



GIOVANNA GOUVÊA SPURI DE MIRANDA

**CLASSIFICAÇÃO DE SINAIS DE
ELETROENCEFALOGRAMA UTILIZANDO TÉCNICAS DE
APRENDIZADO DE MÁQUINA PARA CONTROLE DE
CADEIRA DE RODAS**

LAVRAS – MG

2026

GIOVANNA GOUVÊA SPURI DE MIRANDA

**CLASSIFICAÇÃO DE SINAIS DE ELETROENCEFALOGRAMA UTILIZANDO
TÉCNICAS DE APRENDIZADO DE MÁQUINA PARA CONTROLE DE CADEIRA
DE RODAS**

Dissertação de mestrado apresentada à
Universidade Federal de Lavras como parte
das exigências do Programa de Pós-graduação
em Engenharia de Sistemas e Automação, para
obtenção do título de Mestre

Prof. Dr. Wilian Soares Lacerda
Orientador

LAVRAS – MG

2026

GIOVANNA GOUVÊA SPURI DE MIRANDA

**CLASSIFICAÇÃO DE SINAIS DE ELETROENCEFALOGRAMA UTILIZANDO
TÉCNICAS DE APRENDIZADO DE MÁQUINA PARA CONTROLE DE CADEIRA
DE RODAS**

Dissertação de mestrado apresentada à
Universidade Federal de Lavras como parte
das exigências do Programa de Pós-graduação
em Engenharia de Sistemas e Automação, para
obtenção do título de Mestre

APROVADA em 27 fevereiro 2026.

Prof. Dr. Wilian Soares Lacerda	UFLA
Prof. Dr. Henrique Luis Moreira Monteiro	UFLA
Prof. Dr. Cristiano Leite Castro	UFMG

Documento assinado digitalmente
 **WILIAN SOARES LACERDA**
Data: 31/03/2026 09:13:44-0300
Verifique em <https://validar.iti.gov.br>

Prof. Dr. Wilian Soares Lacerda
Orientador

**LAVRAS – MG
2026**

**Ficha Catalográfica elaborada pelo Sistema de Geração
de Ficha Catalográfica da Biblioteca Universitária da UFLA, com
dados informados pelo(a) próprio(a) autor(a).**

Miranda, Giovanna Gouvea Spuri.

Classificação De Sinais De Eletroencefalograma Utilizando Técnicas De
Aprendizado De Máquina Para Controle De Cadeira De Rodas / Giovanna Gouvea
Spuri Miranda. - 2026.

213 p. : il.

Orientador: Wilian Soares Lacerda

Dissertação (Mestrado Acadêmico) - Universidade Federal de Lavras, 2026.
Bibliografia.

1. Algoritmos Inteligentes. 2. Interface-Cerebro-Computador. 3. Cadeira de
rodas. I. Lacerda, Wilian Soares . II. Universidade Federal de Lavras. III. Título.

AGRADECIMENTOS

À Universidade Federal de Lavras (UFLA) e ao Programa de Pós-Graduação em Engenharia de Sistemas e Automação, pela oportunidade de realização deste curso de mestrado e por toda a infraestrutura disponibilizada para o desenvolvimento desta pesquisa.

Ao meu orientador, Prof. Dr. Wilian Soares Lacerda, pela orientação, confiança e pelos ensinamentos que foram fundamentais para a conclusão deste trabalho.

Aos membros da banca examinadora, Prof. Dr. Henrique Luis Moreira Monteiro (UFLA) e Prof. Dr. Cristiano Leite Castro (UFMG), pelas contribuições e sugestões que enriqueceram este trabalho.

À Fundação de Amparo à Pesquisa do Estado de Minas Gerais (FAPEMIG), pelo apoio financeiro que viabilizou a realização desta pesquisa.

À minha família, pelo apoio incondicional, incentivo e compreensão em todos os momentos.

Aos colegas do laboratório e do programa de pós-graduação, pelo companheirismo e troca de experiências.

A todos que, direta ou indiretamente, contribuíram para a realização deste trabalho.

RESUMO

Este trabalho investigou a classificação de sinais de eletroencefalografia (EEG) de imagética motora do Projeto Milo para aplicações em interfaces cérebro-computador, comparando duas estratégias de decisão: (i) uma arquitetura plana multiclasse (Repouso, Esquerda e Direita) e (ii) uma arquitetura em cascata composta por duas etapas (Repouso $\times$ Movimento e, condicionado a Movimento, Esquerda $\times$ Direita). O pipeline experimental adotou divisão em nível de sessão para mitigação de vazamento de dados (treinamento 55%, validação 20% e teste 25%), pré-processamento temporal com remoção de tendência linear e filtragem, seguido de segmentação em janelas de 2 s (500 amostras a 250 Hz). A extração de características baseou-se em geometria riemanniana, incluindo a estimativa de matrizes de covariância regularizadas, o cálculo da média log-euclidiana exclusivamente no conjunto de treinamento e a projeção no espaço tangente, com posterior remoção de redundância e seleção univariada de atributos. O desbalanceamento entre classes foi tratado apenas no conjunto de treinamento, avaliando-se técnicas de reamostragem como *Random Over-Sampling* e *Random Under-Sampling*, *Synthetic Minority Over-sampling Technique* (SMOTE), *Borderline Synthetic Minority Over-sampling Technique*, *Adaptive Synthetic Sampling* (ADASYN), *Tomek Links*, *Edited Nearest Neighbors* (ENN), *NearMiss* e abordagens híbridas, como *Synthetic Minority Over-sampling Technique combined with Tomek Links* e *Synthetic Minority Over-sampling Technique combined with Edited Nearest Neighbors*. Foram treinados e comparados os classificadores *Multilayer Perceptron* (MLP), *Support Vector Machine* (SVM), *Random Forest*, *Extra Trees*, *Gradient Boosting*, *Extreme Gradient Boosting* (XGBoost) e um comitê por votação suave, com otimização de hiperparâmetros via *random search* (100 iterações) e validação cruzada estratificada em cinco dobras, adotando-se o $F1_{\text{macro}}$ como critério principal, além da estimativa de intervalos de confiança de 95% por *bootstrap*. A melhor solução global em métricas baseadas em rótulo foi obtida na arquitetura plana com SVM e SMOTE-ENN (kernel RBF, $C = 0,1$, $\gamma = 10^{-4}$, ponderação balanceada e 13 atributos), alcançando acurácia de $0,826 \pm 0,013$, acurácia balanceada de $0,825 \pm 0,013$, $F1_{\text{macro}} = 0,822 \pm 0,014$ e $MCC = 0,748 \pm 0,019$; na arquitetura em cascata, o melhor de-

sempenho foi obtido pelo modelo *Extra Trees* com SMOTE ($F1_{\text{macro}} = 0,796 \pm 0,015$), porém inferior ao melhor cenário plano segundo o critério de otimização adotado.

Palavras-chave: algoritmos inteligentes; cadeira de rodas; eletroencefalograma; interface cérebro-computador; sinais cerebrais.

ABSTRACT

This work investigated the classification of motor imagery electroencephalography (EEG) signals from the Milo Project for brain–computer interface applications, comparing two decision strategies: (i) a flat multiclass architecture (Rest, Left, and Right) and (ii) a cascade architecture composed of two stages (Rest $\times$ Movement and, conditioned on Movement, Left $\times$ Right). The experimental pipeline adopted a session-level data split to mitigate data leakage (55% training, 20% validation, and 25% testing), temporal preprocessing with linear trend removal and filtering, followed by segmentation into 2 s windows (500 samples at 250 Hz). Feature extraction was based on Riemannian geometry, including the estimation of regularized covariance matrices, the computation of the log-Euclidean mean exclusively on the training set, and projection onto the tangent space, followed by redundancy removal and univariate feature selection. Class imbalance was addressed only on the training set, evaluating resampling techniques such as Random Over-Sampling and Random Under-Sampling, Synthetic Minority Over-sampling Technique (SMOTE), Borderline Synthetic Minority Over-sampling Technique, Adaptive Synthetic Sampling (ADASYN), Tomek Links, Edited Nearest Neighbors (ENN), NearMiss, and hybrid approaches, such as Synthetic Minority Over-sampling Technique combined with Tomek Links and Synthetic Minority Over-sampling Technique combined with Edited Nearest Neighbors. The classifiers Multilayer Perceptron (MLP), Support Vector Machine (SVM), Random Forest, Extra Trees, Gradient Boosting, Extreme Gradient Boosting (XGBoost), and a soft-voting ensemble were trained and compared, with hyperparameter optimization via random search (100 iterations) and five-fold stratified cross-validation, adopting $F1_{\text{macro}}$ as the primary criterion, in addition to estimating 95% confidence intervals via bootstrap. The best overall solution in label-based metrics was obtained with the flat architecture using SVM and SMOTE-ENN (RBF kernel, $C = 0.1$, $\gamma = 10^{-4}$, balanced class weighting, and 13 features), achieving an accuracy of 0.826 ± 0.013 , balanced accuracy of 0.825 ± 0.013 , $F1_{\text{macro}} = 0.822 \pm 0.014$, and $MCC = 0.748 \pm 0.019$; in the cascade architecture, the best performance was achieved by the Extra Trees model with SMOTE ($F1_{\text{macro}} = 0.796 \pm 0.015$), although inferior to the best flat scenario according to the adopted optimization criterion.

Keywords: brain-computer interface (BCI); brain signals; electroencephalography (EEG); intelligent algorithms; wheelchair.

INDICADORES DE IMPACTO

O impacto social do trabalho alinha-se diretamente com os Objetivos de Desenvolvimento Sustentável da ONU, particularmente o ODS 10 (Redução das Desigualdades) e ODS 3 (Saúde e Bem-Estar), ao contribuir para desenvolvimento de tecnologias que promovem autonomia e inclusão de pessoas com deficiências neuromotoras severas. A viabilidade técnica demonstrada beneficia diretamente indivíduos com esclerose lateral amiotrófica, lesões medulares e síndrome do encarceramento, populações historicamente excluídas do acesso a tecnologias de mobilidade avançadas devido aos altos custos dos sistemas comerciais disponíveis. A redução de barreiras tecnológicas e econômicas promove maior equidade no acesso a soluções assistivas, contribuindo para construção de sociedade mais inclusiva. A relevância para políticas públicas evidencia-se na demonstração de viabilidade de implementação de tecnologias assistivas avançadas em sistemas de saúde pública e programas governamentais de distribuição de equipamentos para pessoas com deficiência. Os resultados fornecem base técnica para elaboração de especificações técnicas em editais de compras públicas de cadeiras de rodas inteligentes, contribuindo para modernização do Sistema Único de Saúde (SUS) na oferta de tecnologias assistivas. A metodologia desenvolvida pode embasar políticas de fomento à pesquisa e desenvolvimento de tecnologias assistivas nacionais, reduzindo dependência de importações e fortalecendo a cadeia produtiva nacional de equipamentos médicos e assistivos. A disseminação dos resultados através de publicações científicas e apresentações em eventos especializados contribui para formação de recursos humanos especializados em interfaces cérebro-computador, área estratégica para competitividade nacional em tecnologias de saúde digital. A metodologia desenvolvida pode ser incorporada em disciplinas de pós-graduação em engenharia biomédica e ciência da computação, multiplicando o impacto educacional da pesquisa. Parcerias com institutos de pesquisa e universidades internacionais podem ser estabelecidas com base nos resultados obtidos, fortalecendo a inserção internacional da pesquisa brasileira em BCI.

IMPACT INDICATORS

The social impact of this work is directly aligned with the United Nations Sustainable Development Goals, particularly SDG 10 (Reduced Inequalities) and SDG 3 (Good Health and Well-Being), by contributing to the development of technologies that promote autonomy and inclusion for individuals with severe neuromotor disabilities. The demonstrated technical feasibility directly benefits individuals with amyotrophic lateral sclerosis, spinal cord injuries, and locked-in syndrome—populations that have historically been excluded from access to advanced mobility technologies due to the high costs of commercially available systems. Reducing technological and economic barriers promotes greater equity in access to assistive solutions, contributing to the construction of a more inclusive society. The relevance to public policy is evidenced by the demonstration of the feasibility of implementing advanced assistive technologies within public healthcare systems and government programs for the distribution of equipment to persons with disabilities. The results provide a technical basis for the development of technical specifications in public procurement calls for intelligent wheelchairs, contributing to the modernization of the Brazilian Unified Health System (SUS) in the provision of assistive technologies. The developed methodology can support policies that foster research and development of national assistive technologies, reducing dependence on imports and strengthening the national production chain of medical and assistive equipment. The dissemination of results through scientific publications and presentations at specialized events contributes to the training of human resources specialized in brain–computer interfaces, a strategic area for national competitiveness in digital health technologies. The developed methodology can be incorporated into graduate-level courses in biomedical engineering and computer science, multiplying the educational impact of the research. Partnerships with international research institutes and universities can be established based on the obtained results, strengthening the international presence of Brazilian research in BCI.

LISTA DE FIGURAS

Figura 2.1 – Posições para a colocação de eletrodos de EEG segundo o Sistema Internacional 10-20.	26
Figura 2.2 – Geração de campos elétricos por correntes sinápticas em células piramidais	27
Figura 2.3 – Sistema 10-20	28
Figura 2.4 – Geração de sinais grandes de EEG por atividade sincronizada.	29
Figura 2.5 – Ritmos no tálamo impulsionando ritmos no córtex cerebral.	30
Figura 2.6 – Ritmos do EEG durante o sono.	32
Figura 2.7 – Medições bipolares e unipolares	34
Figura 4.1 – Pipeline geral do sistema de ICC implementado no projeto Milo	40
Figura 4.2 – Interface de aquisição OpenBCI	41
Figura 4.3 – Configuração dos eletrodos segundo o sistema internacional 10-20	42
Figura 4.4 – Paradigma experimental implementado para coleta de dados	42
Figura 4.5 – Protocolo experimental demonstrando a sequência temporal	43
Figura 4.6 – Interface gráfica do sistema Milo	44
Figura 4.7 – Segmentação temporal dos dados	45
Figura 5.1 – Fluxo do pipeline de processamento e classificação de EEG	48
Figura 5.2 – Fluxos de classificação: multiclasse e cascata	49
Figura 6.1 – Matriz de confusão percentual do MLP	102
Figura 6.2 – Matriz de confusão percentual do MLP sem e com balanceamento	108
Figura 6.3 – Matriz de confusão percentual do Extra Trees sem balanceamento	114
Figura 6.4 – Matrizes de confusão percentuais do Extra Trees na configuração plana	119
Figura 6.5 – Curvas ROC do Extra Trees na configuração plana	120
Figura 6.6 – Curvas Precision–Recall do Extra Trees na configuração plana	120
Figura 6.7 – Matrizes de confusão percentuais do Extra Trees na configuração em cascata	124
Figura 6.8 – Curvas ROC do Extra Trees na configuração em cascata	124
Figura 6.9 – Curvas Precision–Recall do Extra Trees na configuração em cascata	125
Figura 6.10 – Matriz de confusão percentual do Random Forest sem balanceamento	131
Figura 6.11 – Matrizes de confusão percentuais do Random Forest na configuração plana	134
Figura 6.12 – Curvas ROC do Random Forest na configuração plana	135
Figura 6.13 – Curvas Precision–Recall do Random Forest na configuração plana	135

Figura 6.14 – Matriz de confusão percentual do Gradient Boosting	141
Figura 6.15 – Matriz de confusão percentual do SVM sem balanceamento	146
Figura 6.16 – Matrizes de confusão percentuais do SVM na configuração plana	149
Figura 6.17 – Curvas ROC do SVM na configuração plana	150
Figura 6.18 – Curvas Precision–Recall do SVM na configuração plana	151
Figura 6.19 – Matrizes de confusão percentuais do SVM na configuração cascata	154
Figura 6.20 – Curvas ROC do SVM na configuração cascata	155
Figura 6.21 – Curvas Precision–Recall do SVM na configuração cascata	156
Figura 6.22 – Matriz de confusão percentual do XGBoost sem balanceamento	162
Figura 6.23 – Matrizes de confusão percentuais do XGBoost na configuração plana	165
Figura 6.24 – Curvas ROC do XGBoost na configuração plana	165
Figura 6.25 – Curvas Precision–Recall do XGBoost na configuração plana	166
Figura 6.26 – Matrizes de confusão percentuais do XGBoost na configuração cascata	168
Figura 6.27 – Curvas ROC do XGBoost na configuração cascata	169
Figura 6.28 – Curvas Precision–Recall do XGBoost na configuração cascata	169
Figura 6.29 – Matriz de confusão percentual do Comitê	174
Figura 6.30 – Curvas ROC do Comitê na configuração plana	178
Figura 6.31 – Curvas Precision–Recall do Comitê na configuração plana	178
Figura 6.32 – Matrizes de confusão percentuais do Comitê na configuração plana	179
Figura 6.33 – Curvas ROC do Comitê na configuração cascata	182
Figura 6.34 – Curvas Precision–Recall do Comitê na configuração cascata	182
Figura 6.35 – Matrizes de confusão percentuais do Comitê na configuração cascata	183

LISTA DE TABELAS

Tabela 4.1 – Estrutura dos dados brutos conforme registrados pelo sistema OpenBCI . . .	44
Tabela 4.2 – Estrutura dos dados processados após segmentação e rotulação	45
Tabela 5.1 – Dimensionalidade das características Riemannianas em função do número de canais.	56
Tabela 5.2 – Configuração do procedimento de otimização de hiperparâmetros através de busca aleatória com validação cruzada. Os valores foram estabelecidos buscando equilíbrio entre capacidade de exploração do espaço de hiperpa- râmetros e custo computacional.	77
Tabela 5.3 – Síntese das métricas de avaliação utilizadas no pipeline de classificação BCI.	92
Tabela 5.4 – Configuração do pipeline de classificação e avaliação.	96
Tabela 5.5 – Critérios de filtragem para composição do comitê de classificadores.	97
Tabela 6.1 – Desempenho do classificador MLP sem técnica de reamostragem externa .	99
Tabela 6.2 – Desempenho por classe do MLP	100
Tabela 6.3 – Métricas por etapa da configuração em cascata — MLP	101
Tabela 6.4 – Áreas sob as curvas ROC e PR por classe e configuração — MLP	103
Tabela 6.5 – Métricas de calibração por classe e configuração — MLP	104
Tabela 6.6 – Comparação do MLP sem e com Random Oversampling	106
Tabela 6.7 – Desempenho por classe do MLP com e sem balanceamento	107
Tabela 6.8 – Áreas sob as curvas ROC e PR por classe — MLP sem e com balanceamento	109
Tabela 6.9 – Métricas de calibração por classe — MLP sem e com balanceamento . . .	110
Tabela 6.10 – Síntese comparativa das configurações do MLP	110
Tabela 6.11 – Desempenho do classificador Extra Trees sem técnica de balanceamento . .	112
Tabela 6.12 – Desempenho por classe do Extra Trees sem balanceamento	113
Tabela 6.13 – Métricas por etapa da configuração em cascata — Extra Trees	115
Tabela 6.14 – Áreas sob as curvas ROC e PR por classe e configuração — Extra Trees sem balanceamento	116
Tabela 6.15 – Métricas de calibração por classe e configuração — Extra Trees sem balan- ceamento	116
Tabela 6.16 – Comparação do Extra Trees em configuração plana sem e com ADASYN .	118

Tabela 6.17 – Desempenho por classe do Extra Trees em configuração plana sem e com ADASYN	118
Tabela 6.18 – Áreas sob as curvas ROC e PR por classe — Extra Trees (plana): sem balanceamento versus ADASYN	120
Tabela 6.19 – Métricas de calibração por classe — Extra Trees (plana): sem balanceamento versus ADASYN	121
Tabela 6.20 – Comparação do Extra Trees em cascata sem e com SMOTE	122
Tabela 6.21 – Desempenho por classe do Extra Trees em cascata sem e com SMOTE . . .	123
Tabela 6.22 – Áreas sob as curvas ROC e PR por classe — Extra Trees (cascata): sem balanceamento versus SMOTE	125
Tabela 6.23 – Métricas de calibração por classe — Extra Trees (cascata): sem balanceamento versus SMOTE	126
Tabela 6.24 – Síntese comparativa das configurações do Extra Trees	126
Tabela 6.25 – Desempenho do classificador Random Forest sem técnica de balanceamento	129
Tabela 6.26 – Desempenho por classe do Random Forest sem balanceamento	130
Tabela 6.27 – Métricas por etapa da configuração em cascata – Random Forest	130
Tabela 6.28 – Áreas sob as curvas ROC e PR por classe e configuração — Random Forest sem balanceamento	132
Tabela 6.29 – Métricas de calibração por classe e configuração — Random Forest sem balanceamento	132
Tabela 6.30 – Comparação do Random Forest plano sem e com SMOTE	133
Tabela 6.31 – Desempenho por classe do Random Forest plano sem e com SMOTE . . .	134
Tabela 6.32 – Áreas sob as curvas ROC e PR por classe — Random Forest (plana): sem balanceamento versus SMOTE	135
Tabela 6.33 – Métricas de calibração por classe — Random Forest (plana): sem balanceamento versus SMOTE	136
Tabela 6.34 – Síntese comparativa das configurações do Random Forest	136
Tabela 6.35 – Desempenho do classificador Gradient Boosting sem técnica de reamostragem externa	138
Tabela 6.36 – Desempenho por classe do Gradient Boosting	139
Tabela 6.37 – Métricas por etapa da configuração em cascata – Gradient Boosting	140

Tabela 6.38 – Áreas sob as curvas ROC e PR por classe e configuração — Gradient Boosting	142
Tabela 6.39 – Métricas de calibração por classe e configuração — Gradient Boosting . . .	142
Tabela 6.40 – Desempenho do classificador SVM sem técnica de balanceamento	143
Tabela 6.41 – Desempenho por classe do SVM sem balanceamento	144
Tabela 6.42 – Métricas por etapa da configuração em cascata — SVM	145
Tabela 6.43 – Áreas sob as curvas ROC e PR por classe e configuração — SVM sem balanceamento	147
Tabela 6.44 – Métricas de calibração por classe e configuração — SVM sem balanceamento	147
Tabela 6.45 – Comparação do SVM plano sem e com SMOTEENN	148
Tabela 6.46 – Desempenho por classe do SVM plano sem e com SMOTEENN	149
Tabela 6.47 – Comparação do SVM em cascata sem e com SMOTEENN	152
Tabela 6.48 – Desempenho por classe do SVM em cascata sem e com SMOTEENN . . .	153
Tabela 6.49 – Áreas sob as curvas ROC e PR por classe — SVM cascata sem balancea- mento e com SMOTEENN	156
Tabela 6.50 – Métricas de calibração por classe — SVM cascata sem balanceamento e com SMOTEENN	157
Tabela 6.51 – Síntese comparativa das configurações do SVM	157
Tabela 6.52 – Métricas de desempenho global do XGBoost nas configurações plana e cascata sem balanceamento, com intervalos de confiança bootstrap de 95%	159
Tabela 6.53 – Métricas de desempenho do XGBoost por classe nas configurações plana e cascata sem balanceamento	160
Tabela 6.54 – Métricas por etapa da configuração em cascata — XGBoost	160
Tabela 6.55 – Áreas sob as curvas ROC e PR por classe e configuração — XGBoost sem balanceamento	162
Tabela 6.56 – Métricas de calibração por classe e configuração — XGBoost sem balan- ceamento	163
Tabela 6.57 – Comparação do XGBoost plano sem balanceamento e com SMOTEENN .	164
Tabela 6.58 – Desempenho por classe do XGBoost plano sem balanceamento e com SMO- TEENN	164
Tabela 6.59 – Áreas sob as curvas ROC e PR por classe — XGBoost plano sem balance- amento e com SMOTEENN	166

Tabela 6.60 – Comparação do XGBoost cascata sem balanceamento e com SMOTEENN	167
Tabela 6.61 – Desempenho por classe do XGBoost cascata sem balanceamento e com SMOTEENN	168
Tabela 6.62 – Áreas sob as curvas ROC e PR por classe — XGBoost cascata sem balanceamento e com SMOTEENN	170
Tabela 6.63 – Síntese comparativa das configurações do XGBoost	170
Tabela 6.64 – Desempenho do classificador Comitê sem técnica de reamostragem externa	172
Tabela 6.65 – Desempenho por classe do Comitê	173
Tabela 6.66 – Áreas sob as curvas ROC e PR por classe e configuração — Comitê	175
Tabela 6.67 – Métricas de calibração por classe e configuração — Comitê	175
Tabela 6.68 – Comparação do Comitê em plana sem e com ADASYN	176
Tabela 6.69 – Desempenho por classe do Comitê em plana sem e com ADASYN	177
Tabela 6.70 – Áreas sob as curvas ROC e PR por classe — Comitê (plana): sem balanceamento versus ADASYN	177
Tabela 6.71 – Métricas de calibração por classe — Comitê (plana): sem balanceamento versus ADASYN	179
Tabela 6.72 – Comparação do Comitê em cascata sem e com Random Oversampling . . .	180
Tabela 6.73 – Desempenho por classe do Comitê em cascata sem e com Random Oversampling	181
Tabela 6.74 – Áreas sob as curvas ROC e PR por classe — Comitê (cascata): sem balanceamento versus Random Oversampling	181
Tabela 6.75 – Métricas de calibração por classe — Comitê (cascata): sem balanceamento versus Random Oversampling	183
Tabela 6.76 – Síntese comparativa das configurações do Comitê	184
Tabela 6.77 – Maiores valores por classificador na configuração plana (média, desvio-padrão e IC 95% <i>bootstrap</i>). Em cada linha, destaca-se o maior valor; para Brier Score, considera-se o menor valor.	185
Tabela 6.78 – Desempenho dos classificadores na configuração em cascata (média, desvio-padrão e IC 95% <i>bootstrap</i>). Em cada linha, destaca-se o maior valor; para Brier Score, considera-se o menor valor.	186

Tabela 6.79 – Cenário sem balanceamento: comparação computacional entre as configurações plano e cascata para todos os modelos avaliados.	189
Tabela 6.80 – Configuração plana: impacto computacional do balanceamento restrito aos melhores pares modelo–técnica, comparados ao sem balanceamento do mesmo modelo.	190
Tabela 6.81 – Configuração em cascata: impacto computacional do balanceamento restrito aos melhores pares modelo–técnica, comparados ao sem balanceamento do mesmo modelo.	192

SUMÁRIO

1	INTRODUÇÃO	16
1.1	Contextualização e Motivação	17
1.2	Problema de Pesquisa	18
1.3	Objetivos	19
1.4	Justificativa	21
1.5	Estrutura do Trabalho	22
2	INTERFACES CÉREBRO-COMPUTADOR (ICC)	23
2.1	Tipos de ICC: invasivos e não-invasivos	23
2.2	Aplicações práticas em tecnologias assistivas	24
2.3	Desafios técnicos e limitações atuais	24
2.4	Estado da arte em BCI para controle de dispositivos	25
2.5	Princípios Neurofisiológicos e Eletroencefalografia	26
3	REVISÃO DOS ESTUDOS SOBRE CLASSIFICAÇÃO DE SINAIS EEG	35
4	PROJETO MILO: ICC PARA CONTROLE DE CADEIRA DE RODAS	40
4.1	Coleta de Dados Eletroencefalográficos	40
5	METODOLOGIA	47
5.1	Divisão dos Dados	49
5.2	Pré-processamento Temporal	50
5.3	Segmentação em Janelas Temporais	51
5.4	Geometria Riemanniana e Espaço Tangente	54
5.5	Remoção de Características Redundantes	56
5.6	Balanceamento de Classes (Reamostragem do Conjunto de Treinamento)	57
5.7	Seleção Automática do Método de Ranqueamento de Características	59
5.8	Modelos de Classificação	63
5.9	Otimização de Hiperparâmetros	76
5.10	Arquitetura Plana versus Arquitetura em Cascata	80
5.11	Métricas de Avaliação e Comparação Estatística	83
6	RESULTADOS E DISCUSSÃO	98
6.1	Perceptron Multicamadas	98
6.2	Extra Trees	111

6.3	Random Forest	128
6.4	Gradient Boosting	137
6.5	Máquinas de Vetores de Suporte	142
6.6	XGBoost	158
6.7	Comitê de Classificadores	171
6.8	Avaliação Comparativa dos Modelos Desenvolvidos	185
6.9	Impacto computacional	187
7	CONCLUSÃO	195
	REFERÊNCIAS	198

1 INTRODUÇÃO

A tecnologia assistiva tem desempenhado um papel crucial na melhoria da qualidade de vida de pessoas com deficiências motoras, permitindo maior autonomia e inclusão social (COOK; POLGAR, 2020). Nesse contexto, as Interfaces Cérebro-Computador (ICC), também conhecidas como *Brain-Computer Interfaces* (BCI) em inglês, destacam-se como uma tecnologia promissora, especialmente para indivíduos com severas limitações motoras, como pessoas com esclerose lateral amiotrófica (ELA), lesões medulares ou síndrome do encarceramento (MILLAN et al., 2010; NICOLAS-ALONSO; GOMEZ-GIL, 2012).

As ICCs baseadas em eletroencefalograma (EEG) têm ganhado particular atenção devido à sua natureza não invasiva, portabilidade, custo relativamente acessível e boa resolução temporal (LOTTE et al., 2018a; CLERC; BOUGRAIN; LOTTE, 2016; ABIRI et al., 2019). Estas interfaces captam a atividade elétrica do cérebro através de eletrodos posicionados no couro cabeludo, permitindo a interpretação e tradução desses sinais em comandos para dispositivos externos, como cadeiras de rodas, próteses ou sistemas de comunicação (MAK; WOLPAW, 2009; DALY; WOLPAW, 2008).

As primeiras pesquisas em BCI surgiram na década de 1970, quando Vidal (1973) propôs o termo "Interface Cérebro-Computador" e demonstrou a viabilidade de utilizar sinais cerebrais para controlar um cursor em um computador. Desde então, o campo progrediu substancialmente, impulsionado pelos avanços em neurociência, processamento de sinais e aprendizado de máquina. Wolpaw et al. (2002) apresentaram um dos primeiros sistemas BCI confiáveis para comunicação alternativa, permitindo que pessoas com paralisia severa controlassem aplicações de computador.

Apesar dos avanços significativos nessa área, o processamento e a classificação precisa dos sinais de EEG ainda representam desafios consideráveis devido à sua natureza não estacionária, baixa relação sinal-ruído e alta variabilidade entre indivíduos (BASHASHATI et al., 2007; LOTTE et al., 2007). Em particular, a detecção confiável de padrões cerebrais associados à imaginação motora, uma técnica amplamente utilizada em ICCs, requer algoritmos robustos capazes de extrair características relevantes e classificar corretamente os sinais, mesmo em condições adversas (PFURTSCHELLER et al., 2006; NEUPER et al., 2005).

Os algoritmos de aprendizado de máquina têm demonstrado grande potencial para enfrentar esses desafios, oferecendo métodos capazes de identificar padrões complexos em sinais

de EEG (LOTTE et al., 2018a; CRAIK; HE; CONTRERAS-VIDAL, 2019). A aplicação de técnicas como Redes Neurais Artificiais (RNA), Máquinas de Vetores de Suporte (do inglês, *Support Vector Machine* (SVM)), Floresta Aleatória (do inglês, *Random Forest* (RF)), Árvores Extremamente Aleatorizadas (do inglês, *Extra Trees*) e métodos baseados em boosting, como *Gradient Boosting* e *Extreme Gradient Boosting* (XGBoost), tem melhorado significativamente o desempenho dos sistemas de BCI (LOTTE et al., 2007; RASHID et al., 2020).

1.1 Contextualização e Motivação

Segundo dados da Organização Mundial da Saúde, mais de 1 bilhão de pessoas vivem com algum tipo de deficiência, representando aproximadamente 15% da população mundial (ORGANIZATION; BANK, 2011). No Brasil, de acordo com o Censo de 2022, cerca de 18,6 milhões de pessoas possuem alguma deficiência que afeta severamente sua vida diária, sendo que aproximadamente 3,5 milhões necessitam de cadeiras de rodas (IBGE, 2022). Esses números evidenciam a necessidade de tecnologias assistivas que promovam maior autonomia e inclusão social para essa parcela significativa da população.

