreita afinidade com o escopo da pesquisa vigente. Esses dispositivos inteligentes incorporam sensores avançados, como acelerômetros e giroscópios, que podem ser utilizados para monitorar movimentos e detectar quedas. Portanto, mesmo que a maioria dos estudos se concentre na detecção de quedas em idosos,

as técnicas de aprendizado de máquina, particularmente as redes neurais artificiais, aplicadas nesses contextos podem ser adaptadas e aprimoradas para atender às necessidades de pesquisa no contexto militar.

III.2 Análise dos Trabalhos Relacionados

Nesta seção, uma análise abrangente dos artigos relacionados é conduzida, abordando a quantidade de publicações anuais, as palavras-chave mais frequentes, os dispositivos e sensores mais utilizados, bem como as arquiteturas de redes neurais predominantes.

- **Quantidade de publicações anuais:** O gráfico na Figura III.1 revela que a quantidade de artigos publicados por ano sobre detecção de quedas com o uso de *smartphone* e *smartwatch* aumentou consideravelmente ao longo dos anos. Possuindo apenas um artigo em 2015, o número cresceu para sete em 2020, e permaneceu em um patamar elevado nos anos subsequentes. Isso sugere um crescente interesse e foco da comunidade de pesquisa no desenvolvimento de tecnologias e métodos de detecção de quedas. Embora a tendência geral seja de crescimento, também podemos observar flutuações anuais. Por exemplo, houve um declínio em 2017, quando o número de artigos caiu para zero. Essa queda pode indicar uma diminuição temporária no interesse acerca do tema, mas o fato de que a pesquisa retomou e cresceu nos anos subsequentes sugere que a área permanece relevante.

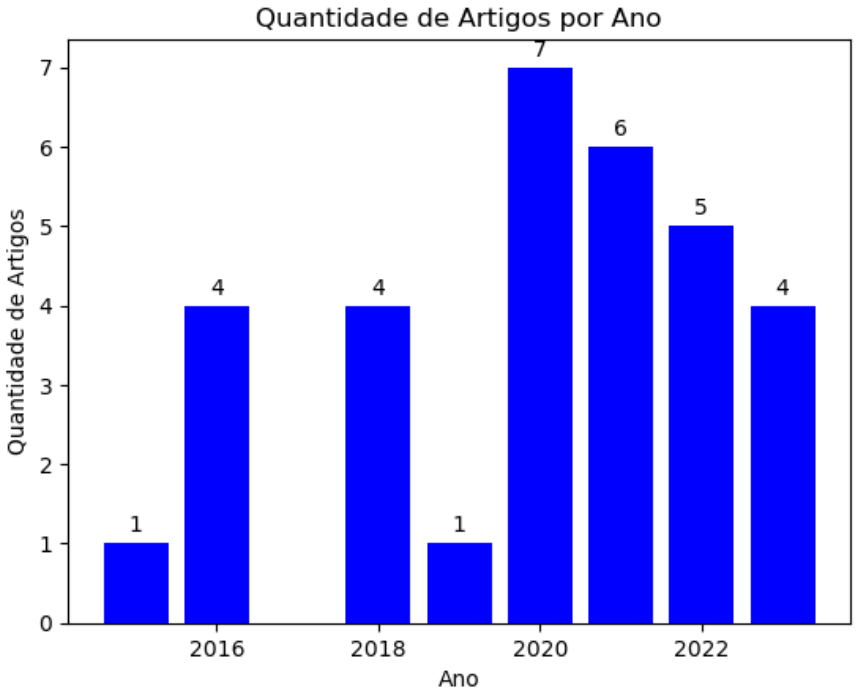


Figura III.1: Artigos por ano

- **Palavras-chave mais frequentes:** Palavras-chave em artigos científicos são termos ou frases

que resumem os principais tópicos, temas e conceitos abordados no artigo. Elas fornecem um contexto valioso para os revisores e editores durante o processo de avaliação de artigos, ajudando-os a determinar a relevância do trabalho em relação ao campo de estudo.

- **Dispositivos e sensores mais utilizados:**

  Verifica-se, por meio da Figura III.2, que o *smartphone* foi o dispositivo mais amplamente utilizado nos sistemas desenvolvidos nos estudos relacionados, contando com mais de 20 artigos dedicados exclusivamente ao seu uso. Adicionalmente, foi identificado que em 3 artigos houve uma abordagem conjunta, incorporando tanto o *smartphone* quanto o *smartwatch* como componentes essenciais. Esses dados destacam a predominância do *smartphone* como plataforma principal neste campo de pesquisa, ao mesmo tempo em que sinalizam a necessidade de uma maior exploração das potencialidades dos *smartwatches*, sugerindo que ainda há muito a se aprofundar na investigação envolvendo esse dispositivo.

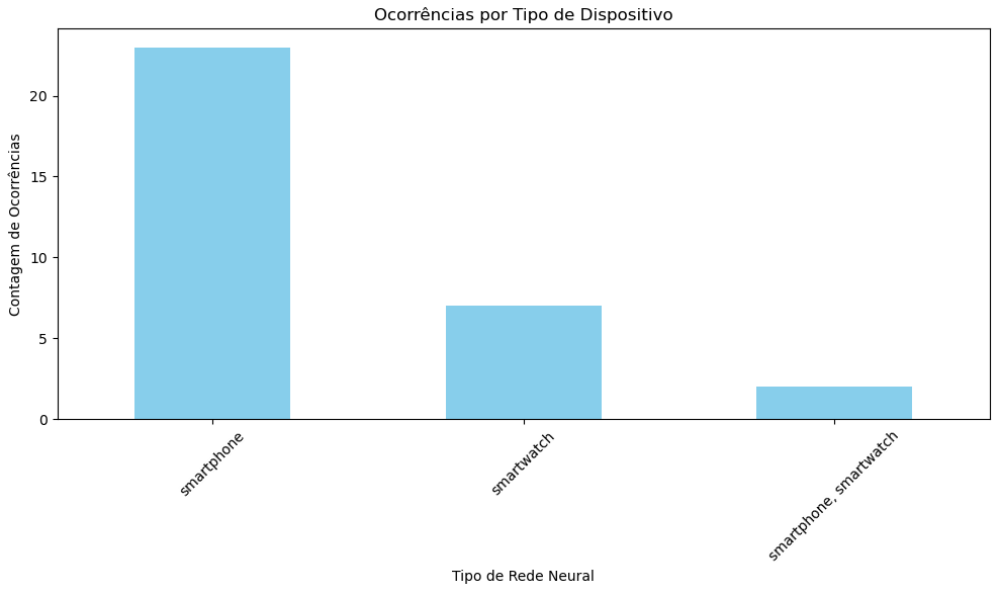


Figura III.2: Ocorrências por Tipo de Dispositivo

Quanto aos sensores empregados nos artigos relacionados, a Figura III.3 demonstra que os dados provenientes do acelerômetro estão entre os mais frequentemente utilizados como entrada para os algoritmos de Aprendizado de Máquina. Em muitos casos, eles são combinados com dados de giroscópios e, ocasionalmente, integrados com informações de outros sensores.

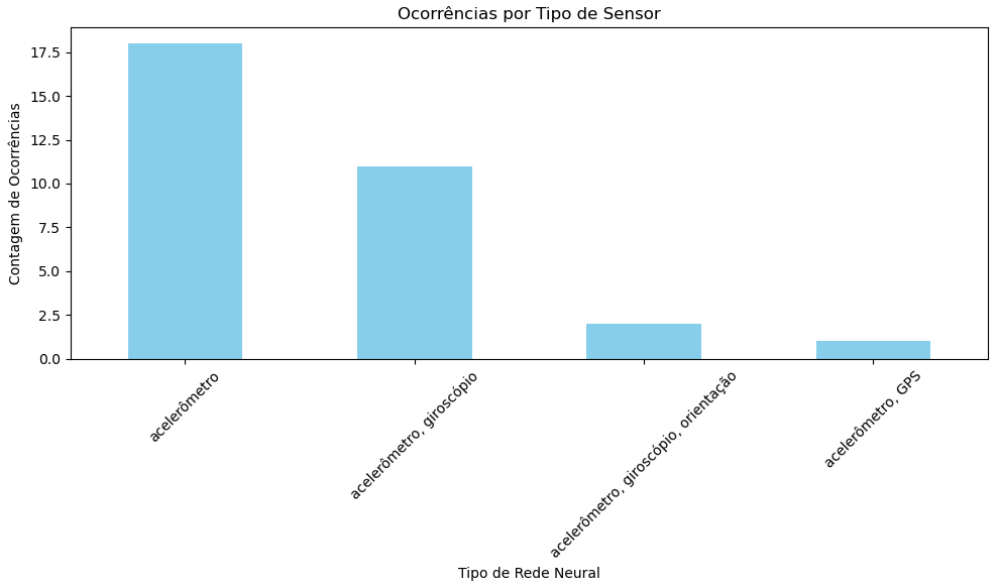


Figura III.3: Ocorrências por Tipo de Sensor

- **Arquiteturas de redes neurais predominantes:**

  Para facilitar a análise da Figura III.4, todas as técnicas de aprendizado de máquina que não se enquadram como redes neurais foram agrupadas como "Algoritmos de ML". Ao observar a figura, notamos que as redes neurais utilizadas foram: Rede Neural Perceptron de Multicamadas (MLP), CNN e Rede Neural Recorrente (RNN). Frequentemente empregadas individualmente e, ocasionalmente, em combinação umas com as outras.

  Entre as redes neurais mais utilizadas, a MLP é reconhecida pela sua capacidade de modelar relações complexas nos dados, tornando-a uma escolha comum em muitos estudos. A CNN possui duas variantes: a CNN2D, que se destaca pela sua eficácia na análise de dados de imagem, encontrando aplicação na detecção de quedas por meio de informações visuais; e a CNN1D, frequentemente utilizada quando os dados possuem uma estrutura unidimensional, como séries temporais ou sequências. Por outro lado, a RNN é frequentemente escolhida para capturar dependências sequenciais nos dados, o que é crucial em cenários de detecção de quedas que envolvem séries temporais.

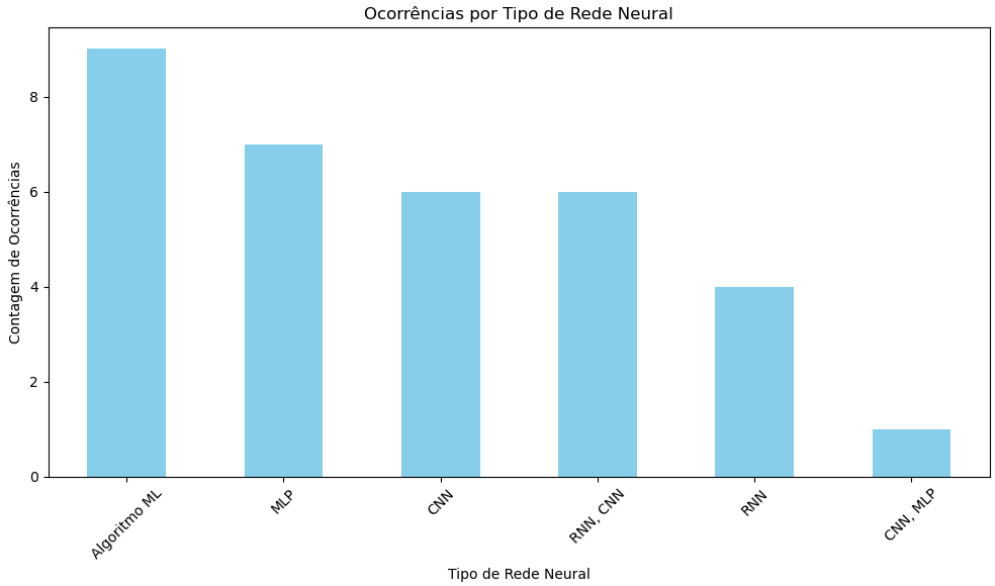


Figura III.4: Ocorrências por Tipo de Rede Neural

III.3 Trabalhos Relevantes

A Tabela III.2 apresenta os 32 artigos selecionados para esta análise bibliográfica. Dentre eles, destacam-se, pela maior quantidade de citações e relevância, os 12 trabalhos a seguir.

1. **Mauldin et al. [2018]** apresentam o sistema *SmartFall*, um aplicativo *Android* que utiliza dados de acelerômetro coletados por *smartwatches* para detectar quedas em tempo real. O sistema avalia métodos tradicionais de aprendizado de máquina (*Support Vector Machine* e *Naive Bayes*) e modelos baseados em *Deep Learning*, demonstrando que o modelo de *Deep Learning* supera os métodos tradicionais devido à sua capacidade de extrair características diretamente dos dados brutos do acelerômetro. Os experimentos alcançaram resultados de acurácia de 99% em dados offline e sensibilidade de 86% em testes em tempo real.

   O *SmartFall* foi testado em três conjuntos de dados: (1) *Smartwatch*, coletado com sete voluntários simulando atividades diárias e quedas (correr, sentar, quedas frontais, laterais, entre outras); (2) *Notch*, um sensor de pulso que também incluiu atividades similares; e (3) *Farseeing*, uma base de dados real com quedas de idosos. O sistema utilizou o *Microsoft Band* 2 como dispositivo principal para coleta de dados, com frequência de amostragem ajustada a 31.25 Hz. A arquitetura *IoT* do *SmartFall* inclui camadas que permitem a integração de outros sensores e armazenamento de dados local para preservar a privacidade.

2. **Khojasteh et al. [2018]** propuseram um método de detecção de quedas utilizando um sensor de acelerômetro 3D (3DACC) em um dispositivo vestível de pulso. O objetivo é aprimorar os sistemas de detecção de quedas focados em idosos, otimizando a ergonomia e reduzindo

os falsos positivos. O sistema utiliza um modelo baseado em redes neurais *feedforward* e explora alternativas, como *Support Vector Machine*, árvores de decisão e sistemas baseados em regras. Para validar os métodos, foram utilizados diversos conjuntos de dados públicos, incluindo o *UMAFall* (dados simulados), *DaLiac* (atividades diárias), *Epilepsy* (movimentos de alta atividade) e o *Farseeing* (quedas reais de idosos com sensores localizados nas costas e coxas).

Os experimentos envolveram voluntários jovens e idosos, sendo as quedas simuladas comparadas às reais, indicando diferenças significativas entre os dois cenários. Por exemplo, o modelo neural apresentou melhores resultados em quedas simuladas com precisão de 94%, enquanto para quedas reais a precisão foi inferior devido à maior variabilidade nos dados. O estudo utilizou otimização de limiares com algoritmos genéticos, ajustando os valores de detecção de picos para diferentes contextos, com resultados estáveis e limiares otimizados entre 2.5 e 3.09g.

Os dispositivos utilizados incluíram sensores comerciais de pulso para coleta dos dados e algoritmos implementados em ambientes móveis, destacando-se pela eficiência energética e pela baixa complexidade computacional dos modelos baseados em regras.

3. **Şengül et al. [2022]** elaboraram um sistema de detecção de quedas baseado em *smartwatches*, utilizando sensores de acelerômetro e giroscópio. O modelo explora redes neurais de memória de curto e longo prazo bidirecionais para classificação de atividades, incluindo quedas, caminhadas, corridas, agachamentos e sentar-se. Dados de 15 voluntários foram coletados (idade média de 22.3 anos), com um total de 824 amostras, em um intervalo de amostragem de 50 Hz. As atividades simuladas incluem quedas de pé e de uma cadeira. Para lidar com a escassez de dados, foi empregada uma técnica de aumento de dados baseada em interpolação cúbica *Hermite*.

   Os experimentos indicaram que a combinação de dados de acelerômetro e giroscópio resultou em uma precisão de 99.59% com validação cruzada *leave-one-activity-out* e 97.35% com validação *leave-one-subject-out*. Em um cenário de classificação binária (quedas vs. outras atividades), o modelo alcançou precisão perfeita quando ambos os sensores foram utilizados. O sistema foi implementado em um *Sony SmartWatch 3 SWR50*, com os cálculos sendo realizados em um servidor em nuvem para reduzir o consumo de energia.

4. **Qian [2019]** idealizaram um sistema distribuído baseado em uma arquitetura de aprendizado profundo hierárquica (DHNN) para detecção de quedas. O sistema combina dispositivos vestíveis, incluindo *smartwatches*, *smartphones* e palmilhas inteligentes, para coletar e processar dados localmente e em servidores na nuvem. A abordagem protege a privacidade dos dados

ao evitar o envio de dados sensíveis para a nuvem, mantendo os cálculos principais nos dispositivos locais e delegando as camadas mais profundas do modelo para o servidor.

Os experimentos envolveram 10 voluntários, que realizaram 10 atividades diárias e 5 tipos de quedas. Os dados foram coletados a uma frequência de 30 Hz, utilizando acelerômetros e giroscópios dos dispositivos, além de sensores de pressão das palmilhas. Com 3.380 amostras no total, o modelo combinou dados de diferentes dispositivos, alcançando 99.7% de acurácia, 99.86% de especificidade e 98.77% de sensibilidade ao integrar os três sensores. A configuração hierárquica foi validada contra abordagens centralizadas e demonstrou resultados comparáveis em termos de precisão, com menor sobrecarga computacional nos *smartphones*.

A arquitetura mostrou-se altamente eficiente, tanto em convergência quanto em uso de memória e processamento. Ao adotar diferentes protocolos de treinamento (*Round Robin* e Assíncrono), foi possível acomodar dispositivos com recursos limitados, mantendo a robustez do modelo. A abordagem hierárquica também melhorou a generalização, ao incorporar dados de múltiplos dispositivos.

5. **Kerdegari et al. [2015]** apresentam o *Smart Fall Detection*, um sistema de detecção de quedas baseado em um aplicativo *Android*, que utiliza redes MLP para classificar padrões de quedas e atividades diárias, a partir de dados de acelerômetro. Quando uma queda é detectada, o sistema envia um pedido de ajuda via SMS e, se disponível, a localização do usuário utilizando GPS. A aplicação é independente e projetada para ser portátil e acessível para idosos.

   O *Smart Fall Detection* foi avaliado utilizando dados coletados de 50 voluntários (18 homens e 32 mulheres) com idade média de 32 anos, altura média de 165 cm e peso médio de 66 kg. Os participantes realizaram 9 atividades diárias e 11 tipos de quedas, totalizando 1.000 amostras registradas por um acelerômetro triaxial montado na cintura. Para pré-processar os dados, foi utilizada uma técnica de janelamento deslizante, com janelas de 1 segundo e sobreposição de 0.5 segundo, garantindo que os sinais fossem adequadamente segmentados.

   Os experimentos mostraram que o sistema alcançou uma sensibilidade de 93.18%, especificidade de 88.88% e acurácia de 91.25% em um total de 80 testes realizados por 4 voluntários. A implementação utilizou *smartphones Android* equipados com acelerômetros, incluindo modelos como *HTC Incredible S* e *Samsung Galaxy*, com resultados consistentes entre dispositivos.

6. **Deutsch and Burgsteiner [2016]** apresentam um sistema de assistência baseado em *smartwatch* chamado *Carrie*, projetado para monitorar e auxiliar idosos em atividades diárias e emergências. O sistema inclui funcionalidades de detecção de quedas, reconhecimento de inatividade incomum e lembretes de medicação, além de permitir alertas manuais em situações de emer-

gência. Para a detecção de quedas, o sistema utiliza dados de aceleração de um *smartwatch Pebble* em conjunto com um *smartphone* conectado via *Bluetooth*.

Os experimentos envolveram 7 participantes saudáveis, com idades entre 14 e 83 anos, que realizaram atividades como caminhar, correr, sentar-se, levantar-se e quedas simuladas em direções frontal, lateral e traseira. Os dados foram coletados a 50 Hz pelo acelerômetro 3D do *smartwatch*, e as quedas foram classificadas por redes MLP com uma camada oculta de 10 neurônios. Três abordagens de extração de características foram testadas, sendo que a terceira, com 8 neurônios de entrada, apresentou os melhores resultados em sensibilidade e precisão.

O sistema envia alertas via SMS a contatos de emergência, incluindo a localização exata obtida pelo GPS, e oferece lembretes de medicação no *smartwatch*. A eficiência energética foi melhorada ao limitar a transmissão de dados via *Bluetooth* apenas para eventos suspeitos, aumentando a duração da bateria do *smartwatch* de 19 para até 60 horas.

7. **Ghate [2021]** investigam abordagens híbridas de aprendizado profundo para o reconhecimento de atividades humanas utilizando sensores embutidos em *smartphones*, como acelerômetros e giroscópios. Entre os modelos propostos, destaca-se o DeepCNN-RF, que integra redes neurais convolucionais para extração automática de características com *Random Forest* para classificação. Além disso, foram implementadas combinações de CNN com Redes Recorrentes de Memória de Longo e Curto Prazo (LSTM) e Unidade Recorrente com Portas (GRU), explorando a captura de dependências temporais em dados de séries temporais.

   Os experimentos foram conduzidos com dois conjuntos de dados públicos: o *UCI HAR*, que inclui 30 participantes realizando 6 atividades de vida diária, e o *WISDM*, com dados de 36 participantes em 6 atividades. As atividades foram registradas com sensores de *smartphones* em frequências de 20 Hz e 50 Hz, aplicando técnicas de pré-processamento, como filtros de ruído e janelas deslizantes com sobreposição de 50%. O modelo DeepCNN-RF obteve os melhores resultados, alcançando 98,2% de acurácia no *UCI HAR* e 97,77% no *WISDM*.

   Os modelos propostos mostraram-se superiores a abordagens individuais de aprendizado profundo, como CNN ou LSTM isoladas, devido à combinação de extração robusta de características com classificadores eficientes. A técnica de regularização *dropout* e a normalização em lotes foram utilizadas para reduzir o sobreajuste, enquanto o otimizador Adam garantiu a convergência eficiente durante o treinamento.

8. **Rashidpour et al. [2016]** apresentam um sistema de detecção de quedas baseado no Sistema de Inferência Neuro-Fuzzy Adaptativo, utilizando sensores de acelerômetro e giroscópio embutidos em *smartphones*. O método visa diferenciar com precisão quedas de atividades diárias

que possuem características semelhantes, como corridas ou saltos rápidos. A abordagem combina aprendizado de máquina com lógica fuzzy, adaptando-se às características individuais de movimento dos usuários.

Os dados utilizados foram extraídos do conjunto de dados *Mobifall*, que contém registros de 4 tipos de quedas e 9 atividades de vida diária. A coleta foi realizada com um *smartphone Samsung Galaxy S3*, equipado com um acelerômetro e giroscópio triaxiais LSM330DLC, com frequências de amostragem de 87 Hz e 200 Hz, respectivamente. Os testes incluíram 11 voluntários (6 homens e 4 mulheres) com idades entre 22 e 36 anos. O sistema Neuro-Fuzzy Adaptativo alcançou sensibilidade e especificidade de 100%, destacando-se pela robustez em diferenciar quedas de atividades intensas sem causar falsos positivos.

9. **Hou et al. [2018]** exibem um algoritmo de reconhecimento de quedas utilizando *Support Vector Machine* aplicado a sensores de aceleração gravitacional embutidos em *smartphones*. O método busca distinguir quedas de atividades diárias como caminhar, subir e descer escadas e permanecer parado. Para melhorar a qualidade dos dados, foi utilizado um filtro mediano para reduzir ruídos e suavizar os sinais capturados pelos sensores.

   Os dados foram coletados por meio de *smartphones* posicionados na cintura de voluntários, local estratégico por ser o centro de gravidade do corpo humano. As características extraídas incluíram aceleração resultante, ângulo de inclinação, média, variância, curtose e assimetria. Após a extração, foi utilizada a técnica de Análise de Componentes Principais (PCA) para reduzir a dimensionalidade dos dados de 21 para 7 dimensões, garantindo melhor desempenho do classificador.

   Os experimentos foram conduzidos com participantes simulando quedas e atividades diárias. O modelo *Support Vector Machine* alcançou uma acurácia de 96.07%, superando redes neurais generalizadas e probabilísticas em experimentos comparativos. Além disso, o estudo abordou desafios como a detecção de quedas em atividades similares, como sentar-se rapidamente ou deitar-se, que podem levar a falsos positivos.

10. **Vidigal et al. [2015]** exploram o uso de redes neurais artificiais para detecção de quedas em idosos, empregando sensores de aceleração integrados em *smartphones*. O trabalho compara três arquiteturas: Perceptron Multicamadas, Função de Base Radial e Mapas Auto-organizáveis, a fim de determinar a mais eficaz para o problema de detecção de quedas. Além disso, foi criada uma base de dados personalizada, composta por eventos simulados de quedas e atividades diárias, que pode ser reutilizada em futuras pesquisas.

    A coleta de dados envolveu 6 voluntários, incluindo 4 homens e 2 mulheres, realizando 11 movimentos controlados, sendo 7 representativos de atividades diárias e 4 de quedas. Os

smartphones foram posicionados no braço e na cintura dos participantes para representar os locais mais comuns em que esses dispositivos são carregados. No total, foram geradas 1.222 amostras, cada uma com 86 valores de entrada e uma saída que indica queda ou não. O treinamento utilizou o algoritmo de retropropagação, com variações no número de neurônios na camada oculta. Os resultados mostraram que a rede MLP apresentou o melhor desempenho, com uma acurácia de 96.4%, sensibilidade de 85% e especificidade de 98.3%. O sistema foi implementado em *smartphones Android* e inclui módulos de coleta, treinamento, detecção e comunicação, permitindo o envio de alertas com a localização do usuário em casos de quedas detectadas.

11. **Mauldin et al. [2020]** abordam o uso de modelos de aprendizado profundo em conjunto (*ensembles*) para detecção de quedas utilizando dados de sensores de dispositivos vestíveis, como *smartwatches*. Os experimentos focaram na limitação de conjuntos de dados pequenos e avaliaram abordagens baseadas em redes neurais recorrentes RNN, comparando os resultados de modelos individuais com aqueles gerados por *ensembles*.

    Dois conjuntos de dados foram utilizados: (1) o *SmartFall*, coletado de 14 voluntários (idades de 21 a 60 anos) usando o *smartwatch Microsoft Band 2*, com quedas simuladas (frontal, traseira, lateral direita e esquerda) e atividades de diária como caminhar, correr e sentar; e (2) o *UniMiB SHAR*, composto por dados de 30 voluntários (idades de 18 a 60 anos) usando *smartphones*. O primeiro conjunto de dados apresentou 528 quedas e 6.572 segmentos Atividades da Vida Diária (AVD), enquanto o segundo incluiu 710 quedas e 486 AVD.

    Os modelos individuais baseados em RNN alcançaram uma precisão de 58% e sensibilidade de 98% no conjunto de dados *SmartFall*. Já o *ensemble*utilizando uma abordagem de *stacking*, composto por 4 modelos com 30 neurônios, superou os modelos individuais, atingindo 92.2% de precisão e 99% de sensibilidade. No *UniMiB*, a mesma arquitetura obteve 85% de precisão e 99% de sensibilidade, demonstrando maior robustez em relação a outros métodos, especialmente na redução de falsos positivos.

    O sistema foi validado em um experimento real com cinco voluntários, mostrando que o *feedback* do usuário sobre falsos positivos pode melhorar significativamente a precisão do modelo. Além disso, o uso de *ensembles* mitigou problemas associados ao treinamento em conjuntos de dados pequenos, como sobreajustes.

12. **Tsai et al. [2015]** propuseram o sistema *Gesture-Aware Fall Detection (GAFD)*, que utiliza *smartphones* para detecção de quedas baseados em gestos, explorando sensores de aceleração triaxial em diferentes posições do corpo (braço, peito, cintura e coxa). O sistema emprega uma rede neural de retropropagação para analisar gestos históricos e criar um modelo empírico

personalizado, melhorando a precisão da detecção.

Os dados foram coletados de 6 voluntários (4 homens e 2 mulheres), com idades entre 20 e 30 anos, que realizaram 10 categorias de atividades, incluindo quedas frontais, traseiras e laterais, bem como AVD, como caminhar, correr, sentar e levantar-se. O modelo neural foi configurado com uma camada oculta de 135 neurônios, aprendendo a partir de 250 grupos de dados para cada gesto ou queda. A frequência de amostragem dos dados do acelerômetro foi ajustada dinamicamente para otimizar a coleta.

Os experimentos mostraram que a posição do dispositivo influencia significativamente a precisão. A coxa apresentou os melhores resultados, com uma taxa média de precisão de 96.3% para detecção de quedas e 95.65% para reconhecimento de posturas, enquanto o braço teve o pior desempenho devido à maior variação de movimento. A Sensibilidade e a especificidade também variaram conforme a posição, sendo 99.15% e 95.91%, respectivamente, para dispositivos na coxa.