Entre as pessoas com deficiência motora severa, muitas preservam suas funções cognitivas intactas, o que torna as interfaces cérebro-computador particularmente relevantes como meio para restabelecer algum grau de independência (KUBLER; HOLZ; BOTREL, 2013; MILLAN et al., 2010). Quando essas interfaces são utilizadas para controlar cadeiras de rodas, podem proporcionar mobilidade autônoma para indivíduos que, de outra forma, dependeriam completamente de cuidadores para locomoção (GALÁN et al., 2008; MILLAN et al., 2009).

Os desafios enfrentados por pessoas com limitações motoras severas, como aquelas com síndrome do encarceramento, esclerose lateral amiotrófica ou lesão medular alta, são substanciais. Estas condições frequentemente resultam em uma capacidade extremamente limitada ou inexistente de movimento voluntário, apesar da preservação da função cognitiva (LAUREYS et al., 2005). Para estes indivíduos, a dependência de cuidadores para tarefas básicas como locomoção representa não apenas uma perda de autonomia, mas também um impacto significativo na qualidade de vida e bem-estar psicológico (BIRBAUMER, 2006).

As interfaces de controle tradicionais para cadeiras de rodas, como *joysticks*, controles de sopro-e-sucção ou tabuleiros de comandos, apresentam limitações consideráveis para usuários com paralisia severa. Estudo realizado por Simpson, LoPresti e Cooper (2008) indicou que

aproximadamente 40% dos usuários de cadeiras de rodas tradicionais enfrentam dificuldades operacionais significativas, incluindo limitações associadas às habilidades cognitivas e motoras necessárias para o controle convencional. Essas dificuldades são amplificadas em indivíduos com condições neurológicas progressivas, como Esclerose Lateral Amiotrófica (ELA), ou em casos de paralisia severa (BIRBAUMER; COHEN, 2006; KUBLER; HOLZ; BOTREL, 2013).

Além disso, as tecnologias assistivas convencionais frequentemente falham em atender às necessidades de pessoas com deficiências motoras mais graves, criando uma lacuna tecnológica que afeta diretamente a qualidade de vida desses indivíduos (COOK; POLGAR, 2020). Nesse cenário, o desenvolvimento de interfaces cérebro-computador para controle de cadeiras de rodas representa não apenas uma inovação tecnológica, mas também uma resposta a uma necessidade social urgente (MILLAN et al., 2010).

A relevância social e o impacto potencial na qualidade de vida dos usuários constituem motivações centrais para o desenvolvimento de BCIs para cadeiras de rodas.

A viabilidade econômica de soluções BCI de baixo custo representa um fator motivacional adicional. Sistemas comerciais de EEG médico podem custar dezenas de milhares de dólares, tornando-os inacessíveis para a maioria dos potenciais usuários, particularmente em países de baixa e média renda (QIAN et al., 2018) demonstraram que sistemas baseados em plataformas de código aberto como o OpenBCI podem reduzir o custo em até 90% comparado a sistemas comerciais, mantendo uma qualidade de sinal adequada para aplicações de BCI.

1.2 Problema de Pesquisa

Apesar do potencial promissor das interfaces cérebro-computador baseadas em EEG para controle de cadeiras de rodas, vários desafios técnicos e práticos ainda limitam sua aplicação generalizada. Um dos principais desafios é a classificação precisa e em tempo real dos sinais de EEG, especialmente aqueles relacionados à imaginação motora ou a artefatos como piscadas de olhos, que podem ser utilizados como sinais de controle (LOTTE et al., 2007; MULLER et al., 2008).

Os algoritmos de aprendizado de máquina oferecem soluções potenciais para esse problema, mas a escolha e configuração adequadas desses algoritmos para aplicações específicas de BCI ainda representam um campo aberto de investigação (LOTTE et al., 2018a; CRAIK; HE; CONTRERAS-VIDAL, 2019). Diferentes métodos podem apresentar vantagens distintas

em termos de precisão, velocidade de processamento, adaptabilidade a diferentes usuários e robustez a ruídos e interferências (LOTTE et al., 2007; BASHASHATI et al., 2007).

Nos últimos cinco anos, os sistemas BCI para cadeiras de rodas têm incorporado cada vez mais técnicas de aprendizado profundo. Schirrmester et al. (2017) demonstraram que redes neurais convolucionais podem melhorar significativamente a classificação de sinais de EEG relacionados à imaginação motora. Além disso, Zhang et al. (2021) integraram sistemas BCI com tecnologias de condução autônoma, criando soluções híbridas que equilibram o controle do usuário com a navegação assistida inteligente.

Nesse contexto, este trabalho propõe investigar a seguinte questão de pesquisa: quais algoritmos de aprendizado de máquina e quais configurações de hiperparâmetros apresentam melhor desempenho na classificação de sinais de EEG aplicados ao controle de cadeiras de rodas, considerando métricas de avaliação como acurácia, precisão, recall e F1-score.

Para responder a essa questão, serão analisados e comparados diferentes modelos de aprendizado de máquina aplicados à classificação de sinais de EEG de imagética motora, incluindo Multilayer Perceptron (MLP), Máquinas de Vetores de Suporte (SVM), Árvores Extremamente Aleatorizadas (Extra Trees), Random Forest, métodos baseados em Gradient Boosting, Extreme Gradient Boosting (XGBoost) e um comitê de classificação por votação suave (soft voting), composto pelos modelos com melhor desempenho individual. Adicionalmente, investiga-se o impacto do desbalanceamento entre classes por meio de técnicas de reamostragem aplicadas exclusivamente ao conjunto de treinamento. Por fim, são comparadas duas estratégias de decisão, a arquitetura plana multiclasse e a arquitetura em cascata em duas etapas, com otimização de hiperparâmetros realizada via random search e validação cruzada estratificada, adotando-se o $F1_{macro}$ como critério de seleção.

1.3 Objetivos

1.3.1 Objetivo Geral

O objetivo geral deste trabalho é avaliar e comparar o desempenho de modelos de aprendizado de máquina na classificação de sinais de EEG de imagética motora para aplicações em BCI, visando identificar a configuração mais robusta e adequada ao controle de cadeiras de rodas.

1.3.2 Objetivos Específicos

Para alcançar o objetivo geral, foram definidos os seguintes objetivos específicos:

- Preparar os dados do Projeto Milo (AHAMEZ; TEAM, 2024), fazendo o particionamento por sessão em treino/validação/teste, imputação e normalização aplicadas de forma consistente para evitar vazamento de informação;
- Realizar o pré-processamento dos sinais de EEG, incluindo remoção de tendência, filtragem notch em 60 Hz e filtragem passa-banda em 4–40 Hz, seguido de segmentação em janelas temporais de 2 s com sobreposição configurável por conjunto;
- Extrair características discriminativas por meio de técnicas baseadas em geometria riemanniana, estimando matrizes de covariância regularizadas e projetando-as no espaço tangente, complementadas por remoção de características redundantes e seleção automática de atributos;
- Investigar estratégias para mitigação do desbalanceamento de classes, aplicando reamostragem exclusivamente no conjunto de treinamento com métodos de subamostragem, sobreamostragem e combinações híbridas (*Random Over/UnderSampling*, SMOTE, Borderline-SMOTE, ADASYN, Tomek Links, ENN, NearMiss, SMOTE–Tomek e SMOTE–ENN).
- Implementar e comparar modelos de classificação (MLP, SVM, *Random Forest*, *Extra Trees*, *Gradient Boosting*, XGBoost) e um comitê por votação suave, avaliando também duas arquiteturas de decisão: (i) classificação plana multiclasse e (ii) classificação em cascata em duas etapas;
- Otimizar hiperparâmetros por *random search* com validação cruzada estratificada, adotando $F1_{\text{macro}}$ como métrica de seleção, e estimar desempenho final com métricas complementares (acurácia, acurácia balanceada, $F1_{\text{macro}}$, *recall* por classe e *MCC*), incluindo intervalos de confiança por *bootstrap*;
- Identificar a melhor configuração final (modelo, arquitetura de decisão, seleção de características e estratégia de balanceamento) a partir da comparação sistemática dos cenários avaliados.

1.4 Justificativa

A relevância deste trabalho fundamenta-se em três pilares principais: o impacto social potencial das tecnologias assistivas baseadas em BCI, a contribuição científica para o avanço do conhecimento sobre classificação de sinais de EEG e a aplicabilidade prática dos resultados obtidos.

No aspecto social, o desenvolvimento de interfaces cérebro-computador mais precisas e eficientes para controle de cadeiras de rodas pode transformar significativamente a vida de pessoas com deficiências motoras severas, promovendo maior autonomia, inclusão e qualidade de vida (KUBLER; HOLZ; BOTREL, 2013; MILLAN et al., 2010). Numa perspectiva mais ampla, esse tipo de tecnologia assistiva contribui para a construção de uma sociedade mais inclusiva, alinhando-se com os Objetivos de Desenvolvimento Sustentável da ONU, particularmente com o objetivo 10, que visa reduzir as desigualdades (United Nations, 2015).

Do ponto de vista científico, a análise comparativa de diferentes algoritmos de aprendizado de máquina para classificação de sinais de EEG contribui para o avanço do conhecimento em um campo interdisciplinar que une neurociência, engenharia biomédica e computação (LOTTE et al., 2018a; CRAIK; HE; CONTRERAS-VIDAL, 2019). A investigação sistemática do efeito de diferentes configurações de hiperparâmetros pode revelar *insights* valiosos sobre o comportamento desses algoritmos quando aplicados a dados neurofisiológicos, potencialmente beneficiando outras aplicações além do controle de cadeiras de rodas (LOTTE et al., 2007; BASHASHATI et al., 2007).

Quanto à aplicabilidade prática, os resultados deste trabalho podem fornecer diretrizes concretas para desenvolvedores de interfaces cérebro-computador, ajudando-os a selecionar os algoritmos e configurações mais adequados para suas aplicações específicas (CLERC; BOUGRAIN; LOTTE, 2016) destacam a importância de desenvolver soluções adaptadas às realidades socioeconômicas locais, especialmente considerando que a maior parte dos avanços tecnológicos em BCI ocorre em países desenvolvidos, frequentemente ignorando as necessidades e limitações de países em desenvolvimento. Além disso, a metodologia desenvolvida para comparação de algoritmos pode servir como um *framework* para avaliação de novas abordagens de classificação de sinais de EEG que venham a surgir no futuro.

1.5 Estrutura do Trabalho

Este trabalho está organizado da seguinte forma:

O Capítulo 2 compõe a fundamentação teórica deste trabalho. Nele são abordados os principais conceitos relacionados às interfaces cérebro-computador (BCI) e à eletroencefalografia (EEG).

Os Capítulos 4 e 5 descrevem os materiais e métodos utilizados no desenvolvimento do estudo, incluindo a caracterização do banco de dados, as etapas de pré-processamento e balanceamento dos sinais, a implementação dos algoritmos de classificação e os critérios empregados para a avaliação de desempenho.

O Capítulo 6 apresenta os resultados obtidos e a respectiva análise comparativa, discutindo o desempenho dos diferentes algoritmos e configurações avaliadas, além do impacto das técnicas de balanceamento de dados aplicadas.

Por fim, o Capítulo 7 contempla as considerações finais do trabalho, sintetizando as principais conclusões alcançadas, destacando suas limitações e propondo direções para investigações futuras.

2 INTERFACES CÉREBRO-COMPUTADOR (ICC)

Uma ICC pode ser definida como um sistema de comunicação que não depende das vias normais de saída do cérebro, como nervos periféricos e músculos, estabelecendo um canal direto entre o cérebro e dispositivos externos (WOLPAW, 2013). Segundo Craik, He e Contreras-Vidal (2019), o principal objetivo de uma BCI é traduzir a atividade neural em comandos para controlar dispositivos externos, permitindo que pessoas com deficiências motoras graves possam se comunicar e interagir com o ambiente ao seu redor.

Os princípios fundamentais de operação de uma BCI incluem a aquisição de sinais cerebrais, pré-processamento para remoção de ruídos, extração de características relevantes, classificação dos padrões identificados e, finalmente, a tradução desses padrões em comandos para dispositivos externos Lotte et al. (2018a). De acordo com Clerc, Bougrain e Lotte (2016), o funcionamento eficaz de uma BCI depende não apenas da tecnologia empregada, mas também da capacidade do usuário em modular deliberadamente sua atividade cerebral.

2.1 Tipos de ICC: invasivos e não-invasivos

As ICCs podem ser classificadas principalmente em dois tipos: invasivas e não-invasivas, dependendo do método de aquisição de sinais cerebrais (MAK; WOLPAW, 2009). As ICCs invasivas utilizam eletrodos implantados cirurgicamente no tecido cerebral ou sobre a superfície cortical, garantindo sinais de alta qualidade e resolução espacial. Conforme Guger, Grünwald e Xu (2023), esses sistemas podem ser classificados em: eletrocorticografia (ECoG), que registra a atividade neural na superfície do córtex; e microeletrodos intracorticais, que registram potenciais de ação de neurônios individuais.

Por outro lado, as ICCs não-invasivas, como destacado por Guger, Grünwald e Xu (2023), adquirem sinais cerebrais sem necessidade de procedimentos cirúrgicos. Entre as técnicas mais comuns estão a eletroencefalografia (EEG), magnetoencefalografia (MEG), ressonância magnética funcional (fMRI) e espectroscopia de infravermelho próximo funcional (fNIRS). O EEG destaca-se pela portabilidade, baixo custo, alta resolução temporal e facilidade de uso, características que o tornam a modalidade mais utilizada em aplicações práticas de BCI. A escolha entre sistemas invasivos e não-invasivos depende de diversos fatores, incluindo a aplicação

pretendida, requisitos de desempenho, considerações éticas e a condição do usuário (GUGER; GRÜNWALD; XU, 2023).

2.2 Aplicações práticas em tecnologias assistivas

As ICCs têm demonstrado grande potencial em tecnologias assistivas, especialmente para pessoas com deficiências neuromotoras severas, como esclerose lateral amiotrófica, lesão da medula espinhal e síndrome do encarceramento (MILLAN et al., 2010). De acordo com Albert, Simeral e Thompson (2023), uma aplicação promissora é o controle de cadeiras de rodas através de sinais EEG, utilizando paradigmas como imagética motora e piscadas de olhos para gerar comandos como avançar, girar e parar.

Rodrigues, Cardoso e Nicolau (2024) destacam que as ICCs também são aplicadas em sistemas de comunicação alternativa, permitindo que pacientes com graves deficiências motoras possam se expressar através de seleção de letras, palavras ou ícones. Além disso, conforme observado por Fernandes et al. (2024), as ICCs têm sido utilizadas no controle de próteses e órteses, na reabilitação neurológica e na operação de dispositivos domésticos inteligentes, ampliando a autonomia de pessoas com limitações físicas.

Notavelmente, o projeto Milo, desenvolvido como um protótipo de cadeira de rodas controlada por ondas cerebrais, representa um avanço significativo nesse campo, utilizando sinais de imagética motora para possibilitar a navegação autônoma para pessoas com síndrome do encarceramento (ALBERT; SIMERAL; THOMPSON, 2023).

2.3 Desafios técnicos e limitações atuais

Apesar dos avanços significativos, as ICCs ainda enfrentam importantes desafios técnicos e limitações. Um dos principais obstáculos, conforme destacado por Lotte et al. (2018a), é a variabilidade dos sinais cerebrais entre diferentes sessões e entre diferentes usuários, o que dificulta a generalização dos algoritmos de classificação. A baixa relação sinal-ruído dos sinais EEG também representa um desafio considerável para a extração confiável de características (CLERC; BOUGRAIN; LOTTE, 2016).

Segundo Craik, He e Contreras-Vidal (2019), as ICCs não-invasivas baseadas em EEG enfrentam limitações como baixa resolução espacial, suscetibilidade a artefatos (movimentos

oculares, atividade muscular, interferência elétrica) e desconforto após uso prolongado. Além disso, o extenso tempo de treinamento necessário para que os usuários aprendam a modular adequadamente seus sinais cerebrais pode limitar a aplicabilidade das ICCs em contextos cotidianos.

As ICCs invasivas, por outro lado, enfrentam desafios relacionados ao procedimento cirúrgico, biocompatibilidade dos materiais, estabilidade dos eletrodos a longo prazo e riscos de infecção (MAK; WOLPAW, 2009). De acordo com Guger, Grünwald e Xu (2023), a degradação do sinal ao longo do tempo devido à formação de tecido cicatricial ao redor dos eletrodos implantados também representa uma limitação significativa para o uso prolongado desses sistemas.

2.4 Estado da arte em BCI para controle de dispositivos

Avanços recentes em algoritmos de processamento de sinais e aprendizado de máquina têm impulsionado o desenvolvimento de ICCs mais precisas e robustas para controle de dispositivos. Craik, He e Contreras-Vidal (2019) destacam que técnicas de aprendizado profundo, como redes neurais convolucionais (CNNs) e redes recorrentes (RNNs), têm demonstrado desempenho superior em comparação aos métodos clássicos de classificação, especialmente na decodificação de padrões complexos de atividade cerebral.

De acordo com Rodrigues, Cardoso e Nicolau (2024), sistemas BCI híbridos, que combinam diferentes paradigmas de controle (por exemplo, imagética motora e potenciais evocados) ou diferentes modalidades de aquisição de sinais (por exemplo, EEG e fNIRS), têm emergido como uma abordagem promissora para aumentar a robustez e a confiabilidade do controle de dispositivos.

Segundo Albert, Simeral e Thompson (2023), o desenvolvimento de interfaces de usuário amigáveis e protocolos de treinamento mais eficientes tem contribuído para melhorar a experiência do usuário e reduzir o tempo necessário para dominar o controle de dispositivos através de ICCs. O projeto recente de uma cadeira de rodas controlada por BCI utilizando imagética motora e piscadas de olhos, como descrito por Albert, Simeral e Thompson (2023), representa um avanço significativo no campo, incorporando recursos como aplicativo para cuidadores e sistema de frenagem dependente da frequência cardíaca para garantir a segurança do usuário.

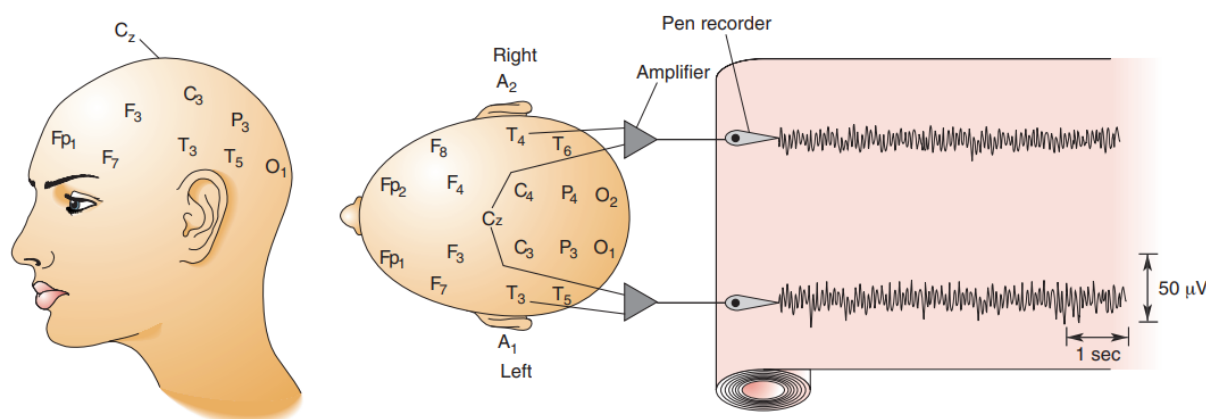
Fernandes et al. (2024) relatam progressos significativos no desenvolvimento de algoritmos adaptativos que podem se ajustar às variações nos padrões cerebrais do usuário ao longo do tempo, melhorando a estabilidade e a robustez do controle. Além disso, como observado por a integração de ICCs com sistemas de navegação assistida e tecnologias de automação pode aumentar a utilidade desses sistemas para usuários com deficiências severas.

2.5 Princípios Neurofisiológicos e Eletroencefalografia

A eletroencefalografia (EEG) representa um método não invasivo e indolor para o registro da atividade elétrica cerebral, constituindo uma das técnicas mais relevantes para a investigação neurofisiológica (BEAR; CONNORS; PARADISO, 2007). O registro obtido, denominado eletroencefalograma, permite a visualização de padrões rítmicos da atividade cerebral, oferecendo aplicações tanto na pesquisa neurocientífica quanto no diagnóstico de condições neurológicas como a epilepsia.

O eletroencefalograma mede as pequenas flutuações de tensão produzidas pela atividade elétrica do córtex cerebral, que constitui aproximadamente 80% da massa cerebral (BEAR; CONNORS; PARADISO, 2007). Estas flutuações são captadas por eletrodos colocados no couro cabeludo do paciente, posicionados em locais padronizados conforme ilustrado na Fig. 2.1.

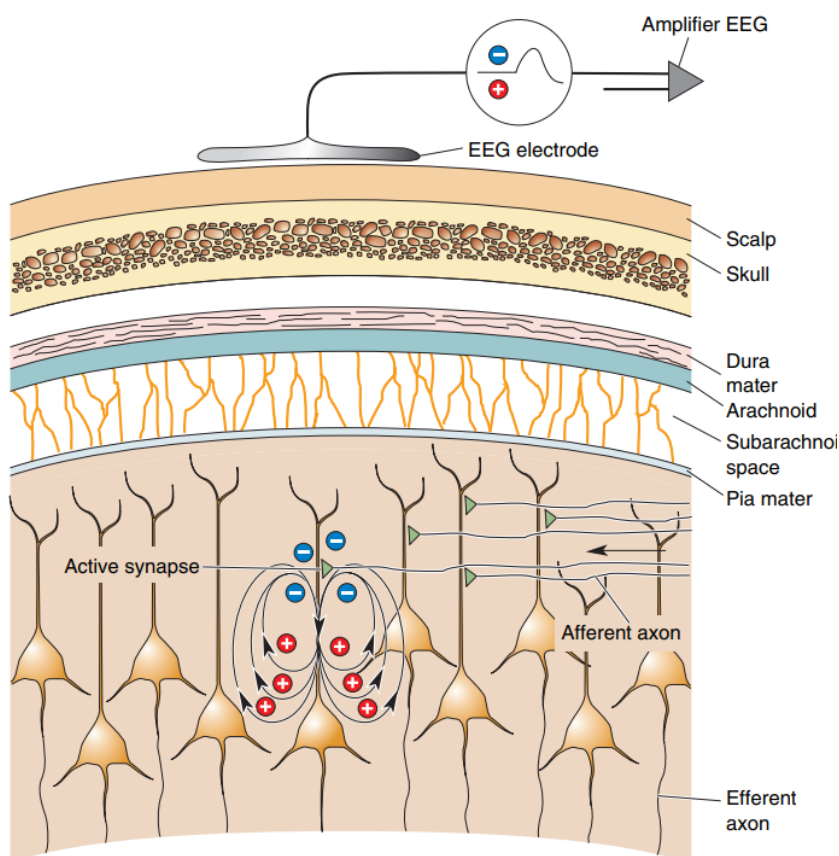
Figura 2.1 – Posições padronizadas para a colocação de eletrodos de EEG segundo o Sistema Internacional 10-20. Os eletrodos são fixados em posições específicas do couro cabeludo e conectados a amplificadores. Cada registro mede a diferença de potencial entre dois pontos.



Fonte: Bear, Connors e Paradiso (2007)

Os potenciais elétricos registrados pelo EEG são gerados principalmente pelas correntes sinápticas que fluem durante a excitação dos dendritos de neurônios piramidais do córtex cerebral (BEAR; CONNORS; PARADISO, 2007). A contribuição elétrica de um único neurônio é extremamente pequena, e o sinal deve atravessar várias camadas de tecidos não-neurais (incluindo as meninges, o líquido cefalorraquidiano, os ossos do crânio e a pele) para alcançar os eletrodos, como ilustrado na Fig. 2.2.

Figura 2.2 – Geração de campos elétricos por correntes sinápticas em células piramidais. A corrente flui para dentro do dendrito através de uma sinapse ativa, deixando uma leve negatividade no fluido extracelular. Estas correntes sinápticas, quando somadas em milhares de neurônios, produzem sinais detectáveis na superfície do escalpo.



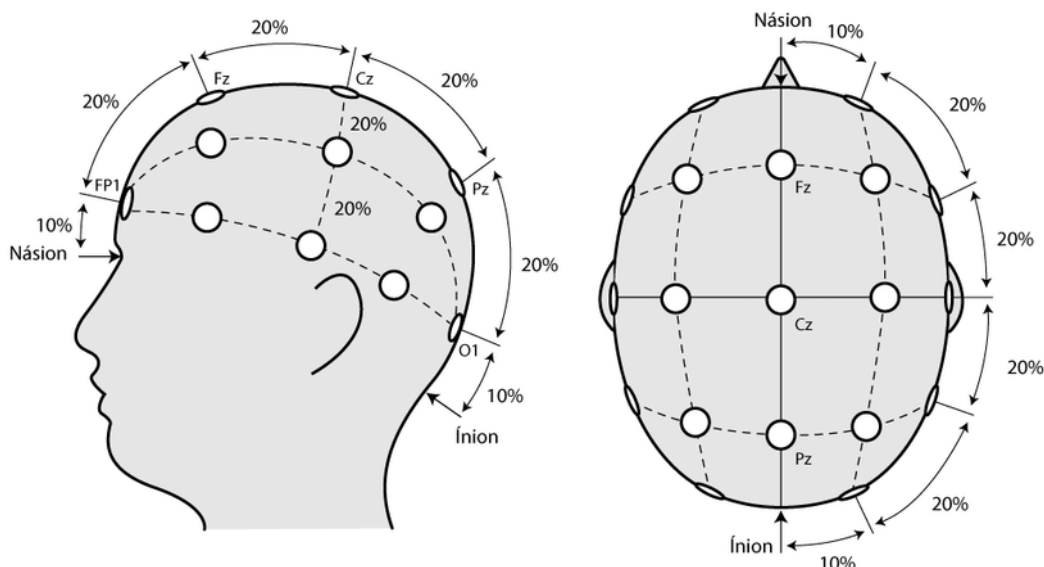
Fonte: Bear, Connors e Paradiso (2007)

Para realizar um exame de EEG, os eletrodos são fixados no couro cabeludo com uma pasta condutora que garante uma conexão de baixa resistência. As flutuações de tensão captadas são geralmente de pequena amplitude, tipicamente na faixa de algumas dezenas de microvolts (μV), e precisam ser amplificadas para análise. Aproximadamente duas dúzias de eletrodos são fixados em posições padronizadas na cabeça e conectados a bancos de amplificadores e dis-

positivos de registro. Diferentes regiões do cérebro - anterior e posterior, esquerda e direita - podem ser examinadas selecionando-se os pares de eletrodos apropriados (BEAR; CONNORS; PARADISO, 2007).

O sistema internacional 10-20 representa o método padronizado mais utilizado mundialmente para posicionamento de eletrodos, garantindo a reprodutibilidade dos registros entre diferentes laboratórios e sessões (BEAR; CONNORS; PARADISO, 2007). O nome "10-20" refere-se às distâncias percentuais entre eletrodos adjacentes, que correspondem a 10% ou 20% do comprimento total de determinados arcos sobre o crânio conforme pode ser observado na Fig. 2.3.

Figura 2.3 – Sistema 10-20



Fonte: Espinelli (2011)

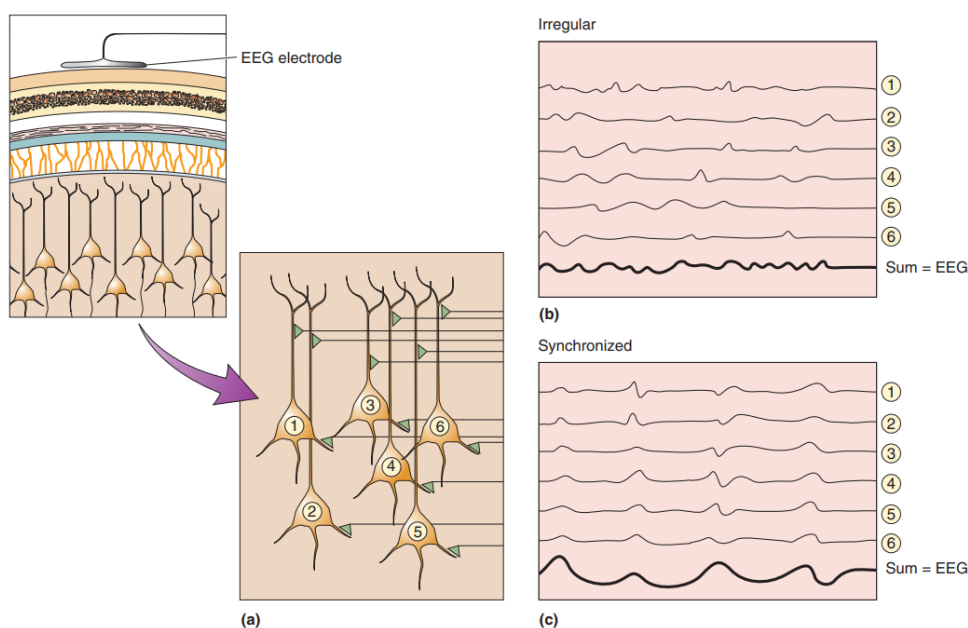
Neste sistema, cada eletrodo recebe uma identificação alfanumérica que indica sua localização sobre o escalpo. As letras referem-se à região cortical subjacente:

F: Frontal	T: Temporal
C: Central (região sobre o sulco central)	Fp: Frontopolar
P: Parietal	A: Auricular (eletrodos de referência colocados nos lóbulos das orelhas)
O: Occipital	

Os números que acompanham as letras seguem uma convenção específica: números ímpares (1, 3, 5, 7) indicam posições no hemisfério esquerdo, números pares (2, 4, 6, 8) no hemisfério direito, e a letra "z"(zero) representa posições na linha média sagital.

Um princípio fundamental da eletroencefalografia é que a amplitude do sinal de EEG depende fortemente do grau de sincronização da atividade dos neurônios subjacentes (BEAR; CONNORS; PARADISO, 2007). Quando um grupo de células é excitado simultaneamente, os pequenos sinais se somam para gerar um grande sinal na superfície. No entanto, quando cada célula recebe a mesma quantidade de excitação, mas distribuída no tempo, os sinais somados são pequenos e irregulares, como demonstrado na Fig. 2.4.

Figura 2.4 – Geração de sinais grandes de EEG por atividade sincronizada. (a) Em uma população de células piramidais localizadas sob um eletrodo de EEG, cada neurônio recebe muitas entradas sinápticas. (b) Se as entradas disparam em intervalos irregulares, as respostas das células piramidais não são sincronizadas, e a amplitude do EEG resultante é pequena. (c) Se as mesmas entradas disparam dentro de uma janela de tempo estreita, as respostas das células piramidais são sincronizadas, resultando em um EEG de maior amplitude.

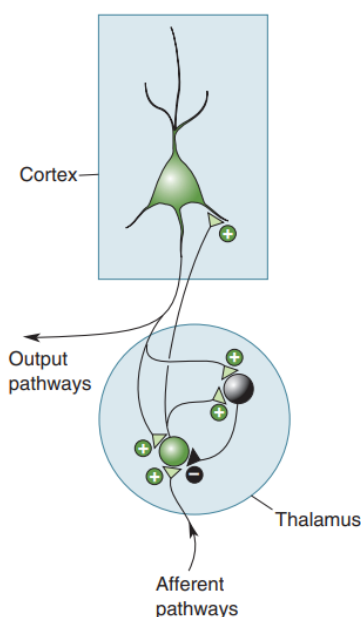


Fonte: Bear, Connors e Paradiso (2007)

Se a excitação sincronizada deste grupo de células é repetida várias vezes, o EEG resultante consistirá em ondas rítmicas de grande amplitude. Os sinais rítmicos do EEG são frequentemente descritos em termos de sua amplitude relativa, sugerindo o quão sincronizada é a atividade neuronal subjacente.