Os artigos citados apresentam soluções que se relacionam diretamente com os objetivos propostos nesta dissertação. Cada estudo contribui com abordagens específicas que, juntas, formam um panorama abrangente sobre o tema em questão. As soluções discutidas nos artigos refletem avanços significativos em áreas relacionadas, como técnicas de modelagem, métodos de análise e estratégias de implementação, proporcionando um ponto de partida sólido para a condução da pesquisa apresentada.

A relevância desses artigos reside em como suas soluções se alinham às lacunas identificadas no campo de estudo. Muitos deles abordam problemas semelhantes, sugerindo metodologias que podem ser adaptadas ou combinadas para atender aos objetivos desta dissertação. Por exemplo, enquanto alguns trabalhos enfatizam a precisão das técnicas empregadas, outros destacam a eficiência computacional ou a aplicabilidade em cenários reais. Essa diversidade permite uma análise comparativa e crítica, essencial para fundamentar as escolhas metodológicas realizadas.

Portanto, os artigos relevantes formam a base teórica e prática para o desenvolvimento desta pesquisa, estabelecendo um diálogo constante entre o que já foi produzido e o que se propõe como avanço científico. Apesar dos avanços encontrados na literatura, identificou-se uma lacuna significativa relacionada à falta de conjuntos de dados voltados para cenários militares, que envolvam condições específicas como o uso de equipamentos e armas de fogo. Esse vazio evidenciado reforça a importância e o ineditismo desta pesquisa, que busca preencher tal lacuna ao propor um método robusto e específico para a detecção de quedas em contextos de operações militares. Essa interação não apenas garante a relevância e a originalidade do trabalho, mas também consolida sua fundamentação acadêmica, contribuindo para aplicações práticas de grande impacto.

Tabela III.2: Artigos selecionados

Artigo	**Citações**	**Dispositivo**	**Sensor**	**Rede Neural**
Smartfall: A smartwatch-based fall detection system using deep learning	152	smartwatch	acelerômetro	SVM, RNN
Improving fall detection using an on-wrist wearable accelerometer	96	smartwatch	acelerômetro	DT, SVM
Deep learning based fall detection using smartwatches for healthcare applications	21	smartwatch	acelerômetro, giroscópio	KNN, DT, SVM
Wearable Computing with Distributed Deep Learning Hierarchy: A Study of Fall Detection	19	smartphone, smartwatch	acelerômetro, giroscópio	CNN, MLP
A pervasive neural network based fall detection system on smart phone	19	smartphone	acelerômetro, GPS	MLP
A smartwatch-based assistance system for the elderly performing fall detection, unusual inactivity recognition and medication reminding	16	smartphone	acelerômetro	MLP
Hybrid deep learning approaches for smartphone sensor-based human activity recognition	14	smartphone	acelerômetro	CNN, RNN
Fall detection using adaptive neuro-fuzzy inference system	12	smartphone	acelerômetro	ANFIS
An SVM fall recognition algorithm based on a gravity acceleration sensor	11	smartphone	acelerômetro	SVM
Elder falls detection based on artificial neural networks	6	smartphone	acelerômetro, giroscópio	MLP, SOM

Continua na próxima página

Tabela III.2 – Continuação da Página Anterior

Artigo	**Citações**	**Dispositivo**	**Sensor**	**Rede Neural**
Ensemble Deep Learning on Wearables Using Small Datasets	6	smartphone	Acelerômetro, giroscópio	RNN, CNN
Gesture-aware fall detection system: Design and implementation	5	smartphone	acelerômetro	MLP
Personalized Fall Detection System	5	smartphone	acelerômetro	RNN
A Smartphone Application for Fall Detection Using Accelerometer and ConvLSTM Network	3	smartwatch	acelerômetro	CNN, RNN
Cnn-based fall detection strategy with edge computing scheduling in smart cities	3	smartphone	acelerômetro	CNN
Fall detection using accelerometer-based smartphones: Where do we go from here?	3	smartphone	acelerômetro,giroscópio	CNN,RNN
An Accurate Fall Detection System for the Elderly People Using Smartphone Inertial Sensors	3	smartphone	acelerômetro, giroscópio	MLP, SVM
Design of elderly fall detection system using CNN	3	smartphone	acelerômetro	CNN
Deep Forest-Based Fall Detection in Internet of Medical Things Environment	1	smartphone	acelerômetro, giroscópio, orientação	Deep Forest
Human Fall Detection using Built-in Smartphone Accelerometer	1	smartphone	Acelerômetro, giroscópio, orientação	SVM, CNN
Online Testing in Machine Learning Approach for Fall Detection	1	smartphone	acelerômetro, giroscópio	RF, SVM, KNN

Continua na próxima página

Tabela III.2 – Continuação da Página Anterior

Artigo	**Citações**	**Dispositivo**	**Sensor**	**Rede Neural**
Fall detection system: Signal analysis in reducing false alarms using built-in tri-axial accelerometer	1	smartphone	acelerômetro	MLP
An Intelligent Mobile App for Fall Detection	1	smartphone	acelerômetro,giroscópio	MLP
Smartphone Sensor-Based Fall Detection Using Machine Learning Algorithms	0	smartphone	acelerômetro	Deep Stacked Autoencoder
Hyper parameter Optimization of Deep Learning Models for Fall Recognition using Tree-structured Parzen Estimator	0	smartphone	acelerômetro, giroscópio	CNN, RNN
Intelligent fall detection system using traditional and non-traditional machine learning algorithm based on MQTT	0	smartphone	acelerômetro	RNN
Wearable Devices with Recurrent Neural Networks for Real-Time Fall Detection	0	smartphone	acelerômetro, giroscópio, orientação	CNN
Smartphone-Based Human Activity and Fall Recognition Using Deep Feature Extraction and Machine-Learning Classifiers	0	smartphone	acelerômetro, giroscópio, orientação	CNN, RNN
A low-power HAR method for fall and high-intensity ADLs identification using wrist-worn accelerometer devices	0	smartphone	acelerômetro, giroscópio	MLP
Human activity detection using smartphone sensor data in real-time based on deep learning approaches	0	smartphone	acelerômetro	RNN
ResNet-based Network for Recognizing Daily and Transitional Activities based on Smartphone Sensors	0	smartphone	acelerômetro, giroscópio, orientação	CNN

Continua na próxima página

Tabela III.2 – Continuação da Página Anterior

Artigo	**Citações**	**Dispositivo**	**Sensor**	**Rede Neural**
Fusion of Body Sensors' Data and Video Images in Assistive Technology	0	smartphone, smartwatch	acelerômetro	CNN, RNN

Capítulo IV Metodologia

Este capítulo está dividido em seis seções e aprofunda a metodologia utilizada para gerar os resultados que são apresentados no capítulo V. Na seção IV.1, aborda-se a problemática da pesquisa e o funcionamento do algoritmo de detecção de quedas. Na Seção IV.2, são descritos o procedimento de coleta de dados e o conjunto de dados *IPqM-Fall*, produzido a partir dessas coletas. Na Seção IV.3, são apresentadas as justificativas para a escolha da rede neural CNN1D. Na Seção IV.4, detalha-se o tratamento e a análise dos dados provenientes do conjunto de dados *IPqM-Fall*. Na Seção IV.5, são apresentados todos os cenários experimentais considerados, englobando as combinações de variáveis inerciais. Por fim, na Seção IV.6, são exibidos os resultados da otimização bayesiana de hiperparâmetros para as arquiteturas CNN1D e suas respectivas implementações.

IV.1 Design do Sistema

O Sistema de Detecção de Baixas que constitui o projeto Combatente do Futuro é dividido em três subsistemas: 1 – Preparação de Dados, 2 – Detecção de Queda e 3 – Classificação de Queda (Figura IV.1). Esta Dissertação aborda a pesquisa e os resultados relacionados aos subsistemas 1 e 2.

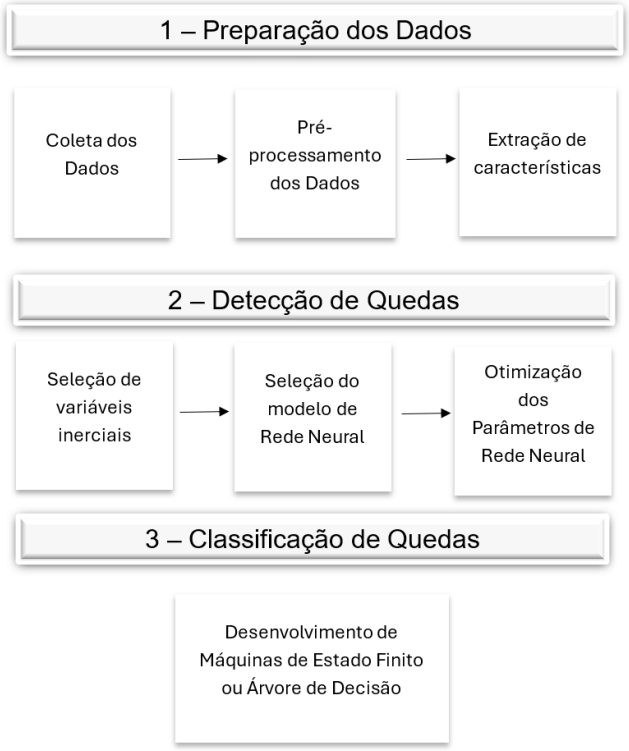


Figura IV.1: Esboço do Sistema de Detecção de Baixas

.

O problema de detecção de quedas pode ser abordado como uma tarefa de aprendizado de máquina, especificamente dentro da área de classificação. As atividades diárias, atividades militares e quedas são tratadas como séries temporais. Uma série temporal é uma sequência de observações feitas em ordem cronológica; neste caso, essas observações referem-se aos dados de aceleração linear e angular captados por dispositivos vestíveis selecionados. Na solução proposta, uma rede neural é treinada em um conjunto de dados contendo as atividades mencionadas.

Durante a operação, a rede neural classifica cada segmento de dados utilizando a técnica de janela deslizante, que divide a série temporal em janelas de tamanho fixo. As atividades diárias, militares e quedas são segmentadas em duas categorias: quedas e não quedas. A rede neural classifica cada janela de dados de aceleração como uma queda ou não queda.

No caso específico deste problema, a janela deslizante é empregada para identificar o padrão exato de uma queda. O padrão de uma queda é caracterizado por uma mudança repentina na aceleração do combatente. A técnica de janelamento permite que a rede neural identifique essa variação abrupta, mesmo que ela ocorra em um curto intervalo de tempo.

Em outras palavras, a rede neural é treinada especificamente para reconhecer os padrões distintos de uma queda, marcados por mudanças bruscas tanto na aceleração linear quanto angular do soldado. O objetivo é que a rede neural detecte essas alterações súbitas nos padrões normais de movimento, mesmo que ocorram em períodos curtos. Matematicamente, seja $x_t{}_{t=1}^{T}$ a série temporal

dos dados de aceleração, onde x_t é a aceleração no tempo t. A rede neural é treinada em um conjunto de dados composto por N janelas de dados. Cada janela é representada por um vetor y_i, onde $i = 1, 2, \ldots, N$. O objetivo do algoritmo de detecção de queda é classificar cada janela de dados como queda ou não queda. A rede neural é treinada para estimar uma função $f(x_t)$ que mapeia os dados de aceleração para uma probabilidade de queda para a janela i, conforme mostrado na Equação IV.1, onde o *limiar de decisão* é uma função não linear que mapeia os valores de entrada para um valor de saída de 0 ou 1.

$$\Pr(y_i = 1) = \text{Limiar de decisão}(f(x_t)), \tag{IV.1}$$

Nesta pesquisa, foram realizados experimentos para determinar quais componentes das forças extraídas das variáveis inerciais dos sensores de acelerômetro e giroscópio capturam melhor o comportamento de uma queda. Esses experimentos ajudarão a identificar qual conjunto de variáveis deve ser utilizado. Além disso, o limiar de decisão é um parâmetro crucial na classificação de quedas. Se determinado de forma aleatória, pode gerar previsões menos assertivas. Portanto, o limiar de decisão foi calibrado utilizando a técnica de otimização bayesiana para definir o valor ideal para a classificação de quedas.

Dado o explicado e observado na subseção de análise de dados, uma queda geralmente dura cerca de cinco segundos. Dessa maneira, o algoritmo de detecção de queda, destinado a funcionar em tempo real durante operações militares, seguirá o seguinte procedimento:

1. O dispositivo vestível, por meio de seus sensores de acelerômetro e giroscópio, registrará continuamente os dados das atividades realizadas pelo soldado.

2. Simultaneamente à coleta dos dados, será iniciada uma janela deslizante com passo de um segundo. Quando o intervalo atingir cinco segundos, ele será considerado uma janela, e os dados dessa janela serão processados. Por exemplo, as três primeiras janelas de uma operação: a primeira começa no tempo 0s e termina no tempo 5s, a segunda começa no tempo 1s e termina no tempo 6s, e a terceira começa no tempo 2s e termina no tempo 7s. A cada segundo, uma nova janela será iniciada.

3. Os dados de cada janela serão organizados em vetores e enviados para classificação pela rede neural.

A técnica de janela deslizante busca encontrar o padrão exato de uma queda, que foi previamente apresentado à rede neural durante o treinamento. Se uma queda iniciar no segundo 2 e terminar no segundo 7, o início será capturado pela primeira janela (0 a 5 segundos), uma parte pela segunda

(1 a 6 segundos) e o padrão completo será capturado apenas na terceira janela (2 a 7 segundos). Assim, o algoritmo detectará a queda somente ao final da terceira janela.

Portanto, a detecção de quedas ocorre com um pequeno atraso, dependendo do momento em que o padrão de queda é identificado. No entanto, considerando o conceito de "hora de ouro"e os "dez minutos de platina", essa detecção é suficientemente rápida para melhorar a consciência situacional do C2 e permitir a rápida mobilização da equipe de resgate. A Figura IV.2 ilustra esse processo, onde cada janela é representada por uma linha pontilhada. A janela roxa captura o início do padrão de queda, enquanto a janela vermelha identifica o padrão completo.

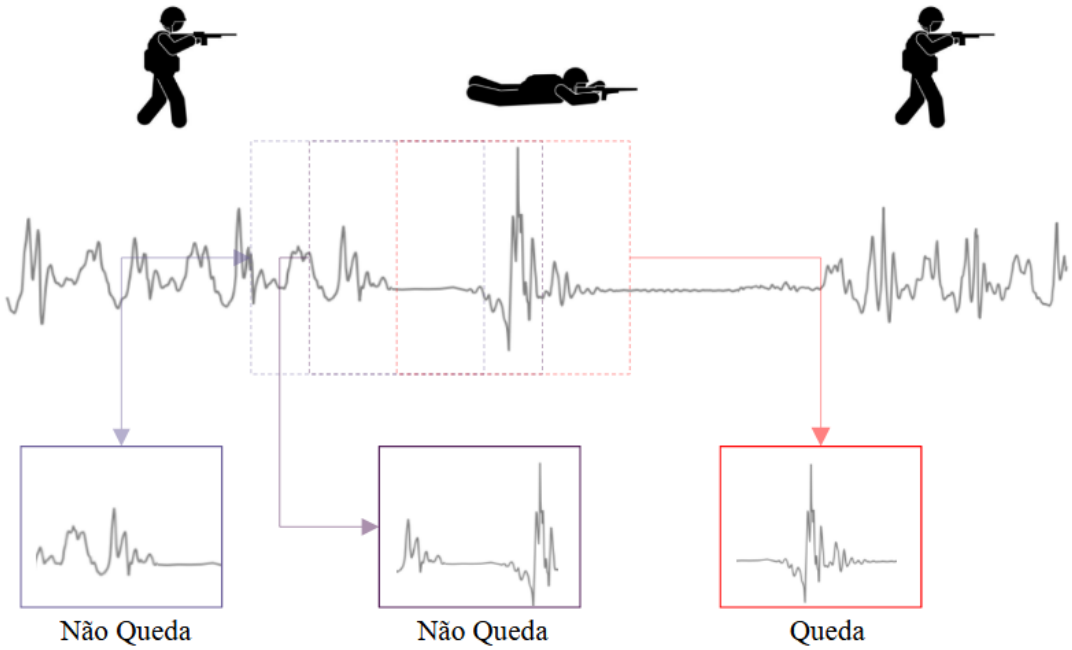


Figura IV.2: Funcionamento do algoritmo de detecção de quedas do projeto Combatente do Futuro

IV.2 Conjunto de Dados *IPqM_Fall*

Nesta seção, fornecemos uma visão geral do conjunto de dados *IPqM_Fall*, detalhando a metodologia de coleta e organização dos dados para detecção de quedas. O objetivo foi criar uma base de dados abrangente para a análise de atividades de vida diária, operações militares e quedas, utilizando variáveis inerciais coletadas por dispositivos vestíveis e *smartphones*. As seções a seguir abordam os dispositivos utilizados e as variáveis coletadas (Seção IV.2.1), a metodologia de coleta de dados (Seção IV.2.2), os participantes voluntários (Seção IV.2.3), as atividades coletadas (Seção IV.2.4) e a organização do conjunto de dados (Seção IV.2.5). Cada subseção oferece uma visão sobre os métodos empregados para garantir a integridade e a utilidade dos dados para o treinamento e a validação de modelos de aprendizado de máquina para detecção de quedas.

IV.2.1 Dispositivos Utilizados e Variáveis Coletadas

Os dispositivos vestíveis selecionados pelo projeto "Combatente do Futuro"para a coleta de dados e subsequente uso em operações foram o *smartwatch* Samsung Galaxy Watch 4 e o *smartphone* LG Velvet. O Galaxy Watch 4 é um dispositivo vestível que, além de conectar-se aos celulares, também é capaz de operar de forma independente, registrando dados biométricos mesmo quando não está pareado com um celular.

O Galaxy Watch 4 e o LG Velvet estão equipados com uma série de sensores, incluindo acelerômetro, barômetro, bússola, sensor geomagnético, giroscópio, entre outros.

As variáveis que foram coletadas dos militares voluntários e que serão observadas em experimentos utilizando Redes Neurais são as variáveis dos sensores de acelerômetro e giroscópio, que são:

- Aceleração Linear: Obtida através do sensor de acelerômetro, mede a taxa de variação da velocidade linear de um soldado. Fornece informações sobre a aceleração linear ao longo dos eixos x, y e z.
- Aceleração Angular: Captada pelo sensor de giroscópio, avalia as mudanças na velocidade angular do corpo ao longo do tempo. Fornece informações sobre a aceleração angular nos eixos x, y e z.

IV.2.2 Coleta de Dados

O processo de coleta de dados com voluntários militares para compor o conjunto de dados *IPqM-Fall* foi submetido à Plataforma Brasil `https://conselho.saude.gov.br/plataforma-brasil-conep`, a plataforma unificada brasileira para registros de pesquisas com seres humanos, e recebeu aprovação do Comitê de Ética do Hospital Naval Marcílio Dias, sob o protocolo CAAE 75570623.3.0000.5256.

O método de coleta foi projetado para capturar dados de múltiplos indivíduos simultaneamente. Cada militar voluntário utilizou três dispositivos durante as atividades: um *smartphone Android* no bolso do peito do uniforme militar, além de dois *smartwatches WearOS*, um no pulso esquerdo e outro no pulso direito (Figura IV.3). Aplicativos de amostragem personalizados foram desenvolvidos e instalados tanto nos *smartphones* quanto nos *smartwatches*, juntamente com um aplicativo instrutor instalado em um *tablet Android* utilizado pelo coordenador do experimento. Assim, foram desenvolvidos dois aplicativos de amostragem: um para dispositivos móveis e outro para dispositivos vestíveis.



Figura IV.3: Posição dos dispositivos utilizados no corpo.

O aplicativo de amostragem conecta-se e recebe comandos do aplicativo do instrutor via HTTP utilizando *Server Sent Events* (SSE), permitindo que o aplicativo do instrutor envie simultaneamente mensagens de início de exercícios. Isso permite que todos os dispositivos conectados comecem a amostrar ao mesmo tempo, sem necessidade de intervenção dos voluntários. A mensagem de início inclui um código para identificar o exercício iniciado e a duração do período de coleta de dados.

Após a coleta, o aplicativo armazena os dados localmente em um sistema de gerenciamento de banco de dados. Os dados incluem o identificador, *timestamp*, variáveis coletadas, localização no corpo (peito, pulso direito ou esquerdo) e o tipo de atividade realizada. Posteriormente, os dados são disponibilizados para *download*.

Os experimentos foram conduzidos em grupos de 3 a 5 militares. Os voluntários realizaram atividades de três categorias distintas: Atividades de Vida Diária, Atividades de Operações Militares e Atividades de Quedas. Todas as atividades foram supervisionadas por um especialista em educação física e por um militar treinado em atividades militares e manejo de armas de fogo. Durante as atividades, os aplicativos de amostragem foram programados para vibrar assim que o instrutor iniciasse o comando de gravação da atividade, sinalizando para os voluntários o início da execução.

Os dispositivos utilizados para amostragem foram configurados para capturar dados na taxa máxima utilizando a configuração *SENSOR_DELAY_FASTEST*, conforme recomendado na documentação do *Android* Android [2024]. Como esperado, as taxas de amostragem variaram entre os dispositivos. O *smartphone*, devido ao seu maior poder de processamento, foi capaz de atingir uma frequência de amostragem mais alta. Para a aceleração linear, o telefone teve uma taxa média de amostragem de 221 Hz com desvio padrão de 23,4 Hz, enquanto os *smartwatches* atingiram 92,6 Hz com desvio padrão de 1,25 Hz. Para a velocidade angular, o *smartphone* apresentou uma taxa média de 219,2 Hz com desvio padrão de 23,2 Hz, em comparação com a média de 91,7 Hz com

desvio padrão de 1,25 Hz dos *smartwatches*. Os dados coletados são fornecidos juntamente com seu *timestamp* em milissegundos, permitindo a subamostragem ou até mesmo interpolações para atingir taxas de amostragem mais constantes ou mais baixas, conforme a preferência do usuário.

IV.2.3 Participantes Voluntários

Para construir o conjunto de dados *IPqM-Fall*, coletamos dados de 15 militares voluntários. Para serem incluídos no experimento de coleta de dados, os militares deveriam fazer parte da escala de serviço armado diário no IPqM, um serviço realizado em dias predeterminados em que o militar porta uma arma de fogo e sua principal função é a salvaguarda das instalações militares. Além disso, os voluntários deveriam ter participado de treinamentos de tiro de fuzil e pistola, serem habilitados para o serviço armado diário com ambas as armas e ter mais de um ano de serviço, para garantir um nível mínimo de expertise. Portanto, foram excluídos voluntários inaptos para o serviço armado e o treinamento físico militar.

Foram selecionados militares de ambos os sexos e de várias patentes, conforme mostrado na Tabela IV.1. A seleção de militares de diferentes patentes visa garantir uma diversidade de experiências e habilidades, proporcionando uma amostra mais representativa da população militar.

Tabela IV.1: Idade, patente, sexo e informações antropométricas dos militares voluntários.

ID	Patente	Sexo	Idade	Altura (m)	Peso (kg)
1	Marinheiro	Masculino	23	1,63	68
2	Sargento	Masculino	34	1,96	88
3	Sargento	Feminino	36	1,64	68
4	Sargento	Masculino	30	1,71	70
5	Tenente	Masculino	32	1,75	75
6	Sargento	Feminino	31	1,65	74
7	Tenente	Masculino	34	1,74	80
8	Cabo	Masculino	30	1,70	75
9	Tenente	Masculino	38	1,70	95
10	Sargento	Masculino	37	1,71	70
11	Marinheiro	Masculino	24	1,73	79
12	Cabo	Masculino	27	1,71	56
13	Cabo	Feminino	28	1,59	56
14	Tenente	Feminino	29	1,64	58
15	Sargento	Masculino	33	1,71	87

O pessoal militar voluntário foi instruído a se apresentar para o exercício de coleta de dados usando o uniforme OP1, OP2 ou AD Marinha do Brasil [2024], conforme mostrado na Figura IV.4, para possibilitar o uso do *smartphone* no bolso do peito.



Figura IV.4: Uniformes OP1, AD e OP2, respectivamente.

IV.2.4 Atividades Coletadas

Conforme mencionado anteriormente na seção IV.2.2, três tipos distintos de atividades foram coletados: Atividades da Vida Diária, que são tarefas básicas realizadas no cotidiano; Atividades de Operações Militares, que envolvem tarefas realizadas em operações militares e serviços que utilizam armas de fogo; e Atividades de Queda, que incluem situações envolvendo perda de equilíbrio e contato do corpo com o chão. Essas atividades foram selecionadas para cobrir uma ampla gama de situações vivenciadas por um soldado, incluindo aquelas que poderiam ser confundidas com atividades de queda, com o objetivo de obter um conjunto de dados diversificado para treinar redes neurais ou outras técnicas de aprendizado de máquina.

As AVD coletadas foram as seguintes: 1) em pé, 2) caminhando, 3) correndo, 4) pulando, 5) sentado em uma cadeira, 6) levantando-se de uma cadeira, 7) subindo uma ladeira, 8) descendo uma ladeira, 9) correndo subindo uma ladeira, 10) correndo descendo uma ladeira, 11) subindo escadas, 12) descendo escadas, 13) pulando degraus. O número de repetições e a duração dessas atividades, juntamente com as instruções, são detalhados na Tabela IV.2.

Tabela IV.2: Atividades da Vida Diária.

Código	**Atividade**	**Rep.**	**Dur.**	**Descrição**
ADL_1	Em pé	1	3 min	De pé, realizando movimentos sutis
ADL_2	Caminhando	1	3 min	Caminhando durante toda a atividade
ADL_3	Correndo	3	30 seg	Correndo durante toda a atividade
ADL_4	Pulando	3	30 seg	De pé, realizando saltos curtos a cada 2 segundos
ADL_5	Subindo escadas	3	10 seg	Subindo escadas
ADL_6	Descendo escadas	3	10 seg	Descendo escadas
ADL_7	Sentado na cadeira	6	5 seg	Começa de pé, senta em uma cadeira
ADL_8	Levantando-se da cadeira	6	5 seg	Começa sentado em uma cadeira, levanta-se
ADL_11	Caminhada ladeira acima	3	30 seg	Caminhando subindo uma ladeira
ADL_12	Caminhada ladeira abaixo	3	30 seg	Caminhando descendo uma ladeira
ADL_13	Corrida ladeira acima	3	15 seg	Correndo subindo uma ladeira
ADL_14	Corrida ladeira abaixo	3	15 seg	Correndo descendo uma ladeira
ADL_15	Pulando degraus	3	10 seg	Subindo escadas e pulando um degrau por vez

As atividades ADL_1, ADL_4, ADL_5, ADL_6, ADL_11, ADL_12, ADL_13 e ADL_14 também foram realizadas com porte de fuzil, sendo consideradas atividades de operações militares com o mesmo número e duração de repetições indicados na Tabela IV.2. As atividades ADL_13 e ADL_14 com uso de fuzil não são corridas convencionais, mas atividades de engajamento rápido.

As Atividades de Operações Militares (OM) foram realizadas com o porte de fuzil, com exceção da atividade de rastejar. As atividades coletadas foram as seguintes: 1) Varredura com caminhada, 2) Varredura com engajamento rápido, 3) Transição de posição de tiro de pé para ajoelhado, 4) Transição de caminhada para posição de tiro ajoelhado, 5) Transição de corrida para posição de tiro ajoelhado, 6) Transição de pé para posição de tiro deitado, 7) Transição de caminhada para posição de tiro deitado, 8) Transição de corrida para posição de tiro deitado, 9) Rastejar. O número e a duração das atividades, bem como as descrições, são detalhados na Tabela IV.3.

Tabela IV.3: Atividades de Operações Militares.

Código	**Atividade**	**Rep.**	**Dur.**	**Descrição**
OM_1	Varredura (Caminhada)	1	3 min	Caminhando enquanto faz varredura (apontando o rifle em um ângulo de cobertura de aproximadamente 120 graus)
OM_2	Varredura (Engajamento Rápido)	3	45 seg	Engajamento rápido enquanto faz varredura (apontando o rifle em um ângulo de cobertura de aproximadamente 120 graus)
OM_3	Posição de Tiro Ajoelhado (De Pé)	3	10 seg	Começa de pé, transição para posição de tiro ajoelhado
OM_4	Posição de Tiro Ajoelhado (Caminhada)	3	1 min	Começa caminhando, transição para posição de tiro ajoelhado
OM_5	Posição de Tiro Ajoelhado (Corrida)	3	45 seg	Começa correndo, transição para posição de tiro ajoelhado
OM_6	Posição de Tiro Deitado (De Pé)	3	10 seg	Começa de pé, transição para posição de tiro deitado
OM_7	Posição de Tiro Deitado (Caminhada)	3	1 min	Começa caminhando, transição para posição de tiro deitado
OM_8	Posição de Tiro Deitado (Corrida)	3	45 seg	Começa correndo, transição para posição de tiro deitado
OM_9	Rastejar	3	50 seg	Começa na posição de rastejo e realiza a atividade pelo tempo determinado

As Atividades de Queda foram realizadas de duas formas, com porte de fuzil e sem o porte de fuzil, como segue: 1) queda para frente terminando deitado de costas, 2) queda para frente terminando deitado de bruços, 3) queda para trás, 4) queda para o lado direito, 5) queda para o lado esquerdo. As quedas laterais foram alternadas, com um grupo de militares voluntários realizando a queda para o lado esquerdo enquanto outro grupo realizava a queda para o lado direito. A quantidade, duração e descrições das atividades são detalhadas na Tabela IV.4.

Tabela IV.4: Atividades de Queda.

Código	Atividade	Reps	Dur.	Descrição
FALL_1	Queda Frontal (Deitado de Costas)	10	10 seg	Começa de pé e termina a queda deitado de costas
FALL_2	Queda Frontal (Deitado de Bruços)	10	10 seg	Começa de pé e termina a queda deitado de bruços
FALL_3	Queda Traseira	10	10 seg	Começa de pé e termina a queda deitado de costas
FALL_4	Queda Lateral Direita	5	10 seg	Começa de pé e termina a queda deitado de lado (direita)
FALL_5	Queda Lateral Esquerda	5	10 seg	Começa de pé e termina a queda deitado de lado (esquerda)

Para todas as atividades que envolvem o uso de um fuzil, visando minimizar o risco de disparo acidental devido a erros no descarregamento da arma e evitar danos ao equipamento durante as atividades, foi tomada a decisão de utilizar um fuzil *Airsoft* M4 como substituto da arma real. O uso do fuzil *Airsoft* não comprometerá os resultados, pois o modelo escolhido,*Airsoft* M4 4003MG JG (Figura IV.5), possui dimensões e peso semelhantes aos de um fuzil M4 real. Por exemplo, um fuzil Colt M4 *Carbine* (Figura IV.5) pesa 2,7 kg quando descarregado e tem um comprimento de 81,28 cm, enquanto o modelo *Airsoft* selecionado pesa 2,8 kg e tem um comprimento total de 78 cm.

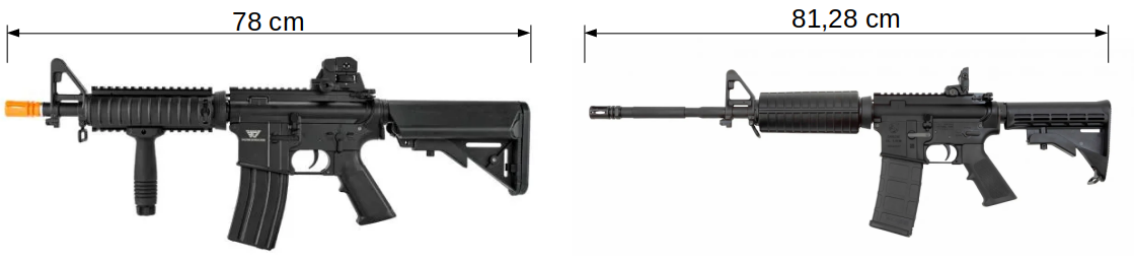


Figura IV.5: Fuzis Airsoft M4 4003MG e Colt M4 Carbine respectivamente.

IV.2.5 Organização do Conjunto de Dados

O conjunto de dados coletados está disponibilizado em `https://zenodo.org/records/12760391` e está organizado da seguinte forma: há 15 diretórios correspondentes a cada voluntário militar. Dentro de cada diretório, há um arquivo chamado *README* e três subdiretórios nomeados como *chest*, *left*, e *right*, que contêm dados coletados pelo *smartphone* (no peito) e pelos *smartwatches* nos pulsos esquerdo e direito. O arquivo *README* contém informações sobre o voluntário militar, que foram registradas na Tabela IV.1, incluindo o número de atividades e repetições realizadas pelo voluntário. Cada subdiretório contém os seguintes arquivos:

- sampling.csv: Contém o ID da atividade, bem como o nome do exercício correspondente ao ID. Além disso, indica a posição do sensor no corpo e os *timestamps* de início e término da gravação da atividade. Possui o campo *withRifle*, que usa o número 0 para indicar que a atividade foi gravada sem fuzil e o número 1 para indicar que foi gravada com fuzil. Através do ID da atividade, é possível identificar as atividades registradas nos arquivos acceleration.csv e angular_speed.csv. O campo id do arquivo sampling.csv corresponde ao campo *sampling* nos arquivos acceleration.csv e angular_speed.csv, funcionando de maneira semelhante a uma chave estrangeira.

- acceleration.csv e angular_speed.csv: Contêm o campo *sampling*, que identifica a atividade registrada, o *timestamp* e os componentes x, y e z de cada observação.

IV.3 Escolha das Arquiteturas de Redes Neurais

Conforme discutido na análise bibliométrica dos trabalhos relacionados (Seção III.2), as arquiteturas de redes neurais amplamente utilizadas na detecção de quedas e reconhecimento de atividades humanas incluem Redes Neurais Recorrentes (RNN), Redes Neurais Convolucionais (CNN) e Perceptrons Multicamadas (MLP). Cada uma dessas arquiteturas tem suas particularidades, sendo adequadas para diferentes tipos de problemas e contextos operacionais. No entanto, para este projeto, que visa desenvolver um sistema de detecção de quedas embarcado em dispositivos portáteis como *smartwatches* e *smartphones*, torna-se crucial avaliar a viabilidade de cada uma dessas arquiteturas em relação às limitações de hardware, como autonomia da bateria, capacidade de processamento e memória. A seguir, apresentamos a avaliação das três arquiteturas:

- Redes Neurais Recorrentes (RNN): São especialmente eficazes em lidar com séries temporais devido à sua capacidade de manter uma "memória"ao longo do tempo. Isso as torna uma escolha natural para o reconhecimento de padrões sequenciais, como a detecção de quedas, onde a análise da sequência dos dados é crucial. No entanto, apesar dessa vantagem, as RNNs apresentam uma desvantagem significativa: seu alto custo computacional. A complexidade inerente dessas redes exige maior poder de processamento, o que impacta diretamente a autonomia de dispositivos portáteis.

  Além disso, o treinamento das RNNs pode ser mais difícil, especialmente em ambientes com dados ruidosos, como os provenientes de atividades de vida diária, operações militares e quedas. Nessas situações, o movimento brusco e as variações rápidas nos dados capturados por sensores vestíveis aumentam a dificuldade do aprendizado adequado da rede.

  Dado o contexto de uso militar, onde as operações podem durar várias horas e os dispositivos precisam operar de forma autônoma, o uso de RNNs foi descartado. O alto consumo de

energia e a latência no processamento tornariam inviável sua implementação embarcada em dispositivos como *smartwatches* e *smartphones*, que possuem recursos computacionais limitados e necessitam de alta eficiência energética para suportar longas missões.

- Perceptrons Multicamadas (MLP): São uma arquitetura simples e eficiente, frequentemente utilizada para tarefas de classificação em problemas não lineares. No entanto, apresentam limitações quando aplicados a séries temporais, pois não capturam de maneira nativa a dependência entre dados sequenciais. Como o problema de detecção de quedas envolve a análise de padrões temporais, os MLPs não se mostram adequados para essa aplicação.

  Embora possam ser utilizados com sucesso em uma variedade de problemas, sua aplicação no contexto de séries temporais exige transformações nos dados de entrada, o que aumenta a complexidade do sistema sem agregar significativamente à precisão do modelo. Dado que a segmentação dos dados em séries temporais é um componente central da pesquisa, a utilização de MLPs foi descartada em favor de abordagens mais adequadas a esse tipo de dado.

- Redes Neurais Convolucionais (CNN): As CNNs unidimensionais, têm se mostrado uma escolha promissora para a análise de séries temporais. Essas redes são capazes de capturar padrões locais dentro dos dados de entrada através da aplicação de convoluções sobre segmentos temporais, o que as torna ideais para a detecção de quedas, onde pequenas variações nos dados de aceleração e velocidade angular podem indicar a ocorrência de um evento crítico.

  Além disso, CNNs possuem uma complexidade computacional intermediária, significativamente menor que a das RNNs, e podem ser implementadas de forma eficiente em dispositivos com recursos limitados, como o *smartwatch* Samsung Galaxy Watch 4 utilizado nesta pesquisa. A capacidade de processamento desses dispositivos é limitada, e redes convolucionais oferecem um bom equilíbrio entre precisão e eficiência, além de serem menos exigentes em termos de consumo de energia.

  Por essas razões, a escolha recaiu sobre as CNNs unidimensionais, que apresentam a vantagem de capturar padrões em séries temporais com menor custo computacional e maior eficiência energética. Esta arquitetura permite que o algoritmo seja implementado de maneira embarcada, prolongando a autonomia do dispositivo e garantindo a operação contínua durante as longas operações militares.

A escolha da arquitetura de rede neural deve, portanto, considerar não apenas a precisão e robustez do modelo na detecção de quedas, mas também os requisitos práticos impostos pela aplicação militar, como tempo de operação prolongado e limitação de recursos computacionais. As CNNs unidimensionais, por sua capacidade de trabalhar eficientemente com séries temporais, seu menor custo

computacional e sua eficiência energética, emergem como a solução mais adequada para o problema de detecção de quedas em dispositivos portáteis, atendendo aos requisitos do projeto "Combatente do Futuro". A implementação final desse sistema embarcado visa melhorar a eficácia das operações militares, proporcionando uma detecção rápida e confiável de quedas, com impacto mínimo sobre a autonomia dos dispositivos utilizados.

IV.4 Pré-Processamento

O pré-processamento dos dados é uma etapa em que os dados brutos referentes a determinados problemas são transformados em um formato adequado para análise posterior. É uma etapa obrigatória no processo de análise de dados e modelagem estatística. A principal finalidade dessa fase é garantir que os dados estejam limpos, consistentes e prontos para serem utilizados pelos algoritmos de aprendizado de máquina, técnicas estatísticas ou outras abordagens de análise de dados. Nesta pesquisa, o pré-processamento foi dividido em três etapas: extração de características, análise dos dados e preparação dos dados.

IV.4.1 Extração de características

Os sensores de acelerômetro e giroscópio dos dispositivos vestíveis registram medições nas três componentes principais: x, y e z, que correspondem à aceleração linear e à aceleração angular. Além dessas componentes individuais, extraímos a força resultante, variável que chamamos de magnitude. A magnitude combina as três componentes em uma única medida de força, representando a aceleração total aplicada a um objeto, levando em consideração tanto a direção quanto a intensidade de cada eixo.

Conforme discutido na seção de análise bibliométrica (Capítulo III), diversos estudos científicos utilizam a magnitude como variável principal ou até mesmo como única variável na detecção de quedas. Portanto, torna-se fundamental extrair essa variável dos dados coletados, para que seu desempenho seja avaliado em um dos 12 cenários experimentais testados nesta pesquisa.

A magnitude é calculada pela seguinte equação:

$$R = \sqrt{a_x^2 + a_y^2 + a_z^2} \tag{IV.2}$$

onde a_x, a_y e a_z representam as componentes de aceleração em cada eixo.

IV.4.2 Análise e Preparação dos Dados

Para aprofundar a compreensão dos diversos padrões de atividades coletadas no IPqM-Fall, realizamos uma análise dos dados obtidos do pessoal militar. A análise foi baseada na magnitude

da aceleração linear medida pelos sensores de acelerômetro nos dispositivos vestíveis usados durante as atividades coletadas. A magnitude é uma variável que combina os valores das três componentes do movimento, facilitando a visualização das variações em qualquer eixo, o que a torna ideal para identificar mudanças significativas no comportamento.

Nesta seção, apresentamos tanto a análise dos dados quanto o processamento subsequente realizado com base nos resultados obtidos. As atividades de queda sem fuzil exibem padrões consistentes (Figura IV.6), marcados por um pico acentuado na magnitude da aceleração no momento em que o corpo atinge o solo. Na maioria dessas atividades, esse pico varia entre 70 e 130, sendo geralmente seguido por um ou dois picos menores, com valores frequentemente acima de 40. As quedas com fuzil apresentam um padrão semelhante, porém com magnitudes ligeiramente mais baixas. Esses padrões característicos das quedas com e sem fuzil são essenciais para identificar eventos de queda, dado o comportamento uniforme observado em diferentes instâncias dessa atividade.

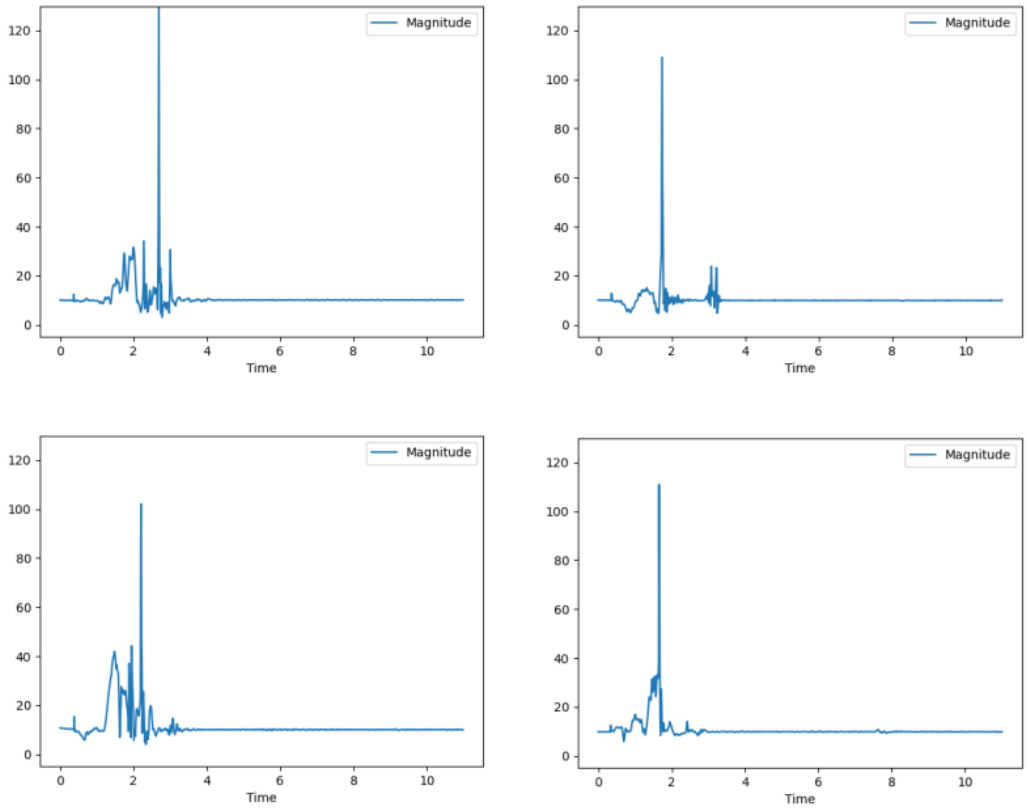


Figura IV.6: Atividades de queda registradas pelo *smartwatch* esquerdo do ID 3: Fall_1, Fall_2, Fall_3 e Fall_5, respectivamente.

Em contraste, as atividades da vida diária AVD e as operações militares que não envolvem transição para a posição de tiro deitado não apresentam padrões tão bem definidos (Figura IV.7), exibindo variações mais significativas entre si. Ainda assim, essas atividades possuem características suficientemente distintas das quedas, o que facilita a diferenciação entre os dois tipos de eventos. Isso permite a adoção de uma abordagem de classificação binária, dividindo as atividades em duas

classes principais: "Atividades de Queda"e "Atividades Não-Queda". A primeira classe inclui todas as quedas, tanto com quanto sem fuzil, enquanto a segunda engloba as AVDs e as operações militares sem quedas.

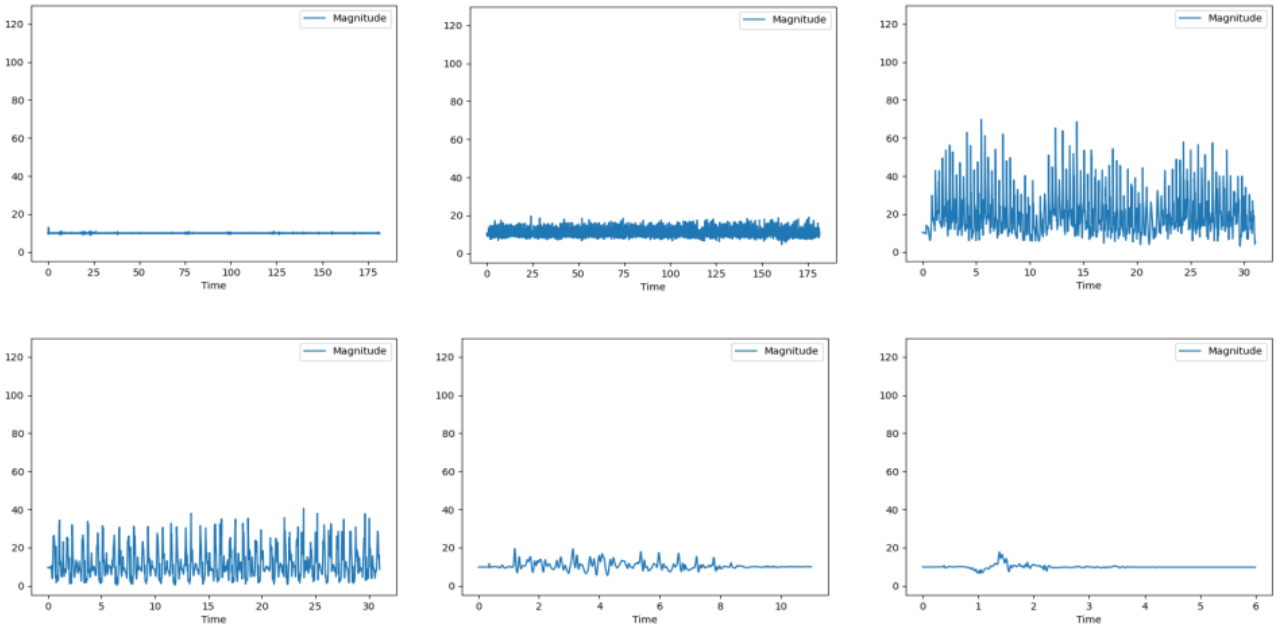


Figura IV.7: Atividades de queda registradas pelo *smartwatch* esquerdo do ID 3: ADL_1, ADL_2, ADL_3, ADL_4, ADL_5 e ADL_7, respectivamente.

Para fins de classificação, a classe rotulada como "quedas"foi identificada com o rótulo 0, enquanto a classe "não-quedas"recebeu o rótulo 1. Esse esquema de rotulagem binária foi utilizado para treinar a RNN unidimensional.

Como mencionado, as atividades que envolvem quedas, tanto com quanto sem um fuzil, apresentam padrões semelhantes, embora haja variações na magnitude. Da mesma forma, que uma mesma atividade ADL realizada com ou sem o fuzil também exibe padrões semelhantes, mas com diferenças significativas na magnitude.(Figura IV.8).

Em geral, as atividades de queda têm uma duração de cinco segundos, seguidas por um padrão estável que exibe uma magnitude média de aceleração de aproximadamente 10 nos cinco segundos finais, refletindo a inatividade do corpo após a queda. Assim, a janela de tempo estabelecida para a detecção de quedas é de cinco segundos, correspondendo, em média, a 480 pontos de observação coletados pelo *smartwatch* e 1020 pontos de observação registrados pelo *smartphone*. A média de aceleração linear e angular é similar, uma vez que as frequências de coleta de dados são próximas.

Para gerar os conjuntos de dados de treino, teste e validação, as atividades de queda coletadas foram segmentadas em vetores de dados com 480 observações (*smartwatch* direito e esquerdo) e 1020 observações (*smartphone* no peito), referentes à série temporal de cinco segundos. Foram descartados os dados relacionados ao corpo em contato com o chão, mantendo apenas aqueles

referentes ao impacto do corpo. Após definir o tamanho dos vetores para as quedas, para garantir a mesma dimensionalidade nos dados de entrada das redes neurais testadas, o tamanho dos vetores correspondentes às outras atividades do *IPqM-Fall* também foi fixado em 480 e 1250 observações.

Nas atividades com duração de 10 segundos ou mais, exceto pelas exceções que serão abordadas posteriormente, os dados foram segmentados da seguinte maneira: considerando, por exemplo, a atividade de corrida, que tem uma duração de 30 segundos, ela foi dividida em seis vetores de dados correspondentes aos intervalos de 0 a 5 segundos, 5 a 10 segundos e assim por diante. No entanto, a quantidade total de observações em uma determinada atividade pode variar, tanto para mais quanto para menos, devido à frequência de coleta e às imprecisões manuais de marcação de tempo. Com uma média de 480 observações definida e considerando essa variação, a quantidade de vetores de dados por atividade obedeceu à seguinte regra de descarte: tomando como exemplo a atividade de corrida mencionada, se houve 2395 observações, o último vetor de dados terá um tamanho de 395. Para manter a uniformidade no tamanho dos dados, esses vetores foram descartados. Uma condição semelhante é aplicada quando a quantidade total de observações ultrapassa 2400.

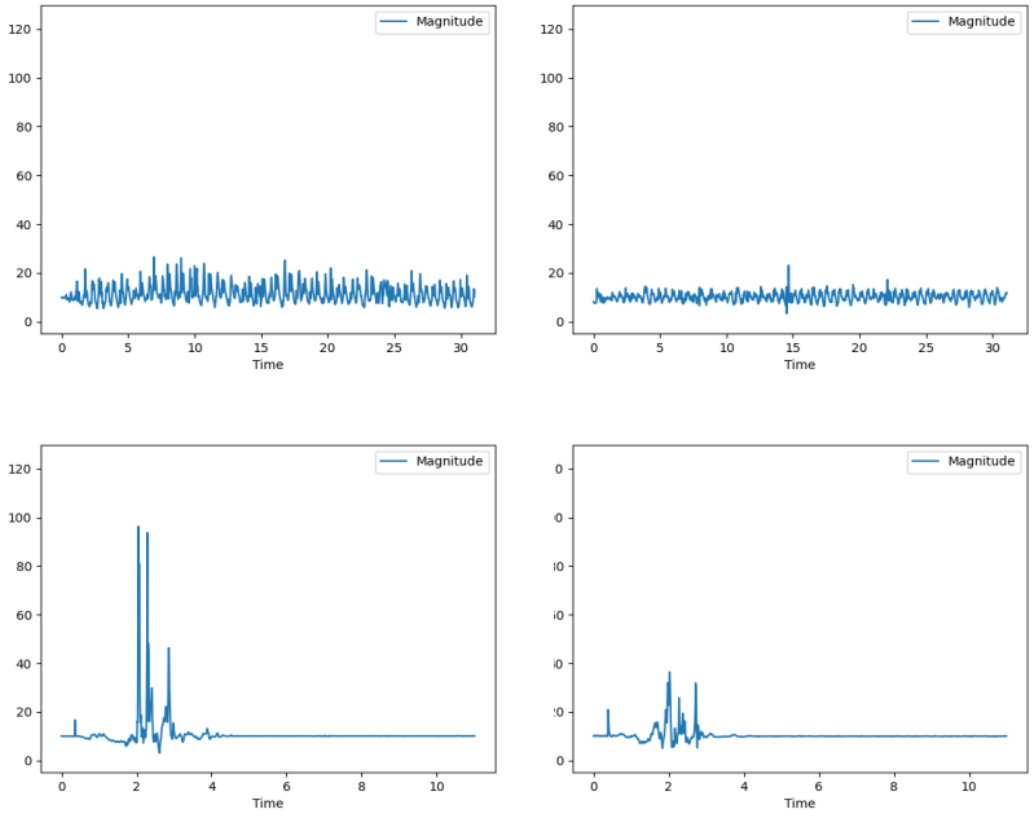


Figura IV.8: Atividades de queda registradas pelo *smartwatch* esquerdo de ID 3: ADl_12, ADL_12_withRifle, Fall_3 e Fall_3_withRifle, respectivamente.

Há uma diferença na magnitude das atividades que envolvem caminhar e correr com e sem um fuzil (ADL_2, ADL_3, OM_1 e OM_2), com a magnitude das atividades com um fuzil sendo menor. Isso ocorre porque caminhar com um fuzil (OM_1) é uma caminhada de engajamento,

caracterizada por um ritmo mais lento e controlado. E correr com um fuzil (OM_2) é uma atividade de engajamento em velocidade, não exatamente uma caminhada ou uma corrida e sim uma atividade ritmada , controlada em velocidade superior a OM_1. O peso do fuzil também contribui para essa magnitude reduzida, já que o usuário precisa exercer um maior controle e estabilização durante o movimento.

As Atividades com dados coletados durante testes de dez segundos ou mais, exceto para atividades de transição e as atividades ADL_5, ADL_6 e ADL_15 relacionadas a subir e descer escadas, apresentam um padrão semelhante na variação da magnitude ao longo da duração da atividade. Isso permitiu que essas atividades fossem segmentadas em mais de um conjunto de cinco segundos como mencionado anteriormente. Para as atividades que envolvem escadas, o tamanho do conjunto foi definido utilizando apenas os primeiros cinco segundos da atividade, descartando o restante dos dados coletados, uma vez que a escada onde as atividades foram coletadas era curta, com apenas dez degraus, e, na maioria das atividades coletadas, após cinco segundos, os voluntários militares já haviam completado a subida ou descida das escadas.

Nas atividades de transição para a posição de tiro (OM_3 a OM_8), como a atividade OM_4, que representa a transição da caminhada para a posição de tiro ajoelhado, o foco está no padrão gerado pela mudança de estado entre caminhar e ajoelhar. O objetivo é verificar, por meio dos resultados fornecidos pela rede neural, se essa transição é, de alguma forma, confundida com uma atividade de queda. Para isso, foi selecionado um único vetor de observação de cinco segundos (480 e 1020), que representa o momento da mudança de estado de uma atividade para outra. Assim, o ponto de observação onde ocorre o pico de maior magnitude foi identificado, e foram escolhidas as 240 (510) observações anteriores ao pico e as 240 (510) observações posteriores para a construção do vetor de dados apresentado à rede neural, exceto nos casos em que não foi possível realizar uma distribuição equilibrada.

As atividades de queda e as atividades de transição para a posição de tiro deitado (OM_6, OM_7 e OM_8) exibem padrões semelhantes (Figura IV.9), pois ambas envolvem o corpo atingindo o solo, o que pode confundir a rede neural. Diante disso, a abordagem adotada nos cenários experimentais foi treinar as redes neurais de duas maneiras: em uma, essas atividades foram rotuladas como quedas; e, na outra, como atividades não relacionadas a quedas.

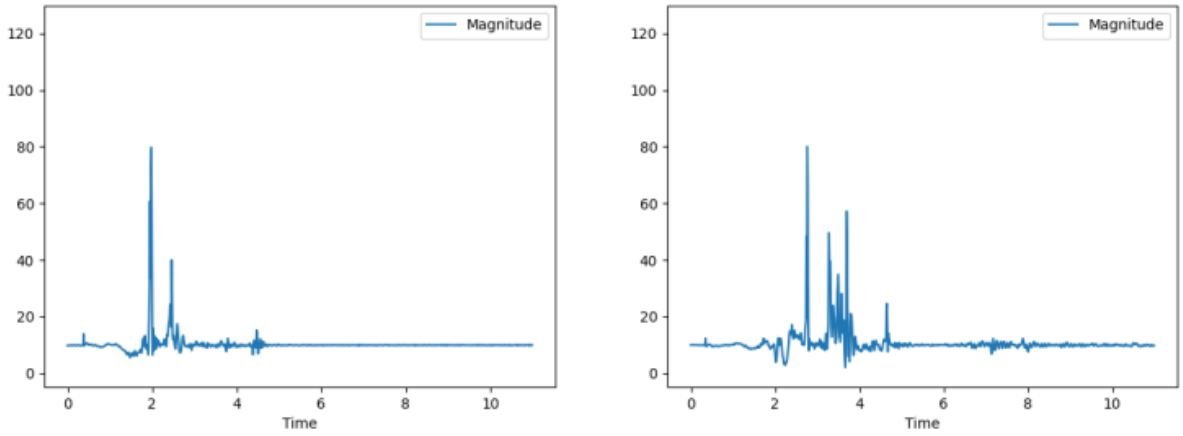


Figura IV.9: Atividades registradas pelo *smartwatch* esquerdo do ID 3: Fall_3 e OM_6, respectivamente.