A atividade rítmica sincronizada no cérebro é geralmente coordenada por uma combinação de mecanismos de marcapasso central e atividade coletiva neuronal. O tálamo, com sua massiva projeção para todo o córtex, pode atuar como um poderoso marcapasso (BEAR; CONNORS; PARADISO, 2007), como ilustrado na Fig. 2.5.

Figura 2.5 – Ritmos no tálamo impulsionando ritmos no córtex cerebral. O tálamo pode gerar atividade rítmica devido às propriedades intrínsecas de seus neurônios e por causa de suas interconexões sinápticas. No tálamo, verde representa uma população de neurônios excitatórios, e preto representa uma população de neurônios inibitórios.



Fonte: Bear, Connors e Paradiso (2007)

Sob certas condições, neurônios talâmicos podem gerar descargas de potenciais de ação rítmicas devido a um conjunto particular de canais iônicos dependentes de tensão. A atividade rítmica de cada neurônio marcapasso talâmico então se sincroniza com muitas outras células talâmicas através de interações sinápticas entre neurônios excitatórios e inibitórios. Estes ritmos coordenados são então transmitidos ao córtex pelos axônios talamocorticais, impondo o padrão rítmico à atividade cortical (BEAR; CONNORS; PARADISO, 2007).

Embora as funções precisas dos ritmos cerebrais permaneçam parcialmente compreendidas, eles oferecem uma janela valiosa para a observação dos estados funcionais do cérebro e têm aplicações importantes tanto na pesquisa quanto na prática clínica, como nas interfaces cérebro-computador, que utilizam diferentes paradigmas para extrair padrões específicos de atividade cerebral que possam ser detectados de forma confiável (BEAR; CONNORS; PARADISO, 2007). Os principais paradigmas incluem:

- **Imagética Motora (MI):** Baseia-se na imaginação mental de movimentos sem a execução física dos mesmos, ativando as mesmas regiões cerebrais responsáveis pelo planejamento e execução de movimentos reais. Este processo resulta em padrões detectáveis

de dessincronização relacionada a eventos (ERD) nos ritmos mu e beta sobre o córtex sensório-motor.

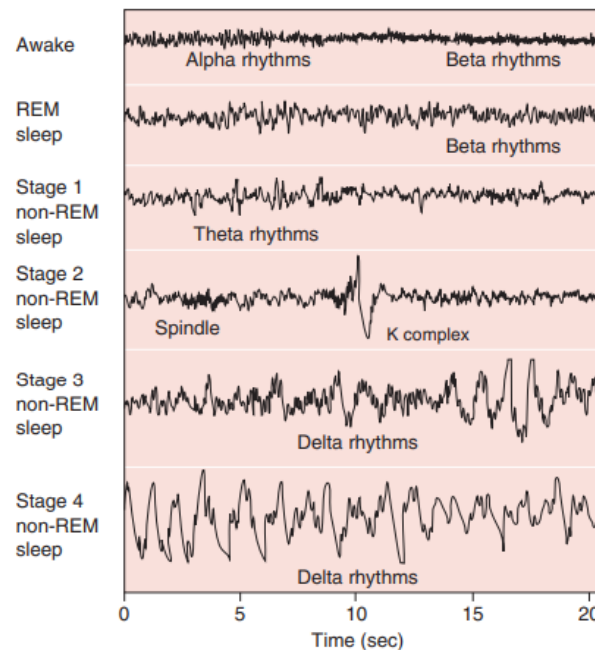
- **Potencial Evocado P300:** Baseia-se no potencial positivo que ocorre aproximadamente 300 milissegundos após a apresentação de um estímulo raro ou significativo, sendo utilizado principalmente em sistemas de comunicação para pacientes com graves deficiências motoras.
- **Potenciais Evocados Visuais de Estado Estacionário (SSVEP):** São respostas oscilatórias contínuas induzidas pela exposição a estímulos visuais piscando em frequências específicas, resultando em atividade cerebral sincronizada com a mesma frequência do estímulo.

Estes paradigmas têm aplicações importantes no desenvolvimento de tecnologias assistivas para pessoas com deficiências motoras graves, permitindo controlar dispositivos externos ou comunicar-se através da atividade cerebral.

2.5.1 Ritmos Cerebrais

Os ritmos do EEG variam dramaticamente e frequentemente correlacionam-se com estados particulares de comportamento (como nível de atenção, sono ou vigília) e patologia (convulsões ou coma). A Fig. 2.6 mostra parte de um EEG normal e os padrões característicos durante diferentes estágios do sono.

Figura 2.6 – Ritmos do EEG durante o sono. As gravações ilustram os ritmos que caracterizam cada estágio do sono, desde o estado de vigília até o sono profundo, demonstrando a variação nas frequências e amplitudes das ondas cerebrais associadas a diferentes estados de consciência.



Fonte: Bear, Connors e Paradiso (2007)

Os ritmos são categorizados por sua faixa de frequência, e cada faixa é nomeada com uma letra grega (BEAR; CONNORS; PARADISO, 2007):

- **Ritmos Beta ($\beta > 14Hz$):** São os mais rápidos e sinalizam um córtex ativado, comumente associados a estados de alerta, vigília e concentração ativa.
- **Ritmos Alfa ($\alpha \rightarrow 8 - 13Hz$):** Oscilam entre 8-13 Hz e são associados a estados de vigília relaxada, especialmente com olhos fechados. O ritmo mu, que também ocorre nesta faixa de frequência mas sobre o córtex sensório-motor, é modulado por atividades motoras reais ou imaginadas.
- **Ritmos Teta ($\theta \rightarrow 4 - 7Hz$):** Variam entre 4-7 Hz e ocorrem durante alguns estados de sono e durante tarefas que exigem atenção sustentada ou processamento de memória.
- **Ritmos Delta ($\delta < 4Hz$):** São bastante lentos, menos de 4 Hz, frequentemente grandes em amplitude, e são uma característica do sono profundo ou estados patológicos como o coma.

- **Ritmos Gama ($\gamma > 30Hz$):** Associados a processamento cognitivo de alta ordem, integração de informações sensoriais e estados de consciência elevada.

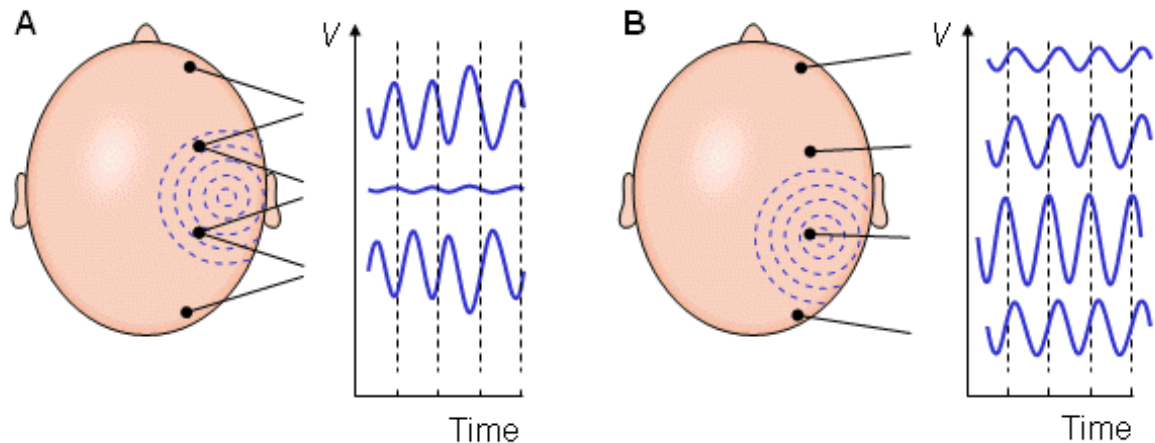
Em geral, ritmos de alta frequência e baixa amplitude estão associados à vigília e estados de alerta, ou às fases de sonho durante o sono. Ritmos de baixa frequência e alta amplitude estão associados a estados de sono sem sonhos ou ao estado patológico de coma (BEAR; CONNORS; PARADISO, 2007).

2.5.2 Tipos de Montagem: Monopolar, Bipolar e Referencial

As montagens de EEG referem-se à configuração de conexão elétrica dos eletrodos para a aquisição de sinais, influenciando diretamente a interpretação dos dados coletados. É importante compreender que o registro do EEG não mede valores absolutos de potencial elétrico, mas sim a diferença de potencial entre dois pontos. Três tipos principais de montagem são utilizados na prática clínica e de pesquisa:

- **Montagem monopolar (unipolar):** Cada eletrodo ativo é comparado a um eletrodo de referência comum ou à média de todos os eletrodos. Esta configuração permite visualizar a amplitude absoluta da atividade em cada localização cortical.
- **Montagem bipolar:** Registra a diferença de potencial entre pares de eletrodos adjacentes, criando cadeias de derivações no sentido ântero-posterior (longitudinal) ou transversal. É particularmente eficaz na detecção de gradientes de potencial locais.
- **Montagem referencial:** Utiliza um eletrodo de referência teoricamente inativo, como o lóbulo da orelha, o mastóide ou a ponta do nariz. A vantagem desta montagem é a preservação da amplitude e polaridade dos sinais.

Figura 2.7 – Medições (A) bipolares e (B) unipolares



Fonte: Espinelli (2011)

A escolha da montagem deve considerar o objetivo do registro, as características do fenômeno a ser estudado e o número de eletrodos disponíveis (BEAR; CONNORS; PARADISO, 2007).

Os sinais EEG são altamente suscetíveis a artefatos e ruídos de diversas origens, representando um desafio significativo para o processamento e interpretação dos dados (BEAR; CONNORS; PARADISO, 2007). Os artefatos podem ser classificados em:

- **Artefatos fisiológicos:** Originados no corpo do sujeito, incluem movimentos oculares, piscadas, atividade muscular, batimentos cardíacos e movimentos corporais.
- **Artefatos não-fisiológicos:** Originados externamente, incluem interferência da rede elétrica (50/60 Hz), ruído de equipamentos eletrônicos e problemas relacionados aos eletrodos.

A mitigação desses artefatos envolve estratégias de prevenção durante a aquisição, como instruções ao sujeito, ambiente adequado, preparação da pele, e técnicas de processamento de sinais pós-aquisição, como filtragem passa-banda, rejeição de segmentos contaminados e análise de componentes independentes.

3 REVISÃO DOS ESTUDOS SOBRE CLASSIFICAÇÃO DE SINAIS EEG

No domínio de imagética motora, o problema não se restringe à distinção entre padrões neurais. O desempenho depende da qualidade da estimação das covariâncias, do desenho do banco de filtros, da estabilidade da representação entre sessões, da estratégia de classificação, do protocolo de validação e da forma como os comandos são incorporados à camada de controle assistivo. Por isso, a literatura trata extração de características, classificação e controle como componentes acoplados de um mesmo sistema (TANGERMANN et al., 2012; LOTTE et al., 2018b; PALUMBO et al., 2021; ZHANG; WANG, 2021).

Na parte de extração de características o trabalho de Ramoser, Müller-Gerking e Pfurtscheller (2000) formalizou o uso de *Common Spatial Patterns* (CSP) para discriminar imagética de mão esquerda e mão direita em ensaios únicos. O método estima filtros espaciais a partir das matrizes de covariância de duas classes, maximizando a variância de uma classe e minimizando a da outra no espaço projetado. Nos experimentos reportados, os melhores resultados de classificação para três sujeitos foram 90,8%, 92,7% e 99,7% (RAMOSER; MÜLLER-GERKING; PFURTSCHELLER, 2000). O estudo consolidou o CSP como uma referência ao mostrar que a estrutura espacial do EEG contém informação discriminativa suficiente para classificação em ensaio único. A limitação do método permanece clara: trata-se de uma formulação originalmente binária, dependente da banda de frequência adotada e sensível a erros de estimação de covariância quando o número de ensaios é reduzido.

Essa limitação foi tratada de forma explícita por Lotte e Guan (2011). Os autores propuseram uma formulação unificada para regularização do CSP, introduzindo restrições na estimação das covariâncias e no critério de extração dos filtros. Em avaliações com bases de competição, os melhores esquemas de regularização superaram o CSP convencional em quase 10% de acurácia mediana e permitiram transferência mais eficiente entre sujeitos (LOTTE; GUAN, 2011). O trabalho mostra que parte da limitação do CSP clássico decorre da instabilidade estatística do problema de estimação. Em aplicações assistivas, esse aspecto é relevante porque pequenas mudanças de distribuição entre sessões podem deslocar a fronteira de decisão e comprometer a consistência dos comandos gerados.

Tangermann et al. (2012) revisaram a *BCI Competition IV* e consolidaram, entre outros, os conjuntos 2a e 2b como referências para comparação metodológica. O conjunto 2a reúne quatro classes de imagética motora com 22 canais de EEG em nove sujeitos, enquanto o 2b re-

duz a aquisição a três derivações bipolares e duas classes (TANGERMANN et al., 2012). Esse trabalho forneceu um protocolo comparativo comum, permitindo distinguir avanços metodológicos de diferenças decorrentes da base de dados, da partição ou da métrica empregada.

A limitação do CSP em banda única foi abordada diretamente por Ang et al. (2012). O método *Filter Bank Common Spatial Pattern* (FBCSP) decompõe o EEG em múltiplas sub-bandas, aplica CSP em cada uma delas e realiza seleção de atributos com base em informação mútua. No *Dataset 2a* e no *Dataset 2b* da *BCI Competition IV*, a abordagem apresentou valores médios de kappa de 0,569 e 0,600, respectivamente, figurando entre os melhores desempenhos da competição (ANG et al., 2012). O FBCSP parte do reconhecimento de que o conteúdo discriminativo da imagética motora não é espectralmente uniforme entre sujeitos. Essa flexibilidade melhora a adaptação ao perfil individual, mas aumenta a dimensionalidade, o custo de seleção de atributos e o risco de sobreajuste. O método tende a apresentar bom desempenho em cenários *offline* controlados, mas sua robustez depende da escolha das sub-bandas, do processo de seleção de características e do protocolo de validação.

A literatura riemanniana deslocou o problema para outro nível de representação. Em Barachant et al. (2012), a matriz de covariância deixa de ser apenas um insumo para filtro espacial e passa a constituir o descritor primário do EEG. Os autores propuseram uma estrutura multiclasse baseada na geometria da variedade das matrizes simétricas definidas positivas. No conjunto 2a da *BCI Competition IV*, o método de projeção no espaço tangente com LDA elevou a acurácia média de 65,1% para 70,2% em relação à referência multiclasse baseada em CSP+LDA (BARACHANT et al., 2012). Esses resultados indicam que a geometria das covariâncias preserva informação que pode se perder sob um tratamento estritamente euclidiano, sobretudo em cenário multiclasse. Em trabalho subsequente, Barachant et al. (2013) estenderam essa lógica a um *kernel* riemanniano para SVM, reforçando a importância da adequação geométrica da representação.

Fang et al. (2022) combinaram essas duas linhas em um método que integra bancos de filtros e projeção riemanniana no espaço tangente. O método FBRTS foi avaliado nos conjuntos 2a e 2b da *BCI Competition IV*, atingindo acurácia média de 77,7% e 86,9%, respectivamente (FANG et al., 2022). Essa formulação trata simultaneamente duas fontes de variabilidade em MI-EEG: a dependência espectral e a instabilidade da representação euclidiana. A limitação apontada pelos autores é o aumento da dimensionalidade e a dependência de seleção adicional

de atributos e ajuste de hiperparâmetros, o que pode restringir escalabilidade ou operação em tempo real quando o pipeline não é rigorosamente controlado.

A revisão de Lotte et al. (2018b) sustenta que não existe um modelo universalmente superior, o desempenho depende da interação entre representação, tamanho amostral, não estacionariedade e protocolo de validação. Para este trabalho, isso implica que a comparação entre SVM, Random Forest, Extra Trees, MLP, Gradient Boosting, XGBoost e comitê só é informativa quando todos os modelos operam sobre um espaço de atributos controlado e sob um protocolo de avaliação capaz de reduzir vazamento de informação e superestimação de desempenho.

Entre os classificadores discriminativos, o SVM é uma escolha recorrente para MI-EEG. Sua formulação por margem máxima se ajusta a cenários em que o vetor de características pode ter dimensionalidade moderada ou alta, enquanto o número de amostras por sessão permanece limitado. No estudo de Hou et al. (2022), que comparou SVM, Random Forest, Naive Bayes, LDA, KNN, XGBoost e AdaBoost em uma tarefa de fusão de características para imagética motora multiclasse, o SVM com *kernel* RBF foi selecionado como melhor classificador, com 85% de acurácia no conjunto de avaliação do *Dataset 2a*. O resultado mostra que o SVM superou, sob o mesmo conjunto de atributos, árvores, métodos de *boosting* e modelos lineares. Luo et al. (2020) mostraram ainda que a própria família SVM pode ser combinada em regime de ensemble. Seu método *ensemble support vector learning* (ESVL), baseado em probabilidades posteriores de múltiplos SVMs treinados sobre descritores complementares, alcançou kappa máximo médio de 0,60 e 0,71 nos conjuntos 2a e 2b da *BCI Competition IV* (LUO et al., 2020). Esses estudos indicam que o SVM é uma alternativa consistente quando os atributos já são informativos e que seu uso não se restringe à classificação plana, podendo também compor estruturas de comitê.

O Random Forest também aparece com frequência na literatura. Esse modelo consegue explorar interações não lineares entre atributos, é relativamente robusto a escalas distintas e tolera redundância moderada entre variáveis sem exigir pressupostos fortes sobre a distribuição dos dados. Steyrl et al. (2016) avaliaram diretamente o uso de Random Forest em BCIs de ritmos sensório-motores e reportaram desempenho estatisticamente superior ao LDA regularizado em aproximadamente 3% em simulação *offline*, além de estabilidade em ampla faixa de parâmetros. O estudo chama atenção para a robustez paramétrica do modelo, aspecto relevante

em bases pequenas e heterogêneas. De modo semelhante, [Thenmozhi e Helen \(2022\)](#) mostraram que um pipeline com seleção de características por *Extreme Gradient Boosting Bayesian Optimization* e classificação por Random Forest atingiu acurácias médias de 94,44% e 88,72% em dois conjuntos públicos, superando métodos como ANOVA, SFS, RFE e LASSO com menor custo computacional ([THENMOZHI; HELEN, 2022](#)). Embora o estudo avalie um pipeline completo e não o Random Forest isoladamente, ele indica que classificadores baseados em árvores são competitivos quando operam sobre atributos bem selecionados.

Analisando os métodos de *boosting*, [Gao et al. \(2016\)](#) empregaram um esquema *multi-class AdaBoost* com *Extreme Learning Machine* e reportaram acurácia média de 79,5% em 10 sujeitos saudáveis. Esse resultado demonstra que estratégias sequenciais de aprendizado conseguem lidar adequadamente com o aumento de complexidade imposto pela classe de repouso, um cenário mais difícil do que a classificação binária convencional. Entretanto, quando comparados diretamente a outros classificadores consolidados, esses métodos nem sempre ocupam a primeira posição. No estudo de [Hou et al. \(2022\)](#), por exemplo, AdaBoost e XGBoost foram superados pelo SVM. Por outro lado, [Thenmozhi e Helen \(2022\)](#) mostraram que o XGBoost pode exercer papel decisivo como mecanismo de seleção de atributos: ao combinarem *Extreme Gradient Boosting Bayesian Optimization* com Random Forest, os autores reportaram acurácias médias de 94,44% e 88,72% nos conjuntos IIIa e IVa, respectivamente, além de menor tempo computacional em comparação com outros quatro métodos de seleção de características. Portanto, os métodos de *boosting* devem ser interpretados, no contexto comparativo, tanto como classificadores candidatos quanto como ferramentas eficazes para revelar estrutura discriminante em espaços de atributos complexos.

No caso dos comitês os ganhos reportados não decorrem apenas da complexidade do modelo, mas da capacidade de combinar erros parcialmente complementares entre diferentes classificadores ou representações. [Luo et al. \(2020\)](#), ao combinarem múltiplos SVMs treinados sobre características complementares baseadas em ERD/ERS e potenciais relacionados ao evento, obtiveram valores médios máximos de Kappa de 0,60 e 0,71 nos conjuntos BCI Competition IV-2a e 2b, respectivamente. Esse resultado evidencia que a agregação de classificadores pode elevar a robustez do sistema mesmo em cenários de poucos canais e variabilidade elevada. Considerando a decodificação *cross-session*, [Pan et al. \(2023\)](#) propuseram o método RAVE, que combina geometria riemanniana, AdaBoost e votação majoritária, e mostraram desempenho su-

perior ao de oito algoritmos concorrentes sob diferentes quantidades de amostras intrassessão disponíveis para treinamento. Esse trabalho demonstra que a combinação entre representação geométrica e aprendizado em comitê pode mitigar a degradação causada pela variabilidade entre sessões. Mais recentemente, [Ahmadi e Mesin \(2024\)](#) reforçaram essa tendência ao propor um modelo de *weighted stacking* com otimização por correlação, obtendo resultados muito elevados em múltiplos conjuntos públicos de referência. Em termos comparativos, esses achados indicam que comitês não são apenas mecanismos de suavização de desempenho, mas uma das frentes empíricas mais promissoras para extração de robustez e generalização em MI-EEG.

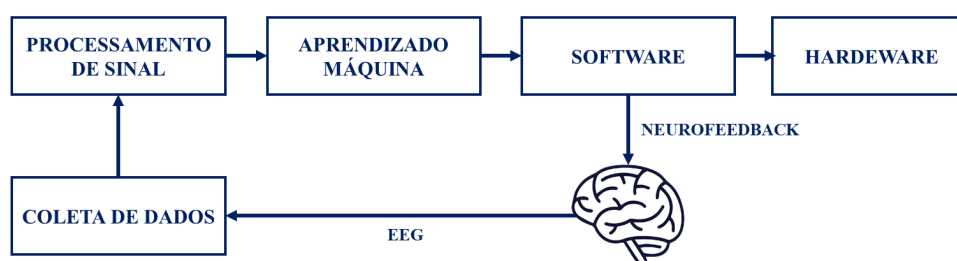
Os estudos revisados indicam quatro aspectos recorrentes. O primeiro é que a etapa de representação do EEG continua sendo determinante. Trabalhos sobre CSP, CSP regularizado, FBCSP e geometria riemanniana mostram que os ganhos mais consistentes surgem quando a estrutura espacial e espectral do sinal é tratada de forma estatisticamente adequada ([RAMOSER; MÜLLER-GERKING; PFURTSCHELLER, 2000](#); [LOTTE](#); [GUAN, 2011](#); [ANG et al., 2012](#); [BARACHANT et al., 2012](#); [BARACHANT et al., 2013](#); [FANG et al., 2022](#)). O segundo é que a literatura não sustenta a superioridade universal de uma única família de classificadores. ([LOTTE et al., 2018b](#); [HOU et al., 2022](#); [STEYRL et al., 2016](#); [HETTIARACHCHI et al., 2018](#); [THENMOZHI; HELEN, 2022](#); [LUO et al., 2020](#); [PAN et al., 2023](#); [AHMADI; MESIN, 2024](#)). O terceiro é que os estudos de cadeira de rodas mostram que o problema prático não consiste apenas em decodificar intenção, mas em integrar essa intenção a camadas de navegação, segurança e correção de erro ([GALÁN et al., 2008](#); [TSUI; GAN; HU, 2011](#); [LONG et al., 2012](#); [DENG et al., 2020](#); [ZHANG; WANG, 2021](#); [PALUMBO et al., 2021](#)). O quarto é que o desempenho de um pipeline depende da coerência entre representação, classificador, validação e arquitetura de decisão.

4 PROJETO MILO: ICC PARA CONTROLE DE CADEIRA DE RODAS

O projeto Milo, desenvolvido por uma equipe multidisciplinar da Universidade McGill em 2019, tem por objetivo o controle de cadeira de rodas através da detecção de padrões específicos de atividade cerebral, utilizando sinais de imaginação motora para navegação através de pensamentos relacionados a movimentos das mãos.

A arquitetura geral do sistema, apresentada na Fig. 4.1, evidencia o fluxo integrado desde a aquisição dos sinais cerebrais até o controle final do hardware da cadeira de rodas, passando por etapas críticas de processamento digital e tomada de decisão automatizada.

Figura 4.1 – Pipeline geral do sistema de interface cérebro-computador implementado no projeto Milo, evidenciando as etapas de aquisição de sinais, processamento digital, classificação por aprendizado de máquina e controle de hardware



Fonte: Adaptado de Albert, Simeral e Thompson (2023)

4.1 Coleta de Dados Eletroencefalográficos

A aquisição de sinais eletroencefalográficos constitui a etapa onde os potenciais elétricos produzidos pela atividade neural são capturados e convertidos em dados digitais adequados para processamento computacional.

4.1.1 Configuração do Hardware de Aquisição

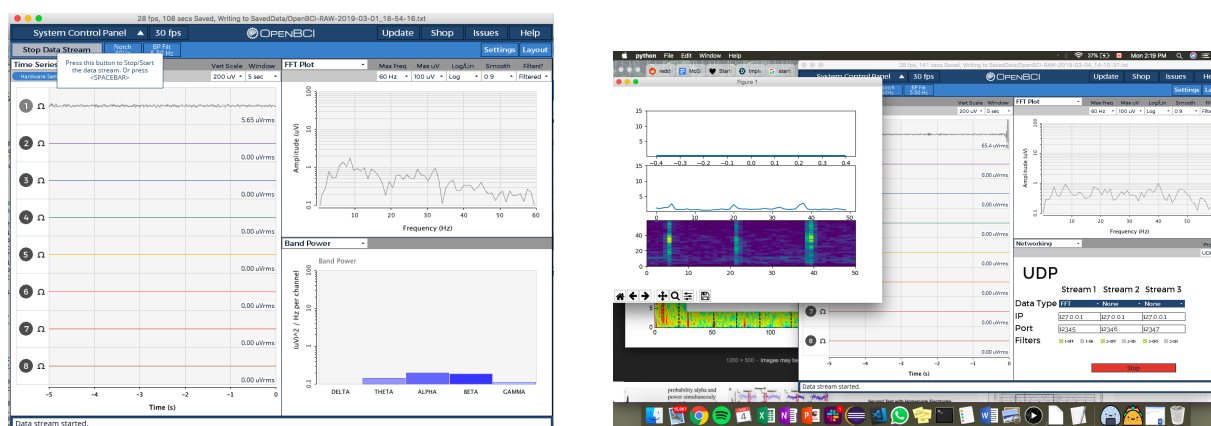
O sistema utiliza a plataforma OpenBCI Cyton como dispositivo principal de aquisição, caracterizada por uma arquitetura de 8 canais com resolução de 32 bits e capacidade de amostragem de até 16.000 Hz. Para este projeto, foi configurada uma frequência de amostragem de 250 Hz, valor selecionado para capturar adequadamente as características espectrais dos ritmos cerebrais de interesse, que tipicamente se concentram na faixa de 1 a 50 Hz.

O princípio físico da eletroencefalografia baseia-se na medição das diferenças de potencial elétrico geradas pela atividade sincronizada de populações neuronais no córtex cerebral.

Quando grupos extensos de neurônios se ativam simultaneamente, produzem campos elétricos detectáveis por eletrodos posicionados sobre o couro cabeludo. O sistema OpenBCI amplifica estes sinais de baixa amplitude, tipicamente variando entre 10 e 100 microvolts, convertendo-os em valores digitais processáveis computacionalmente.

A Fig. 4.2 demonstra a interface de aquisição OpenBCI durante operação típica, apresentando a visualização simultânea dos sinais temporais, análise espectral em tempo real e distribuição de potência por bandas de frequência neurológica relevantes.

Figura 4.2 – Interface de aquisição OpenBCI mostrando sinais temporais de oito canais EEG, análise FFT em tempo real e distribuição de potência por bandas de frequência (Delta, Theta, Alpha, Beta, Gamma) durante sessão experimental

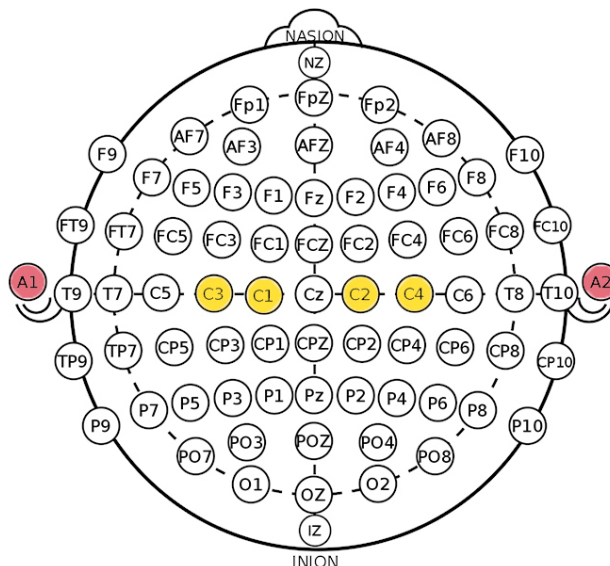


Fonte: Albert, Simeral e Thompson (2023)

A configuração dos eletrodos segue o sistema internacional 10-20, conforme ilustrado na Fig. 4.3, quatro eletrodos ativos de ouro tipo *cup* foram posicionados nas regiões C1, C2, C3 e C4 do córtex sensoriomotor, área cerebral diretamente responsável pelo planejamento e controle de movimentos voluntários. Adicionalmente, dois eletrodos de referência foram posicionados nos lóbulos das orelhas para estabelecer um potencial elétrico de referência comum, procedimento padrão que permite a medição precisa das diferenças de tensões geradas pela atividade cerebral.

Para garantir a qualidade da interface eletrodo-pele, foi utilizada pasta condutiva Ten20 EEG de grau médico, para assegurar baixa impedância elétrica e estabilidade do contato durante toda a sessão experimental. Esta pasta especializada contém componentes que facilitam a condução elétrica e mantém a umidade necessária para preservar a integridade do sinal ao longo do tempo.

Figura 4.3 – Configuração dos eletrodos segundo o sistema internacional 10-20, com posicionamento estratégico sobre o córtex sensoriomotor nas posições C1, C2, C3 e C4, além de eletrodos de referência nos lóbulos auriculares

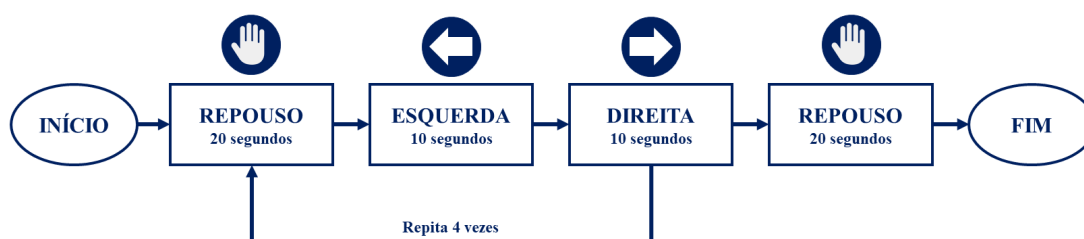


Fonte: Albert, Simeral e Thompson (2023)

4.1.2 Protocolo Experimental de Coleta

O protocolo experimental foi desenvolvido para treinar os usuários na geração de padrões neurais distintos e consistentes associados à imaginação motora. O paradigma implementado alterna entre períodos de repouso mental e períodos de imaginação específica de movimentos, criando um conjunto de dados rotulados. A Fig. 4.4 apresenta a estrutura temporal do protocolo experimental utilizado para coleta dos dados de treinamento, demonstrando a sequência metodológica de estados mentais.

Figura 4.4 – Paradigma experimental implementado para coleta de dados de treinamento, demonstrando a alternância sistemática entre períodos de repouso e imaginação motora com duração específica para cada condição

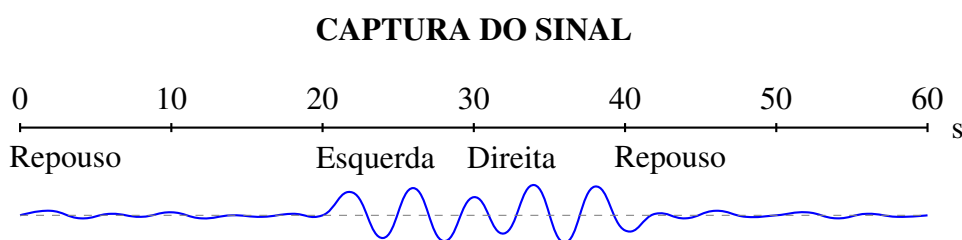


Fonte: Adaptado de Albert, Simeral e Thompson (2023)

A Fig. 4.5 ilustra o processo temporal de captura dos sinais durante uma sessão experimental típica, demonstrando como os diferentes estados mentais são sequenciados ao longo

do tempo e como as características espectrais do sinal EEG variam conforme cada condição experimental específica.

Figura 4.5 – Protocolo experimental demonstrando a sequência temporal de captura de sinais eletroencefalográficos durante diferentes estados mentais, com alternância entre repouso e imaginação motora da mão esquerda e direita

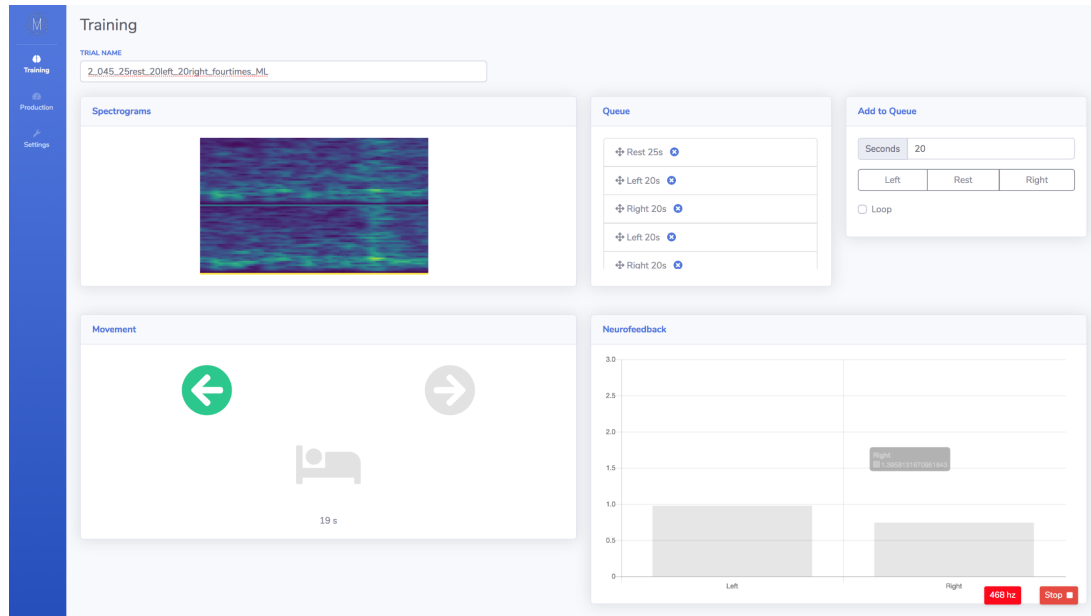


Cada sessão experimental inicia com um período de repouso de 20 segundos, durante o qual o participante é instruído a relaxar mentalmente e evitar pensamentos relacionados a movimentos. Este período serve como linha de base neurofisiológica para comparação estatística com os estados ativos de imaginação motora. Subsequentemente, apresentam-se blocos de 10 segundos cada, durante os quais o participante recebe instruções visuais claras através da interface gráfica, apresentada na Fig. 4.6, para imaginar movimentos específicos da mão esquerda ou direita. Essa interface gráfica combina visualização espectral em tempo real, métricas de classificação, status de sensores e controles operacionais em uma apresentação unificada e intuitiva.

Durante os períodos de imaginação motora, os participantes são orientados a visualizar vividamente o movimento solicitado, mantendo concentração mental consistente e focada. Esta tarefa mental específica produz mudanças mensuráveis e reproduzíveis na atividade elétrica do córtex motor, manifestando-se especificamente através da dessincronização do ritmo μ . Este fenômeno neurofisiológico caracteriza-se pela redução da potência espectral na faixa de 7 a 13 Hz quando neurônios motores se ativam para planejamento de movimentos, mesmo na ausência de execução motora real.

A interface gráfica permite visualização em tempo real dos espectrogramas, possibilitando monitoramento contínuo da qualidade dos sinais eletroencefalográficos e identificação imediata de problemas técnicos como desconexão de eletrodos ou interferências externas. Os indicadores de confiança do classificador fornecem *feedback* instantâneo sobre a certeza das decisões de classificação, permitindo ao usuário ajustar sua técnica de imaginação motora para otimizar progressivamente a performance do sistema.

Figura 4.6 – Interface gráfica do sistema Milo, integrando visualização espectral, métricas de classificação, status de sensores e controles operacionais em uma apresentação unificada



Fonte: Albert, Simeral e Thompson (2023)

4.1.3 Estrutura e Organização dos Dados

Os dados brutos coletados são organizados em formato tabular onde cada linha representa uma amostra temporal única e cada coluna corresponde a um canal de aquisição específico ou informação auxiliar de sincronização. A estrutura original preserva todas as informações de aquisição, incluindo contadores temporais, dados de acelerômetro triaxial e horários precisos para sincronização posterior entre diferentes sistemas. A Tab. 4.1 apresenta um exemplo representativo da estrutura dos dados brutos conforme registrados pelo sistema OpenBCI, incluindo todos os campos disponíveis durante a aquisição em tempo real.

Tabela 4.1 – Estrutura dos dados brutos conforme registrados pelo sistema OpenBCI, demonstrando organização temporal dos canais eletroencefalográficos e informações auxiliares de sincronização

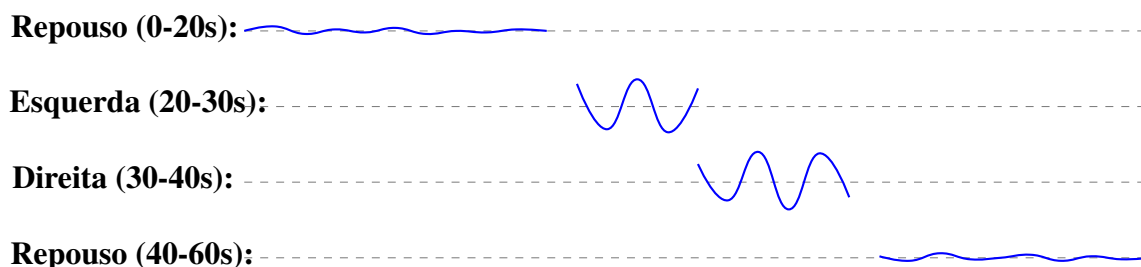
Índice	1	2	3	4	5	6	7	8	X (G)	Y (G)	Z (G)	Hora
0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.000	0.000	0.000	17:31:02.331
1	33957.30	-80948.56	-187500.02	-187500.02	-187500.02	-80863.49	-187500.02	-187500.02	0.150	0.060	1.008	17:31:02.809
2	33968.50	-80787.49	-187500.02	-187500.02	-187500.02	-80687.91	-187500.02	-187500.02	0.000	0.000	0.000	17:31:02.809
3	33979.10	-80683.07	-187500.02	-187500.02	-187500.02	-80548.84	-187500.02	-187500.02	0.000	0.000	0.000	17:31:02.809
4	33954.98	-80802.31	-187500.02	-187500.02	-187500.02	-80686.76	-187500.02	-187500.02	0.000	0.000	0.000	17:31:02.809
5	33964.57	-80911.21	-187500.02	-187500.02	-187500.02	-80827.17	-187500.02	-187500.02	0.000	0.000	0.000	17:31:02.809
						⋮						
253	34047.18	-81059.13	-187500.02	-187500.02	-187500.02	-81133.12	-187500.02	-187500.02	0.000	0.000	0.000	17:31:03.772
254	33985.00	-81115.08	-187500.02	-187500.02	-187500.02	-81178.49	-187500.02	-187500.02	0.152	0.048	1.004	17:31:03.772
255	33953.44	-81254.63	-187500.02	-187500.02	-187500.02	-81354.96	-187500.02	-187500.02	0.000	0.000	0.000	17:31:03.772
0	34011.20	-81258.02	-187500.02	-187500.02	-187500.02	-81372.80	-187500.02	-187500.02	0.000	0.000	0.000	17:31:03.772
1	34025.70	-81048.30	-187500.02	-187500.02	-187500.02	-81133.43	-187500.02	-187500.02	0.000	0.000	0.000	17:31:03.772

Cada linha da tabela representa uma amostra temporal única com informações distribuídas em 13 campos principais: contador cíclico (0-255) que permite detecção automática de amostras perdidas durante transmissão, oito canais de dados EEG expressos em microvolts, três eixos de acelerômetro para detecção de artefatos de movimento da cabeça, e o horário da colta.

O processo de rotulação considera o estado mental que o participante deveria estar executando em cada momento temporal específico, baseado no protocolo experimental predefinido (Fig.4.4). A Fig. 4.7 ilustra como os dados contínuos são segmentados em épocas específicas correspondentes a cada condição experimental, demonstrando o processo metodológico de extração de segmentos temporais para análise estatística posterior.

Figura 4.7 – Segmentação temporal dos dados eletroencefalográficos em épocas correspondentes aos diferentes estados mentais, demonstrando a extração de segmentos específicos para análise estatística e treinamento de classificadores

SEGMENTAÇÃO DE DADOS



Após o processamento inicial e segmentação temporal, os dados são reorganizados em formato otimizado para análises de aprendizado de máquina, conforme exemplificado na Tab. 4.2. Esta transformação estrutural remove informações auxiliares desnecessárias para classificação e adiciona rótulos experimentais correspondentes às condições específicas de imaginação motora.

Tabela 4.2 – Estrutura dos dados processados após segmentação e rotulação, demonstrando formato otimizado para treinamento de algoritmos de classificação de imaginação motora

Identificador	1	2	3	4	5	6	7	8	Rótulo
1553462177628	-102.0581	-5.4985	79.4604	-57.2652	1.8775	-4.4703	15.2439	13.5228	Esquerda
1553462177629	-176.6458	-23.5140	15.9815	48.4362	3.0398	-3.5092	18.3284	-4.8503	Esquerda
1553462177629	169.7168	-6.8843	-24.7881	66.0271	-19.6248	0.8494	-3.2410	10.7065	Esquerda
1553462177630	41.0825	-4.2021	-84.0426	-22.2400	-4.2468	0.6258	-1.4305	-0.3129	Esquerda
1553462177631	64.4848	-20.6530	-4.3586	-62.0708	6.8843	-10.0136	-0.3800	-9.9242	Esquerda

O identificador temporal utiliza o padrão *timestamp UNIX*, que emprega uma contagem de 32 bits para registrar segundos transcorridos desde a data de referência (1º de janeiro de 1970), permitindo conversões precisas para formatos de data e hora legíveis e facilitando sincronização entre diferentes sistemas computacionais.

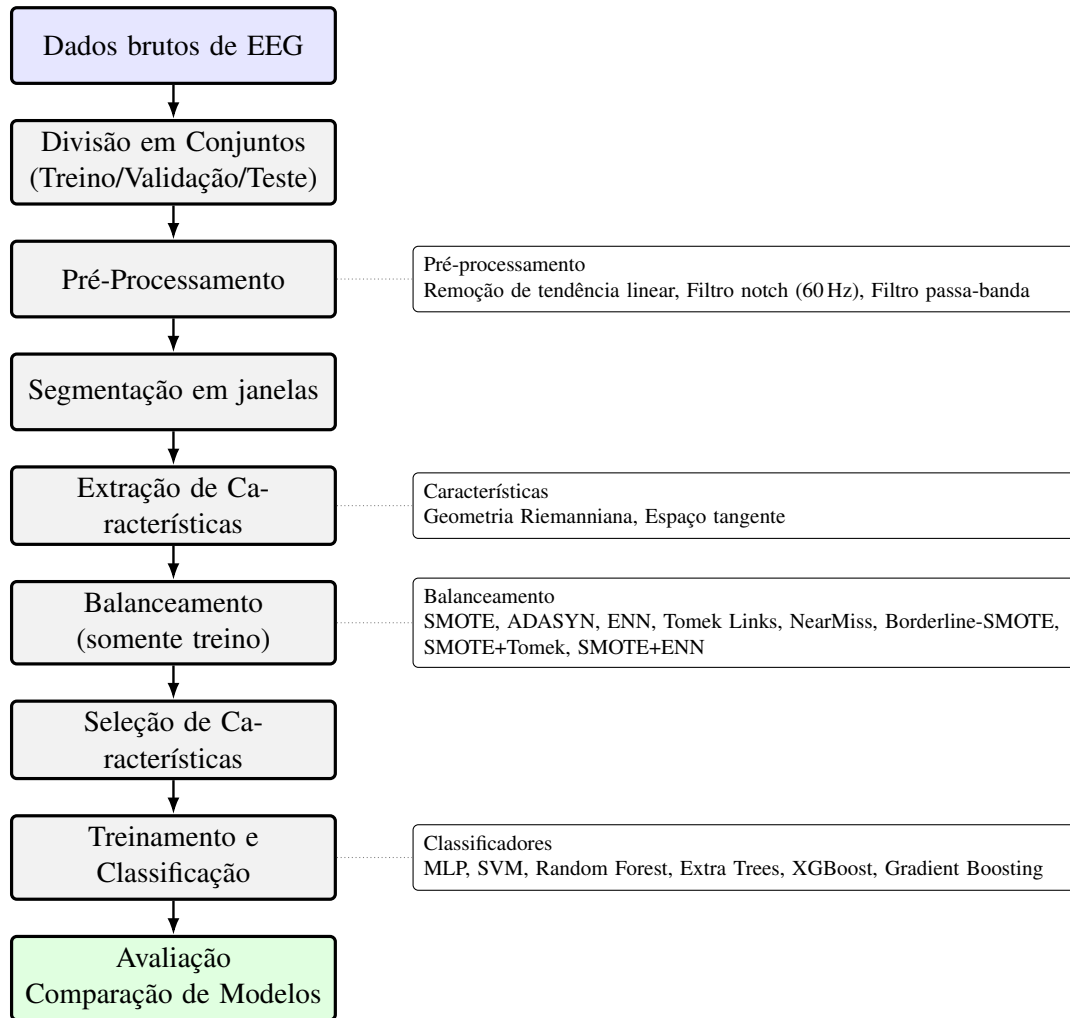
A estrutura simplificada e otimizada mantém exclusivamente as informações essenciais dos oito canais EEG enquanto adiciona uma coluna de rótulos categóricos que identificam precisamente a condição experimental correspondente a cada amostra. Este formato padronizado facilita a implementação de algoritmos de aprendizado supervisionado e permite análises estatísticas eficientes para caracterização detalhada dos padrões neurais associados à imaginação motora.

5 METODOLOGIA

Este capítulo descreve a etapa de classificação dos sinais de imaginação motora, abordando duas estratégias distintas: (i) a classificação multiclasse direta, envolvendo três classes simultaneamente (movimento à direita, movimento à esquerda e repouso); e (ii) a classificação hierárquica em cascata, que decompõe o problema em dois níveis binários sequenciais (repouso versus movimento, seguido de direita versus esquerda).

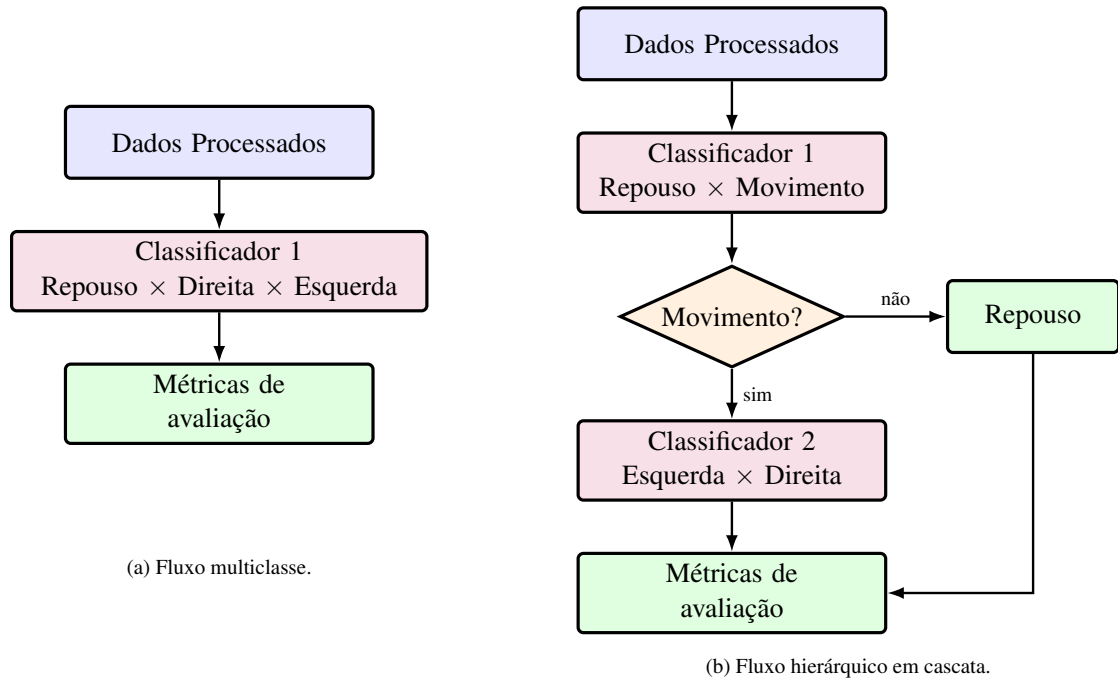
Para ambas as abordagens, foram avaliados seis algoritmos de aprendizado de máquina: Perceptron Multicamadas (MLP), Máquinas de Vetores de Suporte (SVM), Floresta Aleatória (*Random Forest*), *Extra Trees*, *Gradient Boosting* e *XGBoost*. Na comparação entre modelos, foram aplicados testes de significância estatística, permitindo identificar diferenças estatisticamente relevantes no desempenho dos algoritmos.

Figura 5.1 – Fluxo do pipeline proposto para classificação de sinais de EEG: divisão dos dados, pré-processamento, segmentação, extração e seleção de características, técnicas de balanceamento e treinamento de classificadores, seguido de avaliação e comparação de desempenho.



A Fig. 5.1 sintetiza o fluxo adotado, desde a aquisição dos dados brutos até a avaliação final dos modelos. As Figs. 5.2a e 5.2b detalham, respectivamente, os fluxos específicos para a classificação multiclasse e para a classificação hierárquica em cascata.

Figura 5.2 – Comparação entre estratégias de classificação: (a) abordagem multiclasse, em que um único modelo discrimina simultaneamente esquerda, direita e repouso; (b) abordagem em cascata, em que um primeiro classificador separa repouso de movimento e, condicionado à detecção de movimento, um segundo classificador discrimina esquerda de direita.



5.1 Divisão dos Dados

Brookshire et al. (2024), Lee et al. (2024) demonstram que a divisão baseada em segmentos (onde janelas de um mesmo sujeito ou sessão podem aparecer em conjuntos distintos) conduz a estimativas infladas de desempenho, uma vez que o modelo pode aprender características específicas do sujeito ou sessão ao invés de padrões generalizáveis da tarefa. Essa metodologia garante que todas as janelas temporais derivadas de uma mesma sessão sejam atribuídas exclusivamente a um único conjunto.

Neste trabalho, o processo de divisão ocorre em duas etapas sequenciais com estratificação por classe para preservar a distribuição original de classes em cada conjunto. Primeiramente, as sessões são divididas entre treinamento (55%) e não-treinamento (45%). Em seguida, o conjunto não-treinamento é subdividido em validação (44,4%, representa 20% do total) e teste (55,6%, representa 25% do total). Esta abordagem de divisão em nível de sessão é consistente com as recomendações de Varoquaux et al. (2017) e Brookshire et al. (2024) para prevenção de vazamento de dados.

5.2 Pré-processamento Temporal

O pré-processamento temporal é aplicado individualmente a cada sessão de aquisição, preservando a independência estatística entre os dados de diferentes sessões e prevenindo o vazamento de informações que ocorre quando o pré-processamento é realizado sobre o conjunto completo de dados (BROOKSHIRE et al., 2024). Esta etapa compreende três operações sequenciais: remoção de tendência linear, filtragem notch e filtragem passa-banda. Conforme recomendado na literatura de processamento de sinais eletrofisiológicos (WIDMANN; SCHRÖGER; MAESS, 2015; DELORME et al., 2023), todos os filtros são aplicados bidirecionalmente, anulando distorções de fase mediante a aplicação do filtro no sentido direto e reverso sobre os dados.

5.2.1 Remoção de Tendência Linear

A remoção de tendência linear elimina componentes de deriva de baixa frequência presentes no sinal. Estas derivas são causadas por artefatos de movimento de eletrodos, variações lentas na impedância de contato ou potenciais eletroquímicos na interface eletrodo-pele (JIANG; BIAN; TIAN, 2019). A operação consiste no ajuste de uma reta pelo método dos mínimos quadrados aos dados de cada canal, seguido da subtração desta reta do sinal original. Esta operação é aplicada ao sinal contínuo de cada sessão, antes da filtragem temporal, reduzindo componentes de deriva que poderiam afetar as estimativas de potência espectral e covariância nas etapas subsequentes.

A remoção de tendência linear atua de forma complementar ao filtro passa-alta subsequente. Enquanto o filtro passa-alta atenua componentes de frequência abaixo do corte inferior (4 Hz), a remoção de tendência elimina especificamente a componente DC e derivas lineares que poderiam saturar a resposta do filtro ou introduzir transientes indesejados (COHEN, 2014).

5.2.2 Filtragem Notch

A interferência da rede elétrica constitui uma fonte de ruído em registros de EEG. No Brasil, a frequência da rede elétrica é de 60 Hz. Para remoção desta interferência, utiliza-se um filtro notch de resposta ao impulso infinita centrado em 60 Hz, com fator de qualidade $Q = 30$. O fator de qualidade determina a seletividade do filtro, sendo definido como a razão entre a

frequência central e a largura de banda a -3 dB. Valores típicos na literatura variam entre 25 e 35 (DELORME et al., 2023; MITRA; BOKIL, 2007). Com $Q = 30$, a largura de banda do notch é de aproximadamente 2 Hz, proporcionando rejeição da interferência sem atenuação excessiva das frequências adjacentes.

5.2.3 Filtragem Passa-Banda

Após a filtragem notch, aplica-se um filtro passa-banda *Butterworth* de quarta ordem com frequências de corte de 4 Hz e 40 Hz. A escolha desta faixa espectral fundamenta-se na preservação das bandas relevantes para imaginação motora: ritmos theta (4–8 Hz), ritmos mu (8–13 Hz) e beta (13–30 Hz), que apresentam dessincronização relacionada a eventos durante imaginação motora, e a porção inferior da banda gamma (30–40 Hz) (PFURTSCHELLER; NEUPER, 2001).

Blankertz et al. (2008) utilizaram faixas entre 8–30 Hz para análise dos ritmos mu e beta enquanto Tangermann et al. (2012), Altaheri et al. (2023) adotaram faixas mais amplas como 4–40 Hz ou 0,5–40 Hz para preservar componentes theta e gamma. A faixa 4–40 Hz foi adotada considerando que: (i) o corte inferior em 4 Hz remove artefatos de movimento e deriva sem eliminar componentes theta potencialmente informativos; (ii) o corte superior em 40 Hz atenua artefatos eletromiográficos e ruído de alta frequência (AL-QAZZAZ et al., 2023).

5.3 Segmentação em Janelas Temporais

Após a divisão em conjuntos, cada sessão é segmentada em janelas temporais de 2 segundos (500 amostras a 250 Hz). A duração de 2 segundos representa um compromisso entre resolução temporal e estabilidade estatística da estimativa de covariância, sendo um valor adotado em estudos de imaginação motora (LOTTE et al., 2018a; GAUR et al., 2021). Janelas curtas (menores que 1 segundo) podem não capturar adequadamente a dinâmica dos ritmos sensorio-motores, enquanto janelas longas (maiores que 4 segundos) reduzem a resolução temporal e aumentam a probabilidade de não-estacionariedade dentro da janela.

A segmentação permite configuração independente de sobreposição para cada conjunto. A sobreposição é expressa como proporção do comprimento da janela, e o deslocamento temporal entre janelas consecutivas é calculado conforme a Equação 5.1

$$\Delta t = (1 - p_{\text{overlap}}) \times T_{\text{janela}} \quad (5.1)$$

onde p_{overlap} representa a proporção de sobreposição ($0 \leq p_{\text{overlap}} < 1$) e T_{janela} a duração da janela em amostras (500). Uma sobreposição de 75% resulta em deslocamento de 125 amostras (0,5 segundo), enquanto sobreposição de 85% resulta em deslocamento de 75 amostras (0,3 segundo).

Gaur et al. (2021), Lee e Jeong (2023) reportam sobreposições entre 50% e 90% para treinamento, com valores menores para avaliação. A sobreposição diferenciada entre conjuntos segue o princípio de que maior sobreposição no conjunto de treinamento funciona como forma de aumento de dados, enquanto sobreposições menores nos conjuntos de validação e teste preservam maior independência entre as amostras avaliadas (LOTTE et al., 2018a).

Os valores utilizados neste trabalho foram: 85% de sobreposição para o conjunto de treinamento e 40% para os conjuntos de validação e teste. Após testes experimentais, esses valores foram escolhidos para aumentar a quantidade de amostras de treinamento, permitindo maior diversidade de exemplos para o aprendizado, mas sem inserir dependência excessiva entre amostras nos conjuntos de avaliação. Essa escolha gerou um banco de dados com 6 550 amostras de treino, 589 amostras de validação e 776 amostras de teste

5.3.1 Atribuição de Rótulos e Critério de Homogeneidade

A atribuição de rótulos às janelas segue o critério de pureza majoritária. O rótulo de cada janela é determinado pela classe majoritária presente na janela. A pureza do rótulo é definida como a fração de amostras pertencentes à classe majoritária, calculada conforme a Equação 5.2

$$\text{Pureza} = \frac{n_{\text{majoritária}}}{n_{\text{total}}} \quad (5.2)$$

onde $n_{\text{majoritária}}$ representa o número de amostras da classe majoritária e n_{total} o número total de amostras na janela (500).

O limiar de homogeneidade adotado foi de 80%. Este valor difere de limiares mais conservadores (90%) reportado por Lotte et al. (2018a), porém é complementado pela zona de exclusão temporal. A zona de exclusão é verificada para cada janela candidata: se qualquer amostra da janela está a menos de 75 amostras (0,3 segundo a 250 Hz) de um ponto de transição

de rótulo, a janela é descartada. Este critério é aplicado após a verificação de homogeneidade, constituindo um filtro adicional para garantir consistência temporal dos rótulos. Este procedimento reduz a contaminação por estados transitórios que não correspondem a nenhuma das classes de interesse (LEMM et al., 2011).

5.3.2 Controle de Qualidade das Janelas

O controle de qualidade das janelas constitui etapa de rejeição automática de artefatos, aplicada individualmente a cada janela candidata após a segmentação. Os critérios de rejeição incluem: detecção de amplitude baseada em estatísticas da sessão, verificação de variância mínima por canal, detecção de saturação, detecção de segmento plano e verificação de curtose.

O limiar de amplitude para detecção de artefatos é calculado individualmente para cada sessão, utilizando estatísticas baseadas na mediana e no desvio absoluto mediano (MAD). Para cada sessão, calcula-se a amplitude máxima absoluta em cada instante temporal (máximo sobre canais), obtendo-se uma distribuição de amplitudes máximas. O limiar superior é definido conforme a Equação 5.3

$$\tau_{\text{superior}} = \text{Med}(A) + k \cdot 1,4826 \cdot \text{MAD}(A) \quad (5.3)$$

onde A representa o vetor de amplitudes máximas por amostra, $\text{Med}(\cdot)$ denota a mediana, $\text{MAD}(\cdot)$ o desvio absoluto mediano, k o fator multiplicativo, e 1,4826 o fator de consistência que torna o MAD um estimador consistente do desvio-padrão para distribuições normais (ROUSSEEUW; CROUX, 1993).

Valores típicos na literatura de rejeição automática de artefatos em EEG variam entre 3 e 5 desvios (NOLAN; WHELAN; REILLY, 2010; JAS et al., 2017). Limiares próximos a 3σ são mais conservadores e podem rejeitar segmentos com atividade neural de amplitude elevada; limiares próximos a 5σ são mais permissivos e podem reter alguns artefatos de baixa amplitude. A escolha de $k = 5$ favorece a preservação de dados em detrimento da rejeição completa de artefatos, assumindo que a classificação subsequente pode ser parcialmente robusta a ruído residual.

Canais com variância próxima de zero indicam falha de contato ou canal desconectado. Para cada janela, calcula-se a variância temporal de cada canal. Se qualquer canal apresenta

variância inferior a 10^{-12} , a janela é rejeitada. Este limiar é definido como um valor numérico próximo à precisão de máquina para dados em ponto flutuante de dupla precisão, detectando apenas canais efetivamente sem sinal.

5.4 Geometria Riemanniana e Espaço Tangente

A abordagem Riemanniana para classificação de EEG explora propriedades geométricas das matrizes de covariância que são invariantes a transformações de escala e rotação. Esta invariância confere tolerância a artefatos de amplitude e variabilidade inter-sessão (CONGEDO; BARACHANT; BHATIA, 2017).

Para cada janela temporal contendo N_c canais e T amostras temporais, estima-se a matriz de covariância $\Sigma \in \mathbb{R}^{N_c \times N_c}$. A estimativa amostral da covariância é dada pela Equação 5.4

$$\hat{\Sigma}_{\text{sample}} = \frac{1}{T-1} X X^T \quad (5.4)$$

onde $X \in \mathbb{R}^{N_c \times T}$ representa a matriz de dados da janela com média removida por canal. Quando o número de amostras temporais T é comparável ao número de canais N_c , a matriz de covariância amostral torna-se mal-condicionada ou singular (LEDOIT; WOLF, 2004).

Para garantir que a matriz de covariância seja positiva definida, aplica-se regularização por mistura com a matriz identidade escalada. A matriz regularizada é calculada conforme a Equação 5.5

$$\hat{\Sigma}_{\text{reg}} = (1 - \lambda) \hat{\Sigma}_{\text{sample}} + \lambda \cdot \frac{\text{tr}(\hat{\Sigma}_{\text{sample}})}{N_c} \cdot I \quad (5.5)$$

onde λ é o coeficiente de regularização, $\text{tr}(\cdot)$ denota o traço da matriz, e I a matriz identidade de dimensão $N_c \times N_c$. O coeficiente de regularização adotado é $\lambda = 10^{-6}$, um valor baixo que preserva a estrutura da covariância amostral enquanto garante definibilidade positiva.

Adicionalmente, após a regularização, verifica-se o menor autovalor da matriz resultante. Se o menor autovalor é inferior a 10^{-10} , adiciona-se $\varepsilon \cdot I$ à matriz, onde $\varepsilon = \max(10^{-8}, |\lambda_{\min}| + 10^{-8})$. Este procedimento garante que a matriz final seja positiva definida para todas as operações subsequentes (logaritmo matricial, raiz quadrada matricial).