A magnitude ao longo do tempo, evidenciada na análise das atividades de queda, sugere que a análise no domínio da frequência também pode ser útil. Essa análise poderia, hipoteticamente, caracterizar com mais precisão as atividades de queda e distingui-las das ADLs e das atividades de operações militares. Portanto, a aplicação de uma Transformada de Fourier aos conjuntos de dados e o uso desses dados para treinar uma rede neural poderia resultar em uma classificação mais eficiente do que no domínio do tempo. Com isso em mente, foi realizada uma comparação entre os resultados de classificação de quedas nos domínios da frequência e do tempo.

Para a geração de dados no domínio da frequência, foi aplicada uma Transformada de Fourier a cada vetor de dados com 480 e 1020 observações do domínio do tempo, subtraindo da transformada o nível DC, que representa o componente de frequência zero, ou seja, a parte do sinal que não varia ao longo do tempo e representa a média do sinal. Dessa forma, os vetores de dados resultantes foram reduzidos para metade do tamanho das observações do vetor no domínio do tempo.

Para a execução de todos os cenários experimentais (Seção IV.5), os dados pré-processados para o treinamento das redes neurais foram divididos em três conjuntos: treinamento, teste e validação. Os conjuntos de teste e validação corresponderam cada um a 20% das amostras, enquanto o conjunto de treinamento corresponderam a 60% das amostras. A otimização bayesiana dos hiperparâmetros (Seção IV.6) foi realizada utilizando os dados de treinamento e validação. O conjunto de teste não foi apresentado à rede durante a otimização ou o treinamento subsequente, sendo usado apenas para fazer previsões e avaliar o desempenho dos modelos.

Todos os passos de preparação do conjunto de dados para o treinamento da rede neural e o Pipeline em código Python para a reprodução da preparação de dados e dos experimentos apresentados nesta dissertação estão disponíveis no repositório `https://github.com/AILAB-CEFET-RJ/fall-detect`.

IV.5 Cenários Experimentais e Abordagem de Rotulação

Nesta seção, descrevemos os cenários experimentais conduzidos para a detecção de quedas de soldados usando as redes neurais selecionadas. Projetamos os experimentos para avaliar a eficácia de diferentes combinações de sensores (acelerômetro e giroscópio) e os componentes das variáveis inerciais derivadas desses sensores na detecção precisa de quedas.

Realizar testes com diferentes combinações de variáveis inerciais é necessário para determinar quais delas contribuem de forma mais significativa para a precisão dos algoritmos de detecção de quedas. A escolha das variáveis corretas, como a magnitude ou os componentes x, y e z da aceleração linear e angular, pode impactar diretamente a capacidade do modelo de identificar quedas de forma mais assertiva. Cada combinação de variáveis pode capturar diferentes características dos movimentos associados a uma queda, e testar todas essas possibilidades ajuda a identificar quais delas proporcionam a melhor performance para o modelo.

Além disso, testar a eficácia de cada combinação permite entender a influência de diferentes parâmetros inerciais no comportamento do modelo em diferentes cenários. Por exemplo, pode ser que a magnitude da aceleração angular em combinação com a aceleração linear forneça informações suficientes para uma detecção precisa, enquanto em outros casos os componentes x, y e z separados podem fornecer mais detalhes sobre os padrões de movimento.

Ao conduzir esses experimentos, buscamos encontrar a combinação ideal de variáveis inerciais que maximize a assertividade dos algoritmos e minimize os falsos positivos e falsos negativos.

Os cenários de treinamento que testamos foram os seguintes:

1. Usando a magnitude da aceleração linear e a magnitude da aceleração angular individualmente: Neste cenário, as redes neurais são treinadas separadamente usando apenas a magnitude da aceleração linear e a magnitude da aceleração angular. Neste cenário, as redes neurais treinadas serão nomeadas da seguinte forma: Sc_1_acc e Sc_1_gyr para as redes CNN1D treinadas com os dados de aceleração linear e angular, respectivamente.

2. Usando os componentes x, y e z da aceleração linear e da aceleração angular individualmente: Aqui, as redes neurais são treinadas separadamente, cada uma usando os três componentes (x, y e z) da aceleração linear e da aceleração angular.Neste cenário, as redes neurais treinadas serão nomeadas da seguinte forma: Sc_2_acc e Sc_1_gyr para as redes CNN1D treinadas com os dados de aceleração linear e angular, respectivamente.

3. Usando a magnitude da aceleração linear combinada com a magnitude da aceleração angular: Neste experimento, as redes neurais são treinadas com a magnitude da aceleração linear e a magnitude da aceleração angular simultaneamente.Neste cenário, as rede neural treinada será

nomeada como Sc_3.

4. Utilizando os componentes x, y e z da aceleração linear combinados com os componentes x, y e z da aceleração angular: Neste cenário, as redes neurais são treinadas com os componentes x, y e z da aceleração linear e os componentes x, y e z da aceleração angular juntos. A rede neural treinada neste cenário será nomeada como Sc_4.

Os experimentos foram conduzidos para os dados coletados do peito (*smartphone*) e dos pulsos direito e esquerdo (*smartwatches*). Testar sensores em diferentes posições foi realizado para verificar se há variações no desempenho da detecção de quedas dependendo da localização do sensor. Além disso, testar ambos os pulsos é essencial para determinar se a posição das mãos ao segurar uma arma de fogo (fuzil) influencia a detecção de quedas.

Os testes para todos os cenários mencionados foram conduzidos com dados tanto no domínio do tempo quanto no domínio da frequência. As abreviações das redes neurais para cada cenário são acompanhadas por um "T"para o domínio do tempo ou um "F"para o domínio da frequência. Por exemplo: Sc_1_acc_T para a rede CNN1D treinada com dados de aceleração linear do cenário 1 no domínio do tempo e Sc_1_acc_F para a mesma rede treinada no domínio da frequência.

Todos os cenários foram executados com duas abordagens de rotulação distintas, que chamamos de binário_1 e binário_2. Em ambas as abordagens, trabalhamos com classes binárias: a classe "quedas"e a classe "não quedas", conforme mencionado na seção IV.4.2. No entanto, a abordagem binário_1 considera as atividades OM_6, OM_7 e OM_8 como atividades de queda, e essas foram atribuídas ao rótulo 0. Em contraste, na abordagem binário_2, as atividades OM_6, OM_7 e OM_8 são consideradas como não quedas e foram atribuídas ao rótulo 1. As diferentes abordagens são necessárias para avaliar como as atividades de transição para a posição de tiro deitado são interpretadas pela rede neural.

IV.6 Implementação das Redes Neurais

Nesta etapa, que corresponde à implementação das Redes CNN1D para os diversos cenários experimentais, as redes foram implementadas com seus hiperparâmetros estruturados através da técnica de otimização bayesiana, com o objetivo de encontrar os melhores parâmetros e a arquitetura de rede para obter respostas otimizadas tanto no domínio do tempo quanto no domínio da frequência, garantindo que cada cenário experimental foi treinado com a melhor configuração de rede, possibilitando uma comparação justa dos desempenhos obtidos. Todas as modelagens foram realizadas por meio do *framework* de código aberto *Keras*, pertencente ao *TensorFlow*. A otimização bayesiana foi executada por meio do *framework* de código aberto *Optuna*. Ambos os *frameworks* foram desenvolvidos utilizando a linguagem Python.

Para cada arquitetura de cada cenário experimental, foi definido um mesmo espaço de busca que se adéqua às demandas da arquitetura CNN1D. O parâmetro *trial* foi configurado em 20. O parâmetro *trial* do *Optuna* representa o número de iterações ou experimentos realizados durante a busca pelos melhores hiperparâmetros. É uma medida da quantidade de vezes que o algoritmo de otimização bayesiana explora o espaço de busca para encontrar a melhor combinação de hiperparâmetros que maximize o MCC do modelo.

No espaço de busca são explorados os seguintes hiperpârametros:

1. Quantidade de Filtros: Refere-se ao número de filtros aplicados em cada camada convolucional. Cada filtro é responsável por aprender diferentes características dos dados de entrada, sendo fundamental para a extração de informações complexas e diversificadas. As Tabelas IV.6, IV.7, IV.8, IV.9, IV.10 e IV.11 apresentam a quantidade de filtros na primeira camada convolucional. A cada camada atribuída na arquitetura implementada, o número de filtros é o dobro da camada anterior.

   Exemplo: Se a primeira camada convolucional utiliza 32 filtros, a segunda camada terá 64, a terceira 128, e assim por diante.

2. Tamanho do *Kernel*: O *kernel* é o filtro que passa pela entrada para extrair características locais. O tamanho define a janela de observação, influenciando a capacidade da rede de detectar padrões menores ou maiores nos dados.

3. Quantidade de Camadas Convolucionais: Indica o número de camadas convolucionais na arquitetura da CNN. Mais camadas convolucionais permitem que a rede capture padrões mais complexos e hierárquicos, mas podem aumentar o custo computacional.

4. Quantidade de Camadas Densas: São as camadas totalmente conectadas após as camadas convolucionais. A quantidade de camadas densas influencia o processo de decisão da rede, combinando as características extraídas para fazer predições.

5. Quantidade de neurônios: Um número maior de neurônios aumenta a capacidade da rede de aprender representações mais detalhadas, mas pode levar a um risco maior de sobreajuste. O numero de neurônios implementado é o mesmo para cada camada densa atribuída a arquitetura.

6. Taxa de *Dropout*: *Dropout* é uma técnica de regularização que "desativa"aleatoriamente neurônios durante o treinamento para evitar que a rede dependa excessivamente de algumas conexões. A taxa de *dropout* especifica a proporção de neurônios que serão desativados.

7. Taxa de Aprendizado: Controla o tamanho dos ajustes que a rede faz aos pesos durante o treinamento. Uma taxa de aprendizado maior acelera o aprendizado, mas pode resultar em convergência instável, enquanto uma taxa muito baixa pode tornar o aprendizado lento.

8. Limiar de Decisão: Refere-se ao ponto de corte para decidir a classe de saída, especialmente em problemas de classificação binária. Ele define como as previsões probabilísticas são convertidas em decisões categóricas (ex: 0 ou 1).

Com base nas informações apresentadas, as tabelas que tratam das otimizações bayesianas e da implementação das redes CNN1D identificarão cada hiperparâmetro pelo número descrito anteriormente, facilitando a consulta.

o espaço de busca foi definido conforme a tabela IV.5.

Tabela IV.5: Espaço de Busca CNN1D

Hiperparâmetros	Faixa de busca
1	8 a 600
2	2 a 6
3	2 a 4
4	1 a 3
5	60 a 320
6	0.2 a 0.5
7	0.0001, 0.0003, 0.0006, 0.001, 0.003, 0.006, 0.01
8	0.5 a 0.9

As arquiteturas de CNN1D implementadas para a realização dos experimentos após a execução da otimização bayesiana, utilizando os dados oriundos do *smartwatch* esquerdo, para as abordagens de rótulos binário_1 e binário_2, estão expostas, respectivamente, nas tabelas IV.6 e IV.7.

Tabela IV.6: Arquiteturas CNN1D otimizadas para os dados do *smartwatch* esquerdo, rótulos: binário_1.

Cenário	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Sc1_acc_T	32	5	3	1	68	0.2	0.01	0.9
Sc1_acc_F	466	3	3	3	198	0.3	0.0003	0.6
Sc1_gyr_F	39	5	4	3	67	0.4	0.0003	0.5
Sc1_gyr_T	344	3	2	3	284	0.3	0.0006	0.8
Sc_2_acc_F	301	3	4	2	170	0.2	1	0.6
Sc_2_acc_T	210	5	3	2	213	0.1	0.0006	0.5
Sc_2_gyr_F	78	6	4	2	85	0.2	0.0006	0.7
Sc_2_gyr_T	73	4	3	3	230	0.3	1	0.7
Sc_3_F	136	5	3	2	182	0.2	0.0006	0.7
Sc_3_T	100	3	3	3	180	0.1	0.0003	0.6
Sc_4_F	52	3	2	3	78	0.4	0.0006	0.8
Sc_4_T	300	2	2	1	194	0.4	3	0.5

Tabela IV.7: Arquiteturas CNN1D otimizadas para os dados do *smartwatch* esquerdo, rótulos: binário_2.

Cenário	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Sc1_acc_T	330	5	3	3	114	0.2	0.0006	0.7
Sc1_acc_F	159	5	2	2	115	0.4	0.0006	0.9
Sc1_gyr_F	239	5	2	2	64	0.1	1	0.8
Sc1_gyr_T	220	6	4	1	74	0.2	1	0.8
Sc_2_acc_F	242	3	2	2	79	0.4	0.0003	0.5
Sc_2_acc_T	243	5	3	2	247	0.3	0.0006	0.9
Sc_2_gyr_F	73	4	3	3	230	0.3	1	0.7
Sc_2_gyr_T	27	3	2	1	61	0.4	1	0.7
Sc_3_F	57	2	3	3	217	0.3	1	0.5
Sc_3_T	8	6	2	1	298	0.2	1	0.8
Sc_4_F	214	4	2	2	157	0.1	0.0006	0.9
Sc_4_T	125	4	3	2	174	0.1	0.0003	0.6

As arquiteturas de CNN1D implementadas para a realização dos experimentos após a execução da otimização bayesiana, utilizando os dados oriundos do *smartwatch* direito, para as abordagens de rótulos binário_1 e binário_2, estão expostas, respectivamente, nas tabelas IV.8 e IV.9.

Tabela IV.8: Arquiteturas CNN1D otimizadas para os dados do *smartwatch* direito, rótulos: binário_1.

Cenário	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Sc1_acc_T	14	6	2	2	220	0.3	3	0.7
Sc1_acc_F	72	3	2	2	63	0.1	0.0006	0.8
Sc1_gyr_F	8	5	3	2	203	0.1	6	0.5
Sc1_gyr_T	30	5	3	2	238	0.2	1	0.6
Sc_2_acc_F	20	3	2	3	214	0.4	1	0.7
Sc_2_acc_T	41	3	4	2	100	0.2	0.0006	0.5
Sc_2_gyr_F	44	3	2	2	89	0.2	1	0.9
Sc_2_gyr_T	9	4	4	3	66	0.1	1	0.5
Sc_3_F	283	5	2	1	177	0.5	0.0003	0.6
Sc_3_T	57	6	4	3	250	0.4	0.0006	0.7
Sc_4_F	88	2	3	2	200	0.4	3	0.7
Sc_4_T	40	2	2	2	173	0.1	0.0006	0.9

Tabela IV.9: Arquiteturas CNN1D otimizadas para os dados do *smartwatch* direito, rótulos: binário_2.

Cenário	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Sc1_acc_T	253	6	3	2	206	0.2	0.0006	0.8
Sc1_acc_F	217	5	4	3	290	0.2	0.0006	0.5
Sc1_gyr_F	59	3	3	2	77	0.1	3	0.5
Sc1_gyr_T	299	3	3	3	102	0.2	0.0006	0.8
Sc_2_acc_F	160	4	2	2	163	0.2	0.0003	0.6
Sc_2_acc_T	177	3	3	3	78	0.2	0.0003	0.8
Sc_2_gyr_F	109	4	3	1	144	0.2	0.0006	0.7
Sc_2_gyr_T	15	6	2	2	92	0.5	1	0.6
Sc_3_F	550	5	2	3	92	0.2	1	0.5
Sc_3_T	33	2	3	3	67	0.4	6	0.6
Sc_4_F	88	6	2	3	64	0.5	0.0003	0.5
Sc_4_T	9	6	2	1	74	0.1	0.01	0.9

As arquiteturas de CNN1D implementadas para a realização dos experimentos após a execução da otimização bayesiana, utilizando os dados oriundos do *smartphone* localizado no peito, para as abordagens de rótulos binário_1 e binário_2, estão expostas, respectivamente, nas tabelas IV.10 e IV.11.

Tabela IV.10: Arquiteturas CNN1D otimizadas para os dados do *smartphone* peito, rótulos: binário_1.

Cenário	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Sc1_acc_T	37	3	4	2	107	0.2	3	0.6
Sc1_acc_F	303	3	3	2	64	0.2	0.0001	0.6
Sc1_gyr_F	83	2	2	3	62	0.5	1	0.5
Sc1_gyr_T	293	3	2	3	188	0.2	0.0001	0.7
Sc_2_acc_F	82	5	4	2	99	0.2	1	0.8
Sc_2_acc_T	100	4	4	3	89	0.1	0.0003	0.9
Sc_2_gyr_F	566	4	3	1	294	0.2	0.0003	0.5
Sc_2_gyr_T	256	6	2	3	137	0.4	0.0006	0.7
Sc_3_F	287	5	2	2	226	0.2	0.0003	0.7
Sc_3_T	8	5	3	1	255	0.2	0.0006	0.6
Sc_4_F	242	3	3	3	76	0.2	0.0006	0.7
Sc_4_T	16	3	3	2	134	0.1	3	0.8

Tabela IV.11: Arquiteturas CNN1D otimizadas para os dados do *smartphone* peito, rótulos: binário_2

Cenário	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Sc1_acc_T	12	4	4	1	76	0.1	3	0.7
Sc1_acc_F	185	5	2	1	314	0.1	0.0001	0.6
Sc1_gyr_F	151	5	2	3	103	0.3	0.0003	0.6
Sc1_gyr_T	8	4	2	3	66	0.5	3	0.8
Sc_2_acc_F	39	6	2	2	151	0.3	0.0006	0.5
Sc_2_acc_T	222	2	3	3	155	0.2	0.0001	0.6
Sc_2_gyr_F	40	2	3	2	72	0.1	0.0003	0.5
Sc_2_gyr_T	598	4	3	2	99	0.4	1	0.5
Sc_3_F	159	3	4	3	190	0.3	0.0003	0.6
Sc_3_T	241	3	2	2	105	0.5	0.0006	0.7
Sc_4_F	112	3	2	3	62	0.4	1	0.5
Sc_4_T	38	5	4	2	171	0.2	0.0001	0.5

A implementação das redes CNN1D abordada neste capítulo foi realizada com a otimização bayesiana dos hiperparâmetros, garantindo que cada cenário experimental fosse configurado com a melhor combinação de parâmetros. O processo incluiu a definição de um espaço de busca para os principais hiperparâmetros, como a quantidade de filtros, camadas convolucionais, neurônios nas camadas densas e a taxa de aprendizado, entre outros mencionados.

As tabelas apresentadas ao longo desta seção resumem as arquiteturas otimizadas para cada cenário experimental, tanto para o domínio do tempo quanto para o domínio da frequência. Todas as arquiteturas mencionadas foram implementadas e executadas, e os resultados obtidos podem ser conferidos no Capítulo V.

Capítulo V Experimentos

Este capítulo apresenta os resultados dos treinamentos realizados para cada cenário experimental, comparando diferentes domínios, posições e abordagens de rotulação, e culminando na definição da melhor arquitetura para o projeto Combatente do Futuro. O capítulo está organizado como segue. A Seção V.1 apresenta a introdução. Na Seção V.2 são apresentados os resultados obtidos com os dados coletados pelo *smartwatch* no braço esquerdo. Na Seção V.3, os resultados adquiridos com os dados do *smartwatch* no braço direito. Na Seção V.4, os resultados obtidos com os dados do *smartphone* no peito. Por fim, a Seção V.5 compara os resultados das três posições e indica a configuração ideal de rede neural e dados para o projeto Combatente do Futuro.

V.1 Introdução

As redes convolucionais 1D para cada cenário foram implementadas de acordo com as arquiteturas descritas em IV.6. Em cada cenário experimental, as redes neurais foram submetidas a 20 treinamentos, cada um com 25 épocas. Os 20 treinamentos para cada cenário experimental foram realizados com o objetivo de comparar o desempenho dos modelos, permitindo avaliar a variabilidade nos resultados obtidos em diferentes execuções. Essa abordagem é essencial para entender a influência da aleatoriedade presente no processo de treinamento, como a inicialização dos pesos e as trajetórias de otimização, que podem afetar o desempenho final da rede. A realização de múltiplas execuções permite uma análise mais robusta da estabilidade e da consistência do modelo em diversos cenários de treinamento.

Após a conclusão dos treinamentos das redes neurais, os conjuntos de dados de teste de cada cenário experimental, que foram mantidos isolados durante o processo de treinamento, foram utilizados para avaliar a capacidade de generalização das redes desenvolvidas. A aplicação desses conjuntos de dados de teste oferece uma avaliação imparcial e objetiva do desempenho dos modelos em dados não vistos previamente. Isso permite verificar a habilidade das redes em transferir o conhecimento adquirido durante o treinamento para realizar a classificação de quedas com precisão, além de avaliar sua capacidade em lidar com dados desconhecidos, simulando situações do mundo real.

A principal métrica utilizada para avaliar a previsão das redes neurais treinadas foi o MCC,

escolhida por proporcionar uma avaliação mais eficaz em conjuntos de dados desbalanceados, como é o caso do conjunto *IPqM-FALL*. O MCC foi empregado para determinar qual cenário experimental apresentou melhor desempenho. Além disso, utilizamos de forma complementar as métricas de Sensibilidade, Especificidade e Precisão, a fim de oferecer uma visão abrangente dos resultados obtidos. A escolha dessas métricas visa não apenas a avaliação completa dos modelos, mas também a padronização de parâmetros que possam ser comparados em futuros trabalhos derivados e aprimoramentos deste estudo.

Além das métricas citadas, foram utilizados testes estatísticos para comparar o desempenho das redes neurais treinadas. O primeiro passo no processo foi realizar o teste de normalidade de *Shapiro-Wilk*, que verifica se os dados seguem uma distribuição normal [Hanusz et al., 2016]. Os resultados indicaram que a distribuição dos dados obtidos a partir dos desempenhos das redes CNN1D não era compatível com uma distribuição normal. Como alguns testes estatísticos tradicionais, como o teste t, assumem normalidade dos dados, essa constatação indicou a necessidade de empregar testes não paramétricos para garantir a validade das comparações.

Os testes não paramétricos são adequados quando os dados não seguem uma distribuição normal, pois eles não dependem de suposições sobre a forma da distribuição dos dados. Nesse contexto, dois testes foram selecionados: o teste de *Mann-Whitney U* e o teste de *Kruskal-Wallis*, ambos voltados para a comparação de amostras independentes, mas com finalidades diferentes.

O teste de *Mann-Whitney U* é utilizado para comparar duas amostras independentes, verificando se elas provêm da mesma distribuição. Este teste mede se uma amostra tende a ter valores maiores ou menores do que a outra, sendo uma alternativa não paramétrica ao teste t para amostras independentes [McKnight and Najab, 2010a].

Já o teste de *Kruskal-Wallis* é uma extensão do teste de *Mann-Whitney U* e é utilizado quando se deseja comparar três ou mais amostras independentes. Ele testa a hipótese de que pelo menos uma das amostras difere das outras em termos de localização central (mediana), sem assumir que os dados sigam uma distribuição específica. Este teste é uma alternativa não paramétrica à análise de variância (ANOVA) e é apropriado para verificar se os desempenhos das redes variam de forma estatisticamente significativa. [McKnight and Najab, 2010b].

Os testes de *Mann-Whitney U* e *Kruskal-Wallis H* são ferramentas estatísticas que utilizam hipóteses nulas, estatísticas U e H, e p-valores para diferenciar amostras. O p-valor indica a significância das diferenças observadas, e valores abaixo de 0.05 sugerem a rejeição da hipótese nula, apontando para uma diferença estatisticamente significativa entre as amostras. A estatística U mede a frequência com que os valores de uma amostra são maiores do que os valores da outra, enquanto a estatística H é calculada a partir das ordens dos dados nas amostras. Esses elementos são essenciais para identificar divergências entre amostras em contextos onde a normalidade dos dados não

pode ser presumida. Esses testes forneceram uma base para a comparação dos desempenhos das redes neurais, independentemente da falta de normalidade nos dados, garantindo que as conclusões estatísticas fossem válidas.

Em relação aos resultados de uma matriz de confusão, no contexto do projeto do combatente do futuro, o ideal é que a presença de Falsos Negativos seja nula. Isso se justifica pelo pior cenário possível, no qual um combatente cai durante uma operação tática e o algoritmo de detecção não consegue identificar a queda. Nesse sentido, a presença de Falsos Positivos não é tão preocupante inicialmente, uma vez que se pressupõe que o algoritmo de classificação a ser desenvolvido para identificar se uma queda é uma baixa ou uma queda operacional será capaz de filtrar tais indicações falsas. Portanto, uma Rede Neural que seja capaz de discriminar com maior precisão os Falsos Negativos é a ideal para o projeto.

Para realizar uma análise mais detalhada, neste capítulo os dados foram divididos em três grupos distintos: cenários com o *smartwatch* no pulso esquerdo, cenários com o *smartwatch* no pulso direito e cenários com o *smartphone* posicionado no peito. A análise foi conduzida separadamente para cada um desses grupos. Após a avaliação individual de cada grupo, foi realizada uma comparação entre as três posições (direita, esquerda e peito), a fim de identificar o melhor domínio (frequência ou tempo) e as variáveis inerciais mais adequadas para a previsão de quedas, determinando qual configuração oferece o melhor desempenho em termos de detecção de quedas de soldados.

V.2 Resultados - *Smartwatch* esquerdo

Esta seção está dividida em 3 seções, onde a Seção V.2.1 apresenta os resultados para o *smartwatch* do lado esquerdo com os rótulos binário_1, a Seção V.2.2 apresenta os resultados para o *smartwatch* do lado esquerdo com os rótulos binário_2, e a seção V.2.3 apresenta a comparação entre as duas abordagens.

V.2.1 Resultados - binário_1

A Figura V.7 apresenta resultado do teste de *Kruskal-Wallis* para a métrica MCC para os 20 treinamentos executados para os 12 cenários experimentais, utilizando os dados coletados com o *smartwatch* instalado no pulso esquerdo dos militares voluntários para a abordagem de rótulo binário_1 .

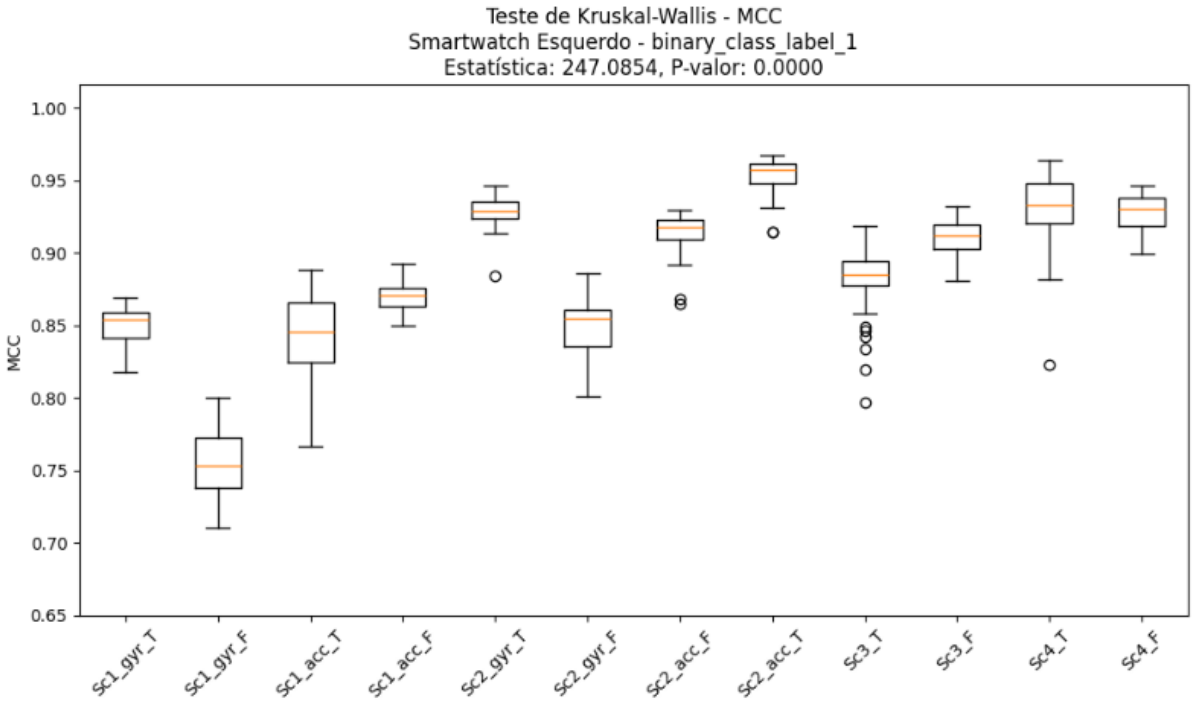


Figura V.1: Teste de *Kruskal-Wallis*, lado esquerdo, binário_1.