O procedimento adotado neste trabalho utiliza regularização com coeficiente fixo complementada por correção explícita de autovalores, priorizando a garantia de definibilidade positiva sobre a otimalidade estatística do estimador.

5.4.1 Média Log-Euclidiana

A média das matrizes de covariância do conjunto de treinamento é calculada utilizando a métrica log-euclidiana. Nesta métrica, a média é computada no espaço dos logaritmos matriciais, conforme a Equação 5.6

$$\bar{\Sigma} = \exp \left(\frac{1}{n} \sum_{i=1}^n \log(\Sigma_i) \right) \quad (5.6)$$

onde $\log(\cdot)$ e $\exp(\cdot)$ denotam o logaritmo e a exponencial matriciais, respectivamente, e Σ_i representa a matriz de covariância da i -ésima janela. A média log-euclidiana possui solução analítica fechada, diferentemente da média Riemanniana sob a métrica afim-invariante que requer algoritmo iterativo (ARSIGNY et al., 2007).

A média é calculada exclusivamente com as matrizes de covariância do conjunto de treinamento. Durante a aplicação aos conjuntos de validação e teste, utiliza-se a mesma matriz de referência, sem recálculo. Este procedimento previne vazamento de informação do conjunto de teste para o processo de transformação das características (RODRIGUES; JUTTEN; CONGEDO, 2019).

5.4.2 Projeção no Espaço Tangente

O espaço tangente é um espaço vetorial Euclidiano que aproxima localmente a variedade no ponto de referência. Esta linearização permite que classificadores desenvolvidos para espaços Euclidianos sejam aplicados às características projetadas (BARACHANT et al., 2012).

Dada a matriz de referência $\bar{\Sigma}$ (média log-euclidiana do conjunto de treinamento), a projeção de uma matriz Σ para o espaço tangente é calculada pelo mapeamento logarítmico conforme a Equação 5.7

$$S = \bar{\Sigma}^{1/2} \log \left(\bar{\Sigma}^{-1/2} \Sigma \bar{\Sigma}^{-1/2} \right) \bar{\Sigma}^{1/2} \quad (5.7)$$

onde $\bar{\Sigma}^{1/2}$ e $\bar{\Sigma}^{-1/2}$ denotam a raiz quadrada matricial e sua inversa, respectivamente. A matriz tangente resultante S é simétrica, permitindo sua vetorização mediante extração dos elementos da matriz triangular superior (incluindo a diagonal).

Para vetorização, os elementos fora da diagonal são multiplicados por $\sqrt{2}$ para preservar a norma de Frobenius durante a conversão para vetor, conforme a Equação 5.8

$$v_k = \begin{cases} S_{ii} & \text{se } k \text{ corresponde a elemento diagonal} \\ \sqrt{2} \cdot S_{ij} & \text{se } k \text{ corresponde a elemento } i < j \end{cases} \quad (5.8)$$

Para N_c canais, a dimensionalidade do vetor de características é $d = N_c(N_c + 1)/2$. A Tab. 5.1 apresenta a dimensionalidade para diferentes configurações de canais.

Tabela 5.1 – Dimensionalidade das características Riemannianas em função do número de canais.

Canais (N_c)	Fórmula	Dimensões
4	$4 \times 5/2$	10
8	$8 \times 9/2$	36

5.5 Remoção de Características Redundantes

Características com variância próxima de zero não contribuem para discriminação entre classes e podem introduzir instabilidade numérica. Tais características são removidas quando sua variância é inferior a um limiar pré-estabelecido, conforme a Equação 5.9

$$\text{Var}(x_j) < \varepsilon, \quad \varepsilon = 10^{-10} \quad (5.9)$$

onde x_j representa o vetor contendo todos os valores da j -ésima característica no conjunto de dados.

Características com correlação linear elevada introduzem redundância. O coeficiente de correlação de Pearson entre cada par de características (i, j) é calculado conforme a Equação 5.10

$$\rho_{ij} = \frac{\sum_{n=1}^N (x_{i,n} - \bar{x}_i)(x_{j,n} - \bar{x}_j)}{\sqrt{\sum_{n=1}^N (x_{i,n} - \bar{x}_i)^2} \sqrt{\sum_{n=1}^N (x_{j,n} - \bar{x}_j)^2}} \quad (5.10)$$

onde N representa o número total de amostras, $\bar{x}_i$ e $\bar{x}_j$ denotam as médias das características i e j . Pares de características com correlação absoluta superior a 0,95 são identificados como redundantes (KUHN; JOHNSON, 2013), nestes casos, uma das características do par é removida.

5.6 Balanceamento de Classes (Reamostragem do Conjunto de Treinamento)

O desbalanceamento entre classes afeta diretamente a capacidade de generalização de classificadores, pois induz o modelo a minimizar o erro global à custa da sub-representação das classes minoritárias, comprometendo métricas sensíveis à classe, como recall e F1-score. Para reduzir esse viés durante o treinamento, foram empregadas técnicas de reamostragem apenas no conjunto de treino, assegurando que os conjuntos de validação e teste permanecessem inalterados, de modo a garantir uma avaliação estatisticamente consistente e livre de vazamento de informação. Neste trabalho foram utilizadas técnicas de adição e remoção de amostras

5.6.1 Técnicas de Subamostragem

As técnicas de subamostragem reduzem a cardinalidade da classe majoritária por meio da remoção seletiva de instâncias. A subamostragem aleatória (*Random Undersampling*) consiste na eliminação estocástica de exemplos da classe majoritária até que se atinja uma distribuição mais equilibrada (LEMAÎTRE; NOGUEIRA; ARIDAS, 2017). Embora computacionalmente eficiente, essa abordagem incorre no risco de descartar informações discriminativas potencialmente relevantes para a fronteira de decisão.

A remoção de pares de Tomek (*Tomek Links*) identifica pares de instâncias pertencentes a classes distintas que constituem mutuamente seus vizinhos mais próximos (TOMEK, 1976). A eliminação desses pares ambíguos visa clarificar a região de fronteira entre classes, removendo exemplos ruidosos ou sobrepostos que dificultam a discriminação. Essa técnica opera sob o princípio de que instâncias situadas em regiões de alta sobreposição interclasse tendem a degradar o desempenho de classificadores.

O método *Edited Nearest Neighbours* (ENN) remove instâncias da classe majoritária cuja classificação, determinada pela maioria de seus k vizinhos mais próximos, difere de seu rótulo verdadeiro (WILSON, 1972). Essa abordagem de limpeza de dados elimina exemplos potencialmente ruidosos ou situados em regiões de sobreposição, resultando em fronteiras de

decisão mais suaves. O parâmetro k determina o grau de conservadorismo da remoção, sendo valores maiores associados a critérios mais restritivos.

A família de métodos *NearMiss* seleciona instâncias da classe majoritária com base em sua proximidade às instâncias minoritárias (MANI; ZHANG, 2003). A variante NearMiss-1, empregada neste estudo, retém as instâncias majoritárias cuja distância média aos k vizinhos mais próximos da classe minoritária é mínima. Essa estratégia visa preservar instâncias majoritárias informativas para a definição da fronteira de decisão, descartando aquelas localizadas em regiões distantes da interface entre classes.

5.6.2 Técnicas de Sobreamostragem

As técnicas de sobreamostragem ampliam a representação da classe minoritária por meio da geração de novas instâncias. A sobreamostragem aleatória (*Random Oversampling*) replica instâncias existentes da classe minoritária, constituindo a abordagem mais elementar dessa categoria (LEMAÎTRE; NOGUEIRA; ARIDAS, 2017). Embora preserve a distribuição original dos dados minoritários, essa técnica pode induzir sobreajuste ao introduzir múltiplas cópias idênticas de instâncias.

O algoritmo *Synthetic Minority Over-sampling Technique (SMOTE)* gera instâncias sintéticas por interpolação linear entre exemplos minoritários e seus k vizinhos mais próximos da mesma classe (CHAWLA et al., 2002). Para cada instância minoritária selecionada, uma nova amostra é criada ao longo do segmento de reta que conecta essa instância a um de seus vizinhos, com posição determinada por um fator aleatório no intervalo $[0, 1]$. Essa estratégia amplia a região de cobertura da classe minoritária no espaço de características, potencialmente melhorando a capacidade de generalização do modelo.

A variante *Borderline-SMOTE* concentra a geração de instâncias sintéticas em exemplos minoritários situados próximos à fronteira de decisão (HAN; WANG; MAO, 2005). Instâncias minoritárias são classificadas como seguras, limítrofes ou ruidosas com base na proporção de vizinhos pertencentes à classe majoritária. A geração de amostras sintéticas ocorre preferencialmente a partir de instâncias limítrofes, sob a premissa de que essas regiões são mais críticas para o desempenho do classificador. Os parâmetros k e m controlam, respectivamente, o número de vizinhos para interpolação e para classificação do tipo de instância.

O algoritmo *Adaptive Synthetic Sampling* (ADASYN) adapta a densidade de amostras sintéticas geradas em função da dificuldade de aprendizado local (HE et al., 2008). Instâncias minoritárias cercadas por maior proporção de exemplos majoritários recebem peso superior na geração de amostras sintéticas. Essa distribuição adaptativa visa concentrar o esforço de sobreamostragem em regiões onde a classe minoritária é sub-representada ou onde a fronteira de decisão é mais difusa.

5.6.3 Técnicas Híbridas

As técnicas híbridas combinam sobreamostragem e subamostragem em sequência, buscando mitigar limitações inerentes a cada abordagem isolada (BATISTA; PRATI; MONARD, 2004). O método *SMOTE-Tomek* aplica inicialmente SMOTE para sobreamostragem da classe minoritária, seguido da remoção de pares de Tomek que possam ter sido introduzidos ou evidenciados pelo processo de geração sintética. Essa combinação visa equilibrar a distribuição de classes enquanto remove instâncias ambíguas na região de fronteira.

O método *SMOTE-ENN* combina a geração de instâncias sintéticas por SMOTE com a limpeza de dados via *Edited Nearest Neighbours* (BATISTA; PRATI; MONARD, 2004). Após a sobreamostragem, o algoritmo ENN remove instâncias de ambas as classes cujos rótulos divergem da classificação por vizinhança. Essa abordagem tende a produzir fronteiras de decisão mais definidas ao eliminar instâncias potencialmente ruidosas, tanto originais quanto sintéticas.

As técnicas baseadas em vizinhança requerem que o parâmetro k seja inferior ao número de instâncias na menor classe, constituindo uma restrição para a viabilidade do algoritmo (FERNÁNDEZ et al., 2018). Valores de k iguais ou superiores à cardinalidade da classe minoritária impossibilitam a identificação de k vizinhos distintos, tornando a configuração inválida.

5.7 Seleção Automática do Método de Ranqueamento de Características

A alta dimensionalidade dos sinais EEG e a presença de características irrelevantes comprometem o desempenho de classificadores em BCIs (THENG; BHOYAR, 2024). Este trabalho propõe a seleção automática do método de ranqueamento mais adequado através de validação empírica, comparando três abordagens complementares: ANOVA F-statistic, informação mútua e regularização L1. A estratégia adaptativa justifica-se pela observação de que diferentes

métodos apresentam desempenho variável conforme as características do conjunto de dados (BUYUKKECECI; OKUR, 2023).

5.7.1 Procedimento de Comparação

A comparação emprega validação cruzada estratificada com $K = 3$ partições, garantindo representação proporcional das classes em cada fold (REFAEILZADEH; TANG; LIU, 2009). O classificador LDA avalia cada método devido à sua solução analítica fechada, que elimina variabilidade de processos iterativos e permite que diferenças observadas reflitam as características dos métodos de ranqueamento.

O desempenho de cada método s é quantificado pela média do F1-Score macro nas partições conforme Equação 5.11

$$\text{Score}(s) = \frac{1}{K} \sum_{k=1}^K F1_{\text{macro},k}(s) \quad (5.11)$$

O método com maior score é selecionado automaticamente. Empates, definidos como diferenças inferiores a 0,5 pontos percentuais, priorizam ANOVA por sua estabilidade numérica e menor custo computacional.

5.7.2 ANOVA F-statistic

A estatística F quantifica a razão entre a variabilidade das médias entre classes e a variabilidade observada dentro de cada classe conforme Equação 5.12

$$F_j = \frac{\sum_{c=1}^C n_c (\bar{x}_{j,c} - \bar{x}_j)^2 / (C - 1)}{\sum_{c=1}^C \sum_{n \in \mathcal{D}_c} (x_{j,n} - \bar{x}_{j,c})^2 / (N - C)} \quad (5.12)$$

onde C é o número de classes, n_c o número de amostras da classe c , $\bar{x}_{j,c}$ a média da característica j na classe c , $\bar{x}_j$ a média global, $\mathcal{D}_c$ o conjunto de índices das amostras da classe c e N o total de amostras.

Valores elevados de F indicam que as médias entre classes diferem substancialmente em relação à dispersão interna, sugerindo alto poder discriminativo. Em sinais EEG, características espectrais como a potência em bandas de frequência frequentemente apresentam médias distin-

tas entre estados mentais diferentes, tornando o método ANOVA eficaz para identificar essas diferenças (HASTIE; TIBSHIRANI; FRIEDMAN, 2009).

A principal limitação reside em sua natureza univariada e linear. Processos neurais subjacentes à imaginação motora podem manifestar padrões espaciais complexos onde a informação discriminativa emerge da combinação de múltiplos eletrodos, e não de canais isolados. Além disso, relações não lineares entre a atividade cerebral e as classes de movimento, comuns em EEG devido à natureza dos geradores corticais, não são capturadas pela estatística F (ALOTAIBY et al., 2023). Apesar dessas limitações, o baixo custo computacional e a robustez a ruído tornam o método adequado para sistemas BCI com restrições temporais.

5.7.3 Informação Mútua

A informação mútua mede a dependência estatística entre uma característica X_j e a variável de classe Y , quantificando a redução na incerteza sobre a classe ao conhecer o valor da característica pela Equação 5.13

$$I(X_j; Y) = H(Y) - H(Y|X_j) \quad (5.13)$$

onde $H(Y)$ é a entropia da classe e $H(Y|X_j)$ a entropia condicional dado X_j . Quando $I = 0$, a característica e a classe são estatisticamente independentes; valores maiores indicam maior capacidade discriminativa.

Diferentemente da correlação linear, a informação mútua captura dependências arbitrariamente complexas (COVER; THOMAS, 2006). Esta propriedade é relevante para sinais EEG, onde a relação entre potenciais corticais e estados cognitivos frequentemente envolve não linearidades. Por exemplo, a dessincronização relacionada a eventos (ERD) em bandas mu e beta pode apresentar comportamento não monotônico com a intensidade da imaginação motora, relação que métodos lineares subestimariam.

A estimação utiliza o método k -NN de Kraskov et al. (KRASKOV; STÖGBAUER; GRASSBERGER, 2004), implementado em *scikit-learn*, que se adapta à densidade local dos dados sem assumir distribuições paramétricas. O custo computacional superior ao ANOVA é justificado quando as relações entre características e classes são intrinsecamente não lineares. Entretanto, assim como o ANOVA, o método avalia cada característica individualmente, igno-

rando possíveis redundâncias entre características correlacionadas, situação comum em EEG devido à condução de volume que propaga sinais para eletrodos vizinhos.

5.7.4 Regularização L1 (LASSO)

O LASSO integra a seleção de características ao processo de treinamento, avaliando-as simultaneamente através da minimização (TIBSHIRANI, 1996) conforme Equação 5.14

$$\min_{\mathbf{w}, b} \left[\frac{1}{N} \sum_{n=1}^N \mathcal{L}(y_n, f(\mathbf{x}_n; \mathbf{w}, b)) + \lambda \|\mathbf{w}\|_1 \right] \quad (5.14)$$

onde $\mathcal{L}$ é a perda logística, λ o parâmetro de regularização e $\|\mathbf{w}\|_1 = \sum_j |w_j|$ a norma L1 do vetor de pesos.

A penalidade L1 possui a propriedade de forçar coeficientes de características menos relevantes exatamente para zero durante a otimização. Características com magnitudes $|w_j|$ maiores contribuem mais para a discriminação. Para problemas multiclasse com esquema um-contra-resto, o score considera a maior contribuição entre todas as classes conforme descrito na Equação 5.15

$$\text{Score}(x_j) = \max_c |w_{j,c}| \quad (5.15)$$

A natureza multivariada do LASSO é útil onde a condução de volume gera alta correlação entre eletrodos adjacentes. Enquanto métodos univariados podem selecionar múltiplas características redundantes de uma mesma região cortical, o LASSO tende a reter apenas as mais informativas, produzindo subconjuntos mais compactos.

A principal limitação é a instabilidade de seleção em presença de características altamente correlacionadas: pequenas perturbações nos dados podem resultar na seleção de diferentes características do mesmo grupo correlacionado (ZOU; HASTIE, 2005). Além disso, o método requer padronização prévia dos dados, pois características com magnitudes numéricas maiores receberiam penalidades desproporcionais.

5.7.5 Determinação do Número de Características

O número k de características selecionadas é determinado pelo critério de representatividade percentual, que oferece adaptabilidade automática à estrutura informacional do conjunto

de dados. Os scores são normalizados para representar contribuições proporcionais:

$$s_j^{\text{norm}} = \frac{s_j}{\sum_{i=1}^d s_i} \quad (5.16)$$

Seleciona-se o menor subconjunto cuja importância acumulada atinja o limiar de 85%:

$$k = \min \left\{ n : \sum_{j=1}^n s_{\pi(j)}^{\text{norm}} \geq 0,85 \right\} \quad (5.17)$$

onde π ordena as características por score decrescente. O valor é restrito ao intervalo $[1, 50]$ para prevenir casos extremos.

Esta abordagem adapta-se automaticamente às características do problema: em conjuntos onde a informação discriminativa concentra-se em poucas bandas de frequência ou regiões corticais específicas, um número reduzido de características é selecionado; quando a informação distribui-se uniformemente, como em tarefas que envolvem múltiplas áreas cerebrais simultaneamente, mais características são necessárias para atingir o mesmo nível de representatividade.

5.8 Modelos de Classificação

O sistema de classificação emprega seis algoritmos multiclasse: Perceptron Multicamadas (MLP), Máquinas de Vetores de Suporte (SVM), Floresta Aleatória, Extra Trees, Gradient Boosting e XGBoost. A otimização dos hiperparâmetros foi conduzida mediante busca aleatória com validação cruzada estratificada. Com base nos resultados individuais, implementou-se um sistema de comitê por votação suave que agrega as predições dos modelos com melhor desempenho, seguindo a abordagem de *ensemble learning* preconizada por Zhou (2012).

5.8.1 Perceptron Multicamadas

O Perceptron Multicamadas constitui uma rede neural artificial *feedforward* capaz de aprender representações não lineares por meio da composição de transformações afins e funções de ativação. A arquitetura compreende camadas de neurônios artificiais organizados hierarquicamente, incluindo camada de entrada para recepção das características, camadas ocultas para extração de representações intermediárias e camada de saída para produção das probabilidades de classificação (HAYKIN, 2009; GOODFELLOW; BENGIO; COURVILLE, 2016).

A ativação de cada neurônio i em uma camada oculta l é computada em duas etapas. A primeira etapa calcula a soma ponderada das entradas conforme a Equação 5.18

$$z_i^{(l)} = \sum_j w_{ij}^{(l)} a_j^{(l-1)} + b_i^{(l)} \quad (5.18)$$

onde $a_j^{(l-1)}$ representa as ativações dos neurônios da camada anterior, $w_{ij}^{(l)}$ denota os pesos sinápticos conectando o neurônio j da camada $l-1$ ao neurônio i da camada l , e $b_i^{(l)}$ corresponde ao termo de viés. Os pesos sinápticos codificam a importância relativa de cada conexão, enquanto o viés possibilita deslocamentos da função de ativação independentemente das entradas.

A segunda etapa aplica a função de ativação ReLU (*Rectified Linear Unit*), proposta por (NAIR; HINTON, 2010), conforme a Equação 5.19

$$a_i^{(l)} = \phi(z_i^{(l)}) = \max(0, z_i^{(l)}) \quad (5.19)$$

Esta função introduz não linearidade essencial para o aprendizado de padrões complexos, permitindo que a rede aproxime funções arbitrárias segundo o teorema da aproximação universal (CYBENKO, 1989). A ReLU apresenta vantagens computacionais sobre funções sigmóides tradicionais, notadamente gradientes não saturantes que facilitam a otimização em redes profundas (GLOROT; BORDES; BENGIO, 2011).

A camada de saída utiliza a função *softmax* para produzir uma distribuição de probabilidade sobre as classes, conforme a Equação 5.20

$$P(y = c | \mathbf{x}) = \frac{\exp(z_c)}{\sum_{c'=1}^C \exp(z_{c'})} \quad (5.20)$$

onde z_c representa a ativação pré-*softmax* (logit) da classe c , C denota o número total de classes, e a exponencial garante probabilidades positivas. A normalização pelo denominador assegura que $\sum_{c=1}^C P(y = c | \mathbf{x}) = 1$, satisfazendo os axiomas de probabilidade.

O treinamento é realizado pelo algoritmo Adam (*Adaptive Moment Estimation*), proposto por (KINGMA; BA, 2015). Este otimizador combina ideias de momento e taxas de aprendizado adaptativas, mantendo médias exponencialmente decaídas dos gradientes (primeiro momento) e dos gradientes quadrados (segundo momento). O ajuste automático da taxa de

aprendizado para cada parâmetro resulta em convergência mais rápida e estável comparada ao gradiente descendente estocástico tradicional.

A função objetivo combina a entropia cruzada com regularização L2 conforme a Equação 5.21

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{CE}} + \alpha \sum_l \|\mathbf{W}^{(l)}\|_F^2 \quad (5.21)$$

onde $\mathcal{L}_{\text{CE}}$ representa a entropia cruzada, definida na Equação 5.22

$$\mathcal{L}_{\text{CE}} = -\frac{1}{N} \sum_{n=1}^N \sum_{c=1}^C y_{n,c} \log P(y = c | \mathbf{x}_n) \quad (5.22)$$

sendo $y_{n,c} \in \{0, 1\}$ a codificação *one-hot* da classe verdadeira da amostra n . O termo $\|\mathbf{W}^{(l)}\|_F^2 = \sum_{i,j} (w_{ij}^{(l)})^2$ corresponde à norma de Frobenius dos pesos da camada l . O parâmetro de regularização α controla o compromisso entre ajuste aos dados de treinamento e simplicidade do modelo, penalizando pesos de magnitude elevada que podem indicar sobreajuste (HASTIE; TIBSHIRANI; FRIEDMAN, 2009; NG, 2004).

A otimização de hiperparâmetros via busca aleatória explorou configurações de arquitetura variando de redes rasas (uma camada) até redes mais profundas (três camadas), com tamanhos de camadas amostrados entre 32 e 512 neurônios. O coeficiente de regularização α foi amostrado na escala logarítmica no intervalo $[10^{-5}, 10^{-1}]$, a taxa de aprendizado inicial no intervalo $[10^{-5}, 10^{-2}]$, e o tamanho do lote entre 32 e 256 amostras. A parada antecipada (*early stopping*) foi configurada para interromper o treinamento após 20 épocas consecutivas sem melhoria na métrica de validação, prevenindo sobreajuste conforme recomendado por (PRECHT, 1998).

5.8.2 Máquinas de Vetores de Suporte

As Máquinas de Vetores de Suporte buscam o hiperplano de separação ótimo que maximiza a margem entre as classes em um espaço de características potencialmente de alta dimensionalidade (CORTES; VAPNIK, 1995; VAPNIK, 1998). A fundamentação teórica baseia-se na teoria do aprendizado estatístico de Vapnik, que estabelece limites de generalização relacionados à margem de separação.

Para o caso binário, o SVM resolve um problema de otimização convexa com restrições. A formulação primal é expressa na Equação 5.23

$$\min_{\mathbf{w}, b, \xi} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{n=1}^N \xi_n \quad (5.23)$$

sujeito às restrições da Equação 5.24

$$y_n(\mathbf{w}^T \phi(\mathbf{x}_n) + b) \geq 1 - \xi_n, \quad \xi_n \geq 0, \quad \forall n \quad (5.24)$$

onde $\mathbf{w}$ denota o vetor normal ao hiperplano, b o termo de interceptação, $\phi(\mathbf{x})$ a transformação implícita para um espaço de características de maior dimensão induzida pelo *kernel*, $y_n \in \{-1, +1\}$ os rótulos das classes, e ξ_n as variáveis de folga que permitem violações suaves da margem.

O termo $\frac{1}{2} \|\mathbf{w}\|^2$ controla a complexidade do modelo, sendo equivalente à maximização da margem geométrica $2/\|\mathbf{w}\|$ entre as classes (SCHÖLKOPF; SMOLA, 2002). O parâmetro de regularização C estabelece o compromisso entre maximização da margem e minimização dos erros de classificação. Valores pequenos de C favorecem margens mais amplas tolerando mais erros, enquanto valores grandes enfatizam a classificação correta dos pontos de treinamento (CRISTIANINI; SHAW-TAYLOR, 2000; HSU; CHANG; LIN, 2003).

Para problemas não linearmente separáveis, o *kernel* RBF (Gaussiano) mapeia implicitamente os dados para um espaço de dimensão infinita, conforme a Equação 5.25

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2) \quad (5.25)$$

onde γ controla a influência de cada amostra de treinamento. Valores pequenos de γ produzem funções de decisão suaves, enquanto valores grandes criam regiões de decisão altamente localizadas. O *kernel* RBF satisfaz o teorema de Mercer, garantindo correspondência a um produto interno em algum espaço de Hilbert, o que assegura convexidade do problema de otimização e unicidade da solução (ARONSZAJN, 1950).

Para problemas com $C > 2$ classes, emprega-se a estratégia um-contra-resto (*one-vs-rest*), que treina C classificadores binários independentes (RIFKIN; KLAUTAU, 2004). A classe

final é determinada pela Equação 5.26

$$\hat{y} = \arg \max_c (\mathbf{w}_c^T \phi(\mathbf{x}) + b_c) \quad (5.26)$$

selecionando a classe cujo classificador produz o maior valor de decisão.

Para compensar o desbalanceamento entre classes, pesos de classe são aplicados automaticamente conforme a Equação 5.27

$$w_c = \frac{N}{C \cdot n_c} \quad (5.27)$$

onde N é o número total de amostras, C o número de classes, e n_c o número de amostras da classe c . Este ajuste modifica o parâmetro de custo efetivo para cada classe, tornando erros em classes minoritárias proporcionalmente mais custosos (HE; GARCIA, 2009).

A otimização de hiperparâmetros explorou o parâmetro de regularização C no intervalo logarítmico $[10^{-2}, 10^3]$, o parâmetro do *kernel* γ em $[10^{-5}, 10^1]$, além de diferentes tipos de *kernel* (linear, polinomial e RBF). A escolha adequada de C e γ depende da separabilidade dos dados e do nível de ruído, conforme discutido por (CHAPELLE et al., 2002).

5.8.3 Floresta Aleatória

A Floresta Aleatória é um método de *ensemble* que constrói múltiplas árvores de decisão e agrega suas predições por votação (BREIMAN, 2001). Este algoritmo combate o sobreajuste característico de árvores individuais através de duas fontes de aleatoriedade: amostragem *bootstrap* dos dados e seleção aleatória de características a cada divisão.

Cada árvore t do *ensemble*, com $t \in \{1, 2, \dots, T\}$, é treinada independentemente em uma amostra *bootstrap* $\mathcal{D}_t$ do conjunto de treinamento original $\mathcal{D}$ (BREIMAN, 1996). A amostragem *bootstrap* seleciona N amostras com reposição, resultando em aproximadamente 63,2% de amostras únicas em cada $\mathcal{D}_t$. As amostras não selecionadas (*out-of-bag*) podem ser utilizadas para estimativa interna do erro de generalização sem necessidade de validação cruzada separada (WOLPERT, 1992).

Durante a construção de cada árvore, a cada nó de decisão, um subconjunto aleatório de m características é amostrado do total de d características disponíveis. Para classificação, a escolha típica é $m = \sqrt{d}$ (BREIMAN, 2001), embora este valor seja otimizado durante a

busca de hiperparâmetros. Esta aleatoriedade decorrelaciona as árvores, reduzindo a variância do *ensemble* sem aumento significativo do viés (LOUPPE, 2014).

A divisão ótima em cada nó minimiza a impureza de Gini, definida na Equação 5.28

$$G(S) = \sum_{c=1}^C p_c(1 - p_c) = 1 - \sum_{c=1}^C p_c^2 \quad (5.28)$$

onde $p_c = n_c/|S|$ é a proporção de amostras da classe c no nó S , sendo n_c o número de amostras dessa classe e $|S|$ o tamanho total do nó. A impureza de Gini quantifica a probabilidade de classificação incorreta sob atribuição aleatória de rótulos segundo a distribuição de classes no nó (BREIMAN et al., 1984). Um nó puro (todas as amostras da mesma classe) apresenta $G(S) = 0$, enquanto máxima impureza ocorre quando as classes estão uniformemente distribuídas.

O ganho de informação para uma divisão do nó pai S_p em filhos esquerdo S_l e direito S_r é calculado conforme a Equação 5.29

$$\Delta G = G(S_p) - \frac{|S_l|}{|S_p|} G(S_l) - \frac{|S_r|}{|S_p|} G(S_r) \quad (5.29)$$

A divisão que maximiza ΔG é selecionada entre as m características candidatas, buscando reduzir maximamente a impureza média ponderada dos nós filhos (LOUPPE, 2014).

A predição final da floresta para uma amostra $\mathbf{x}$ combina as probabilidades de todas as árvores mediante média aritmética, conforme a Equação 5.30

$$P(y = c|\mathbf{x}) = \frac{1}{T} \sum_{t=1}^T P_t(y = c|\mathbf{x}) \quad (5.30)$$

onde $P_t(y = c|\mathbf{x})$ é a probabilidade estimada pela árvore t , calculada como a proporção de amostras de treinamento da classe c na folha onde $\mathbf{x}$ é classificada. Esta média reduz a variância comparada a uma árvore individual, segundo o teorema do limite central aplicado a preditores correlacionados (HASTIE; TIBSHIRANI; FRIEDMAN, 2009).

A otimização explorou o número de árvores T entre 100 e 1000, profundidade máxima entre 5 e 30 (ou irrestrita), mínimo de amostras para divisão entre 2 e 50, mínimo de amostras em folhas entre 1 e 20, e máximo de características entre $\sqrt{d}$, $\log_2(d)$, $0,5d$, $0,8d$ e d . Conforme (OSHIRO; PEREZ; BARANAUSKAS, 2012), mais árvores geralmente melhoram o desempenho assintoticamente, porém com retornos decrescentes além de certo ponto.

5.8.4 Extra Trees

O algoritmo Extra Trees (*Extremely Randomized Trees*) estende a Floresta Aleatória introduzindo aleatoriedade adicional na seleção dos pontos de corte (GEURTS; ERNST; WEHENKEL, 2006). Enquanto a Floresta Aleatória busca o ponto de corte ótimo entre as características candidatas, o Extra Trees amostra pontos de corte aleatoriamente, selecionando o melhor entre estes candidatos.

Para cada nó e cada característica candidata j , o ponto de corte θ_j é amostrado uniformemente do intervalo de valores observados, conforme a Equação 5.31

$$\theta_j \sim \mathcal{U} \left(\min_{\mathbf{x} \in S} (x_j), \max_{\mathbf{x} \in S} (x_j) \right) \quad (5.31)$$

onde S é o conjunto de amostras no nó atual e x_j representa o valor da característica j para a amostra $\mathbf{x}$. Entre os m pares aleatórios $\{(\text{característica}_j, \theta_j)\}$ gerados, seleciona-se aquele que maximiza o ganho de impureza de Gini conforme a Equação 5.29.

A aleatoriedade adicional no ponto de corte, combinada com a aleatoriedade na seleção de características, resulta em maior diversidade entre as árvores do *ensemble* (GEURTS; ERNST; WEHENKEL, 2006). Segundo a teoria de *ensemble*, a redução do erro pode ser decomposta conforme a Equação 5.32

$$E_{\text{ensemble}} \approx \bar{\rho} \sigma^2 + \frac{1 - \bar{\rho}}{T} \sigma^2 \quad (5.32)$$

onde $\bar{\rho}$ é a correlação média entre preditores, σ^2 a variância individual, e T o número de árvores (DIETTERICH, 2000). Os cortes aleatórios reduzem $\bar{\rho}$, diminuindo o primeiro termo e potencialmente melhorando a generalização.

Diferentemente da Floresta Aleatória, o Extra Trees utiliza o conjunto de treinamento completo para cada árvore, sem amostragem *bootstrap* (GEURTS; ERNST; WEHENKEL, 2006). Esta escolha compensa a maior aleatoriedade nos pontos de corte, mantendo mais informação disponível para cada árvore. O Extra Trees tende a apresentar melhor desempenho em problemas com muitas características irrelevantes ou ruidosas, pois a aleatoriedade adicional atua como regularização implícita. Estudos teóricos demonstram que, sob certas condições

de regularidade, o Extra Trees alcança taxas de convergência ótimas mesmo com menor custo computacional que a Floresta Aleatória (BIAU; DEVROYE; LUGOSI, 2008).

Os hiperparâmetros explorados são análogos aos da Floresta Aleatória, mantendo a mesma estrutura de regularização: número de árvores entre 100 e 1000, profundidade máxima entre 5 e 30 (ou irrestrita), mínimo de amostras para divisão entre 2 e 50, mínimo de amostras em folhas entre 1 e 20, e máximo de características variando entre $\sqrt{d}$ e d .

5.8.5 Gradient Boosting

O Gradient Boosting constrói um *ensemble* aditivo de preditores fracos (tipicamente árvores de decisão rasas) treinados sequencialmente para corrigir os erros residuais dos preditores anteriores (FRIEDMAN, 2001). Diferentemente de métodos de *bagging* que treinam preditores independentemente em paralelo, o *boosting* é inerentemente sequencial, com cada novo modelo focando nas amostras mal classificadas pelos modelos anteriores.

O modelo é construído iterativamente através de expansão funcional. Após m iterações, o modelo acumulado é dado pela Equação 5.33

$$F_m(\mathbf{x}) = F_{m-1}(\mathbf{x}) + \eta \cdot h_m(\mathbf{x}) \quad (5.33)$$

onde $F_0(\mathbf{x})$ é uma predição inicial constante (tipicamente a probabilidade marginal das classes), $h_m(\mathbf{x})$ a árvore fraca adicionada na iteração m , e $\eta \in (0, 1]$ a taxa de aprendizado que controla a contribuição de cada árvore. Taxas menores requerem mais iterações mas geralmente resultam em melhor generalização através de regularização implícita (BUHLMANN; HOTHORN, 2007).

A formulação do Gradient Boosting interpreta a construção do *ensemble* como descida de gradiente em um espaço funcional, minimizando uma função de perda diferenciável $\mathcal{L}(y, F(\mathbf{x}))$ (MASON et al., 1999). A cada iteração, a árvore h_m é ajustada para aproximar o negativo do gradiente da perda, expresso na Equação 5.34

$$r_{n,m} = - \left[\frac{\partial \mathcal{L}(y_n, F(\mathbf{x}_n))}{\partial F(\mathbf{x}_n)} \right]_{F=F_{m-1}} \quad (5.34)$$

onde $r_{n,m}$ são denominados pseudo-resíduos e indicam a direção de maior decréscimo da perda para cada amostra de treinamento n .

Para problemas de classificação com C classes, o Gradient Boosting mantém C modelos aditivos separados, um para cada classe. A função de perda *log-loss* (entropia cruzada multi-classe) é definida na Equação 5.35

$$\mathcal{L}(\mathbf{y}, \mathbf{F}(\mathbf{x})) = - \sum_{c=1}^C y_c \log p_c(\mathbf{x}) \quad (5.35)$$

onde $y_c \in \{0, 1\}$ é a codificação *one-hot*, e as probabilidades são obtidas via *softmax* conforme a Equação 5.36

$$p_c(\mathbf{x}) = \frac{\exp(F_c(\mathbf{x}))}{\sum_{c'=1}^C \exp(F_{c'}(\mathbf{x}))} \quad (5.36)$$

Os pseudo-resíduos para a classe c na iteração m são calculados pela Equação 5.37

$$r_{n,m,c} = y_{n,c} - p_{m-1}(y = c | \mathbf{x}_n) \quad (5.37)$$

representando o erro entre o rótulo verdadeiro e a probabilidade predita pelo modelo atual.

A implementação utiliza *Histogram-based Gradient Boosting*, que discretiza características contínuas em *bins* (histogramas) antes do treinamento (KE et al., 2017). Esta estratégia reduz o custo computacional de encontrar divisões ótimas de $O(nd \log n)$ para $O(nd)$, onde n é o número de amostras e d o número de características. A discretização em tipicamente 255 *bins* introduz regularização adicional, tornando o modelo mais robusto a *outliers* e ruído.

As estratégias de regularização empregadas incluem penalização L2 aplicada aos valores preditos nas folhas, parada antecipada mediante monitoramento de métrica de validação, limitação da profundidade das árvores (tipicamente 3–5), e exigência de número mínimo de amostras em folhas (FRIEDMAN, 2001; PRECHELT, 1998).

A otimização de hiperparâmetros explorou a taxa de aprendizado η no intervalo logarítmico $[0,01, 0,3]$, número de iterações entre 100 e 1000, profundidade máxima entre 3 e 7, mínimo de amostras em folhas entre 10 e 50, e coeficiente de regularização L2 entre 0 e 10. Conforme (NATEKIN; KNOLL, 2013), taxas de aprendizado menores combinadas com maior número de árvores tendem a produzir melhor generalização.

5.8.6 XGBoost

O XGBoost (*eXtreme Gradient Boosting*) implementa uma versão otimizada do *gradient boosting* com inovações algorítmicas que melhoram eficiência computacional, controle de sobreajuste e tratamento de dados esparsos (CHEN; GUESTRIN, 2016). Este algoritmo tem dominado competições de aprendizado de máquina devido à combinação de desempenho preditivo, velocidade e flexibilidade.

O XGBoost minimiza uma função objetivo que combina explicitamente perda preditiva e complexidade estrutural do modelo, conforme a Equação 5.38

$$\mathcal{L}(\theta) = \sum_{n=1}^N l(y_n, \hat{y}_n) + \sum_{t=1}^T \Omega(f_t) \quad (5.38)$$

onde l é a função de perda diferenciável (*log-loss* para classificação), $\hat{y}_n$ a predição acumulada para a amostra n , e $\Omega(f_t)$ o termo de regularização aplicado a cada árvore t .

O termo de regularização combina penalizações L1 e L2 na estrutura da árvore, conforme a Equação 5.39

$$\Omega(f) = \gamma T_{\text{folhas}} + \frac{1}{2} \lambda \sum_{j=1}^{T_{\text{folhas}}} w_j^2 + \alpha \sum_{j=1}^{T_{\text{folhas}}} |w_j| \quad (5.39)$$

onde T_{folhas} é o número de folhas terminais na árvore, w_j o peso (score de predição) atribuído à folha j , $\gamma \geq 0$ penaliza o número de folhas favorecendo árvores menores, $\lambda \geq 0$ é o coeficiente de regularização L2 sobre os pesos, e $\alpha \geq 0$ é o coeficiente de regularização L1 sobre os pesos. A regularização L2 (λ) produz suavização dos pesos das folhas, enquanto a L1 (α) induz esparsidade, potencialmente zerando pesos de folhas menos relevantes (HASTIE; TIBSHIRANI; FRIEDMAN, 2009).

O XGBoost utiliza expansão de Taylor de segunda ordem da função de perda para determinar divisões ótimas (CHEN; GUESTRIN, 2016). Para adicionar a árvore t na iteração t , aproxima-se a função objetivo conforme a Equação 5.40

$$\mathcal{L}^{(t)} \approx \sum_{n=1}^N \left[g_n f_t(\mathbf{x}_n) + \frac{1}{2} h_n f_t^2(\mathbf{x}_n) \right] + \Omega(f_t) \quad (5.40)$$

onde $g_n = \partial_{\hat{y}_n} l(y_n, \hat{y}_n)$ é o gradiente da perda e $h_n = \partial_{\hat{y}_n}^2 l(y_n, \hat{y}_n)$ é a hessiana, ambos avaliados na predição atual $\hat{y}_n^{(t-1)}$.

O ganho de informação para uma divisão candidata é calculado conforme a Equação 5.41

$$\text{Ganho} = \frac{1}{2} \left[\frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{(G_L + G_R)^2}{H_L + H_R + \lambda} \right] - \gamma \quad (5.41)$$

onde $G_L = \sum_{i \in I_L} g_i$ e $G_R = \sum_{i \in I_R} g_i$ são as somas dos gradientes nos filhos esquerdo e direito, $H_L = \sum_{i \in I_L} h_i$ e $H_R = \sum_{i \in I_R} h_i$ são as somas das hessianas, e I_L e I_R são os conjuntos de índices das amostras nos filhos. Esta formulação permite encontrar divisões ótimas através de algoritmos gulosos, com garantias de que a divisão escolhida minimiza localmente a função objetivo regularizada.

O XGBoost emprega amostragem estocástica em múltiplos níveis para introduzir regularização adicional e acelerar o treinamento: subamostragem de linhas (cada árvore treinada em fração aleatória das amostras), subamostragem de colunas por árvore, por nível e por divisão (CHEN; GUESTRIN, 2016). Estas estratégias criam um *ensemble* mais diverso, reduzindo a correlação entre preditores e melhorando a generalização (DIETTERICH, 2000). O XGBoost também inclui algoritmo especializado para tratamento de características esparsas, aprendendo automaticamente a direção padrão para amostras com valores faltantes.

A otimização de hiperparâmetros explorou o número de árvores entre 100 e 500, taxa de aprendizado η entre 0,01 e 0,3, profundidade máxima entre 3 e 8, peso mínimo na folha entre 1 e 10, parâmetros de regularização γ entre 0 e 1, λ entre 0,1 e 10, e α entre 0 e 5, além de frações de subamostragem de linhas e colunas entre 0,6 e 1,0. Conforme (NIELSEN, 2016), a combinação de regularização explícita e implícita (via subamostragem) confere ao XGBoost robustez superior a outros métodos de *boosting*.

5.8.7 Sistema de Comitê de Modelos

O sistema implementa agregação das predições de múltiplos classificadores mediante votação suave (*soft voting*). Esta técnica combina as probabilidades estimadas pelos modelos individuais por meio de média aritmética, conferindo maior peso implícito às predições com maior confiança (ZHOU, 2012). Conforme (PRABHAKAR; WON, 2024), métodos de *ensemble* para classificação de sinais eletroencefalográficos frequentemente superam classificadores

individuais, beneficiando-se da diversidade de representações aprendidas por diferentes algoritmos.

A probabilidade agregada para cada classe é calculada como a média das probabilidades dos modelos individuais, conforme a Equação 5.42

$$P_{\text{comitê}}(y = c|\mathbf{x}) = \frac{1}{M} \sum_{m=1}^M P_m(y = c|\mathbf{x}) \quad (5.42)$$

onde M representa o número de modelos no comitê, P_m a probabilidade estimada pelo modelo m , c o índice da classe e $\mathbf{x}$ o vetor de características da amostra. A classe predita corresponde àquela com maior probabilidade agregada.

A inclusão de modelos com sobreajuste ou desempenho insuficiente pode degradar a performance do comitê (ZHOU, 2012). O sistema implementa filtragem baseada em dois critérios complementares. O primeiro critério identifica modelos com discrepância excessiva entre desempenho em treinamento e validação, indicativo de sobreajuste. Conforme Ghasemzadeh, Hillman e Mehta (2024), discrepâncias acentuadas entre métricas de treinamento e validação indicam memorização dos dados em detrimento da capacidade de generalização. O critério é expresso pela Equação 5.43

$$\Delta_m = \text{Score}_{\text{treino}}^{(m)} - \text{Score}_{\text{val}}^{(m)} > \delta_{\text{gap}} \quad (5.43)$$

onde Δ_m representa a diferença entre *scores* de treinamento e validação para o modelo m , e δ_{gap} o limiar máximo aceitável, adotado como 0,25 (25%).

O segundo critério exclui modelos com performance inferior a um limiar mínimo, conforme a Equação 5.44

$$\text{Score}_{\text{val}}^{(m)} < \delta_{\text{min}} \quad (5.44)$$

onde δ_{min} representa a performance mínima aceitável medida pelo F1-Score macro, adotado como 0,40 (40%). Este valor supera o nível de chance para três classes equiprováveis (33,33%), garantindo que apenas modelos com capacidade discriminativa mínima componham o comitê.

Caso todos os modelos sejam reprovados pelos critérios estabelecidos, um mecanismo de *fallback* é ativado, selecionando os três modelos com melhor desempenho em validação para

compor um comitê mínimo, conforme a Equação 5.45

$$\mathcal{C}_{\text{fallback}} = \text{Top}_3 \{m : \text{Score}_{\text{val}}^{(m)}\} \quad (5.45)$$

onde $\mathcal{C}_{\text{fallback}}$ representa o conjunto de modelos selecionados e Top_3 a função que retorna os três modelos com maiores valores de $\text{Score}_{\text{val}}^{(m)}$. Esta estratégia garante que o comitê nunca permaneça vazio, preservando os benefícios de *ensemble* mesmo em cenários com modelos de qualidade limitada (ZHOU, 2012).

5.8.8 Estimação de Intervalos de Confiança via Bootstrap

A quantificação da incerteza nas estimativas de desempenho é realizada através de reamostragem *bootstrap* não paramétrica (EFRON; TIBSHIRANI, 1994). Conforme Mokhtar, El-Rafaie e Mahmoud (2023), o método *bootstrap* percentílico constitui abordagem adequada para estimação de intervalos de confiança de métricas de classificação, não requerendo suposições sobre a distribuição das estatísticas.

Segundo DiCiccio e Efron (1996), ao menos 2000 repetições são necessárias para estimação de intervalos de confiança com precisão adequada. Efron e Tibshirani (1994) recomendam entre 1000 e 2000 reamostras para esta finalidade, enquanto Banjanovic e Osborne (2016) sugerem valores entre 4000 e 5000 para maior estabilidade. O valor de 2000 representa compromisso entre precisão estatística e custo computacional.

Para cada reamostragem $b = 1, \dots, B$, gera-se amostra *bootstrap* através de amostragem com reposição do conjunto de teste, e calcula-se a métrica de interesse $\theta^{(b)}$ na amostra *bootstrap*. O intervalo de confiança de $(1 - \alpha) \times 100\%$ é obtido pelos percentis da distribuição *bootstrap*, conforme a Equação 5.46

$$\text{IC}_{1-\alpha} = [\theta_{(\alpha/2)}, \theta_{(1-\alpha/2)}] \quad (5.46)$$

onde α representa o nível de significância, $\theta_{(\alpha/2)}$ o percentil $(\alpha/2) \times 100$ da distribuição *bootstrap* e $\theta_{(1-\alpha/2)}$ o percentil $(1 - \alpha/2) \times 100$. O valor adotado de $\alpha = 0,05$ resulta nos percentis 2,5% e 97,5% para intervalo de confiança de 95%, conforme prática estabelecida por Davison e Hinkley (1997).

Esta abordagem não paramétrica não assume normalidade das métricas e captura a variabilidade das estimativas de desempenho decorrente da amostragem. Conforme Raschka (2022), o método percentílico aplicado ao conjunto de teste constitui abordagem adequada para estimação de intervalos de confiança de métricas de classificação, quando o modelo permanece fixo e apenas as predições são reamostradas. A implementação utiliza a biblioteca scikit-learn (PEDREGOSA et al., 2011) para os classificadores e a biblioteca XGBoost (CHEN; GUESTRIN, 2016) para o algoritmo homônimo.

5.9 Otimização de Hiperparâmetros

A busca aleatória amostra configurações de hiperparâmetros de distribuições especificadas, em contraste com a busca em grade que avalia todas as combinações de um conjunto discreto de valores. Estudos empíricos demonstram que, para um orçamento computacional fixo, a busca aleatória frequentemente encontra configurações superiores, quando alguns hiperparâmetros têm impacto muito maior que outros (BERGSTRA; BENGIO, 2012). Isto ocorre porque a busca aleatória explora mais valores distintos para cada hiperparâmetro, enquanto a busca em grade pode desperdiçar avaliações em regiões pouco promissoras.

O procedimento é configurado com 100 iterações de amostragem, cada uma representando uma configuração distinta de hiperparâmetros. Para cada configuração θ amostrada, o desempenho é avaliado através de validação cruzada estratificada com 5 partições, conforme a Equação 5.47.

$$\text{Score}(\theta) = \frac{1}{K} \sum_{k=1}^K F1_{\text{macro},k}(\theta) \quad (5.47)$$

Na Equação 5.47, $\text{Score}(\theta)$ representa o desempenho médio da configuração de hiperparâmetros θ , K é o número de partições da validação cruzada, e $F1_{\text{macro},k}(\theta)$ denota o F1-Score macro obtido na partição k quando o modelo é treinado com a configuração θ . O F1-Score macro é calculado como a média aritmética simples dos F1-Scores de cada classe, conferindo peso igual a todas as classes independentemente de seus tamanhos.

A utilização do F1-Score macro como métrica de otimização é apropriada para problemas multiclasse com possível desbalanceamento entre classes. Esta métrica equilibra precisão e revocação para cada classe individualmente antes de calcular a média, evitando que o desempe-

nho em classes majoritárias domine a avaliação. Em contraste, métricas como acurácia global podem ser enganosas quando as classes têm tamanhos muito diferentes.

A configuração ótima de hiperparâmetros é selecionada conforme a Equação 5.48.

$$\theta^* = \arg \max_{\theta \in \Theta} \text{Score}(\theta) \quad (5.48)$$

Na Equação 5.48, θ^* representa a configuração ótima de hiperparâmetros, Θ denota o conjunto de todas as configurações amostradas durante a busca (com $|\Theta| = 100$ neste caso), e $\arg \max$ identifica a configuração que maximiza o score de validação cruzada. Após a identificação da configuração ótima, o modelo final é retreinado utilizando θ^* e todo o conjunto de treinamento disponível.

Tabela 5.2 – Configuração do procedimento de otimização de hiperparâmetros através de busca aleatória com validação cruzada. Os valores foram estabelecidos buscando equilíbrio entre capacidade de exploração do espaço de hiperparâmetros e custo computacional.

Parâmetro	Valor	Descrição
Iterações	100	Número de configurações amostradas aleatoriamente
Folds CV	5	Número de partições na validação cruzada estratificada
Métrica	F1-macro	Critério de otimização (média harmônica balanceada)
Semente	42	Valor fixo para garantir reprodutibilidade dos resultados

A fixação de uma semente aleatória (42) garante reprodutibilidade completa dos experimentos. Dada a mesma semente, o mesmo conjunto de configurações será amostrado e avaliado na mesma ordem, permitindo comparações consistentes entre diferentes execuções e facilitando a depuração de problemas.

A escolha de 100 iterações representa um compromisso pragmático entre exploração do espaço de hiperparâmetros e custo computacional. Para espaços de hiperparâmetros de dimensionalidade moderada (tipicamente 3 a 6 hiperparâmetros), este número de iterações permite razoável cobertura de regiões promissoras sem incorrer em tempo computacional proibitivo. A utilização de 5 folds na validação cruzada oferece estimativas relativamente estáveis de desempenho com variância controlada, sendo um padrão amplamente adotado na literatura de aprendizado de máquina.

5.9.1 Busca Aleatória com Validação Cruzada

O sistema emprega busca aleatória (*random search*) em substituição à busca em grade (*grid search*), que examina todas as combinações possíveis de forma exaustiva. No estudo de Lawhern et al. (2018), a busca aleatória foi utilizada para otimizar arquiteturas de redes neurais convolucionais para EEG, demonstrando convergência para configurações de alto desempenho com número reduzido de avaliações.

Seja $\mathcal{H}$ o espaço de hiperparâmetros e $f(\mathbf{h})$ o desempenho de validação para uma configuração $\mathbf{h} \in \mathcal{H}$. A probabilidade de encontrar uma configuração com desempenho dentro de uma fração ε do ótimo após n tentativas aleatórias é expressa pela Equação 5.49

$$P\left(\max_{i=1}^n f(\mathbf{h}_i) \geq (1 - \varepsilon) \max_{\mathbf{h} \in \mathcal{H}} f(\mathbf{h})\right) = 1 - (1 - p)^n \quad (5.49)$$

onde p representa a proporção do espaço de hiperparâmetros que se encontra dentro de ε do valor ótimo. A Equação 5.49 demonstra que a probabilidade de sucesso cresce exponencialmente com o número de tentativas n , garantindo alta probabilidade de identificar configurações satisfatórias com orçamento computacional moderado.

Para cada hiperparâmetro, define-se uma distribuição apropriada à sua natureza e escala de variação. A distribuição log-uniforme é empregada para parâmetros que variam em ordens de magnitude, como taxas de aprendizado ou coeficientes de regularização, conforme a Equação 5.50

$$\log h \sim \mathcal{U}(\log a, \log b) \implies h \sim \text{LogUniform}(a, b) \quad (5.50)$$

onde a e b definem os limites inferior e superior do intervalo. Esta distribuição aloca probabilidade igual a cada ordem de magnitude, evitando concentração em valores extremos. A distribuição uniforme, expressa na Equação 5.51, aplica-se a parâmetros com efeito aproximadamente linear no desempenho

$$h \sim \mathcal{U}(a, b) \quad (5.51)$$

sendo apropriada para proporções, probabilidades e parâmetros com escala limitada. Para hiperparâmetros discretos, como número de árvores em florestas aleatórias ou dimensões de camadas

em redes neurais, utiliza-se a distribuição inteira conforme a Equação 5.52

$$h \sim \text{RandInt}(a, b) \quad (5.52)$$

Cada configuração amostrada é avaliada através de validação cruzada estratificada com K partições. O conjunto de treinamento é dividido em K subconjuntos mutuamente exclusivos, mantendo as proporções originais das classes em cada partição, conforme a Equação 5.53

$$\mathcal{D}_{\text{treino}} = \bigcup_{k=1}^K \mathcal{D}_k, \quad \mathcal{D}_i \cap \mathcal{D}_j = \emptyset \text{ para } i \neq j \quad (5.53)$$

Para cada iteração k , o modelo é treinado em $K - 1$ partições e validado na partição restante, permitindo calcular o desempenho conforme a Equação 5.54

$$S_k(\mathbf{h}) = \text{Desempenho} \left(\text{Modelo}_{\mathbf{h}} \text{ treinado em } \bigcup_{j \neq k} \mathcal{D}_j, \text{ testado em } \mathcal{D}_k \right) \quad (5.54)$$

O desempenho médio e desvio padrão através das partições fornecem estimativas da performance esperada e sua variabilidade, expressas na Equação 5.55

$$\bar{S}(\mathbf{h}) = \frac{1}{K} \sum_{k=1}^K S_k(\mathbf{h}), \quad \sigma_S(\mathbf{h}) = \sqrt{\frac{1}{K-1} \sum_{k=1}^K (S_k(\mathbf{h}) - \bar{S}(\mathbf{h}))^2} \quad (5.55)$$

onde $\bar{S}(\mathbf{h})$ representa a média dos desempenhos nas K partições e $\sigma_S(\mathbf{h})$ quantifica a variabilidade dessas estimativas. A estratificação é relevante em problemas de classificação para garantir que classes minoritárias estejam adequadamente representadas em cada partição, evitando partições sem amostras de determinadas classes.

O sistema foi configurado com os seguintes parâmetros. O número de iterações foi estabelecido em 100 configurações aleatórias por modelo, balanceando exploração do espaço de hiperparâmetros com custo computacional. O número de partições foi definido como $K = 5$, valor que proporciona compromisso entre viés e variância da estimativa de desempenho. A métrica de otimização selecionada foi o F1-score macro, apropriada para problemas multiclasse potencialmente desbalanceados. A busca emprega computação paralela, avaliando múltiplas configurações simultaneamente para acelerar o processo em sistemas multi-core.

Após avaliar todas as n configurações amostradas, seleciona-se aquela com melhor desempenho médio de validação cruzada, conforme a Equação 5.56

$$\mathbf{h}^* = \arg \max_{\mathbf{h} \in \{\mathbf{h}_1, \dots, \mathbf{h}_n\}} \bar{S}(\mathbf{h}) \quad (5.56)$$

A configuração ótima $\mathbf{h}^*$ é então retreinada no conjunto de treinamento completo antes da avaliação final no conjunto de teste independente, maximizando o uso dos dados disponíveis enquanto mantém avaliação honesta do desempenho.

5.10 Arquitetura Plana versus Arquitetura em Cascata

Este trabalho investiga duas abordagens arquiteturais para classificação de sinais EEG de imaginação motora: a arquitetura plana e a arquitetura hierárquica em cascata. A comparação entre estas abordagens permite avaliar os *trade-offs* entre simplicidade e desempenho.

5.10.1 Arquitetura Plana

A arquitetura plana constitui a abordagem tradicional para problemas multiclasse, na qual um único classificador é treinado para discriminar simultaneamente entre todas as classes de interesse. O modelo aprende fronteiras de decisão que separam diretamente as três condições experimentais: Repouso, Movimento da Mão Esquerda e Movimento da Mão Direita.

Para o problema de classificação com $C = 3$ classes, a arquitetura plana utiliza um classificador $f : \mathbb{R}^d \rightarrow \{0, 1, 2\}$, onde d representa a dimensionalidade do espaço de características. A predição é obtida diretamente através da Equação 5.57

$$\hat{y} = f(\mathbf{x}) = \arg \max_{c \in \{0, 1, 2\}} P(y = c | \mathbf{x}), \quad (5.57)$$

onde $\mathbf{x} \in \mathbb{R}^d$ representa o vetor de características extraídas do sinal EEG e $P(y = c | \mathbf{x})$ denota a probabilidade estimada da classe c dado o vetor de entrada.

Esta abordagem apresenta como vantagens a simplicidade de implementação e a ausência de propagação de erros entre etapas sequenciais. Entretanto, pode apresentar limitações em problemas com estrutura hierárquica natural, onde diferentes classes compartilham características em níveis distintos de abstração (JR.; FREITAS, 2011).

5.10.2 Arquitetura Hierárquica em Cascata

A arquitetura hierárquica em cascata fundamenta-se no princípio de decomposição de problemas complexos em subproblemas mais simples e especializados (JR.; FREITAS, 2011). Esta abordagem é apropriada para BCI de imaginação motora, onde a discriminação entre estados de repouso e movimento pode ser considerada conceitualmente distinta da discriminação entre diferentes tipos de movimento (ALTAHERI et al., 2023).

A classificação em cascata decompõe o problema multiclasse original em duas etapas binárias sequenciais. A primeira etapa (E1) realiza classificação binária entre estados de repouso cerebral e estados de imaginação motora, independentemente da lateralidade. O mapeamento de rótulos para esta etapa é definido pela Equação 5.58

$$y_{E1} = \begin{cases} 0, & \text{se } y = 0 \text{ (Repouso)} \\ 1, & \text{se } y \in \{1, 2\} \text{ (Movimento)} \end{cases}, \quad (5.58)$$

onde y representa o rótulo original multiclasse (0: Repouso, 1: Esquerda, 2: Direita) e y_{E1} o rótulo binário resultante. Esta etapa concentra-se na detecção de ativação motora através de marcadores neurofisiológicos como a dessincronização relacionada a eventos (ERD) nas bandas mu (8–13 Hz) e beta (13–30 Hz).

A segunda etapa (E2) opera exclusivamente sobre amostras classificadas como Movimento em E1, discriminando entre imaginação motora da mão esquerda e direita. O conjunto de dados para E2 é filtrado conforme a Equação 5.59

$$\mathcal{D}_{E2} = \{(\mathbf{x}_i, y_i) \in \mathcal{D} : y_i \in \{1, 2\}\}. \quad (5.59)$$

O mapeamento de rótulos para E2 é expresso pela Equação 5.60

$$y_{E2} = \begin{cases} 0, & \text{se } y = 1 \text{ (Esquerda)} \\ 1, & \text{se } y = 2 \text{ (Direita)} \end{cases}. \quad (5.60)$$

Esta etapa explora diferenças de lateralização cortical, manifestadas através de padrões assimétricos de ERD/ERS entre hemisférios cerebrais (NEUPER; PFURTSCHELLER, 2006). A predição final na arquitetura em cascata integra as decisões de ambas as etapas através do pro-

cesso hierárquico definido pela Equação 5.61

$$\hat{y} = \begin{cases} 0, & \text{se } \hat{y}_{E1} = 0 \\ 1, & \text{se } \hat{y}_{E1} = 1 \text{ e } \hat{y}_{E2} = 0, \\ 2, & \text{se } \hat{y}_{E1} = 1 \text{ e } \hat{y}_{E2} = 1 \end{cases} \quad (5.61)$$

onde $\hat{y}$ representa a predição final multiclasse, $\hat{y}_{E1}$ a predição da primeira etapa e $\hat{y}_{E2}$ a predição da segunda etapa. A etapa E2 é executada condicionalmente, apenas quando E1 prediz movimento ($\hat{y}_{E1} = 1$).

A reconstrução das probabilidades finais segue a regra da probabilidade total, preservando a estrutura condicional da cascata. As probabilidades são calculadas conforme a Equação 5.62

$$\begin{aligned} P(\hat{y} = 0|\mathbf{x}) &= P_{E1}(\text{Repouso}|\mathbf{x}) \\ P(\hat{y} = 1|\mathbf{x}) &= P_{E1}(\text{Movimento}|\mathbf{x}) \times P_{E2}(\text{Esquerda}|\text{Movimento}, \mathbf{x}), \\ P(\hat{y} = 2|\mathbf{x}) &= P_{E1}(\text{Movimento}|\mathbf{x}) \times P_{E2}(\text{Direita}|\text{Movimento}, \mathbf{x}) \end{aligned} \quad (5.62)$$

onde $P_{E1}(\cdot|\mathbf{x})$ e $P_{E2}(\cdot|\text{Movimento}, \mathbf{x})$ representam as distribuições de probabilidade estimadas pelos classificadores da primeira e segunda etapas, respectivamente. Esta formulação garante que as probabilidades finais satisfaçam a propriedade de normalização expressa pela Equação 5.63

$$\sum_{c=0}^2 P(\hat{y} = c|\mathbf{x}) = 1. \quad (5.63)$$

Para amostras classificadas como Repouso em E1, onde a segunda etapa não é executada, as probabilidades de Esquerda e Direita são distribuídas uniformemente como $P_{E1}(\text{Movimento}|\mathbf{x})/2$ cada, assegurando a validade probabilística.

Uma característica da arquitetura em cascata é a possibilidade de especialização independente em cada etapa. Formalmente, seja $\phi_{E1} : \mathbb{R}^{d_0} \rightarrow \mathbb{R}^{d_1}$ a função de seleção de características para E1 e $\phi_{E2} : \mathbb{R}^{d_0} \rightarrow \mathbb{R}^{d_2}$ para E2, onde d_0 é a dimensionalidade original e d_1, d_2 podem diferir. Cada etapa possui processo independente de seleção de características, permitindo que E1 e E2 utilizem subconjuntos potencialmente distintos do espaço de características original. Além disso, a otimização independente de hiperparâmetros permite adaptar algoritmos e parâmetros especificamente para cada subproblema binário, bem como estratégias de *threshold*

customizadas possibilitam ajuste fino do ponto de decisão para otimizar métricas específicas de cada etapa.