O p-valor de 0.0000 e a estatística H de 247.0854 rejeitam a hipótese nula e indicam que o desempenho de cada cenário de treinamento é estatisticamente diferente, o que nos permite, dentro do ambiente do teste de *Kruskal*, identificar o treinamento que apresentou o melhor desempenho. Podemos verificar também que a dispersão dos cenários, em geral, fica entre um MCC de 0.70 e 0.96.

O cenário de melhor desempenho em termos de MCC é o Sc2_acc_T, com uma mediana superior a 0.95 e baixa dispersão, o que demonstra alta consistência nos treinamentos. Outros cenários, como o Sc4_T, também apresentam bons resultados, com MCC elevados, mas o destaque do Sc2_acc_T está na menor variabilidade dos dados, indicando maior confiabilidade no desempenho do modelo. Em contraste, cenários como Sc1_gyr_F e Sc1_acc_T mostram maior dispersão, sugerindo variações significativas nos resultados entre os diferentes treinamentos. O cenário Sc1_gyr_F apresenta o pior desempenho, com uma mediana de MCC inferior a 0.80 e alta dispersão nos valores.

A Tabela V.1 apresenta os resultados dos melhores e piores desempenhos alcançados pelos treinamentos das redes neurais CNN1D para os 12 cenários.

Tabela V.1: Resultados do desempenho dos Treinamento para cada cenário, lado esquerdo, binário_1.

Treinamentos com Melhor Desempenho								
Cenário	MCC	SE	ES	PR	TP	TN	FP	FN
Sc1_acc_F	0.8929	0.9905	0.8805	0.9821	1043	140	19	10
Sc1_acc_T	0.8886	0.9791	0.9371	0.9904	1031	149	10	22
Sc1_gyr_F	0.8001	0.9924	0.7233	0.9596	1045	115	44	8
Sc1_gyr_T	0.8690	0.9839	0.8805	0.9820	1036	140	19	17
Sc_2_acc_F	0.9299	0.9962	0.9057	0.9859	1049	144	15	4
Sc_2_acc_T	0.9677	0.9943	0.9811	0.9971	1047	156	3	6
Sc_2_gyr_F	0.8862	0.9820	0.9182	0.9876	1034	146	13	19
Sc_2_gyr_T	0.9466	0.9905	0.9686	0.9952	1043	154	5	10
Sc_3_F	0.9323	0.9886	0.9560	0.9933	1041	152	7	12
Sc_3_T	0.9186	0.9858	0.9497	0.9924	1038	151	8	15
Sc_4_F	0.9466	0.9905	0.9686	0.9952	1043	154	5	10
Sc_4_T	0.9640	0.9943	0.9748	0.9962	1047	155	4	6
Treinamentos com Pior Desempenho								
Cenário	MCC	SE	ES	PR	TP	TN	FP	FN
Sc1_acc_F	0.8499	0.9896	0.8176	0.9729	1042	130	29	11
Sc1_acc_T	0.0	0.0	1.0000	–	0	159	0	1053
Sc1_gyr_F	0.7102	0.9981	0.5535	0.9367	1051	88	71	2
Sc1_gyr_T	0.8179	0.9668	0.8868	0.9826	1018	141	18	35
Sc_2_acc_F	0.8646	0.9991	0.7799	0.9678	1052	124	35	1
Sc_2_acc_T	0.9142	0.9810	0.9686	0.9952	1033	154	5	20
Sc_2_gyr_F	0.8007	0.9829	0.7799	0.9673	1035	124	35	18
Sc_2_gyr_T	0.8843	0.9753	0.9497	0.9923	1027	151	8	26
Sc_3_F	0.8807	0.9791	0.9245	0.9885	1031	147	12	22
Sc_3_T	0.7966	0.9421	0.9560	0.9930	992	152	7	61
Sc_4_F	0.8995	0.9953	0.8616	0.9794	1048	137	22	5
Sc_4_T	0.8228	0.9810	0.8239	0.9736	1033	131	28	20

- **Análise dos treinamentos com melhor desempenho**: Na Tabela V.1, observa-se que os cenários com menor número de falsos negativos, como Sc_2_acc_F (4 falsos negativos) e Sc_2_acc_T (6 falsos negativos), também apresentam alta sensibilidade (0.9962 e 0.9943, respectivamente) e precisão (0.9859 e 0.9971, respectivamente). Esses cenários são particularmente bons, pois além de manterem alta capacidade de identificar corretamente os positivos, têm uma taxa de erro de falsos negativos muito baixa, o que é desejável para evitar a falha na detecção de eventos críticos.

  Outro destaque é o cenário Sc_4_T, com apenas 6 falsos negativos, sensibilidade de 0.9943 e especificidade de 0.9748. A combinação desses fatores indica um ótimo equilíbrio entre a detecção correta de positivos e a minimização de falsos positivos e negativos, o que se reflete também na alta precisão (0.9961).

- **Análise dos treinamentos com pior desempenho**: Na Tabela V.1, podemos observar os piores resultados de desempenho dos modelos. O cenário Sc1_acc_T se destaca negativamente

por apresentar 1053 falsos negativos, com uma sensibilidade de 0.0, indicando um total fracasso em detectar os eventos positivos. Neste cenário o pior resultado é devido há um treinamento que não convergiu.

Os cenários como Sc1_gyr_F e Sc_2_gyr_T também demonstram desempenhos inferiores, com altas taxas de falsos negativos (71 e 26, respectivamente) e menores precisões. Embora ainda tenham sensibilidades razoáveis (0.9981 e 0.9753), o elevado número de falsos negativos compromete a eficácia desses modelos.

A análise da tabela indica que os cenários no domínio do tempo, especialmente o Sc_2_acc_T, são os mais eficazes, apresentando uma combinação de alta sensibilidade, precisão e baixa ocorrência de falsos negativos. A Figura V.2 apresenta o teste de *Mann-Whitney* comparando os domínios do tempo e da frequência, onde um *boxplot* representa o MCC de todos os treinamentos para os cenários no domínio do tempo, e outro *boxplot* para todos os cenários de treinamento no domínio da frequência.

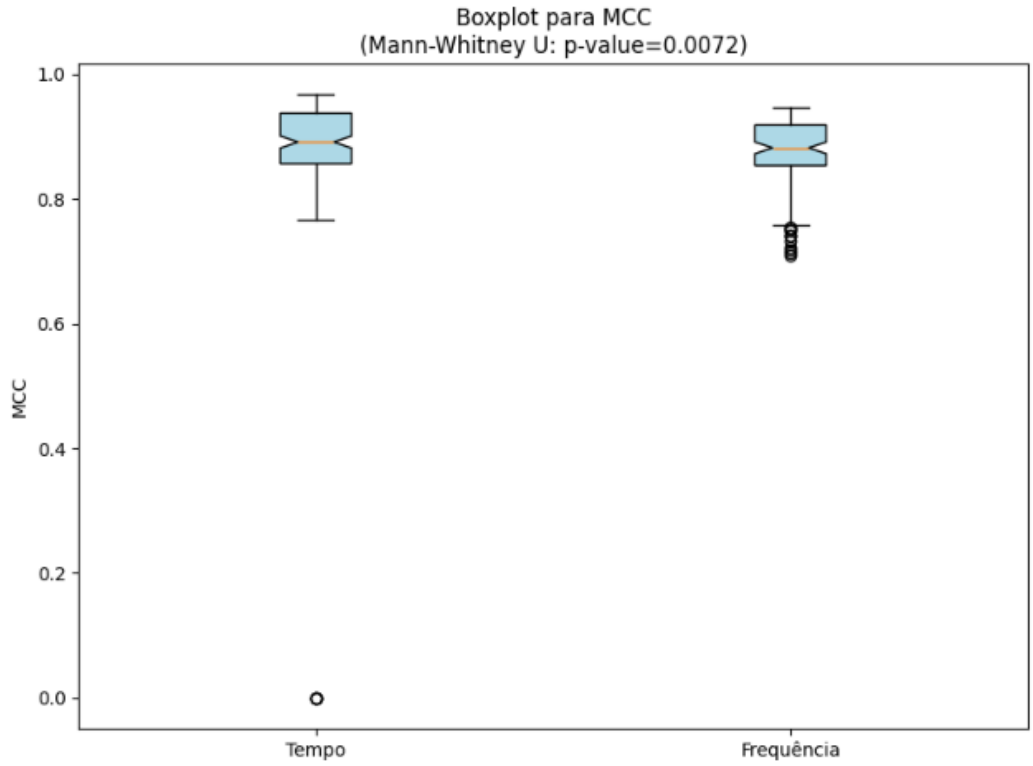


Figura V.2: Teste de *Mann-Whitney* para os domínios da frequência e do tempo, lado esquerdo, binário_1.

,

No teste de *Mann-Whitney*, apresentado na Figura V.2, o p-valor de 0.0072 indica uma diferença estatisticamente significativa entre os dois grupos, ou seja, os desempenhos obtidos nos treinamentos com dados no domínio do tempo e da frequência são distintos. Observando os *boxplots*, percebemos que ambos os grupos apresentam valores elevados de MCC, o que indica bom desempenho geral. No

entanto, o domínio do tempo apresenta menor dispersão nos valores, sugerindo maior consistência nos resultados dos treinamentos, enquanto o domínio da frequência apresenta maior variabilidade, incluindo alguns *outliers*. Isso pode sugerir que, embora o desempenho médio seja similar, os resultados no domínio da frequência são mais variáveis, com alguns casos apresentando desempenho inferior.

Com a análise realizada para o rótulo binário_1, que corresponde à abordagem que considera as atividades de transição para tiro deitado (OM_6 a OM_8) como quedas, o cenário que apresentou o melhor desempenho foi o Sc_2_acc_T. Isso indica que o melhor conjunto de variáveis inerciais a ser utilizado são as componentes x, y e z da aceleração linear. A análise também demonstrou que os dados no domínio do tempo resultaram em um melhor desempenho na detecção de quedas.

V.2.2 Resultados - binário_2

A Figura V.3 apresenta resultado do teste de *Kruskal-Wallis* para a métrica MCC para os 20 treinamentos executados para os 12 cenários experimentais, utilizando os dados coletados com o *smartwatch* instalado no pulso esquerdo dos militares voluntários para a abordagem de rótulo binário_2 .

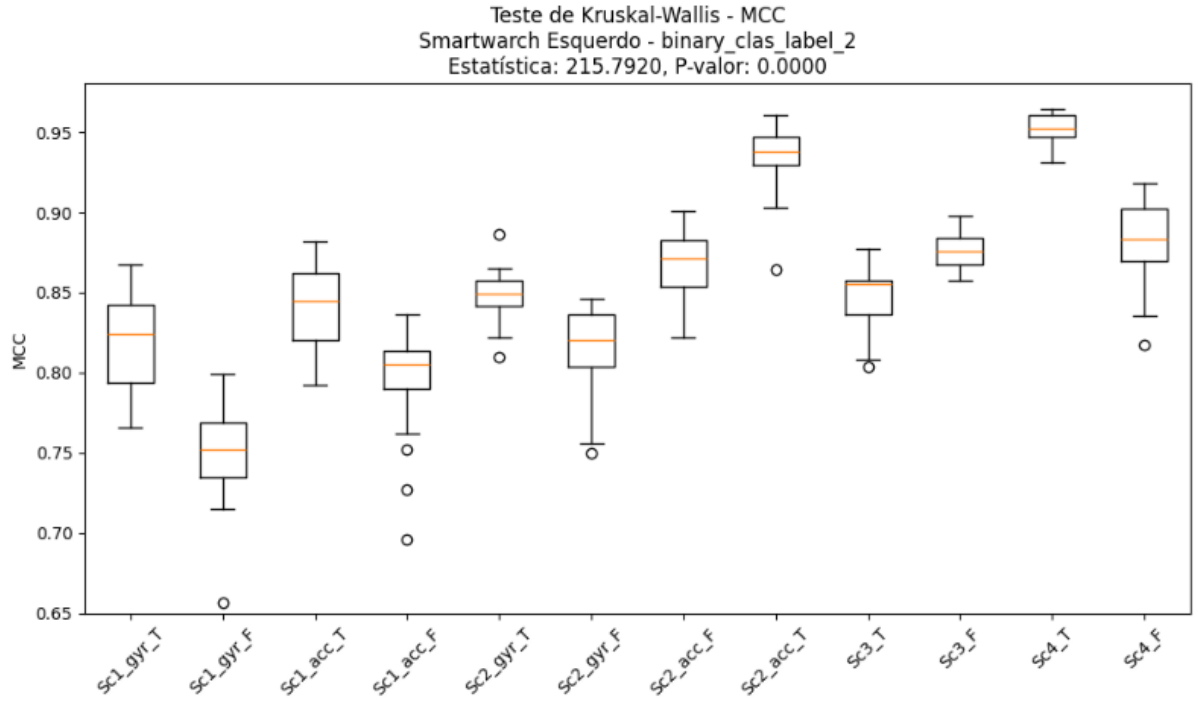


Figura V.3: Teste de *Kruskal-Wallis*, lado esquerdo, binário_2.

O p-valor de 0.0000 e a estatística H de 215.7920 rejeitam a hipótese nula e indicam que o desempenho de cada cenário de treinamento é estatisticamente diferente, o que nos permite, dentro do ambiente do teste de *Kruskal-Wallis*, identificar o treinamento que apresentou o melhor desempenho.

Podemos verificar também que a dispersão dos cenários, em geral, fica entre um MCC de 0.70 e 0.95. O cenário de melhor desempenho em termos de MCC é o Sc4_T, com uma mediana próxima de 0.95 e baixa dispersão, o que demonstra alta consistência nos treinamentos. Outros cenários, como o Sc2_acc_T, também apresentam bons resultados, com MCC elevados e consistência similar. O destaque do Sc4_T está na sua menor variabilidade dos dados, o que indica maior confiabilidade no desempenho do modelo. Em contraste, cenários como o Sc1_gyr_F mostram maior dispersão, sugerindo variações significativas nos resultados entre os diferentes treinamentos.

A Tabela V.2 apresenta os resultados para os melhores e piores desempenhos alcançados pelos modelos de redes neurais CNN1D para os 12 cenários.

Tabela V.2: Resultados do desempenho dos Treinamento para cada cenário, lado esquerdo, binário_2.

Treinamentos com Melhor Desempenho								
Cenário	MCC	SE	ES	PR	TP	TN	FN	FP
Sc1_acc_F	0.8366	0.9797	0.8740	0.9852	1063	111	22	16.0
Sc1_acc_T	0.8817	0.9871	0.8976	0.9880	1071	114	14	13.0
Sc1_gyr_F	0.7991	0.9779	0.8268	0.9797	1061	105	24	22.0
Sc1_gyr_T	0.8672	0.9867	0.8616	0.9793	1039	137	14	22.0
Sc2_acc_F	0.9008	0.9871	0.9291	0.9917	1071	118	14	9.0
Sc2_acc_T	0.9606	0.9954	0.9685	0.9963	1080	123	5	4.0
Sc2_gyr_F	0.8465	0.9797	0.8898	0.9870	1063	113	22	14.0
Sc2_gyr_T	0.8865	0.9871	0.9055	0.9889	1071	115	14	12.0
Sc3_F	0.8979	0.9908	0.8976	0.9881	1075	114	10	13.0
Sc3_T	0.8770	0.9825	0.9213	0.9907	1066	117	19	10.0
Sc4_F	0.9181	0.9889	0.9449	0.9935	1073	120	12	7.0
Sc4_T	0.9648	0.9963	0.9685	0.9963	1081	123	4	4.0
Treinamentos com Pior Desempenho								
Cenário	MCC	SE	ES	PR	TP	TN	FN	FP
Sc1_acc_F	0.6957	0.9207	0.9291	0.9911	999	118	86	9.0
Sc1_acc_T	0.7920	0.9677	0.8740	0.9850	1050	111	35	16.0
Sc1_gyr_F	0.6567	0.8949	0.9528	0.9939	971	121	114	6.0
Sc1_gyr_T	0.7662	0.9945	0.6614	0.9617	1079	84	6	43.0
Sc_2_acc_F	0.8216	0.9797	0.8504	0.9824	1063	108	22	19.0
Sc_2_acc_T	0.8644	0.9714	0.9685	0.9962	1054	123	31	4.0
Sc_2_gyr_F	0.7502	0.9945	0.6378	0.9591	1079	81	6	46.0
Sc_2_gyr_T	0.8097	0.9594	0.9449	0.9933	1041	120	44	7.0
Sc_3_F	0.8578	0.9935	0.8110	0.9782	1078	103	7	24.0
Sc_3_T	0.8040	0.9576	0.9449	0.9933	1039	120	46	7.0
Sc_4_F	0.8174	0.9963	0.7244	0.9686	1081	92	4	35.0
Sc_4_T	0.9307	0.9908	0.9528	0.9944	1075	121	10	6.0

- **Análise dos treinamentos com melhor desempenho**: Na Tabela V.1, destacam-se os cenários Sc2_acc_T e Sc4_T, que apresentam o menor número de falsos negativos (5 e 4, respectivamente), juntamente com alta sensibilidade (0.9954 e 0.9963) e precisão elevada (0.9963 em ambos os casos). Esses cenários mostram excelente capacidade de detectar queda,

minimizando falhas na detecção.

- **Análise dos treinamentos com pior desempenho**: Na Tabela V.1, observa-se que o cenário Sc1_gyr_F apresenta o maior número de falsos negativos entre os cenários de menor desempenho, com 114 falsos negativos e uma sensibilidade de 0.8949. Embora a sensibilidade seja relativamente alta, o número expressivo de falsos negativos compromete a confiabilidade do modelo em situações críticas, onde a falha na detecção deve ser minimizada.

A análise da tabela indica que os cenários no domínio do tempo, especialmente o Sc4_T, são os mais eficazes, apresentando uma combinação de alta sensibilidade, precisão e baixa ocorrência de falsos negativos. A Figura V.4 apresenta o teste de *Mann-Whitney* comparando os domínios do tempo e da frequência, onde um *boxplot* representa o MCC de todos os treinamentos para os cenários no domínio do tempo, e outro *boxplot* para todos os cenários de treinamento no domínio da frequência.

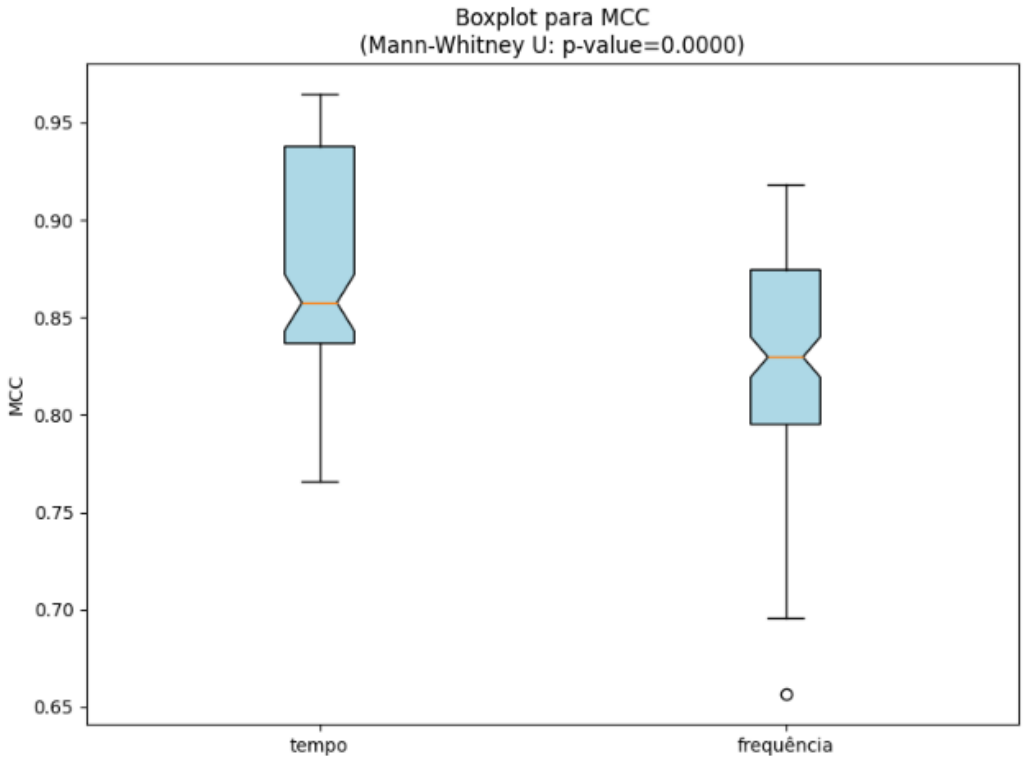


Figura V.4: Teste de *Mann-Whitney* para os domínios da frequência e do tempo, lado esquerdo, binário_2.

,

O *boxplot* da Figura V.4 indica que o domínio do tempo apresenta uma mediana de MCC mais alta em comparação ao domínio da frequência, sugerindo que, em média, os modelos treinados com dados do domínio do tempo obtêm desempenho superior. A amplitude interquartil no domínio do tempo é maior, indicando uma variação mais ampla nos valores de MCC e sugerindo maior

instabilidade ou variabilidade de desempenho. Em contraste, o domínio da frequência mostra menor variabilidade nos valores de MCC, caracterizando uma distribuição mais concentrada em torno de sua mediana, embora um *outlier* indique um caso de desempenho particularmente baixo.

O teste de *Mann-Whitney* foi aplicado para avaliar a significância estatística das diferenças observadas entre os dois domínios. O p-valor igual a 0.0000 indica que a diferença de desempenho entre os domínios é estatisticamente significativa, permitindo rejeitar a hipótese nula de que as distribuições do MCC são iguais. Assim, conclui-se que o domínio do tempo oferece uma vantagem no desempenho dos treinamentos, apesar de sua maior variabilidade.

Com a análise realizada para o rótulo binário_2, que corresponde à abordagem que considera as atividades de transição para tiro deitado (OM_6 a OM_8) como não quedas, o cenário que apresentou o melhor desempenho foi o Sc_4_T. Isso indica que o melhor conjunto de variáveis inerciais a ser utilizado são as componentes x, y e z da aceleração linear combinados com os componentes x, y e z da aceleração angular. A análise também demonstrou que os dados no domínio do tempo resultaram em um melhor desempenho na detecção de quedas.

V.2.3 Comparação entre as abordagens de rotulação - lado esquerdo.

A Figura V.5 apresenta o teste de *Mann-Whitney* comparando as abordagens binário_1 e binário_2. O gráfico exibe dois *boxplots*: um representa o valor do MCC de todos os treinamentos para a abordagem binário_1, e o outro mostra o MCC para todos os treinamentos da abordagem binário_2.

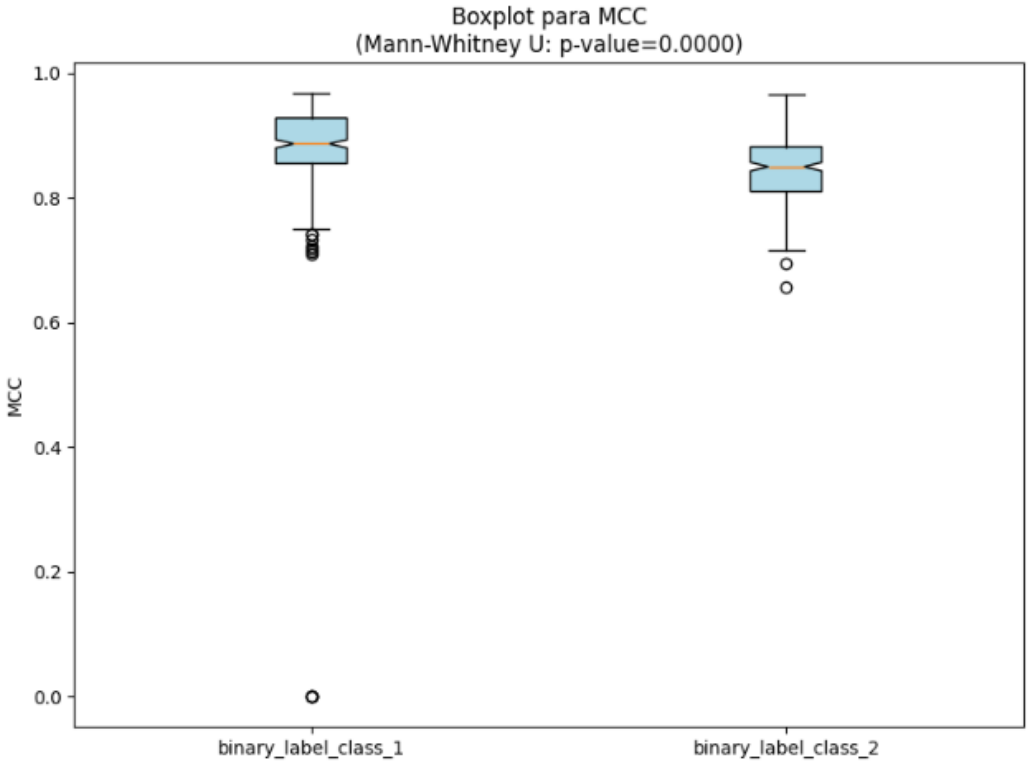


Figura V.5: Teste de *Mann-Whitney* para as abordagens de rotulação, lado esquerdo.

Com um valor de p igual a 0.0000, indicando uma diferença significativa entre as distribuições de MCC das duas abordagens, há evidências de que uma delas apresenta desempenho superior. Os *boxplots* mostram que a mediana dos valores de MCC em ambas as abordagens está próxima de 0.8, sugerindo desempenho consistente. No entanto, algumas diferenças se destacam: na abordagem binário_1, há vários *outliers* abaixo de 0.8, incluindo um valor próximo de zero, sinalizando que, apesar de a maioria dos treinamentos ter sido satisfatória, alguns apresentaram desempenho bem abaixo do esperado. Na abordagem binário_2, também se observam *outliers*, mas em menor quantidade e sem valores tão extremos.

A análise do teste de *Mann-Whitney* confirma que as diferenças observadas entre as distribuições de MCC das duas abordagens são estatisticamente significativas. Conclui-se, assim, que a abordagem binário_1 apresenta uma leve vantagem em termos de desempenho médio em relação à binário_2.

A Figura V.6 apresenta a comparação dos cenários experimentais com melhor desempenho para ambas as abordagens de rotulação. O cenário Sc_2_acc representa o melhor desempenho para a abordagem binário_1, enquanto o cenário Sc4_T destaca-se na abordagem binário_2.

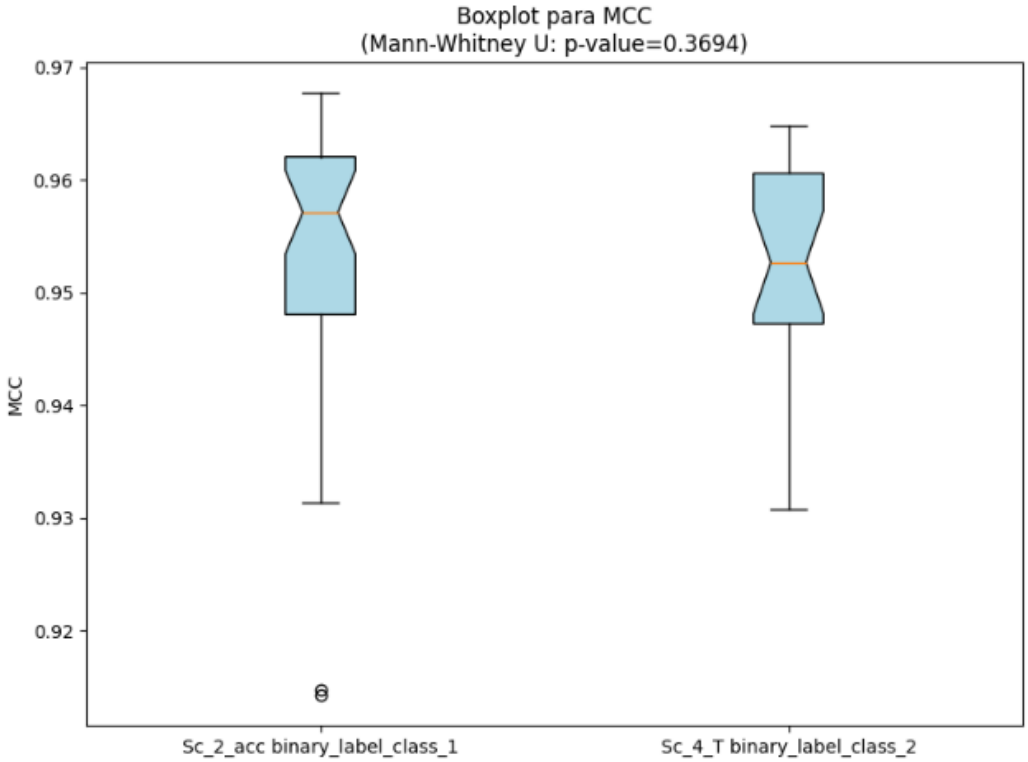


Figura V.6: Teste de *Mann-Whitney* para os melhores cenários experimentais para ambas abordagens de rotulação, lado esquerdo, *binary_class_label_2*.

Apesar de o teste de *Mann-Whitney U* (Figura V.6 indicar uma ausência de diferença estatisticamente significativa entre os cenários (p-valor = 0.3694), observamos que o cenário Sc_2_acc possui uma mediana ligeiramente superior em comparação ao cenário Sc_4_T. Assim, com base na mediana, o cenário Sc_2_acc pode ser considerado o de melhor desempenho entre os dois.

Com base nos resultados apresentados, conclui-se que a abordagem binário_1 apresentou um desempenho ligeiramente superior em termos de valor mediano de MCC, especialmente no cenário Sc_2_acc, que teve o melhor desempenho entre os experimentos comparados. Apesar de ambas as abordagens terem medianas de MCC próximas, a análise de *outliers* e o teste de *Mann-Whitney* mostram que binário_1 possui uma menor variabilidade nos valores de MCC, mesmo considerando que alguns *outliers* alcançam valores abaixo de 0.8. O cenário Sc_2_acc, embora sem diferença estatisticamente significativa em relação ao Sc_4_T da abordagem binário_2, demonstra um desempenho consistentemente superior na comparação direta. Assim, a abordagem binário_1 que corresponde à abordagem que considera as atividades de transição para tiro deitado (OM_6 a OM_8) como quedas, é a mais indicada para o lado esquerdo, com base no desempenho médio e estabilidade dos valores de MCC.

V.3 Resultados - *Smartwatch* direito

Esta seção está dividida em 3 seções, onde a seção V.3.1 apresenta os resultados para o *smartwatch* do lado direito com os rótulos binário_1, a seção V.3.2 apresenta os resultados para o *smartwatch* do lado direito com os rótulos binário_2, e a seção V.3.3 apresenta a comparação entre as duas abordagens.

V.3.1 Resultados - binário_1

A Figura V.7 apresenta resultado do teste de *Kruskal-Wallis* para a métrica MCC para os 20 treinamentos executados para os 12 cenários experimentais, utilizando os dados coletados com o *smartwatch* instalado no pulso direito dos militares voluntários para a abordagem de rótulo binário_1 .

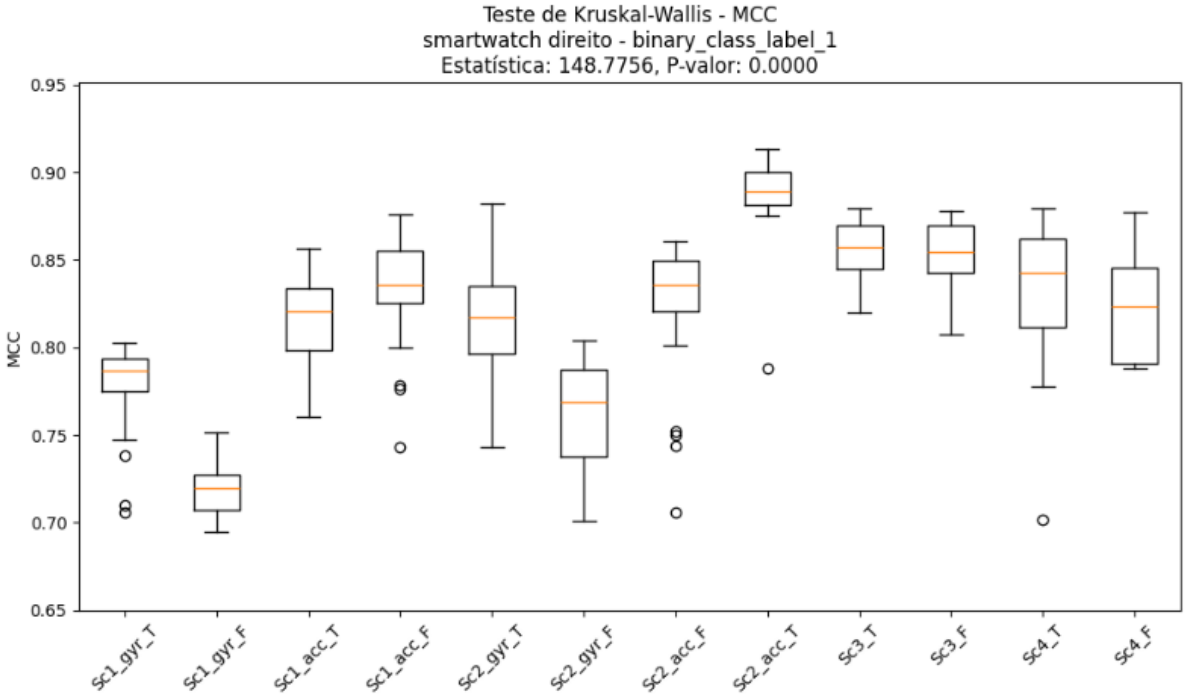


Figura V.7: Teste de *Kruskal-Wallis*, lado direito, binário_1.

O teste de *Kruskal-Wallis* apresentou uma estatística H de 148.7756 e um p-valor de 0.0000, indicando diferenças significativas entre os cenários experimentais e variações relevantes de desempenho entre eles. A análise dos resultados mostra uma diferença notável nos valores de MCC, onde a dispersão dos cenários se concentra entre 0.68 e 0.92. O cenário Sc2_acc_T destacou-se de forma isolada como o melhor, com medianas de MCC próximas a 0.90 e baixa variação, indicando que a maioria dos valores de MCC está concentrada em uma faixa elevada, refletindo um desempenho consistente.

Em contraste, os cenários Sc1_gyr_F e Sc2_gyr_F apresentaram os desempenhos mais baixos,

com medianas de MCC de aproximadamente 0.73 e 0.75, respectivamente. Ambos exibem alta dispersão nos valores de MCC, indicando significativa variabilidade entre os treinamentos e um desempenho inferior em relação aos demais cenários.

A Tabela V.3 resume os resultados dos treinamentos das redes neurais CNN1D para os 12 cenários, destacando os melhores e piores desempenhos alcançados.

Tabela V.3: Resultados do desempenho dos Treinamento para cada cenário, lado direito, binário_1.

Treinamentos com Melhor Desempenho								
Cenário	MCC	SE	ES	PR	TP	TN	FN	FP
Sc1_acc_F	0.8760	0.9817	0.9067	0.9865	1020	136	19	14.0
Sc1_acc_T	0.8565	0.9759	0.9067	0.9864	1014	136	25	14.0
Sc1_gyr_F	0.7514	0.9759	0.7467	0.9639	1014	112	25	38.0
Sc1_gyr_T	0.8025	0.9894	0.7467	0.9644	1028	112	11	38.0
Sc_2_acc_F	0.8608	0.9952	0.8000	0.9718	1034	120	5	30.0
Sc_2_acc_T	0.9131	0.9875	0.9333	0.9903	1026	140	13	10.0
Sc_2_gyr_F	0.8041	0.9865	0.7667	0.9670	1025	115	14	35.0
Sc_2_gyr_T	0.8822	0.9923	0.8533	0.9791	1031	128	8	22.0
Sc_3_F	0.8778	0.9798	0.9200	0.9883	1018	138	21	12.0
Sc_3_T	0.8794	0.9827	0.9067	0.9865	1021	136	18	14.0
Sc_4_F	0.8774	0.9981	0.8067	0.9728	1037	121	2	29.0
Sc_4_T	0.8794	0.9827	0.9067	0.9865	1021	136	18	14.0
Treinamentos com Pior Desempenho								
Cenário	MCC	SE	ES	PR	TP	TN	FN	FP
Sc1_acc_F	0.7431	0.9240	0.9467	0.9917	960	142	79	8.0
Sc1_acc_T	0.7607	0.9798	0.7400	0.9631	1018	111	21	39.0
Sc1_gyr_F	0.6948	0.9731	0.6800	0.9547	1011	102	28	48.0
Sc1_gyr_T	0.7059	0.9913	0.5933	0.9441	1030	89	9	61.0
Sc_2_acc_F	0.7059	0.9981	0.5467	0.9385	1037	82	2	68.0
Sc_2_acc_T	0.7881	0.9981	0.6667	0.9540	1037	100	2	50.0
Sc_2_gyr_F	0.7011	0.9913	0.5867	0.9432	1030	88	9	62.0
Sc_2_gyr_T	0.7431	0.9923	0.6400	0.9502	1031	96	8	54.0
Sc_3_F	0.1529	1.0000	0.0267	0.8768	1039	4	0	146.0
Sc_3_T	0.8200	0.9827	0.8133	0.9733	1021	122	18	28.0
Sc_4_F	0.2540	0.9875	0.1400	0.8883	1026	21	13	129.0
Sc_4_T	0.7014	0.9105	0.9267	0.9885	946	139	93	11.0

- **Análise dos treinamentos com melhor desempenho**: O cenário Sc_2_acc_T se destaca com uma sensibilidade de 0.9875 e especificidade de 0.9333, o que reflete a capacidade deste cenário de identificar corretamente as quedas dos soldados enquanto minimiza a ocorrência de falsos positivos (10) e falsos negativos (13). A precisão de 0.9903 e a acurácia de 0.9807 também reforçam sua eficácia, sendo um dos cenários mais equilibrados. Outro destaque é o cenário Sc_4_F, com uma sensibilidade de 0.9981 e poucos falsos negativos (2), indicando sua capacidade na detecção de quedas.

- **Análise dos treinamentos com pior desempenho**: Os cenários Sc1_gyr_T e Sc_3_F. são os que apresentam maior número de falsos negativos e baixas taxas de especificidade. O

cenário Sc_3_F, por exemplo, possui uma especificidade de (0.0267) taxa de falsos positivos (146), indicando uma não convergência do treinamento, o que torna o cenário inadequado.

A análise da tabela revela que os cenários no domínio do tempo, com destaque para o Sc_2_acc_T, são os mais eficazes. Na Figura V.8, o teste de *Mann-Whitney* compara os domínios do tempo e da frequência. Nela, um *boxplot* representa o MCC de todos os treinamentos para os cenários no domínio do tempo, enquanto outro *boxplot* ilustra os cenários de treinamento no domínio da frequência.

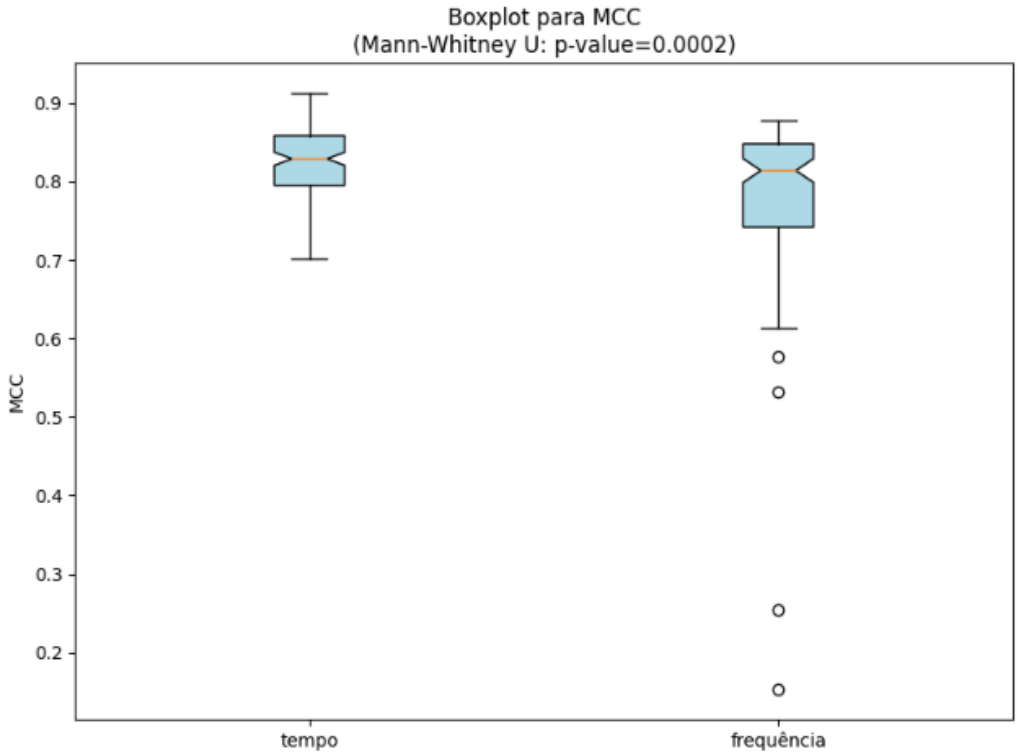


Figura V.8: Teste de *Mann-Whitney* para os domínios do tempo e da frequência, lado direito, binário_1.

O p-valor de 0.0002 indica uma diferença significativa entre os desempenhos nos dois domínios. No domínio do tempo, observa-se uma mediana de MCC superior ao domínio da frequência, o que sugere que o treinamento da rede neural com dados temporais resulta em um desempenho médio mais alto. A variação dos valores de MCC nesse domínio é relativamente pequena, refletindo maior consistência nos resultados. Esse comportamento é evidente na distribuição compacta dos valores no *boxplot*, com poucos *outliers*, o que indica que o modelo conseguiu generalizar bem em quase todos os treinamentos realizados com dados temporais.

Em contraste, o domínio da frequência apresentou uma mediana de MCC inferior e uma distribuição mais dispersa, com vários *outliers* em valores baixos. Essa variabilidade maior sugere que o desempenho dos cenários treinados com dados no domínio da frequência é menos consistente, com experimentos que exibem desempenho consideravelmente abaixo da média. Esses *outliers* indicam

que, em algumas instâncias, o modelo teve dificuldades para extrair padrões úteis a partir dos dados no domínio da frequência, resultando em valores de MCC significativamente mais baixos.

Assim, conclui-se que o uso de dados no domínio do tempo é o mais eficaz para o treinamento de redes para dados obtidos para o pulso direito.

Com a análise realizada para o rótulo binário_1, que corresponde à abordagem que considera as atividades de transição para tiro deitado (OM_6 a OM_8) como quedas, o cenário que apresentou o melhor desempenho foi o Sc_2_acc_T. Isso indica que o melhor conjunto de variáveis inerciais a ser utilizado são as componentes x, y e z da aceleração linear. A análise também demonstrou que os dados no domínio do tempo resultaram em uma melhor desempenho preditivo na detecção de quedas.

V.3.2 Resultados - binário_2

A Figura V.9 apresenta resultado do teste de *Kruskal-Wallis* para a métrica MCC para os 20 treinamentos executados para os 12 cenários experimentais, utilizando os dados coletados com o *smartwatch* instalado no pulso direito dos militares voluntários para a abordagem de rótulo binário_2 .

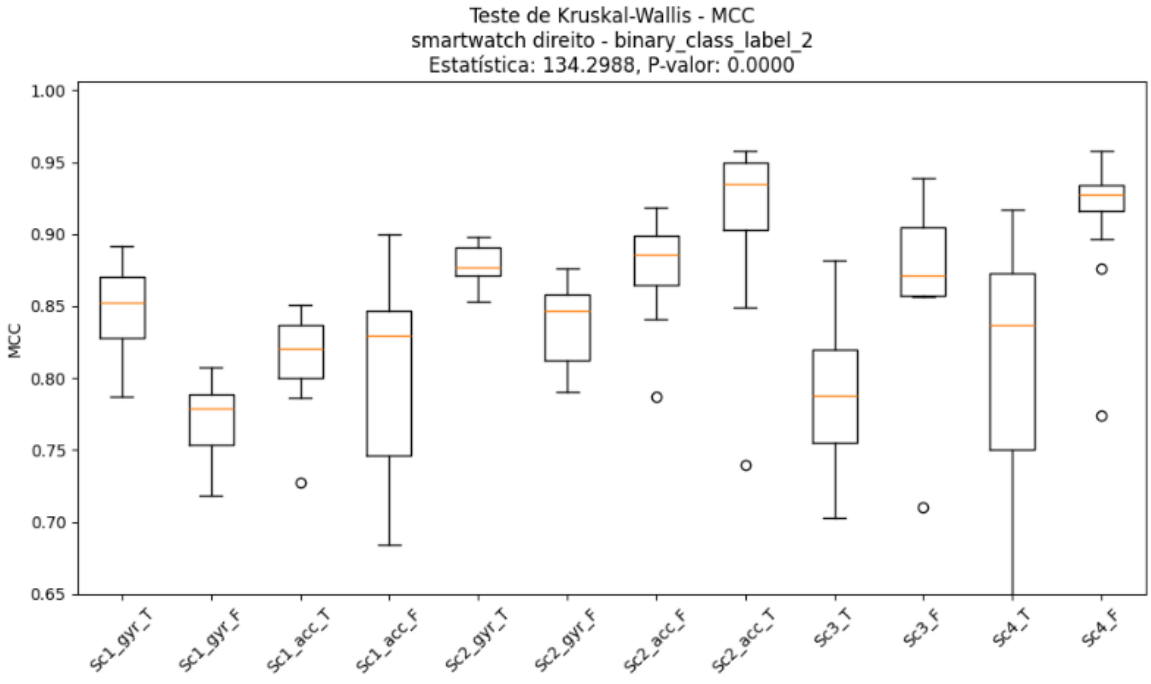


Figura V.9: Teste de *Kruskal-Wallis*, lado direito, binário_2.

O teste de *Kruskal-Wallis* demonstrado na Figura V.9 apresenta um p-valor de 0.0000 e uma estatística H de 134.2988, rejeitando a hipótese nula e confirmando uma diferença estatisticamente significativa entre os cenários de treinamento. O cenário Sc2_acc_T se destaca como o mais consistente, com uma mediana de MCC próxima a 0.95 e baixa dispersão, indicando alta confiabilidade

nos resultados. Essa menor variabilidade sugere que os treinamentos nesse cenário tendem a produzir desempenhos consistentes, proporcionando estabilidade ao modelo. O cenário Sc4_F também se destaca, com uma mediana elevada — ligeiramente inferior a Sc2_acc_T — e uma menor variação, como demonstrado pelo *boxplot* mais compacto, o que também reforça a consistência. Em contraste, o cenário Sc1_gyr_F apresenta uma mediana abaixo de 0.80 e uma ampla dispersão, evidenciando uma grande variabilidade nos desempenhos dos treinamentos, o que compromete a confiabilidade do modelo.

A Tabela V.4 resume os resultados dos treinamentos das redes neurais CNN1D para os 12 cenários, destacando os melhores e piores desempenhos alcançados.

Tabela V.4: Resultados do desempenho dos Treinamento para cada cenário, lado direito, binário_2.

Treinamentos com Melhor Desempenho							
Cenário	MCC	SE	ES	TP	TN	FN	FP
Sc1_acc_F	0.8994	0.9916	0.8974	1063.0	105.0	12.0	12.0
Sc1_acc_T	0.8507	0.9832	0.8803	1054.0	103.0	14.0	14.0
Sc1_gyr_F	0.8076	0.9832	0.8120	1054.0	95.0	22.0	22.0
Sc1_gyr_T	0.8914	0.9888	0.9060	1060.0	106.0	11.0	11.0
Sc_2_acc_F	0.9186	0.9935	0.9145	1065.0	107.0	10.0	10.0
Sc_2_acc_T	0.9579	0.9944	0.9744	1066.0	114.0	3.0	3.0
Sc_2_gyr_F	0.8759	0.9888	0.8803	1060.0	103.0	14.0	14.0
Sc_2_gyr_T	0.8984	0.9869	0.9316	1058.0	109.0	8.0	8.0
Sc_3_F	0.9392	0.9925	0.9573	1064.0	112.0	5.0	5.0
Sc_3_T	0.8820	0.9953	0.8376	1067.0	98.0	19.0	19.0
Sc_4_F	0.9575	0.9953	0.9658	1067.0	113.0	4.0	4.0
Sc_4_T	0.9169	0.9888	0.9487	1060.0	111.0	6.0	6.0
Treinamentos com Pior Desempenho							
Cenário	MCC	SE	ES	TP	TN	FN	FP
Sc1_acc_F	0.4251	1.0	0.1966	1072	23	94	94.0
Sc1_acc_T	0.7276	0.9328	0.9487	1000	111	6	6.0
Sc1_gyr_F	0.7184	0.9618	0.8034	1031	94	23	23.0
Sc1_gyr_T	0.7869	0.9823	0.7863	1053	92	25	25.0
Sc_2_acc_F	0.7871	0.9757	0.8291	1046	97	20	20.0
Sc_2_acc_T	0.7398	0.9291	0.9829	996	115	2	2.0
Sc_2_gyr_F	0.7904	0.9981	0.6667	1070	78	39	39.0
Sc_2_gyr_T	0.8534	0.9813	0.8974	1052	105	12	12.0
Sc_3_F	0.0	1.0	0.0	1072	0	117	117.0
Sc_3_T	0.5024	0.9991	0.2821	1071	33	84	84.0
Sc_4_F	0.7738	0.9991	0.6325	1071	74	43	43.0
Sc_4_T	0.1924	0.2938	0.9915	315	116	1	1.0

- **Análise dos Treinamentos com Melhor Desempenho**: O cenário Sc_2_acc_T destaca-se por uma sensibilidade de 0.9944 e especificidade de 0.9743, demonstrando sua alta capacidade de identificar corretamente quedas dos soldados, enquanto minimiza falsos positivos (3) e falsos negativos (6). Com precisão de 0.9971 e acurácia de 0.9924, este cenário apresenta um dos desempenhos mais equilibrados. Outro ponto forte é o cenário Sc_4_F, com uma

sensibilidade de 0.9953 e apenas cinco falsos negativos, reforçando sua eficácia na detecção de quedas.

- **Análise dos Treinamentos com Pior Desempenho**: Os cenários Sc1_acc_F e Sc_3_F apresentaram baixa convergência, evidenciada pelas baixas taxas de especificidade de 0.1965 e 0 respectivamente. Isso indica uma limitada capacidade de evitar falsos positivos, prejudicando sua eficácia geral.

A análise da tabela revela que os melhores cenários no domínio do tempo e da frequência possuem desempenho parecido. Na Figura V.10, o teste de *Mann-Whitney* compara os domínios do tempo e da frequência. Nela, um *boxplot* representa o MCC de todos os treinamentos para os cenários no domínio do tempo, enquanto outro *boxplot* ilustra os cenários de treinamento no domínio da frequência.

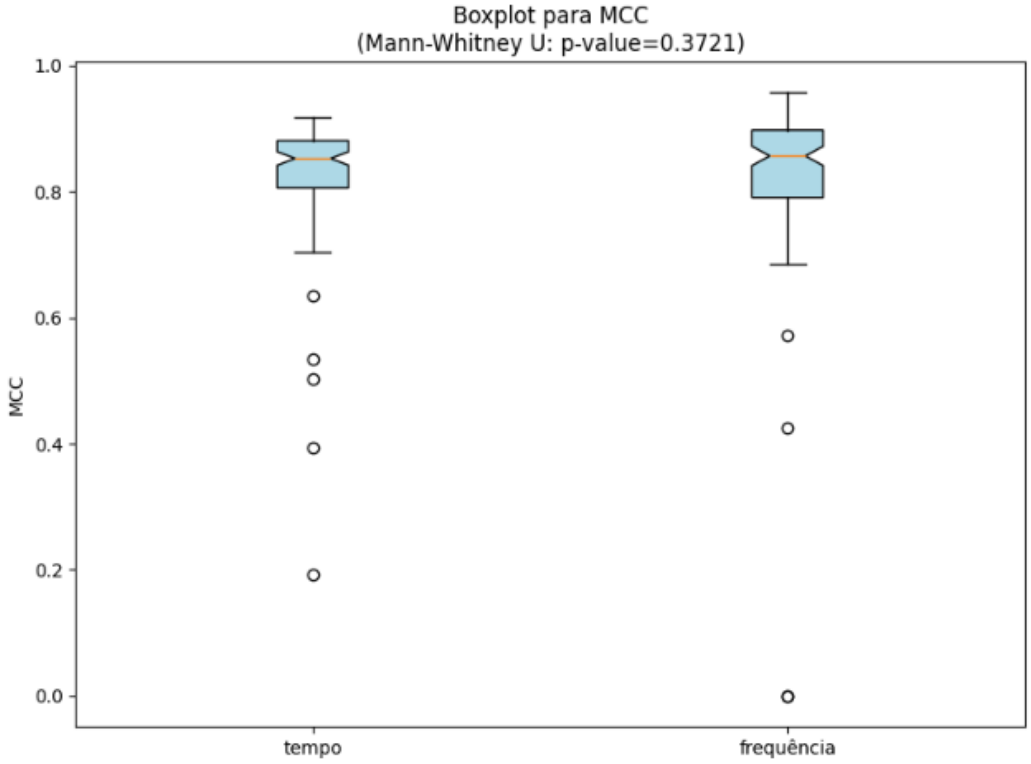


Figura V.10: Teste de *Mann-Whitney* para os domínios do tempo e da frequência, lado direito, binário_2.

O p-valor de 0.3721 indica que não há diferença estatisticamente significativa entre os desempenhos nos domínios do tempo e da frequência. No entanto, observando as medianas de MCC, o domínio do tempo mostra uma leve superioridade em relação ao domínio da frequência. Essa diferença, ainda que não estatisticamente significativa, sugere que o modelo tem uma tendência ligeiramente melhor em desempenho com os dados no domínio do tempo. A distribuição dos valores é mais compacta no domínio do tempo, o que indica maior consistência e menos variabilidade em

relação aos dados de frequência. Assim, o domínio do tempo pode ser considerado como um cenário preferível.

Com a análise realizada para o rótulo binário_2, que corresponde à abordagem que considera as atividades de transição para tiro deitado (OM_6 a OM_8) como não quedas, o cenário que apresentou o melhor desempenho foi o Sc_2_acc_T. Isso indica que o melhor conjunto de variáveis inerciais a ser utilizado são as componentes x, y e z da aceleração linear. A análise também demonstrou que os dados no domínio do tempo resultaram em um melhor desempenho na detecção de quedas.

V.3.3 Comparação entre as abordagens de rotulação - lado direito.

A Figura V.11 apresenta o teste de *Mann-Whitney* comparando as abordagens binário_1 e binário_2. O gráfico exibe dois *boxplots*: um representa o valor do MCC de todos os treinamentos para a abordagem binário_1, e o outro mostra o MCC para todos os treinamentos da abordagem binário_2.

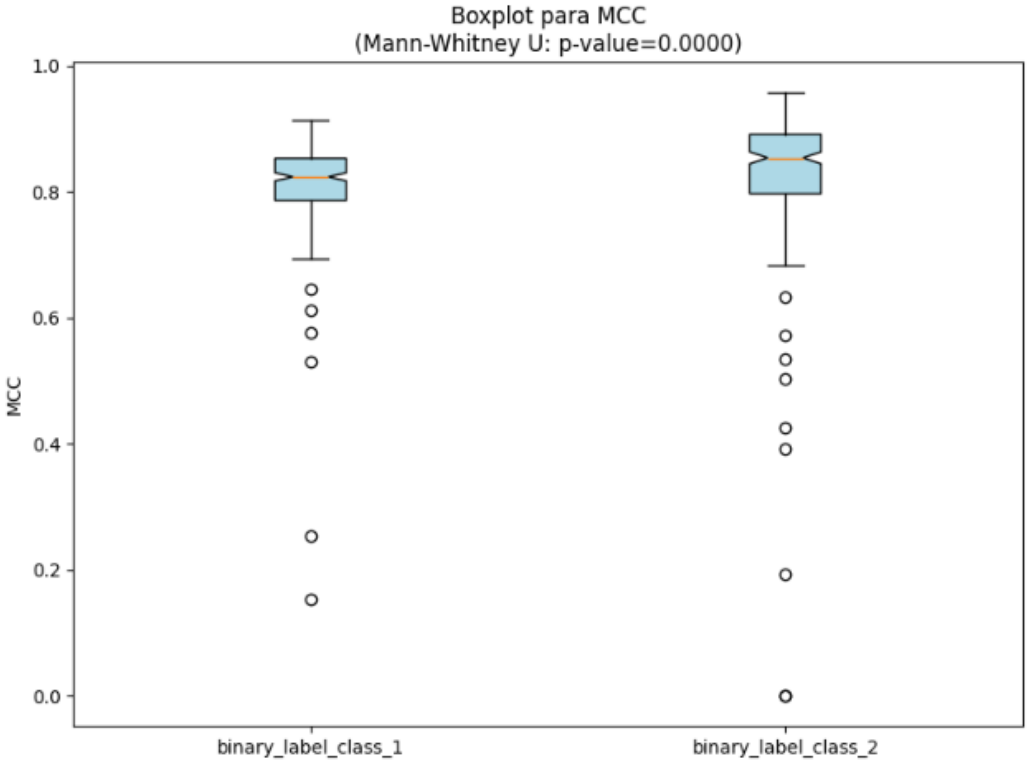


Figura V.11: Teste de *Mann-Whitney* para as abordagens de rotulação, lado direito.