5.10.3 Análise Comparativa

Subproblemas binários tendem a apresentar maior separabilidade linear que o problema multiclasse original, potencialmente resultando em fronteiras de decisão mais robustas. Diferentes características neurais podem ser mais discriminativas para detecção de movimento (E1) versus discriminação de lateralidade (E2), permitindo especialização de características. A estrutura hierárquica reflete o conhecimento neurofisiológico de que detecção de ativação motora e discriminação de lateralidade são processos neurais parcialmente distintos, conferindo maior interpretabilidade ao modelo. Em cenários de produção, a segunda etapa só precisa ser executada para amostras classificadas como movimento, reduzindo o custo computacional médio.

A arquitetura em cascata também apresenta desafios inerentes. Erros cometidos em E1 são irrecuperáveis, propagando-se para a saída final; se E1 classificar incorretamente uma amostra de movimento como repouso, E2 nunca terá oportunidade de corrigi-la. O conjunto de treinamento para E2 é naturalmente menor, contendo apenas amostras de movimento, o que pode afetar a capacidade de generalização. A presença de duas etapas multiplica o espaço de decisões de hiperparâmetros e requer estratégias coordenadas de otimização. A reconstrução correta de probabilidades requer que ambas as etapas forneçam estimativas bem calibradas, condição que nem sempre é garantida.

5.11 Métricas de Avaliação e Comparação Estatística

Esta seção apresenta metodologia para avaliação de desempenho dos classificadores desenvolvidos, organizando-se em três eixos complementares: métricas de classificação derivadas da matriz de confusão, métricas probabilísticas e de calibração, e procedimentos de comparação estatística entre técnicas. A fundamentação matemática de cada métrica é apresentada conjuntamente com sua interpretação prática.

5.11.1 Métricas Derivadas da Matriz de Confusão

Para um problema com C classes distintas, a matriz de confusão corresponde a uma estrutura quadrada $C \times C$ onde cada elemento M_{ij} quantifica o número de amostras verdadeiramente pertencentes à classe i que foram classificadas como pertencentes à classe j . A estrutura geral é expressa pela Equação 5.64

$$\mathbf{M} = \begin{bmatrix} M_{11} & M_{12} & \cdots & M_{1C} \\ M_{21} & M_{22} & \cdots & M_{2C} \\ \vdots & \vdots & \ddots & \vdots \\ M_{C1} & M_{C2} & \cdots & M_{CC} \end{bmatrix} \quad (5.64)$$

onde os elementos da diagonal principal M_{ii} indicam classificações corretas, enquanto elementos fora da diagonal M_{ij} ($i \neq j$) representam erros específicos de classificação, revelando quais classes são sistematicamente confundidas entre si. A soma dos elementos de cada linha corresponde ao suporte da classe n_i , conforme a Equação 5.65

$$n_i = \sum_{j=1}^C M_{ij} \quad (5.65)$$

representando o número total de amostras verdadeiramente pertencentes à classe i no conjunto de dados avaliado.

A acurácia global representa a métrica mais intuitiva de desempenho, correspondendo à proporção de predições corretas sobre o total de amostras avaliadas (PEDREGOSA et al., 2011). Esta métrica é expressa pela Equação 5.66

$$\text{Acurácia} = \frac{\sum_{i=1}^C M_{ii}}{\sum_{i=1}^C \sum_{j=1}^C M_{ij}} = \frac{\sum_{i=1}^C M_{ii}}{N} \quad (5.66)$$

onde o numerador corresponde à soma dos elementos da diagonal principal e N representa o número total de amostras. Esta métrica varia no intervalo $[0, 1]$, onde 1 indica classificação perfeita e $1/C$ corresponde ao desempenho esperado de um classificador aleatório uniforme. Apesar de sua ampla utilização e fácil interpretabilidade, a acurácia apresenta limitação em conjuntos desbalanceados. Conforme demonstrado por Brodersen et al. (2010), em um problema binário onde 95% das amostras pertencem à classe majoritária, um classificador trivial

que sempre prediz esta classe alcança 95% de acurácia sem demonstrar qualquer capacidade discriminativa genuína, tornando a métrica potencialmente enganosa para avaliação.

Para mitigar a sensibilidade ao desbalanceamento, a acurácia balanceada computa a média aritmética das taxas de acerto individuais de cada classe, conforme a Equação 5.67

$$\text{Acurácia Balanceada} = \frac{1}{C} \sum_{i=1}^C \frac{M_{ii}}{n_i} \quad (5.67)$$

onde o termo M_{ii}/n_i representa a sensibilidade (*recall*) da classe i , definida como a proporção de amostras corretamente identificadas dentre todas as amostras verdadeiras daquela classe. Esta formulação atribui peso igual a cada classe independentemente de seu suporte, sendo adequada quando o desempenho em classes minoritárias possui igual relevância prática (BRODERSEN et al., 2010). Para classificação aleatória, a acurácia balanceada esperada é $1/C$ independentemente da distribuição de classes, correspondendo a aproximadamente 33,3% no problema triclasse ($C = 3$) abordado neste trabalho.

Para análise detalhada por classe, definem-se três categorias fundamentais de contagem. Os verdadeiros positivos (VP_c) correspondem às amostras da classe c corretamente classificadas, equivalendo a M_{cc} . Os falsos positivos (FP_c), expressos pela Equação 5.68, representam amostras de outras classes incorretamente atribuídas à classe c

$$FP_c = \sum_{i=1}^C M_{ic} - M_{cc} = \sum_{i \neq c} M_{ic} \quad (5.68)$$

enquanto os falsos negativos (FN_c), definidos pela Equação 5.69, representam amostras da classe c incorretamente classificadas como pertencentes a outras classes

$$FN_c = \sum_{j=1}^C M_{cj} - M_{cc} = \sum_{j \neq c} M_{cj} \quad (5.69)$$

A precisão para a classe c quantifica a proporção de predições positivas que são corretas, sendo expressa pela Equação 5.70

$$\text{Precisão}_c = \frac{VP_c}{VP_c + FP_c} = \frac{M_{cc}}{\sum_{i=1}^C M_{ic}} \quad (5.70)$$

enquanto a sensibilidade (*recall*) quantifica a proporção de amostras da classe corretamente identificadas, conforme a Equação 5.71

$$\text{Sensibilidade}_c = \frac{VP_c}{VP_c + FN_c} = \frac{M_{cc}}{\sum_{j=1}^C M_{cj}} = \frac{M_{cc}}{n_c} \quad (5.71)$$

Ambas as métricas variam no intervalo $[0, 1]$, onde valores próximos a 1 indicam ausência de falsos positivos (precisão perfeita) ou falsos negativos (sensibilidade perfeita), respectivamente. Em sistemas de comunicação assistiva, alta sensibilidade para detecção de intenção de movimento é prioritária para minimizar falsos negativos que frustrariam o usuário, enquanto em sistemas de controle de próteses, alta precisão pode ser preferível para evitar movimentos não intencionados.

O F1-score combina precisão e sensibilidade através da média harmônica, penalizando valores extremadamente desbalanceados entre estas métricas, conforme a Equação 5.72

$$F1_c = 2 \cdot \frac{\text{Precisão}_c \cdot \text{Sensibilidade}_c}{\text{Precisão}_c + \text{Sensibilidade}_c} = \frac{2 \cdot VP_c}{2 \cdot VP_c + FP_c + FN_c} \quad (5.72)$$

A escolha da média harmônica, em detrimento da média aritmética, garante que o F1-score atinge seu valor máximo apenas quando precisão e sensibilidade são simultaneamente elevadas. Um classificador com precisão de 100% mas sensibilidade de 10% obteria média aritmética de 55%, porém F1-score de apenas 18,2%, refletindo mais adequadamente seu desempenho desequilibrado (SASAKI, 2007).

Para sintetizar métricas calculadas individualmente por classe em um único valor escalar, três estratégias de agregação são empregadas (PEDREGOSA et al., 2011; SOKOLOVA; LAPALME, 2009). A macro-média computa a média aritmética simples das métricas individuais de cada classe, conforme a Equação 5.73

$$\text{Métrica}_{\text{macro}} = \frac{1}{C} \sum_{c=1}^C \text{Métrica}_c \quad (5.73)$$

atribuindo peso igual a todas as classes independentemente de seu tamanho. O F1-score macro, adotado como critério primário de otimização neste trabalho, é calculado pela Equação 5.74

$$F1_{\text{macro}} = \frac{1}{C} \sum_{c=1}^C F1_c \quad (5.74)$$

A micro-média agrega todas as contagens de verdadeiros positivos, falsos positivos e falsos negativos antes de calcular a métrica final. A precisão micro é expressa pela Equação 5.75

$$\text{Precisão}_{\text{micro}} = \frac{\sum_{c=1}^C VP_c}{\sum_{c=1}^C (VP_c + FP_c)} \quad (5.75)$$

e a sensibilidade micro pela Equação 5.76

$$\text{Sensibilidade}_{\text{micro}} = \frac{\sum_{c=1}^C VP_c}{\sum_{c=1}^C (VP_c + FN_c)} \quad (5.76)$$

Para problemas multiclasse com classes mutuamente exclusivas, precisão micro, sensibilidade micro e F1-score micro são numericamente equivalentes à acurácia global, atribuindo maior peso implícito a classes com maior número de amostras. A média ponderada pondera explicitamente a contribuição de cada classe pelo seu suporte, conforme a Equação 5.77

$$\text{Métrica}_{\text{ponderada}} = \frac{\sum_{c=1}^C n_c \cdot \text{Métrica}_c}{\sum_{c=1}^C n_c} = \frac{\sum_{c=1}^C n_c \cdot \text{Métrica}_c}{N} \quad (5.77)$$

reconhecendo que classes com maior número de amostras contribuem proporcionalmente mais para o desempenho geral percebido pelo usuário final do sistema. Neste trabalho, o F1-score macro foi adotado como critério primário de otimização de hiperparâmetros, garantindo desempenho equilibrado entre todas as classes de estados mentais independentemente de sua frequência no conjunto de dados, alinhando-se às recomendações de Grandini, Bagli e Visani (2020) para cenários onde todas as classes possuem igual relevância funcional.

O Coeficiente de Correlação de Matthews (MCC) representa uma métrica balanceada que incorpora todos os elementos da matriz de confusão. Para classificação binária, o MCC é definido pela Equação 5.78

$$MCC = \frac{VP \times VN - FP \times FN}{\sqrt{(VP + FP)(VP + FN)(VN + FP)(VN + FN)}} \quad (5.78)$$

onde VP, VN, FP e FN representam verdadeiros positivos, verdadeiros negativos, falsos positivos e falsos negativos, respectivamente. Para classificação multiclasse, o MCC generaliza-se como o coeficiente de correlação de Pearson entre as classes verdadeiras e preditas, tratadas como variáveis categóricas codificadas numericamente (GORODKIN, 2004). O MCC varia no intervalo $[-1, +1]$, onde $MCC = +1$ indica predição perfeita, $MCC = 0$ indica desempenho equivalente ao aleatório, e $MCC = -1$ indica desacordo total. Conforme demonstrado por Chicco e Jurman (2020), o MCC apresenta vantagem sobre F1-score e acurácia por considerar simultaneamente todas as células da matriz de confusão.

5.11.2 Métricas Probabilísticas e Calibração

O escore de Brier quantifica a acurácia das probabilidades preditas, comparando-as diretamente com os resultados observados através do erro quadrático médio (BRIER, 1950). Para classificação multiclasse, é definido pela Equação 5.79

$$\text{Brier} = \frac{1}{N} \sum_{n=1}^N \sum_{c=1}^C (y_{n,c} - p_{n,c})^2 \quad (5.79)$$

onde N representa o número de amostras, C o número de classes, $y_{n,c} \in \{0, 1\}$ é a codificação *one-hot* da classe verdadeira (assumindo valor 1 se a amostra n pertence à classe c e 0 caso contrário), e $p_{n,c}$ é a probabilidade predita pelo modelo para a classe c . O escore de Brier varia teoricamente em $[0, 2]$ para problemas multiclasse, com valores menores indicando melhor qualidade das estimativas probabilísticas. Um classificador perfeitamente calibrado que prediz $p_{n,c} = 1$ para a classe correta e $p_{n,c} = 0$ para as demais alcança $\text{Brier} = 0$, enquanto um classificador que sempre prediz probabilidades uniformes ($p_{n,c} = 1/C$) obtém escore igual a $(C - 1)/C$.

A decomposição proposta por Murphy (1973) permite análise detalhada das fontes de erro, conforme a Equação 5.80

$$\text{Brier} = \underbrace{\text{UNC}}_{\text{Incerteza}} - \underbrace{\text{RES}}_{\text{Resolução}} + \underbrace{\text{REL}}_{\text{Confiabilidade}} \quad (5.80)$$

onde o componente de incerteza (UNC) depende apenas das frequências marginais das classes e representa a variabilidade intrínseca do problema, a resolução (RES) mede a capacidade do

modelo de diferenciar entre grupos com diferentes taxas de ocorrência, e a confiabilidade (REL) quantifica o desvio entre probabilidades preditas e frequências observadas.

A curva ROC (*Receiver Operating Characteristic*) visualiza o compromisso entre taxa de verdadeiros positivos (sensibilidade) e taxa de falsos positivos em diferentes limiares de decisão (FAWCETT, 2006). Para um dado limiar τ , as taxas são calculadas conforme a Equação 5.81

$$\text{TPR}(\tau) = \frac{\text{VP}(\tau)}{\text{VP}(\tau) + \text{FN}(\tau)}, \quad \text{FPR}(\tau) = \frac{\text{FP}(\tau)}{\text{FP}(\tau) + \text{VN}(\tau)} \quad (5.81)$$

A área sob a curva ROC (AUC-ROC) fornece uma medida agregada de desempenho através de todos os limiares possíveis, calculada pela integração expressa na Equação 5.82

$$\text{AUC} = \int_0^1 \text{TPR}(\text{FPR}^{-1}(x)) dx \quad (5.82)$$

A AUC pode ser interpretada probabilisticamente como a probabilidade de que o classificador atribua pontuação mais alta a uma amostra positiva aleatoriamente selecionada do que a uma amostra negativa aleatória (HAND; TILL, 2001), conforme a Equação 5.83

$$\text{AUC} = P(\text{score}(x_+) > \text{score}(x_-)) \quad (5.83)$$

onde $\text{AUC} = 1,0$ indica separação perfeita entre classes, $\text{AUC} = 0,5$ indica desempenho equivalente ao aleatório, e $\text{AUC} < 0,5$ indica previsões sistematicamente invertidas. Para extensão multiclasse, este trabalho adota a estratégia *One-vs-Rest* (OvR), calculando a AUC para cada classe c versus todas as demais classes agrupadas, seguida de agregação via macro-média expressa pela Equação 5.84

$$\text{AUC}_{\text{macro}} = \frac{1}{C} \sum_{c=1}^C \text{AUC}_c \quad (5.84)$$

Em cenários com desbalanceamento pronunciado entre classes, a curva Precisão-Recall apresenta maior informatividade que a curva ROC tradicional, conforme demonstrado por Saito e Rehmsmeier (2015). Esta curva visualiza o compromisso entre precisão e sensibilidade em diferentes limiares de decisão, com as métricas calculadas pela Equação 5.85

$$\text{Precisão}(\tau) = \frac{\text{VP}(\tau)}{\text{VP}(\tau) + \text{FP}(\tau)}, \quad \text{Recall}(\tau) = \frac{\text{VP}(\tau)}{\text{VP}(\tau) + \text{FN}(\tau)} \quad (5.85)$$

A área sob a curva Precisão-Recall (AUC-PR) sumariza o desempenho através da integração expressa na Equação 5.86

$$\text{AUC-PR} = \int_0^1 \text{Precisão}(\text{Recall}^{-1}(r)) dr \quad (5.86)$$

Diferentemente da AUC-ROC, cujo valor de referência aleatório é sempre 0,5, o valor de referência para AUC-PR corresponde à prevalência da classe positiva no conjunto de dados. Para uma classe com prevalência p , um classificador aleatório alcança $\text{AUC-PR} \approx p$, tornando esta métrica mais sensível a melhorias de desempenho em cenários desbalanceados.

As probabilidades produzidas por classificadores frequentemente não refletem adequadamente as probabilidades verdadeiras das classes (NICULESCU-MIZIL; CARUANA, 2005; GUO et al., 2017). Um modelo bem calibrado deve satisfazer a propriedade de que, dentre todas as amostras às quais atribuiu probabilidade de 70% para uma determinada classe, aproximadamente 70% devem efetivamente pertencer a essa classe. A calibração é relevante para sistemas BCI onde decisões baseadas em níveis de confiança são necessárias.

O método de Platt ajusta uma função sigmoide às saídas brutas do classificador (PLATT, 1999), conforme a Equação 5.87

$$P_{\text{cal}}(y = c|\mathbf{x}) = \frac{1}{1 + \exp(A \cdot s_c(\mathbf{x}) + B)} \quad (5.87)$$

onde $s_c(\mathbf{x})$ representa o score bruto para a classe c , $\mathbf{x}$ é o vetor de características, e os parâmetros A e B são estimados por máxima verossimilhança em um conjunto de validação separado. A regressão isotônica oferece abordagem não-paramétrica mais flexível (ZADROZNY; ELKAN, 2002), expressa pela Equação 5.88

$$P_{\text{cal}}(y = c|\mathbf{x}) = f_{\text{iso}}(s_c(\mathbf{x})) \quad (5.88)$$

onde f_{iso} é uma função monotônica não-decrescente obtida pelo algoritmo PAV (*Pool Adjacent Violators*), que resolve o problema de otimização expresso pela Equação 5.89

$$\min_{f_{\text{iso}}} \sum_{n=1}^N (y_n - f_{\text{iso}}(s_n))^2 \quad \text{sujeito a } f_{\text{iso}}(s_i) \leq f_{\text{iso}}(s_j) \text{ se } s_i < s_j \quad (5.89)$$

Neste trabalho, a escolha entre os métodos é realizada automaticamente com base no tamanho do conjunto de calibração, utilizando regressão isotônica quando o número de amostras excede 1000 e método de Platt para amostras menores devido à sua maior estabilidade com dados limitados (NICULESCU-MIZIL; CARUANA, 2005).

O *Expected Calibration Error* (ECE) quantifica o desvio médio entre probabilidades preditas e frequências observadas (NAEINI; COOPER; HAUSKRECHT, 2015). As probabilidades são discretizadas em B intervalos, e para cada intervalo b calculam-se a confiança média ($\bar{p}_b$) e a acurácia observada ($\bar{y}_b$), conforme a Equação 5.90

$$\bar{p}_b = \frac{1}{|I_b|} \sum_{n \in I_b} \hat{p}_n, \quad \bar{y}_b = \frac{1}{|I_b|} \sum_{n \in I_b} y_n \quad (5.90)$$

onde $I_b = \{n : \hat{p}_n \in [(b-1)/B, b/B]\}$ denota o conjunto de amostras cujas probabilidades pertencem ao intervalo b . O ECE é então calculado como a média ponderada dos desvios absolutos, conforme a Equação 5.91

$$\text{ECE} = \sum_{b=1}^B \frac{|I_b|}{N} |\bar{p}_b - \bar{y}_b| \quad (5.91)$$

onde valores menores indicam melhor calibração e um modelo perfeitamente calibrado satisfaz $\bar{p}_b = \bar{y}_b$ para todo intervalo, resultando em $\text{ECE} = 0$.

A Tab. 5.3 apresenta uma síntese das métricas de avaliação empregadas neste trabalho, incluindo seus intervalos de valores, valores de referência para classificação aleatória, interpretações práticas, características de sensibilidade ao desbalanceamento e os contextos de aplicação recomendados para cada métrica.

5.11.3 Comparação Estatística entre Técnicas

A comparação de desempenho entre múltiplas técnicas de balanceamento aplicadas ao mesmo conjunto de dados constitui um cenário de medidas repetidas, no qual os algoritmos avaliados compartilham as mesmas partições de treinamento e teste (DEMSAR, 2006). Este delineamento viola o pressuposto de independência entre observações requerido por testes paramétricos convencionais como ANOVA, demandando procedimentos estatísticos não-paramétricos para inferências válidas.

Tabela 5.3 – Síntese das métricas de avaliação utilizadas no pipeline de classificação BCI.

Métrica	Melhor	Pior	Aleatório	Aspecto Avaliado	Sens. Desbal.	Contexto de Aplicação
Acurácia	1	0	$1/C$	Proporção global de acertos	Sim	Conjuntos balanceados; comunicação com não-especialistas
Acurácia Balanceada	1	0	$1/C$	Média das sensibilidades por classe	Não	Conjuntos desbalanceados; todas classes com igual relevância
Precisão	1	0	—	Confiabilidade das predições positivas	—	Avaliação de predições positivas
Sensibilidade	1	0	—	Taxa de detecção de positivos	—	Aplicações com alto custo de falsos negativos
F1-Score	1	0	—	Equilíbrio precisão-sensibilidade	—	Avaliação binária balanceada
F1-Score Macro	1	0	—	Desempenho médio entre classes	Não	Otimização de hiperparâmetros; avaliação multiclasse
MCC	+1	-1	0	Correlação predição-verdade	Não	Avaliação em cenários de desbalanceamento acentuado
Escore de Brier	0	2	—	Qualidade das probabilidades	—	Aplicações com decisões baseadas em probabilidades
AUC-ROC	1	0	0,5	Capacidade discriminativa global	Não	Avaliação de ranqueamento; comparação entre modelos
AUC-PR	1	0	p	Desempenho em classes minoritárias	Parcial	Foco em desempenho na classe minoritária
ECE	0	1	—	Calibração das probabilidades	—	Avaliação de calibração de probabilidades

Notas: C = número de classes; p = prevalência da classe positiva; Sens. Desbal. = sensibilidade a desbalanceamento de classes. Para métricas sem valor aleatório definido (—), o valor de referência depende da distribuição específica das classes.

O teste de Friedman foi empregado como procedimento omnibus para avaliar a hipótese nula de que todas as k técnicas de balanceamento apresentam desempenho equivalente (FRIEDMAN, 1937; DEMSAR, 2006). Este teste não-paramétrico opera sobre os postos (*ranks*) dos algoritmos em cada partição experimental, sendo robusto a violações de normalidade e homocedasticidade frequentemente observadas em métricas de classificação. O procedimento consiste em ordenar, para cada partição experimental $j = 1, \dots, n$, as k técnicas de acordo com a métrica de desempenho obtida, atribuindo postos de 1 (melhor desempenho) a k (pior desempenho), calcular o posto médio $\bar{R}_i$ de cada técnica i sobre todas as n partições, e computar a estatística de teste de Friedman expressa pela Equação 5.92

$$\chi_F^2 = \frac{12n}{k(k+1)} \left[\sum_{i=1}^k \bar{R}_i^2 - \frac{k(k+1)^2}{4} \right] \quad (5.92)$$

onde n é o número de partições experimentais, k é o número de técnicas comparadas, e $\bar{R}_i$ é o posto médio da técnica i . Sob a hipótese nula de equivalência entre técnicas, χ_F^2 segue aproximadamente uma distribuição qui-quadrado com $k - 1$ graus de liberdade.

Para amostras de tamanho moderado, a correção de Iman-Davenport transforma a estatística em uma distribuição F com maior precisão (GARCÍA et al., 2010), conforme a Equação 5.93

$$F_F = \frac{(n-1)\chi_F^2}{n(k-1) - \chi_F^2} \quad (5.93)$$

que segue distribuição F com $(k-1)$ e $(k-1)(n-1)$ graus de liberdade.

Condicionado à rejeição da hipótese nula pelo teste de Friedman, o teste post-hoc de Nemenyi identifica pares específicos de técnicas com diferenças estatisticamente significativas (NEMENYI, 1963; DEMSAR, 2006). Duas técnicas i e j são consideradas significativamente diferentes ao nível α se a diferença absoluta entre seus postos médios excede a distância crítica (CD), conforme a Equação 5.94

$$|\bar{R}_i - \bar{R}_j| > \text{CD} = q_{\alpha,k} \sqrt{\frac{k(k+1)}{6n}} \quad (5.94)$$

onde $q_{\alpha,k}$ denota o quantil da distribuição da amplitude estudentizada para k tratamentos ao nível de significância α . Os resultados são tipicamente representados através de diagramas de diferença crítica, nos quais os postos médios dos algoritmos são dispostos em um eixo horizontal, e grupos de técnicas sem diferença significativa são conectados por barras horizontais.

Em análises envolvendo comparação de todas as técnicas contra uma condição de controle específica, o procedimento de Holm foi empregado para controle da taxa de erro tipo I familiar (HOLM, 1979). Este procedimento sequencial opera ordenando os m valores- p das comparações pareadas de forma crescente ($p_1 \leq p_2 \leq \dots \leq p_m$), rejeitando a hipótese H_i se e somente se $p_i < \alpha/(m-i+1)$. O procedimento termina quando a primeira hipótese não pode ser rejeitada, mantendo-se todas as hipóteses subsequentes como não rejeitadas. Os testes estatísticos foram conduzidos utilizando a biblioteca `scikit-posthocs` (TERPIŁOWSKI, 2019) para Python, com cálculos auxiliares realizados pela biblioteca `SciPy` (VIRTANEN et al., 2020).

O teste de Friedman assume que as observações dentro de cada bloco são mutuamente independentes e que os tratamentos foram atribuídos aleatoriamente (DEMSAR, 2006). No contexto de validação cruzada, a sobreposição entre conjuntos de treinamento de diferentes par-

tições introduz dependência entre observações que pode inflacionar a taxa de erro tipo I. Conforme observado por Demsar (2006), não existe atualmente teste estatístico amplamente aceito que acomode adequadamente esta dependência, sendo adotada interpretação conservadora dos resultados neste trabalho.

5.11.4 Validação Estatística e Análise Complementar

A validação estatística tem por objetivo verificar se o desempenho observado supera a classificação aleatória de forma estatisticamente significativa. O nível de chance teórico para um problema de classificação com C classes sob distribuição uniforme corresponde à probabilidade de acerto esperada por um classificador sem capacidade discriminativa, expressa pela Equação 5.95

$$P_{\text{chance}} = \frac{1}{C} \quad (5.95)$$

Para o problema triclasse abordado neste trabalho ($C = 3$), $P_{\text{chance}} \approx 0,333$ (33,3%). Conforme Combrisson e Jerbi (2015), comparar o desempenho observado com este nível permite verificar se existe aprendizado genuíno. A estimativa empírica da distribuição de desempenho aleatório foi obtida através de $N_{\text{sim}} = 1000$ simulações de classificação aleatória estratificada.

O teste de permutação avalia a hipótese nula de que não existe relação entre as características extraídas e os rótulos de classe (OJALA; GARRIGA, 2010). Este método não-paramétrico dispensa suposições sobre a distribuição dos dados. O procedimento executou $N_{\text{perm}} = 1000$ permutações aleatórias dos rótulos verdadeiros, mantendo as características inalteradas. Para cada permutação π , o desempenho foi calculado conforme a Equação 5.96

$$S_{\pi} = M(\mathbf{y}_{\pi}, \hat{\mathbf{y}}) \quad (5.96)$$

onde $M(\cdot, \cdot)$ denota a métrica utilizada e $\mathbf{y}_{\pi}$ corresponde aos rótulos permutados. O p-valor empírico foi calculado com correção conservativa (PHIPSON; SMYTH, 2010), conforme a Equação 5.97

$$p = \frac{1 + \sum_{i=1}^{N_{\text{perm}}} 1[S_{\pi_i} \geq S_{\text{obs}}]}{N_{\text{perm}} + 1} \quad (5.97)$$

onde $1[\cdot]$ é a função indicadora. Resultados com $p < 0,05$ foram considerados estatisticamente significativos.

A quantificação da importância das características foi realizada para permitir interpretabilidade dos modelos treinados. Para classificadores baseados em árvores (Random Forest, Extra Trees, Gradient Boosting, XGBoost), foi empregada a importância baseada na redução média de impureza de Gini (BREIMAN, 2001), conforme a Equação 5.98

$$I_G(x_j) = \sum_{t \in \mathcal{T}_j} p(t) \cdot \Delta G(t) \quad (5.98)$$

onde $\mathcal{T}_j$ denota o conjunto de nós onde a característica j é utilizada para divisão, $p(t)$ é a proporção de amostras que alcançam o nó t , e $\Delta G(t)$ é a redução de impureza obtida. Para o Perceptron Multicamadas, a magnitude absoluta média dos pesos da primeira camada foi utilizada como proxy de importância (GARSON, 1991), conforme a Equação 5.99

$$I_W(x_j) = \frac{1}{H_1} \sum_{h=1}^{H_1} |w_{jh}^{(1)}| \quad (5.99)$$

onde H_1 é o número de neurônios na primeira camada oculta e $w_{jh}^{(1)}$ é o peso conectando a característica j ao neurônio h . As importâncias foram normalizadas para totalizar a unidade, permitindo comparação relativa, conforme a Equação 5.100

$$I_{\text{norm}}(x_j) = \frac{I(x_j)}{\sum_{j'=1}^d I(x_{j'})} \quad (5.100)$$

5.11.5 Configurações utilizadas

A implementação computacional e os experimentos foram realizados em um computador equipado com processador Intel® Core™ i7-14700, operando a 2,10 GHz, arquitetura de 64 bits, e 64 GB de memória RAM (63,7 GB utilizáveis), o que possibilita o processamento de grandes volumes de dados e o treinamento de modelos de aprendizado de máquina com elevada demanda computacional. O sistema dispõe ainda de placa gráfica dedicada com 8 GB de memória de vídeo, permitindo suporte adicional a operações paralelas quando aplicável, além de armazenamento interno de aproximadamente 1 TB, suficiente para acomodar bases de dados, resultados intermediários e registros experimentais.

No que se refere ao ambiente de software, a implementação computacional utiliza bibliotecas consolidadas da linguagem Python, incluindo o scikit-learn (versão 1.3 ou superior) para a implementação dos algoritmos de classificação, o imbalanced-learn para aplicação das técnicas de balanceamento de classes, e as bibliotecas NumPy e SciPy para operações numéricas fundamentais.

Adicionalmente, o processamento paralelo é explorado sempre que disponível, por meio da configuração $n_jobs = -1$, visando a redução do tempo de execução, especialmente em procedimentos de validação cruzada e otimização de hiperparâmetros. A Tab. 5.4 apresenta a configuração completa do pipeline de classificação e avaliação utilizado neste trabalho.

Tabela 5.4 – Configuração do pipeline de classificação e avaliação.

Categoria	Parâmetro	Valor
Pré-processamento	Particionamento	Pré-definido (treino/validação/teste)
	Imputação	Mediana do conjunto de treinamento
	Normalização	Padronização Z-score
Otimização	Iterações de busca	100
	Partições de validação cruzada	5 (estratificada)
	Métrica de otimização	F1-score macro
	Semente aleatória	42
Calibração	Método	Seleção automática
	Limiar para regressão isotônica	1000 amostras
Seleção de características	Método	Seleção automática via CV
	Critério	85% da variância explicada
	Limites	[1, 50] características
Comitê	Agregação	Votação suave (<i>soft voting</i>)
	Lacuna máxima treino-validação	25%
	Desempenho mínimo	40%
Intervalos de confiança	Reamostras bootstrap	2000
	Nível de confiança	95%

A Tab. 5.5 detalha os critérios de filtragem para composição do comitê de classificadores, projetados para excluir modelos com problemas de generalização. O mecanismo de contingência é ativado quando todos os modelos são reprovados pelos critérios de filtragem, garantindo que o comitê sempre contenha pelo menos três modelos para manter os benefícios de ensemble.

Tabela 5.5 – Critérios de filtragem para composição do comitê de classificadores.