,

Com um valor de p igual a 0.0000, indicando uma diferença significativa entre as distribuições de MCC das duas abordagens, há evidências de que uma delas apresenta desempenho superior. Os *boxplots* mostram que a mediana da abordagem binário_1 está próxima de 0.8, enquanto a da binário_2 está em torno de 0.85. Embora a binário_2 tenha alcançado a mediana e o melhor

desempenho mais altos, ela também exibe maior variabilidade nos resultados e uma presença mais significativa de *outliers* em comparação com a binário_1.

A Figura V.12 mostra a comparação dos cenários experimentais de melhor desempenho para ambas as abordagens de rotulação, sendo o Sc_2_acc_T o cenário com melhor desempenho para ambas.

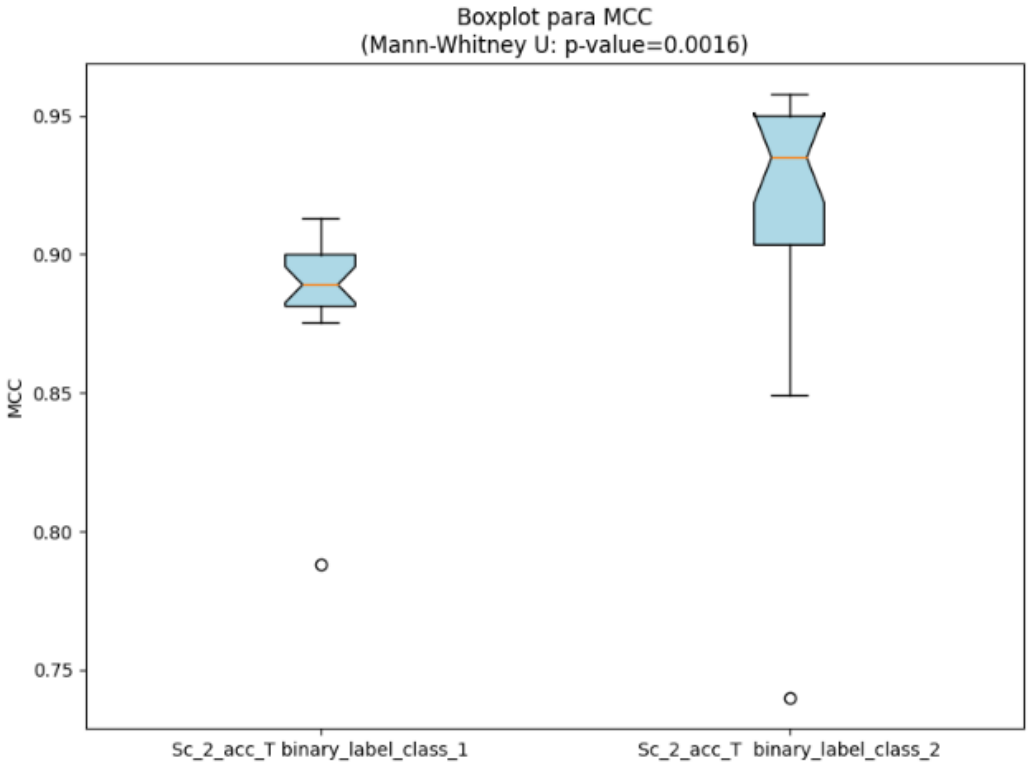


Figura V.12: Teste de *Mann-Whitney* para os melhores cenários de ambas as abordagens de rotulação, lado direito.

,

O teste de *Mann-Whitney*, com um valor de p de 0.0016, indica que a diferença entre as distribuições de MCC das duas abordagens é estatisticamente significativa, sugerindo que uma abordagem apresenta um desempenho superior à outra. Observando os *boxplots*, podemos ver que a abordagem binário_2 tem uma mediana ligeiramente mais alta do que a binário_1, sugerindo que esta abordagem tende a obter um desempenho médio melhor.

No entanto, a abordagem binário_2 apresenta uma variabilidade maior, com uma amplitude interquartil mais ampla e presença de *outliers* mais distantes, indicando que os resultados podem ser mais inconsistentes em comparação à binário_1. Em contrapartida, a binário_1 exibe menor variabilidade, com resultados mais concentrados em torno da mediana, sugerindo maior estabilidade. Contudo, a binário_2 é mais indicada, pois atinge um desempenho máximo mais elevado. Dessa forma, para os dados obtidos pelo *smartwatch* no pulso esquerdo, a abordagem recomendada é a binário_2, que considera as atividades de transição para tiro deitado (OM_6 a OM_8) como não

sendo atividades de queda.

V.4 Resultados - *Smartphone* peito

Esta seção está dividida em 3 seções, onde a seção V.4.1 apresenta os resultados para o *smartphone* do peito com os rótulos binário_1, a seção V.4.2 apresenta os resultados para o *smartphone* do peito com os rótulos binário_2, e a seção V.4.3 apresenta a comparação entre as duas abordagens.

V.4.1 Resultados - binário_1

A Figura V.13 apresenta resultado do teste de *Kruskal-Wallis* para a métrica MCC para os 20 treinamentos executados para os 12 cenários experimentais, utilizando os dados coletados com o *smartphone* portado no bolso do peito dos militares voluntários para a abordagem de rótulo binário_1 .

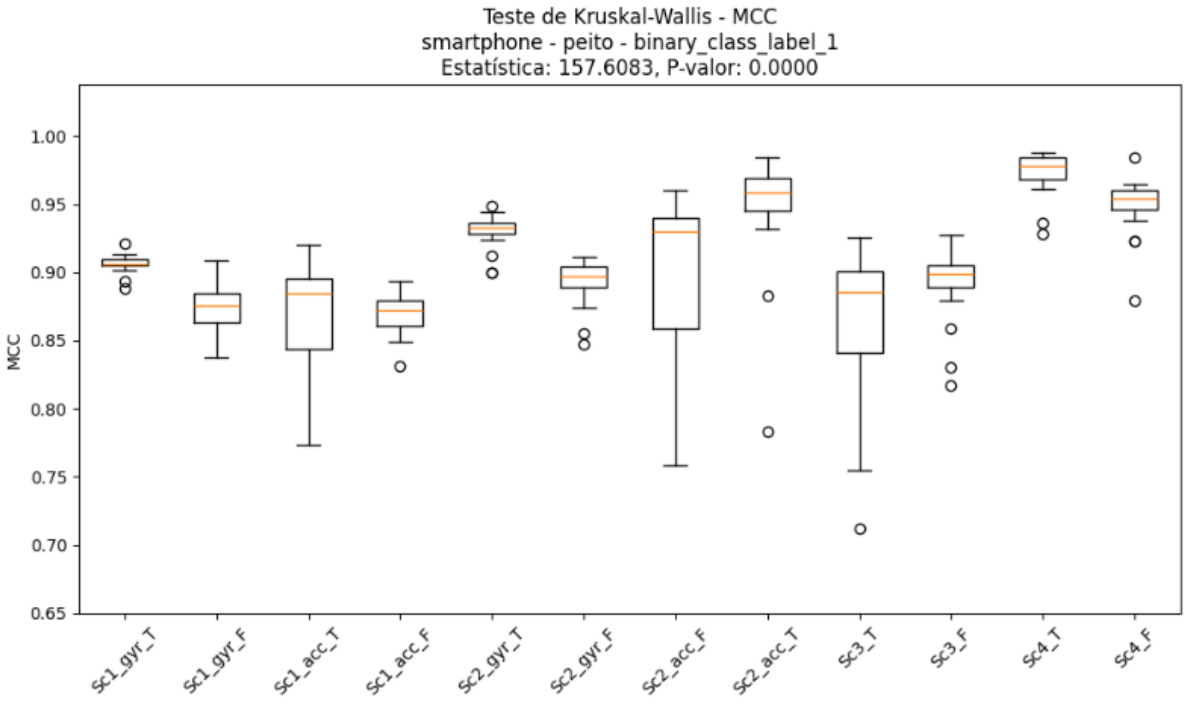


Figura V.13: Teste de *Kruskal-Wallis*, peito, binário_1.

O teste de *Kruskal-Wallis* demonstrado na Figura V.13 apresenta um p-valor de 0.0000 e uma estatística H de 157.6083, rejeitando a hipótese nula e confirmando uma diferença estatisticamente significativa entre os cenários de treinamento. O cenário Sc4_T se destaca como o mais consistente, com uma mediana de MCC próxima a 0.98 e baixa dispersão, indicando alta confiabilidade nos resultados. Essa menor variabilidade sugere que os treinamentos nesse cenário tendem a produzir desempenhos consistentes, proporcionando estabilidade ao modelo. O cenário Sc2_acc_T também se destaca, com uma mediana elevada (0.95). Em contraste, o cenário Sc1_acc_F apresenta uma

mediana abaixo de 0.87 o que compromete a confiabilidade do modelo.

A Tabela V.5 resume os resultados dos treinamentos das redes neurais CNN1D para os 12 cenários, destacando os melhores e piores desempenhos alcançados.

Tabela V.5: Resultados do desempenho dos Treinamento para cada cenário, peito, binário_1.

Treinamentos com Melhor Desempenho								
Cenário	MCC	SE	ES	PR	TP	TN	FP	FN
Sc1_acc_F	0.8934	0.9972	0.8392	0.9788	1062.0	120.0	23.0	3.0
Sc1_acc_T	0.9202	0.9915	0.9231	0.9897	1056.0	132.0	11.0	9.0
Sc1_gyr_F	0.9085	0.9897	0.9161	0.9887	1054.0	131.0	12.0	11.0
Sc1_gyr_T	0.9212	0.9897	0.9371	0.9915	1054.0	134.0	9.0	11.0
Sc_2_acc_F	0.9600	0.9972	0.9510	0.9935	1062.0	136.0	7.0	3.0
Sc_2_acc_T	0.9843	0.9972	0.9930	0.9991	1062.0	142.0	1.0	3.0
Sc_2_gyr_F	0.9114	0.9925	0.9021	0.9869	1057.0	129.0	14.0	8.0
Sc_2_gyr_T	0.9490	0.9925	0.9650	0.9953	1057.0	138.0	5.0	8.0
Sc_3_F	0.9273	0.9953	0.9091	0.9879	1060.0	130.0	13.0	5.0
Sc_3_T	0.9254	0.9897	0.9441	0.9925	1054.0	135.0	8.0	11.0
Sc_4_F	0.9841	1.0000	0.9720	0.9963	1065.0	139.0	4.0	0.0
Sc_4_T	0.9881	0.9991	0.9860	0.9981	1064.0	141.0	2.0	1.0
Treinamentos com Pior Desempenho								
Cenário	MCC	SE	ES	PR	TP	TN	FP	FN
Sc1_acc_F	0.8313	0.9878	0.8042	0.9741	1052.0	115.0	28.0	13.0
Sc1_acc_T	0.7736	0.9981	0.6434	0.9542	1063.0	92.0	51.0	2.0
Sc1_gyr_F	0.8373	0.9991	0.7343	0.9655	1064.0	105.0	38.0	1.0
Sc1_gyr_T	0.8883	0.9878	0.8951	0.9859	1052.0	128.0	15.0	13.0
Sc_2_acc_F	0.0000	1.0000	0.0000	0.8816	1065.0	0.0	143.0	0.0
Sc_2_acc_T	0.7830	0.9324	0.9930	0.9990	993.0	142.0	1.0	72.0
Sc_2_gyr_F	0.8470	0.9915	0.8042	0.9742	1056.0	115.0	28.0	9.0
Sc_2_gyr_T	0.8995	0.9906	0.8951	0.9860	1055.0	128.0	15.0	10.0
Sc_3_F	0.8172	0.9559	0.9510	0.9932	1018.0	136.0	7.0	47.0
Sc_3_T	0.7121	0.9061	0.9720	0.9959	965.0	139.0	4.0	100.0
Sc_4_F	0.8790	0.9718	0.9720	0.9962	1035.0	139.0	4.0	30.0
Sc_4_T	0.9282	0.9925	0.9301	0.9906	1057.0	133.0	10.0	8.0

- **Análise dos Treinamentos com Melhor Desempenho**: O cenário Sc_4_T destaca-se isoladamente entre os demais, apresentando uma sensibilidade de 0.9990 e uma especificidade de 0.9860. Esses valores refletem a alta capacidade deste cenário em identificar corretamente as quedas dos soldados, ao mesmo tempo que minimiza a ocorrência de falsos positivos (2) e falsos negativos (1). A precisão de 0.9981 reforça sua eficácia.

- **Análise dos Treinamentos com Pior Desempenho**: O cenário Sc_2_acc_F apresentou baixa convergência, evidenciada pela especificidade de 0.0. Entre os cenários que convergiram, o cenário Sc_2_acc_T teve o pior desempenho, com uma especificidade de 0.9323 e um total de 72 falsos negativos.

A análise da tabela revela que os cenários no domínio do tempo, com destaque para o Sc_4_T,

são os mais eficazes. Na Figura V.14, o teste de *Mann-Whitney* compara os domínios do tempo e da frequência. Nela, um *boxplot* representa o MCC de todos os treinamentos para os cenários no domínio do tempo, enquanto outro *boxplot* ilustra os cenários de treinamento no domínio da frequência.

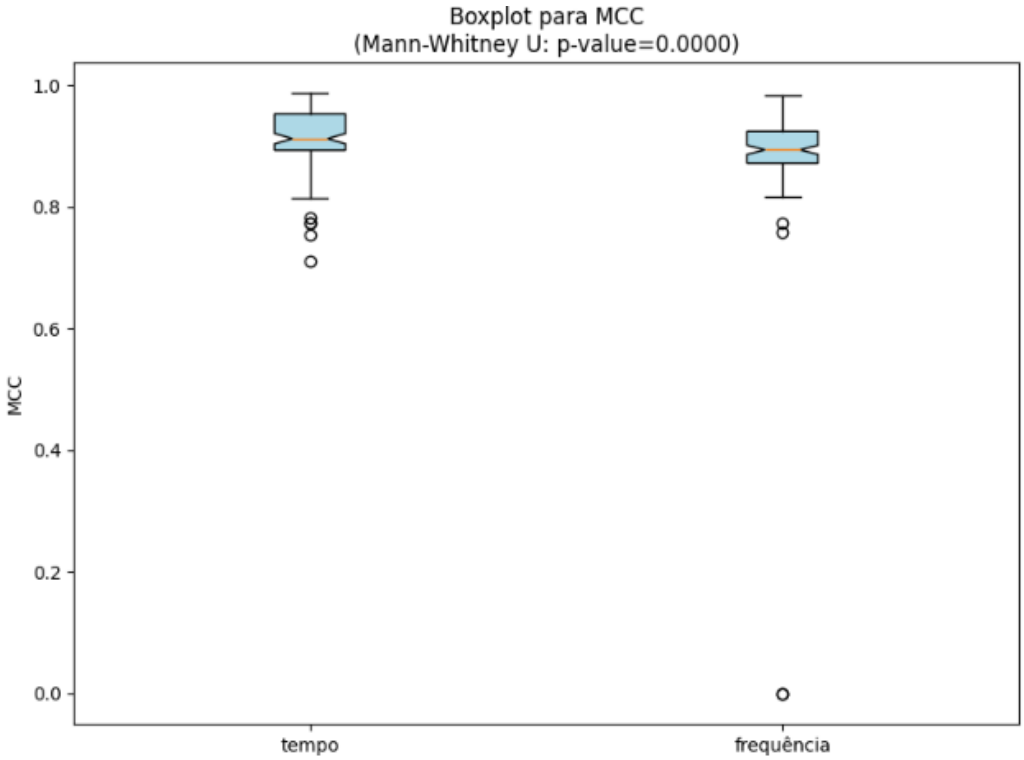


Figura V.14: Teste de *Mann-Whitney* para os domínios do tempo e da frequência, peito, binário_1.

O p-valor de 0.0000 indica uma diferença estatisticamente significativa entre os desempenhos nos domínios do tempo e da frequência. A mediana de MCC é claramente mais alta no domínio do tempo, evidenciando que os dados temporais proporcionaram um desempenho superior. A menor variabilidade dos valores no domínio do tempo, com poucos *outliers*, sugere que o modelo conseguiu generalizar bem em vários treinamentos, apresentando resultados consistentes. Em contrapartida, o domínio da frequência exibiu maior dispersão e mais *outliers*, indicando dificuldades em extrair padrões consistentes.

Com a análise realizada para o rótulo binário_1, que corresponde à abordagem que considera as atividades de transição para tiro deitado (OM_6 a OM_8) como quedas, o cenário que apresentou o melhor desempenho foi o Sc_4_T. Isso indica que o melhor conjunto de variáveis inerciais a ser utilizado são as componentes x, y e z da aceleração linear combinados com as componentes x, y e z da aceleração angular. A análise também demonstrou que os dados no domínio do tempo resultaram em um melhor desempenho na detecção de quedas.

V.4.2 Resultados - binário_2

A Figura V.15 apresenta resultado do teste de *Kruskal-Wallis* para a métrica MCC para os 20 treinamentos executados para os 12 cenários experimentais, utilizando os dados coletados com o *smartphone* portado no bolso do peito dos militares voluntários para a abordagem de rótulo binário_2.

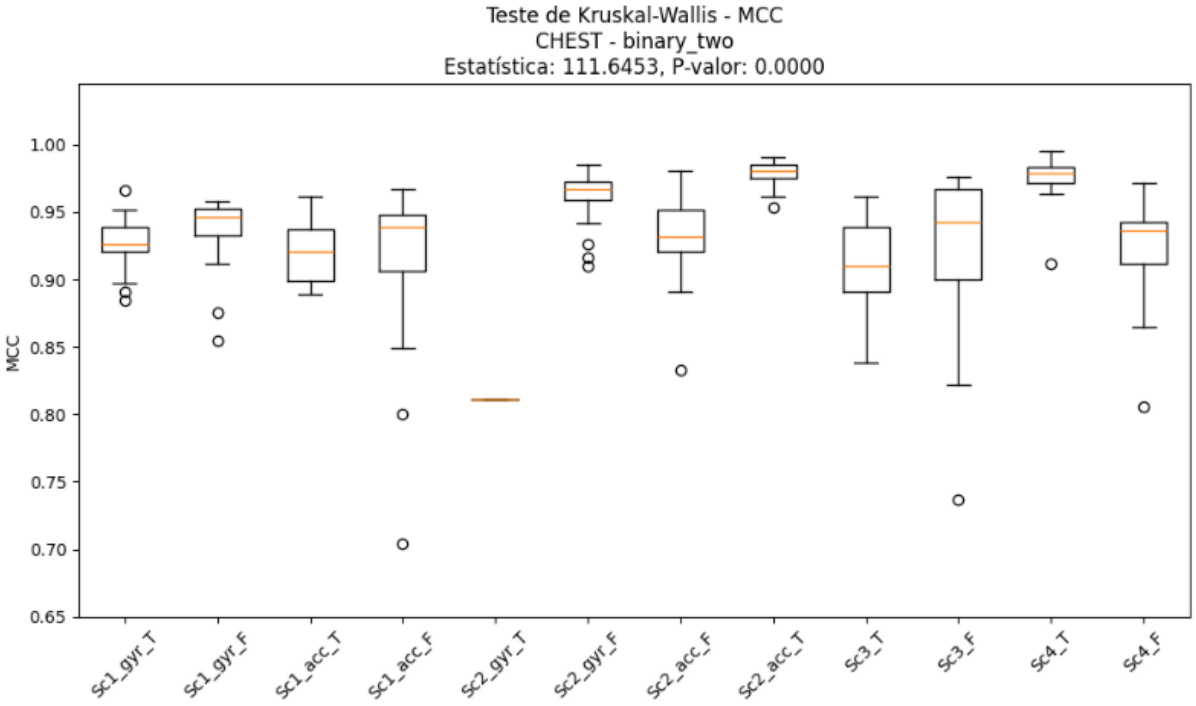


Figura V.15: Teste de *Kruskal-Wallis*, peito, binário_2

O teste de *Kruskal-Wallis*, ilustrado na Figura V.15, apresentou um p-valor de 0.0000 e uma estatística H de 111.6453, rejeitando a hipótese nula e indicando uma diferença estatisticamente significativa entre os cenários de treinamento. Os cenários Sc4_T e Sc2_acc_T se destacam pela consistência, com medianas de MCC próximas de 0.97 e baixa dispersão em seus *boxplots*, sugerindo alta confiabilidade nos resultados. Essa menor variabilidade indica que os treinamentos nesses cenários tendem a produzir desempenhos consistentes, contribuindo para a estabilidade do modelo. Em contrapartida, o cenário Sc_3_T apresentou o pior desempenho, com uma mediana abaixo de 0.91.

A Tabela V.6 resume os resultados dos treinamentos das redes neurais CNN1D para os 12 cenários, destacando os melhores e piores desempenhos alcançados.

Tabela V.6: Resultados do desempenho dos Treinamento para cada cenário, lado peito, binário_2.

Treinamentos com Melhor Desempenho								
Cenário	MCC	SE	ES	PR	TP	TN	FP	FN
Sc1_acc_F	0.9671	0.9954	0.9828	0.9982	1087	114	2	5
Sc1_acc_T	0.9616	0.9973	0.9569	0.9954	1089	111	5	3
Sc1_gyr_F	0.9582	0.9936	0.9828	0.9982	1085	114	2	7
Sc1_gyr_T	0.9665	0.9973	0.9655	0.9963	1089	112	4	3
Sc_2_acc_F	0.9809	0.9982	0.9828	0.9982	1090	114	2	2
Sc_2_acc_T	0.9904	1.0000	0.9828	0.9982	1092	114	2	0
Sc_2_gyr_F	0.9858	0.9982	0.9914	0.9991	1090	115	1	2
Sc_2_gyr_T	0.8116	1.0000	0.6810	0.9672	1092	79	37	0
Sc_3_F	0.9765	0.9963	0.9914	0.9991	1088	115	1	4
Sc_3_T	0.9619	0.9963	0.9655	0.9963	1088	112	4	4
Sc_4_F	0.9714	0.9973	0.9741	0.9973	1089	113	3	3
Sc_4_T	0.9952	1.0000	0.9914	0.9991	1092	115	1	0
Treinamentos com Pior Desempenho								
Cenário	MCC	SE	ES	PR	TP	TN	FP	FN
Sc1_acc_F	0.0	1.0	0.0	0.9039	1092.0	0.0	116.0	0.0
Sc1_acc_T	0.0966	0.9991	0.0172	0.9054	1091.0	2.0	114.0	1.0
Sc1_gyr_F	0.8545	0.9991	0.7586	0.9750	1091.0	88.0	28.0	1.0
Sc1_gyr_T	0.8847	0.9780	0.9741	0.9972	1068.0	113.0	3.0	24.0
Sc_2_acc_F	0.8331	0.9991	0.7241	0.9715	1091.0	84.0	32.0	1.0
Sc_2_acc_T	0.9538	0.9927	0.9828	0.9982	1084.0	114.0	2.0	8.0
Sc_2_gyr_F	0.9099	0.9817	0.9914	0.9991	1072.0	115.0	1.0	20.0
Sc_2_gyr_T	0.8116	1.0	0.6810	0.9672	1092.0	79.0	37.0	0.0
Sc_3_F	0.7372	0.9991	0.5776	0.9570	1091.0	67.0	49.0	1.0
Sc_3_T	0.8388	0.9716	0.9397	0.9934	1061.0	109.0	7.0	31.0
Sc_4_F	0.8061	1.0	0.6724	0.9664	1092.0	78.0	38.0	0.0
Sc_4_T	0.9117	1.0	0.8448	0.9838	1092.0	98.0	18.0	0.0

- **Análise dos Treinamentos com Melhor Desempenho**: Os cenários Sc_4_T e Sc_2_acc_T se destacaram com sensibilidade e especificidade de 0.9913 e 0.9827, respectivamente. Esses valores refletem a capacidade desses cenários de identificar corretamente as quedas dos soldados, ao mesmo tempo que minimizam as ocorrências de falsos positivos (1 e 2, respectivamente) e falsos negativos (0). Estes foram os melhores resultados entre todas as posições e abordagens de rotulação utilizadas nesta pesquisa.

- **Análise dos Treinamentos com Pior Desempenho**: O cenário Sc_1_acc_F, no pior resultado, não apresentou convergência, o que se evidenciou pelas baixas taxas de especificidade e MCC. Entre os cenários que convergiram, o Sc_3_F se destacou negativamente, com uma especificidade de 0.5775 e apresentando 49 falsos positivos.

A análise da tabela revela que os cenários no domínio do tempo, com destaque para o Sc_4_T, são os mais eficazes. Na Figura V.16, o teste de *Mann-Whitney* compara os domínios do tempo e da frequência. Nela, um *boxplot* representa o MCC de todos os treinamentos para os cenários

no domínio do tempo, enquanto outro *boxplot* ilustra os cenários de treinamento no domínio da frequência.

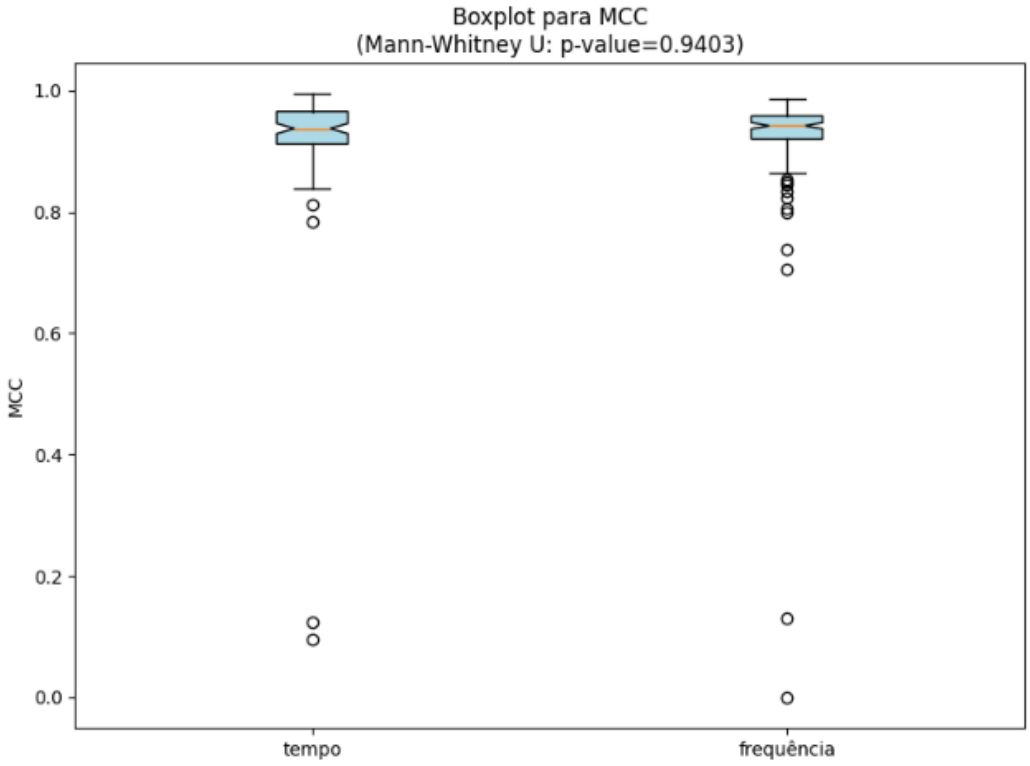


Figura V.16: Teste de *Mann-Whitney* para os domínios do tempo e da frequência, peito, binário_2.

,

O p-valor de 0.9403 indica que não há diferença estatisticamente significativa entre os desempenhos nos domínios do tempo e da frequência. Apesar disso, uma análise visual mostra que a mediana do MCC é ligeiramente mais alta no domínio do tempo. Além disso, a distribuição no domínio do tempo é um pouco mais concentrada, refletindo menor variação nos resultados e sugerindo uma maior consistência em comparação ao domínio da frequência. Portanto, mesmo sem significância estatística, o domínio do tempo pode ser considerado um cenário ligeiramente superior devido à maior estabilidade dos resultados.

Com a análise realizada para o rótulo binário_2, que corresponde à abordagem que considera as atividades de transição para tiro deitado (OM_6 a OM_8) como não quedas, o cenário que apresentou o melhor desempenho foi o Sc_4_T. Isso indica que o melhor conjunto de variáveis inerciais a ser utilizado são as componentes x, y e z da aceleração linear combinados com as componentes x, y e z da aceleração angular. A análise também demonstrou que os dados no domínio do tempo resultaram em um melhor desempenho preditivo na detecção de quedas.

V.4.3 Comparação entre as abordagens de rotulação - Peito.

A Figura V.17 apresenta o teste de *Mann-Whitney* comparando as abordagens binário_1 e binário_2. O gráfico exibe dois *boxplots*: um representa o valor do MCC de todos os treinamentos para a abordagem binário_1, e o outro mostra o MCC para todos os treinamentos da abordagem binário_2.

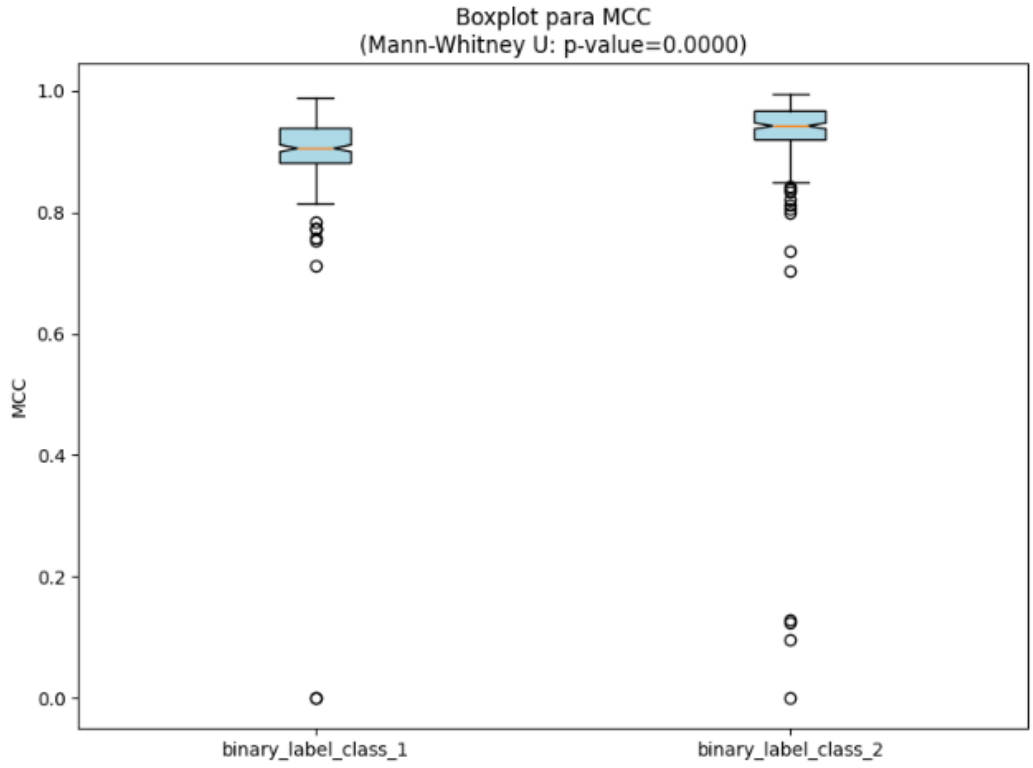


Figura V.17: Teste de *Mann-Whitney* para as abordagens de rotulação, peito.

,

O p-valor igual a 0.0000 indica uma diferença estatisticamente significativa entre as duas distribuições de MCC, sugerindo que uma das abordagens é consistentemente melhor em termos de desempenho. Observa-se que ambas as abordagens apresentam uma mediana alta de MCC, com valores próximos de 1.0, indicando um bom desempenho geral.

A abordagem binário_1 possui uma distribuição mais concentrada, com menor variabilidade e menos *outliers* extremos, sugerindo resultados mais estáveis. Em contraste, a abordagem binário_2 exibe uma maior quantidade de *outliers*, indicando uma variabilidade maior, embora sua mediana esteja em um nível comparável.

A Figura V.18 mostra a comparação dos cenários experimentais de melhor desempenho para ambas as abordagens de rotulação, sendo o Sc_4_T o cenário com melhor desempenho para ambas.

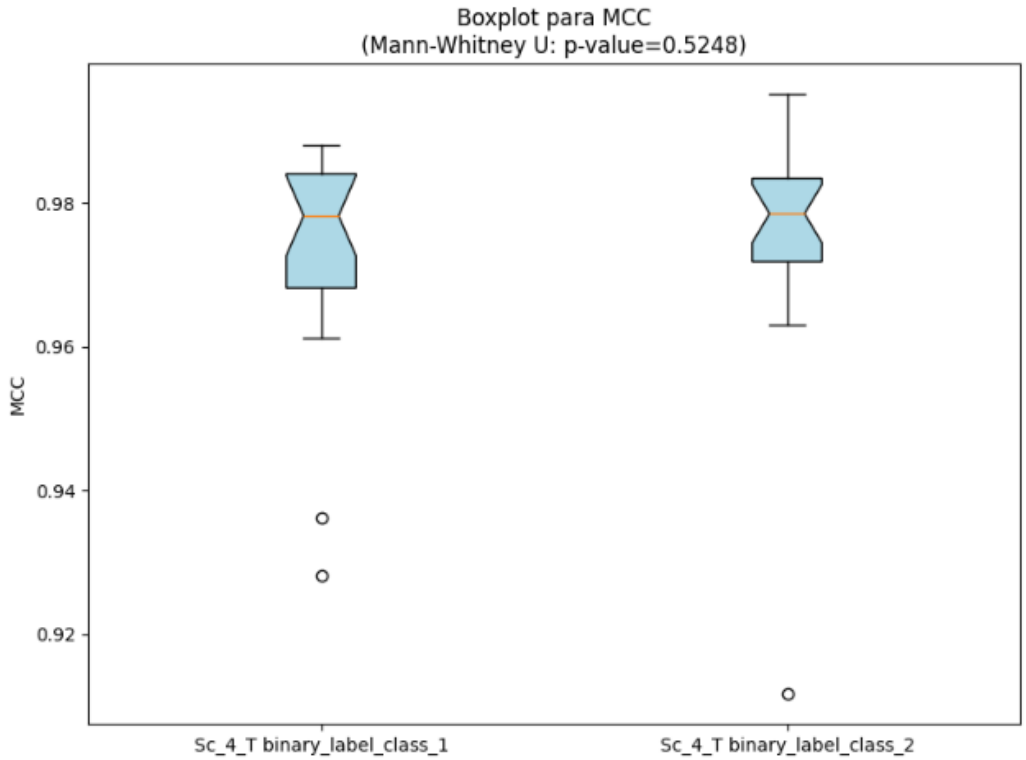


Figura V.18: Teste de *Mann-Whitney* para os melhores cenários de ambas as abordagens de rotulação, peito.

,

O valor de p de 0.5248 indica que não há diferença estatisticamente significativa entre as distribuições de MCC das duas abordagens neste cenário. Em outras palavras, os desempenhos das abordagens binário_1 e binário_2 são similares no contexto do cenário Sc_4_T. Visualmente, os *boxplots* reforçam essa interpretação, uma vez que ambas as abordagens apresentam medianas elevadas e próximas, com variações semelhantes nos resultados. A abordagem binário_1 exibe alguns *outliers* ligeiramente mais baixos, mas esses valores não impactam de forma significativa a distribuição geral. Assim, como não há diferença relevante, ambas as abordagens podem ser consideradas igualmente eficazes neste cenário específico (Sc_4_T).

V.5 Comparação entre as três posições: esquerda, direita e peito.

Conforme discutido nas seções anteriores, os cenários e abordagens de rotulação que apresentaram melhor desempenho nas posições esquerda, direita e peito foram, respectivamente: Sc_2_acc_T – binário_1, Sc_2_acc_T – binário_2 e Sc_4_T para ambas as abordagens. Para determinar qual posição é mais apropriada para detectar quedas em soldados, aplicamos o teste de *Kruskal Wallis* (Figura V.19) nos cenários mencionados para cada posição.

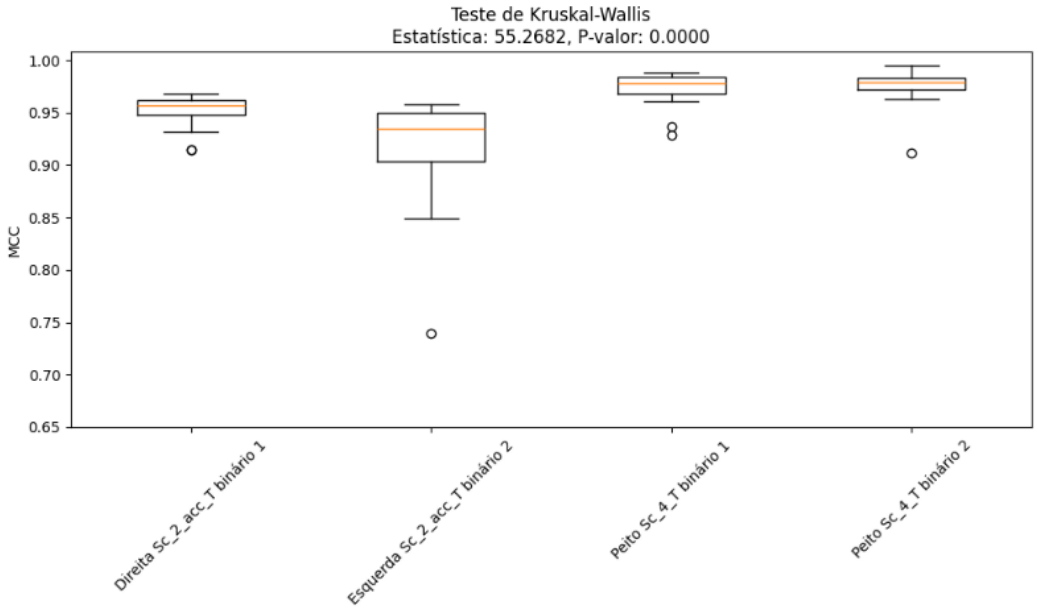


Figura V.19: Teste de *Kruskal Wallis* para os melhores cenários para cada posição.

,

A estatística obtida no teste de *Kruskal-Wallis* foi de 55.2682, com um valor-p de 0.0000, indicando a rejeição da hipótese nula, que assume a inexistência de diferenças significativas entre os cenários. Assim, concluímos que há diferenças estatisticamente significativas entre os desempenhos dos cenários analisados. No cenário Direita Sc_2_acc_T binário_1, destaca-se um desempenho elevado e consistente, com valores de MCC variando entre aproximadamente 0.92 e 0.97. A baixa dispersão e a presença de apenas um *outlier* sugerem estabilidade e confiabilidade desse cenário.

Por outro lado, o cenário Esquerda Sc_2_acc_T binário_2 apresentou maior variabilidade, com o MCC variando entre aproximadamente 0.75 e 0.96. A dispersão mais alta e a presença de *outliers* indicam um desempenho menos consistente em relação aos demais cenários. O cenário Peito Sc_4_T binário_1 demonstrou um desempenho elevado e estável, com MCC variando entre 0.94 e 0.99, mínima dispersão e apenas um *outlier* menor, o que reflete boa confiabilidade e estabilidade nos resultados. Finalmente, o cenário Peito Sc_4_T binário_2 apresentou desempenho semelhante ao anterior, com MCC variando entre 0.94 e 0.9950. Este cenário também demonstrou alta estabilidade, com pouca variação e apenas um *outlier*, evidenciando o melhor desempenho geral. Considerando esses aspectos, a classificação dos cenários em ordem de melhor desempenho é:

1. Peito Sc_4_T binário_2.
2. Peito Sc_4_T binário_1.
3. Direita Sc_2_acc_T binário_1.
4. Esquerda Sc_2_acc_T binário_2.

Como mencionado anteriormente, para o algoritmo de classificação de queda da primeira fase do projeto "Combatente do Futuro", o principal objetivo é maximizar o poder de detecção de quedas, minimizando a ocorrência de falsos positivos (ou seja, eventos incorretamente detectados como quedas). Os verdadeiros positivos não são uma preocupação nesta fase, pois se teoriza que, na segunda fase, o algoritmo de classificação do tipo de queda será capaz de distinguir se a queda foi devido a um ferimento, necessitando de atendimento, ou se foi uma queda operacional. Assim, se uma ação que não seja uma queda for detectada como tal, o erro será mitigado ao se classificar a natureza da queda como não emergencial.

Nesse contexto, considerando as taxas de falsos negativos e positivos, o cenário Sc_4_T da abordagem binário_1 na posição do peito é apontado como o mais adequado, sendo o melhor em termos de MCC. Em seu melhor resultado, apresentou uma quantidade de 0 falsos negativos e 1 falso positivo, enquanto nos 20 treinamentos executados, obteve uma taxa média de 2.35 falsos negativos e 2.75 falsos positivos.

Dessa forma, o modelo a ser implementado no projeto "Combatente do Futuro" é uma rede convolucional unidimensional treinada com os componentes x, y e z da aceleração linear combinados com os componentes x, y e z da aceleração angular, no domínio do Tempo. Neste modelo, as atividades de transição de tiro para tiro deitado (OM_6, OM_7 e OM_8) são consideradas como atividades de queda e a melhor posição para a localização dos sensores de detecção é o peito.

Os resultados exibidos neste capítulo mostram que não há diferença significativa nas desempenhos de treinamento das redes CNN1D para as abordagens binário_1 e binário_2, ou seja, a suposição inicial de que as atividades de transição para tiro deitado não se confundem com atividades de queda durante o treinamento da rede neural foi confirmada. Além disso, os cenários que apresentaram os melhores resultados para cada posição (direito, esquerdo e peito) foram os Sc_4_T e Sc_2_acc_T. Isso indica que as combinações de variáveis mais eficazes para a detecção de quedas são, respectivamente, a combinação dos componentes x, y e z da aceleração linear com os componentes x, y e z da aceleração angular, e apenas os componentes x, y e z da aceleração linear. Esse resultado evidencia que o uso exclusivo da magnitude da aceleração linear, amplamente adotado em estudos biográficos de detecção de queda (Capítulo III), pode ser uma simplificação excessiva que compromete a eficácia da detecção. Também ficou comprovada que para todas a posições os dados no domínio do tempo ofereceram resultados melhores do que os dados no domínio da frequência.

Capítulo VI Conclusão

Neste capítulo, apresentamos as conclusões do trabalho e discutimos possíveis desenvolvimentos futuros. Na Seção VI.1, realizamos uma análise retrospectiva dos resultados alcançados. Na Seção VI.2, apontamos as limitações desta pesquisa e sugerimos algumas direções para trabalhos futuros. Por fim, a Seção VI.3 apresenta aspectos relevantes à reprodutibilidade da pesquisa vigente.

VI.1 Visão Geral dos Resultados

O objetivo deste trabalho foi desenvolver um método baseado em aprendizado de máquina, especificamente utilizando redes neurais, para detectar quedas de soldados durante atividades militares. Essa iniciativa busca apoiar o projeto "Combatente do Futuro" da Marinha do Brasil, cujo objetivo é identificar automaticamente soldados incapacitados em campo de batalha.

Após a realização de uma pesquisa bibliográfica, constatou-se que não havia conjuntos de dados públicos apropriados para a detecção de quedas em cenários de combate, ou seja, envolvendo pessoas com treinamento militar e portando armas de fogo. Por isso, foi necessário coletar dados junto aos militares da Marinha do Brasil, especificamente do Instituto de Pesquisas da Marinha, para criar um conjunto de dados inédito. Esses dados incluem atividades de quedas, atividades de vida diária e operações táticas militares.

A pesquisa bibliográfica também revelou que a comunidade científica utiliza amplamente a magnitude da aceleração linear como variável inercial para treinar redes neurais. No entanto, poucos estudos apresentavam evidências robustas para validar essa abordagem. Isso motivou a realização de experimentos considerando diversos cenários para identificar quais variáveis melhor definem a queda de um soldado. Além disso, devido à alta variabilidade dos sinais coletados, decidiu-se analisar os dados tanto no domínio do tempo quanto no domínio da frequência. Para isso, a Transformada de Fourier foi aplicada, permitindo a extração de padrões no domínio da frequência. Contudo, os cenários experimentais baseados nos dados do domínio da frequência não superaram aqueles baseados no domínio do tempo.

Os dados de cada atividade foram segmentados em intervalos temporais de cinco segundos, simulando o janelamento do algoritmo do projeto "Combatente do Futuro". Cada janela foi processada para extrair características relevantes, que serviram como entrada para os algoritmos de aprendizado

de máquina. Este método capturou padrões de atividades relacionadas a quedas de forma eficiente. O modelo desenvolvido, uma rede convolucional unidimensional, conseguiu identificar quedas simuladas, distinguindo-as de outras atividades.

A arquitetura CNN1D foi escolhida devido às suas vantagens no processamento de sinais unidimensionais, como séries temporais, além de sua menor complexidade computacional em comparação com modelos bidimensionais. Para otimizar o desempenho do modelo, utilizou-se a otimização *bayesiana* para ajustar hiperparâmetros, como o número de filtros, o tamanho do *kernel* e a taxa de aprendizado. Esse processo foi aplicado a cada cenário experimental, garantindo uma comparação justa e identificando o melhor conjunto de hiperparâmetros.

O modelo de CNN1D com melhor desempenho foi treinado com dados coletados no peito do soldado, utilizando os componentes x, y e z da aceleração linear combinados com os mesmos componentes da aceleração angular, no domínio do tempo. Esse modelo alcançou um MCC de 0.9952, sensibilidade de 1.0000, especificidade de 0.9914 e precisão de 0.9914. Esse cenário também considerou transições entre posições de tiro como atividades de queda. Sendo este o mais eficiente para a implementação no algoritmo da fase 1 do projeto "Combatente do Futuro".

VI.2 Limitações e Trabalhos Futuros

Apesar dos resultados alcançados, algumas limitações foram identificadas:

- Condições Controladas de Coleta de Dados: Os dados foram coletados em ambientes simulados, o que pode não refletir completamente a complexidade de situações reais.
- Número Limitado de Participantes: Os experimentos envolveram 15 voluntários, o que limita a generalização dos resultados para populações maiores e mais diversificadas.

Para trabalhos futuros, sugerimos:

- Coleta em Cenários de Exercício de Treinamento de Combate: Coletar os dados em operações militares de treinamento, permitindo uma avaliação prática e contribuindo para a validação externa do modelo proposto.
- Recomenda-se a investigação do desempenho de outras arquiteturas de redes neurais, como MLP, para a análise dos dados abordados neste estudo. Essa abordagem pode incluir a exploração de redes profundas e arquiteturas híbridas, avaliando sua eficácia em comparação com os modelos utilizados. Além disso, testes com diferentes funções de ativação, técnicas de regularização e otimização podem oferecer percepções relevantes sobre a generalização dos modelos. Por fim, a utilização de métodos de *ensemble* pode ser uma estratégia promissora para aprimorar os resultados.

- Prosseguimento do Estudo no Projeto Combatente do Futuro: Prosseguir o estudo desenvolvendo o algoritmo de classificação de queda que classifica a queda do combatente como uma queda oriunda de ferimentos que deixa o combatente inoperante (baixa) ou como uma queda operacional.

- Utilizar Armamentos de Outros Portes: Coletar os dados dos militares voluntários utilizando outros tipos de armamentos empregados em cenários de conflito, como pistola 9mm, submetralhadora e armamento de calibre 12.

- Detecção de Disparo: Estudar se o pulso de aceleração do disparo da arma de fogo, assim como o coice da arma, produz índices de aceleração que se confundem com uma queda durante o treinamento da rede neural. Além disso, estudar se, além da queda, é possível detectar o disparo, evoluindo o sistema para um suporte à saúde do combatente e ao controle de munição.

VI.3 Reprodutibilidade

Para garantir a reprodutibilidade, todos os códigos e dados utilizados estão disponíveis publicamente. O código-fonte pode ser acessado em um dos repositórios GitHub (`https://github.com/AILAB-CEFET-RJ/fall-detect`). Além disso, os conjuntos de dados utilizados nos experimentos estão disponíveis no Zenodo (`https://zenodo.org/records/12760391`). Essas medidas buscam promover a continuidade da pesquisa e a replicação dos resultados obtidos.

Referências

Ministério da Defesa. MD33-M-10: Garantia da lei e da ordem, 2014.

Instituto de Segurança Pública do RJ. Policiais mortos em serviço: série histórica mensal por área de delegacia desde 01/2003. `http://www.ispdados.rj.gov.br/CrimesVida.html`, 2023. Acesso em: 31/05/2023.

Ministério da Defesa. Estudo das operações de glo no período de 1992 - 2021. `https://www.gov.br/defesa/pt-br/assuntos/exercicios-e-operacoes/garantia-da-lei-e-da-ordem`, 2023. Acesso em: 31/05/2023.

CASNAV. Comando e controle o desafio da interoperabilidade. `https://www.marinha.mil.br/casnav/?q=node/118`, 2023. Acesso em: 31/05/2023.

Ministério da Defesa. MD42-M-04: Apoio de saúde em operações conjuntas, 2017.

Sidnei Augusto Mascarenhas. Metodologia científica, 2018.

Warren S McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5:115–133, 1943.

Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958.

Paul Werbos. Beyond regression: New tools for prediction and analysis in the behavioral sciences. *PhD thesis, Committee on Applied Mathematics, Harvard University, Cambridge, MA*, 1974.

Thomas Walter Rauber. Redes neurais artificiais, 2005.

Maria Inês Vasconcellos Furtado. Redes neurais artificiais: uma abordagem para sala de aula. *Ponta Grossa, PR. Atena Editora*, page 19, 2019.

Matheus Souza Sales Aráujo, Cláudia Teresa Teles Farias, and Alexandre Smith Aquer Oliveira. Detecção e classificação de defeitos em chapas de aço carbono utilizando ondas ultrassônicas guiadas de lamb e redes neurais artificiais. In *VII CONNEPI, Congresso Norte Nordeste de Pesquisa e Inovação, Palmas*, 2012.

Yann LeCun, Leon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Zewen Li, Fan Liu, Wenjie Yang, Shouheng Peng, and Jun Zhou. A survey of convolutional neural networks: analysis, applications, and prospects. *IEEE transactions on neural networks and learning systems*, 2021.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.

Martin Andreoni Lopez and Diogo MF Mattos. Resumo de grandes volumes de dados com filtro de bloom: Uma abordagem eficiente para aprendizado profundo com redes neurais convolucionais em fluxos de rede. In *Anais do XXXIX Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, pages 532–545. SBC, 2021.

Ana Caroline Gomes Vargas, Aline Paes, and Cristina Nader Vasconcelos. Um estudo sobre redes neurais convolucionais e sua aplicação em detecção de pedestres. 2016.

Serkan Kiranyaz, Onur Avci, Osama Abdeljaber, Turker Ince, Moncef Gabbouj, and Daniel J Inman. 1d convolutional neural networks and applications: A survey. *Mechanical systems and signal processing*, 151:107398, 2021.

Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical bayesian optimization of machine learning algorithms. *Advances in neural information processing systems*, 25, 2012.

Brian Conolly and IJ Good. A table of discrete fourier transform pairs. *SIAM Journal on Applied Mathematics*, 32(4):810–822, 1977.

Han-Yun Chen and Ching-Hung Lee. Vibration signals analysis by explainable artificial intelligence (xai) approach: Application on bearing faults diagnosis. *IEEE Access*, 8:134246–134256, 2020.

Yafeng Qi, Lin Yang, Bangxu Liu, Li Liu, Yuhong Liu, Qingfeng Zheng, Dameng Liu, and Jianbin Luo. Accurate diagnosis of lung tissues for 2d raman spectrogram by deep learning based on short-time fourier transform. *Analytica Chimica Acta*, 1179:338821, 2021.

Guanis de Barros Vilela Junior, Bráulio Nascimento Lima, Heros Ferreira Ribeiro, Marcelo Francisco Rodrigues, Adriano de Almeida Pereira, José Ricardo Lourenço de Oliveira, Luís Felipe Silio, and Ricardo Pablo Passos. Importância do índice fowlkes-mallows (fmi), do coeficiente de correlação de matthews (mcc) e do índice youden (iy) nos classificadores de inteligência artificial na área da saúde. *Revista CPAQV–Centro de Pesquisas Avançadas em Qualidade de Vida/ Vol*, 14(3):2, 2022.

Ronaldo Cristiano Prati, GEAPA Batista, Maria Carolina Monard, et al. Curvas roc para avaliação de classificadores. *Revista IEEE América Latina*, 6(2):215–222, 2008.

M. Hossin and M.N. Sulaiman. A review on evaluation metrics for data classification evaluations. *International Journal of Data Mining & Knowledge Management Process (IJDKP)*, 5(2):1–10, March 2015. doi: 10.5121/ijdkp.2015.5201.

Taylor R Mauldin, Marc E Canby, Vangelis Metsis, Anne HH Ngu, and Coralys Cubero Rivera. Smartfall: A smartwatch-based fall detection system using deep learning. *Sensors*, 18(10):3363, 2018.

Samad Barri Khojasteh, José R Villar, Camelia Chira, Víctor M González, and Enrique De la Cal. Improving fall detection using an on-wrist wearable accelerometer. *Sensors*, 18(5):1350, 2018.

Gökhan Şengül, Murat Karakaya, Sanjay Misra, Olusola O Abayomi-Alli, and Robertas Damaševičius. Deep learning based fall detection using smartwatches for healthcare applications. *Biomedical Signal Processing and Control*, 71:103242, 2022.

Xiaoye Qian. Wearable computing architecture over distributed deep learning hierarchy: Fall detection study. Master's thesis, Case Western Reserve University, 2019.

Hamideh Kerdegari, Saeid Mokaram, Khairulmizam Samsudin, and Abdul Rahman Ramli. A pervasive neural network based fall detection system on smart phone. *Journal of Ambient Intelligence and Smart Environments*, 7(2):221–230, 2015.

Markus Deutsch and Harald Burgsteiner. A smartwatch-based assistance system for the elderly performing fall detection, unusual inactivity recognition and medication reminding. In *Health Informatics Meets eHealth*, pages 259–266. IOS Press, 2016.

Vasundhara Ghate. Hybrid deep learning approaches for smartphone sensor-based human activity recognition. *Multimedia Tools and Applications*, 80(28):35585–35604, 2021.

Minoo Rashidpour, F Abdali-Mohammadi, and Abdolhossein Fathi. Fall detection using adaptive neuro-fuzzy inference system. *Int. J. Multimed. Ubiquitous Eng*, 11:91–106, 2016.

Mengqi Hou, Haixia Wang, Zechen Xiao, and Guilin Zhang. An svm fall recognition algorithm based on a gravity acceleration sensor. *Systems Science & Control Engineering*, 6(3):208–214, 2018.

Marcelo Vidigal, Mario Lima, and Areolino De Almeida Neto. Elder falls detection based on artificial neural networks. In *2015 Fourteenth Mexican International Conference on Artificial Intelligence (MICAI)*, pages 226–230. IEEE, 2015.

Taylor Mauldin, Anne H Ngu, Vangelis Metsis, and Marc E Canby. Ensemble deep learning on wearables using small datasets. *ACM Transactions on Computing for Healthcare*, 2(1):1–30, 2020.

Pei-Yu Tsai, Yi-Cian Yang, Yi-Jiun Shih, and Hsu-Yang Kung. Gesture-aware fall detection system: Design and implementation. In *2015 IEEE 5th International Conference on Consumer Electronics-Berlin (ICCE-Berlin)*, pages 88–92. IEEE, 2015.

Developer Android. Android sensormanager documentation, 2024. URL `https://developer.android.com/reference/android/hardware/SensorManager#SENSOR_DELAY_FASTEST`. Accessed on 18 June. 2024.

Marinha do Brasil. Regulamentos de uniformes da marinha do brasil (rumb), volume 2, 2024. URL `https://www.marinha.mil.br/com3dn/sites/www.marinha.mil.br.com3dn/files/Rumb%20parte%202_0.pdf`. Acessado em 20 de junho de 2024.

Z. Hanusz, J. Tarasinska, and W. Zielinski. Shapiro–wilk test with known mean. *REVSTAT-Statistical Journal*, 14(1):89–100, 2016. doi: 10.57805/revstat.v14i1.180. URL `https://doi.org/10.57805/revstat.v14i1.180`.

Patrick E. McKnight and Julius Najab. Mann-whitney u test. *The Corsini Encyclopedia of Psychology*, 2010a. doi: 10.1002/9780470479216.corpsy0524. URL `https://doi.org/10.1002/9780470479216.corpsy0524`.

Patrick E. McKnight and Julius Najab. Kruskal-wallis test. *The Corsini Encyclopedia of Psychology*, 2010b. doi: 10.1002/9780470479216.corpsy0491. URL `https://doi.org/10.1002/9780470479216.corpsy0491`.