Critério	Limiar	Ação
Lacuna treino-validação	$> 25\%$	Exclusão por sobreajuste
Desempenho mínimo em validação	$< 40\%$	Exclusão por subajuste
Mecanismo de contingência	—	Inclusão dos 3 melhores modelos

6 RESULTADOS E DISCUSSÃO

Neste capítulo são apresentados e analisados os resultados da classificação obtidos pelos métodos MLP, Random Forest, Extra Trees, SVM, Gradient Boosting, XGBoost e pelo comitê de classificadores, considerando as configurações plana e em cascata. Para cada modelo, foram avaliados diferentes cenários de balanceamento de classes, com o objetivo de analisar o impacto dessas técnicas no desempenho dos classificadores.

Inicialmente, são discutidos os resultados obtidos sem a aplicação de técnicas de balanceamento, estabelecendo uma linha de base comum para todos os métodos avaliados. Em seguida, são analisados os melhores desempenhos alcançados com a aplicação de técnicas de balanceamento, tanto na configuração plana quanto na configuração em cascata, quando aplicável.

Observou-se que o Random Oversampling beneficiou o MLP na configuração plana e o comitê de classificadores na configuração em cascata. A técnica ADASYN favoreceu o Extra Trees e o comitê na configuração plana, enquanto o SMOTE apresentou impacto positivo no Random Forest na configuração plana e no Extra Trees na configuração em cascata. Já a técnica SMOTEENN proporcionou ganhos consistentes para os modelos SVM e XGBoost em ambas as configurações avaliadas. Para os demais modelos e configurações, não foram observadas melhorias relevantes com o uso das técnicas de balanceamento, sendo o melhor desempenho obtido no cenário sem balanceamento.

Por fim, é realizada uma comparação global entre os melhores resultados obtidos por cada modelo, considerando ambas as configurações, com o objetivo de identificar o classificador de melhor desempenho geral. Esses resultados são então confrontados com aqueles reportados na literatura, permitindo avaliar a competitividade da abordagem proposta em relação aos estudos recentes da área.

6.1 Perceptron Multicamadas

Na configuração plana multiclasse, a melhor arquitetura identificada apresentou duas camadas ocultas com 16 e 8 neurônios, respectivamente, função de ativação ReLU, regularização L2 com $\alpha = 0,122$, taxa de aprendizado inicial de $2,22 \times 10^{-4}$ com estratégia adaptativa,

tamanho de lote de 16 amostras, e seleção de 5 atributos via seleção univariada com função de pontuação ANOVA F. O melhor escore de validação cruzada alcançou $F1_{\text{macro}} = 0,558$.

Para a arquitetura cascata, a primeira etapa (Movimento versus Repouso) utilizou duas camadas ocultas com 64 e 32 neurônios, regularização aproximadamente 7,5 vezes superior à do plano ($\alpha = 0,913$), taxa de aprendizado inicial de $3,86 \times 10^{-4}$, e seleção de 15 atributos. O escore de validação cruzada para a primeira etapa foi $F1_{\text{macro}} = 0,687$. A segunda etapa (Esquerda versus Direita), treinada exclusivamente com amostras de movimento, utilizou arquitetura idêntica à primeira etapa, obtendo $F1_{\text{macro}} = 0,745$ em validação cruzada no cenário com oráculo.

6.1.1 Desempenho Global sem Balanceamento

A Tab. 6.1 sintetiza as métricas de desempenho obtidas com intervalos de confiança de 95% derivados de 2000 reamostras por técnica de bootstrap estratificado. Por ela tem-se que a arquitetura cascata teve ganhos de 9,0 pontos percentuais (p.p) em acurácia (de 0,702 para 0,793) e 10,3 p.p em F1-macro (de 0,684 para 0,787) em relação a configuração plana.

Tabela 6.1 – Métricas de desempenho global do MLP nas configurações plana e cascata, com intervalos de confiança bootstrap de 95%

Métrica	Plano		Cascata		Δ
	Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Acurácia	0,702 $\pm$ 0,015	[0,673; 0,731]	0,793 $\pm$ 0,014	[0,765; 0,820]	+0,090
Acurácia Balanceada	0,693 $\pm$ 0,015	[0,663; 0,721]	0,791 $\pm$ 0,014	[0,764; 0,819]	+0,099
F1-macro	0,684 $\pm$ 0,016	[0,652; 0,714]	0,787 $\pm$ 0,014	[0,760; 0,816]	+0,103
F1-weighted	0,692 $\pm$ 0,015	[0,661; 0,721]	0,790 $\pm$ 0,014	[0,762; 0,818]	+0,098
Precisão (macro)	0,690 $\pm$ 0,016	[0,657; 0,721]	0,799 $\pm$ 0,013	[0,773; 0,825]	+0,109
Recall (macro)	0,693 $\pm$ 0,015	[0,663; 0,721]	0,791 $\pm$ 0,014	[0,764; 0,819]	+0,099
MCC	0,557 $\pm$ 0,022	[0,512; 0,599]	0,696 $\pm$ 0,020	[0,657; 0,736]	+0,139
ROC-AUC (macro)	0,833 $\pm$ 0,012	[0,810; 0,855]	0,907 $\pm$ 0,009	[0,889; 0,923]	+0,074
PR-AUC (macro)	0,708 $\pm$ 0,019	[0,672; 0,746]	0,854 $\pm$ 0,013	[0,828; 0,879]	+0,146
Brier Score	0,151 $\pm$ 0,004	[0,144; 0,158]	0,113 $\pm$ 0,006	[0,102; 0,124]	-0,039

Nota: Δ = Cascata – Plano.

A diferença entre os classificadores foi submetida ao teste de McNemar para avaliação de significância estatística das predições pareadas. A análise mostrou assimetria na tabela de contingência: 112 amostras foram corretamente classificadas exclusivamente pelo cascata contra 42 amostras corretas exclusivas do plano, com 503 acertos comuns e 119 erros comuns

(total de 776 amostras). O teste de McNemar com correção de continuidade confirmou a significância estatística desta diferença ($\chi^2 = 30,92$; $p < 10^{-7}$), rejeitando a hipótese nula de equivalência entre as predições das duas configurações mesmo após correção de Holm-Bonferroni para comparações múltiplas entre os classificadores avaliados neste estudo ($\alpha_{\text{ajustado}} = 0,0083$, correspondente a seis comparações).

Na configuração plana, quatro métricas apresentaram intervalo de confiança de 95% com largura superior a 0,10 (aproximadamente 15% da faixa útil): sensibilidade da classe 1 (Esquerda) com largura de 0,124, precisão da classe 1 (0,115), F1 da classe 1 (0,110) e sensibilidade da classe 0 (0,103). Na configuração em cascata, apenas a sensibilidade da classe 1 excedeu este limiar (largura = 0,112). A redução no número de métricas com incerteza elevada no cascata indica que a decomposição hierárquica não apenas melhora as estimativas pontuais, mas também confere maior estabilidade ao processo de classificação.

6.1.2 Desempenho por Classe sem Balanceamento

A Tab. 6.2 detalha o desempenho discriminado por classe, permitindo identificar onde ocorrem os ganhos mais expressivos da arquitetura cascata. O maior ganho do cascata concentra-se na classe 1 (Esquerda), que apresentava desempenho marcadamente inferior na configuração plana. A sensibilidade desta classe aumentou de 0,451 para 0,704 (+25,3 p.p), e o F1-score de 0,509 para 0,724 (+21,6 p.p).

Tabela 6.2 – Métricas de desempenho do MLP por classe nas configurações plana e cascata

Métrica		Plano		Cascata		Δ
		Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Repouso (Classe 0)	F1	0,755 $\pm$ 0,020	[0,717; 0,793]	0,798 $\pm$ 0,019	[0,760; 0,834]	+0,043
	Precisão	0,784 $\pm$ 0,022	[0,741; 0,825]	0,902 $\pm$ 0,019	[0,865; 0,938]	+0,118
	Recall	0,730 $\pm$ 0,027	[0,677; 0,780]	0,716 $\pm$ 0,026	[0,667; 0,766]	−0,013
Esquerda (Classe 1)	F1	0,509 $\pm$ 0,028	[0,452; 0,562]	0,724 $\pm$ 0,022	[0,681; 0,765]	+0,216
	Precisão	0,583 $\pm$ 0,030	[0,527; 0,642]	0,746 $\pm$ 0,024	[0,697; 0,794]	+0,163
	Recall	0,451 $\pm$ 0,031	[0,391; 0,515]	0,704 $\pm$ 0,029	[0,648; 0,760]	+0,253
Direita (Classe 2)	F1	0,788 $\pm$ 0,015	[0,760; 0,817]	0,840 $\pm$ 0,014	[0,813; 0,866]	+0,052
	Precisão	0,703 $\pm$ 0,018	[0,668; 0,740]	0,751 $\pm$ 0,019	[0,713; 0,789]	+0,047
	Recall	0,896 $\pm$ 0,019	[0,858; 0,931]	0,954 $\pm$ 0,013	[0,927; 0,977]	+0,058

Nota: DP = desvio padrão bootstrap. Δ = Cascata – Plano.

A classe 2 (Direita) apresentou desempenho superior em ambas as configurações, com sensibilidade de 0,896 no plano e 0,954 no cascata. Para a classe 0 (Repouso), observa-se comportamento característico de sistemas em cascata: enquanto a precisão aumentou de 0,784 para 0,902 (+11,8 p.p), a sensibilidade apresentou ligeira redução de 0,730 para 0,716 (−1,3 p.p). Este comportamento resulta da configuração da primeira etapa, que opera com sensibilidade elevada para a classe Movimento (0,955), minimizando falsos negativos de intenções de movimento. Consequentemente, a primeira etapa classifica como Movimento algumas amostras de Repouso (falsos positivos), que são subsequentemente alocadas às classes Esquerda ou Direita pela segunda etapa, reduzindo a sensibilidade de Repouso no resultado final. A melhoria na precisão de Repouso (de 0,784 para 0,902) indica que, quando o cascata classifica uma amostra como Repouso, esta classificação é altamente confiável.

6.1.3 Análise por Etapa da Configuração em Cascata sem Balanceamento

A Tab. 6.3 apresenta as métricas de desempenho para cada etapa da configuração em cascata, incluindo a comparação entre os cenários condicionado (primeira etapa real) e ideal (primeira etapa perfeita, ou oráculo) para a segunda etapa.

Tabela 6.3 – Métricas por etapa da configuração em cascata — MLP

Métrica	Etapa 1		E2 Condicionada		E2 Ideal	
	Média ± DP	IC 95%	Média ± DP	IC 95%	Média ± DP	IC 95%
Acurácia	0,868 ± 0,012	[0,845; 0,892]	0,875 ± 0,015	[0,843; 0,903]	0,880 ± 0,015	[0,850; 0,907]
Acurácia Bal.	0,836 ± 0,014	[0,807; 0,864]	0,866 ± 0,015	[0,835; 0,895]	0,874 ± 0,015	[0,846; 0,901]
F1	0,902 ± 0,009	[0,884; 0,921]	0,894 ± 0,013	[0,866; 0,918]	0,897 ± 0,014	[0,868; 0,921]
Precisão	0,855 ± 0,015	[0,827; 0,884]	0,818 ± 0,022	[0,774; 0,858]	0,823 ± 0,022	[0,778; 0,863]
Sensibilidade	0,955 ± 0,009	[0,937; 0,973]	0,984 ± 0,008	[0,968; 0,996]	0,985 ± 0,008	[0,967; 0,996]
MCC	0,713 ± 0,026	[0,662; 0,762]	0,763 ± 0,026	[0,709; 0,812]	0,774 ± 0,026	[0,723; 0,823]
ROC-AUC	0,939 ± 0,009	[0,921; 0,956]	0,938 ± 0,011	[0,916; 0,957]	0,940 ± 0,010	[0,919; 0,958]
PR-AUC	0,958 ± 0,009	[0,939; 0,973]	0,942 ± 0,011	[0,920; 0,962]	0,941 ± 0,011	[0,917; 0,961]
Brier Score	0,114 ± 0,007	[0,101; 0,128]	0,107 ± 0,012	[0,084; 0,132]	0,102 ± 0,012	[0,079; 0,127]

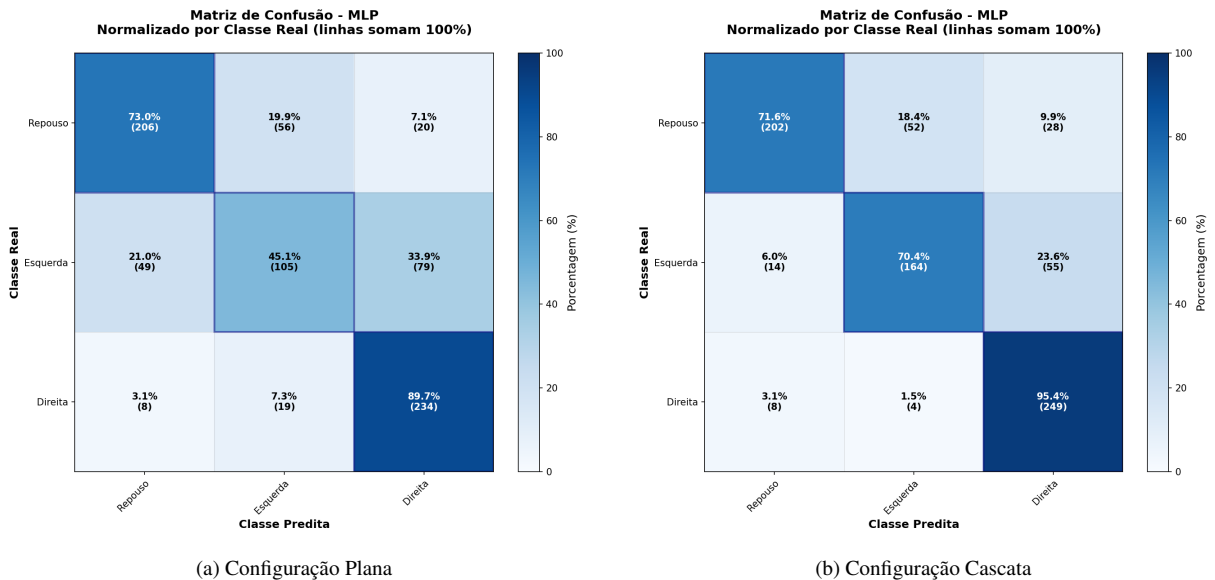
A primeira etapa apresentou capacidade de detecção de movimento adequada, com sensibilidade de 0,955 (IC 95%: [0,937; 0,973]) e ROC-AUC de 0,939. A análise identificou o limiar de 0,52 (superior ao padrão de 0,50), resultando em F1-macro de 0,711 contra 0,696 do limiar padrão, representando melhoria relativa de 2,25%.

A comparação entre os cenários E2 Ideal e E2 Condicionada permite quantificar o impacto da propagação de erros da primeira etapa. A degradação observada foi pequena: $\Delta_{F1} = -0,003$, $\Delta_{\text{acurácia balanceada}} = -0,008$, $\Delta_{\text{precisão}} = -0,005$, e $\Delta_{\text{sensibilidade}} < 0,001$. Esta resiliência pode ser atribuída a dois fatores: (i) a baixa taxa de falsos positivos da primeira etapa em termos absolutos (80 de 282 amostras de Repouso, ou seja, 28,4% das amostras de Repouso, mas apenas 10,3% do total de amostras); e (ii) a possibilidade de que as características espectrais das amostras de Repouso incorretamente encaminhadas à segunda etapa ocupem regiões do espaço de atributos que são razoavelmente bem separadas pelas fronteiras de decisão da segunda etapa.

6.1.4 Matriz de Confusão sem Balanceamento

A Fig. 6.1 compara as matrizes de confusão percentuais das configurações plana e cascata, permitindo visualizar os padrões de erro de cada abordagem. A matriz de confusão da configuração plana (Figura 6.1a) evidencia elevada taxa de confusão da classe Esquerda: 33,91% das amostras desta classe foram incorretamente classificadas como Direita, e 21,03% como Repouso, resultando em apenas 45,06% de acertos. Por outro lado, a classe Direita apresentou sensibilidade de 89,66%, com confusão predominante com Esquerda (7,28%).

Figura 6.1 – Matrizes de confusão percentuais do MLP nas configurações plana (a) e cascata (b). Os valores representam a proporção de cada classe verdadeira distribuída entre as classes preditas (normalização por linha).



Na configuração cascata (Figura 6.1b), a classe Esquerda apresentou melhoria atingindo 70,39% de acertos. A confusão com Direita foi reduzida de 33,91% para 23,61%, enquanto a confusão com Repouso diminuiu de 21,03% para 6,01%. Esta última redução apresenta interesse específico: no cascata, a segunda etapa opera apenas sobre amostras classificadas como movimento pela primeira etapa; portanto, os 6,01% de amostras de Esquerda classificadas como Repouso correspondem integralmente aos falsos negativos da primeira etapa (14 de 233 amostras). A classe Direita manteve desempenho elevado, alcançando 95,40% de acertos. Para a classe Repouso, observou-se ligeira redução na sensibilidade (de 73,05% para 71,63%), acompanhada de redistribuição da confusão: a taxa de confusão com Esquerda reduziu de 19,86% para 18,44%, enquanto a confusão com Direita aumentou de 7,09% para 9,93%.

6.1.5 Análise das Curvas ROC e PR sem Balanceamento

A Tab. 6.4 evidencia o desempenho inferior da classe Esquerda em ambas as métricas na configuração plana (ROC-AUC = 0,715; PR-AUC = 0,594) confirmando sua maior dificuldade de discriminação. Na configuração cascata, esta classe apresentou os maiores ganhos absolutos: +13,7 p.p em ROC-AUC e +21,5 p.p em PR-AUC.

Tabela 6.4 – Áreas sob as curvas ROC e PR por classe e configuração — MLP

Classe	ROC-AUC			PR-AUC		
	Plano	Cascata	Δ	Plano	Cascata	Δ
Repouso	0,921	0,939	+0,018	0,815	0,908	+0,093
Esquerda	0,715	0,852	+0,137	0,594	0,809	+0,215
Direita	0,863	0,931	+0,068	0,715	0,845	+0,130
Macro	0,833	0,907	+0,074	0,708	0,854	+0,146

Nota: Δ = Cascata – Plano.

No presente estudo, a distribuição das classes (Repouso = 282, Esquerda = 233, Direita = 261) configura desbalanceamento moderado (razão máxima $\approx 1,2 : 1$), o que atenua, mas não elimina, a pertinência da PR-AUC como métrica complementar à ROC-AUC. O incremento de PR-AUC macro de 0,708 para 0,854 (+14,6 p.p) indica que o cascata mantém melhor equilíbrio entre precisão e sensibilidade em diversos pontos de operação.

6.1.6 Análise de Calibração sem Balanceamento

A calibração das probabilidades previstas foi avaliada mediante o erro de calibração esperado (ECE) e o erro de calibração máximo (MCE) por classe, utilizando binagem uniforme com 10 intervalos de probabilidade. Conforme observado na Tab. 6.5 na configuração plana, os valores de ECE por classe foram: Repouso = 0,181, Esquerda = 0,062, Direita = 0,127. O MCE da classe Repouso (0,310), localizado no intervalo [0,5–0,6], indica que amostras com probabilidade prevista nesta faixa apresentam discrepância de aproximadamente 0,310 entre a confiança declarada e a frequência observada de acerto.

Tabela 6.5 – Métricas de calibração por classe e configuração — MLP

Classe	ECE		MCE		Bins com MCE	
	Plano	Cascata	Plano	Cascata	Plano	Cascata
Repouso	0,181	0,114	0,310	0,543	[0,5–0,6]	[0,2–0,3]
Esquerda	0,062	0,105	0,332	0,536	[0,5–0,6]	[0,4–0,5]
Direita	0,127	0,119	0,238	0,747	[0,2–0,3]	[0,7–0,8]

Na configuração cascata, a calibração apresentou comportamento misto. O ECE de Repouso melhorou de 0,181 para 0,114, enquanto o ECE de Esquerda deteriorou de 0,062 para 0,105, um aumento de aproximadamente 69%. Esta deterioração pode ser atribuída ao mecanismo de composição de probabilidades no cascata, onde $P(\text{Esquerda}) = P_{E1}(\text{Movimento}) \times P_{E2}(\text{Esquerda}|\text{Movimento})$. A multiplicação de probabilidades tende a concentrar a distribuição em valores mais baixos, e a segunda etapa, treinada apenas em amostras de movimento, pode não estar adequadamente calibrada para o subconjunto de amostras que efetivamente recebe no cenário condicionado.

Os valores elevados de MCE no cascata (0,543 para Repouso, 0,536 para Esquerda, 0,747 para Direita) indicam discrepâncias em faixas específicas de probabilidade, embora a calibração média (ECE) seja aceitável. A classe Direita apresenta o caso mais extremo: no intervalo [0,7–0,8], a discrepância entre probabilidade prevista e frequência observada atinge 0,747, indicando que amostras com alta confiança nesta classe têm frequência de acerto diferente da declarada.

6.1.7 Efeito do Random Oversampling na Configuração Plana

Para o MLP na configuração plana, o Random Oversampling emergiu como a técnica de balanceamento mais efetiva, enquanto a configuração cascata obteve melhor desempenho sem balanceamento externo. Esta seção analisa comparativamente o MLP plano sem balanceamento e com Random Oversampling, investigando os mecanismos pelos quais o balanceamento de classes melhora o desempenho.

A otimização de hiperparâmetros com Random Oversampling convergiu para uma arquitetura distinta da configuração sem balanceamento. O modelo otimizado empregou duas camadas ocultas com 32 e 16 neurônios (versus 16 e 8 na configuração original), função de ativação ReLU, taxa de aprendizado adaptativa, e tamanho de lote de 32 amostras. A regularização L2 apresentou valor de $\alpha = 0,006$, e a seleção de atributos considerou até 33 características. O processo de validação cruzada estratificada com cinco partições alcançou $F1_{\text{macro}} \approx 0,706$, representando melhoria de 26,5% em relação ao score de validação da configuração sem balanceamento ($F1_{\text{macro}} = 0,558$).

A arquitetura mais expressiva combinada com o aumento do conjunto de treinamento via reamostragem permite ao modelo capturar padrões mais complexos nas classes minoritárias sem incorrer em sobreajuste, dado que as amostras sintéticas expandem efetivamente a região de suporte da classe Esquerda no espaço de atributos.

A Tab. 6.6 apresenta a comparação direta entre as configurações, mostrando que o Random Oversampling proporcionou ganhos de 8,0 p.p em acurácia (de 0,702 para 0,782) e 9,0 p.p em F1-macro (de 0,684 para 0,774). A melhoria no MCC (+11,9 p.p), indica que os ganhos não resultam meramente de viés em favor da classe majoritária. A redução do Brier Score de 0,151 para 0,130 (−13,9%) indica melhoria na qualidade das probabilidades preditas.

Tabela 6.6 – Métricas de desempenho global do MLP na configuração plana sem balanceamento e com Random Oversampling, com intervalos de confiança bootstrap de 95%

Métrica	Sem Balanceamento		Random Oversampling		Δ
	Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Acurácia	0,702 $\pm$ 0,015	[0,673; 0,731]	0,782 $\pm$ 0,014	[0,754; 0,811]	+0,080
Acurácia Balanceada	0,693 $\pm$ 0,015	[0,663; 0,721]	0,773 $\pm$ 0,015	[0,745; 0,803]	+0,080
F1-macro	0,684 $\pm$ 0,016	[0,652; 0,714]	0,774 $\pm$ 0,015	[0,745; 0,804]	+0,090
F1-weighted	0,692 $\pm$ 0,015	[0,661; 0,721]	0,778 $\pm$ 0,015	[0,749; 0,808]	+0,086
Precisão (macro)	0,690 $\pm$ 0,016	[0,657; 0,721]	0,791 $\pm$ 0,015	[0,762; 0,819]	+0,101
Recall (macro)	0,693 $\pm$ 0,015	[0,663; 0,721]	0,773 $\pm$ 0,015	[0,745; 0,803]	+0,080
MCC	0,557 $\pm$ 0,022	[0,512; 0,599]	0,676 $\pm$ 0,022	[0,634; 0,719]	+0,119
ROC-AUC (macro)	0,833 $\pm$ 0,012	[0,810; 0,855]	0,875 $\pm$ 0,010	[0,854; 0,896]	+0,042
PR-AUC (macro)	0,708 $\pm$ 0,019	[0,672; 0,746]	0,818 $\pm$ 0,014	[0,789; 0,846]	+0,110
Brier Score	0,151 $\pm$ 0,004	[0,144; 0,158]	0,130 $\pm$ 0,004	[0,123; 0,138]	−0,021

Nota: Δ = Random Oversampling – Sem Balanceamento.

A Tab. 6.7 detalha o desempenho discriminado por classe, permitindo identificar onde e como o Random Oversampling atua. A Classe Esquerda (minoritária) constitui a principal beneficiária do balanceamento, com ganhos em todas as métricas. O F1-score aumentou de 0,509 para 0,698 (+18,9 p.p), representando melhoria relativa de 37,1%. A precisão apresentou o maior ganho absoluto (+24,3 p.p, de 0,583 para 0,826), indicando redução nos falsos positivos desta classe. A sensibilidade também melhorou (+15,4 p.p), embora permaneça a mais baixa entre as três classes (0,605). Este padrão indica que o modelo com balanceamento tornou-se mais conservador ao predizer Esquerda (precisão alta), mas simultaneamente mais sensível a seus padrões característicos.

Tabela 6.7 – Métricas de desempenho do MLP por classe nas configurações sem balanceamento e com Random Oversampling

Métrica		Sem Balanceamento		Random Oversampling		Δ
		Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Repouso (Classe 0)	F1	0,755 $\pm$ 0,020	[0,717; 0,793]	0,806 $\pm$ 0,016	[0,774; 0,837]	+0,051
	Precisão	0,784 $\pm$ 0,022	[0,741; 0,825]	0,746 $\pm$ 0,020	[0,708; 0,787]	−0,038
	Recall	0,730 $\pm$ 0,027	[0,677; 0,780]	0,876 $\pm$ 0,020	[0,837; 0,911]	+0,146
Esquerda (Classe 1)	F1	0,509 $\pm$ 0,028	[0,452; 0,562]	0,698 $\pm$ 0,026	[0,648; 0,751]	+0,189
	Precisão	0,583 $\pm$ 0,030	[0,527; 0,642]	0,826 $\pm$ 0,026	[0,773; 0,874]	+0,243
	Recall	0,451 $\pm$ 0,031	[0,391; 0,515]	0,605 $\pm$ 0,033	[0,541; 0,670]	+0,154
Direita (Classe 2)	F1	0,788 $\pm$ 0,015	[0,760; 0,817]	0,819 $\pm$ 0,016	[0,785; 0,850]	+0,031
	Precisão	0,703 $\pm$ 0,018	[0,668; 0,740]	0,800 $\pm$ 0,020	[0,759; 0,838]	+0,097
	Recall	0,896 $\pm$ 0,019	[0,858; 0,931]	0,839 $\pm$ 0,023	[0,793; 0,881]	−0,057

Nota: DP = desvio padrão bootstrap. Δ = Random Oversampling – Sem Balanceamento.

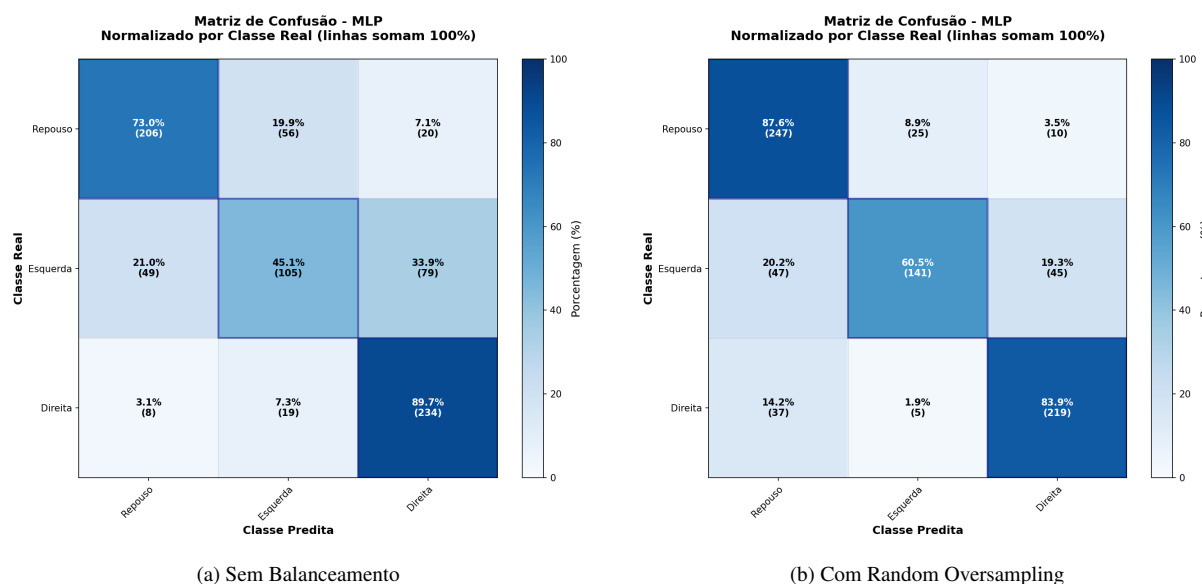
Na Classe Repouso A sensibilidade aumentou de 0,730 para 0,876 (+14,6 p.p), enquanto a precisão reduziu de 0,784 para 0,746 (−3,8 p.p). O F1-score resultante melhorou de 0,755 para 0,806 (+5,1 p.p), indicando que o ganho em sensibilidade compensou a perda em precisão. Este comportamento sugere que o modelo balanceado tornou-se mais agressivo na detecção de Repouso, capturando amostras que anteriormente eram incorretamente classificadas como movimento.

A Classe Direita apresentou comportamento inverso ao Repouso. A precisão aumentou de 0,703 para 0,800 (+9,7 p.p), enquanto a sensibilidade reduziu de 0,896 para 0,839 (−5,7 p.p). O F1-score melhorou de 0,788 para 0,819 (+3,1 p.p). A redução da sensibilidade indica que o modelo balanceado tornou-se mais conservador ao classificar amostras como Direita, cedendo parte de sua dominância sobre o espaço de predições para as demais classes. Este constitui um efeito esperado do balanceamento: ao aumentar a representação das classes minoritárias no treinamento, as fronteiras de decisão se deslocam, reduzindo a região atribuída à classe originalmente dominante.

A dinâmica observada pode ser sintetizada como uma reconfiguração do espaço de decisão: sem balanceamento, o modelo favorecia fortemente a classe Direita (maior sensibilidade), em detrimento das demais; com balanceamento, as fronteiras de decisão foram redistribuídas de forma mais equitativa, resultando em melhor desempenho agregado às custas de uma pequena redução na sensibilidade da classe Direita.

A comparação das matrizes de confusão feita na Fig. 6.2 confirma que a técnica de balanceamento beneficiou o classificador MLP na configuração plana. Para a Classe Repouso, a taxa de acerto aumentou de 73,05% para 87,59% (+14,54 p.p). As confusões com as classes de movimento foram reduzidas: Repouso→Esquerda caiu de 19,86% para 8,87% (−11,0 p.p) e Repouso→Direita de 7,09% para 3,55% (−3,5 p.p). O modelo balanceado discrimina Repouso das classes de movimento com eficácia superior.

Figura 6.2 – Matrizes de confusão percentuais do MLP na configuração plana sem balanceamento (a) e com Random Oversampling (b). Os valores representam a proporção de cada classe verdadeira distribuída entre as classes preditas (normalização por linha).



Para a Classe Esquerda, a taxa de acerto aumentou de 45,06% para 60,52% (+15,46 p.p). A melhoria resultou principalmente da redução da confusão Esquerda→Direita, que caiu de 33,91% para 19,31% (−14,6 p.p). A confusão Esquerda→Repouso permaneceu estável (21,03% versus 20,17%).

Para a Classe Direita, a taxa de acerto reduziu de 89,66% para 83,91% (−5,75 p.p). Os erros redistribuíram-se significativamente: a confusão Direita→Repouso aumentou de 3,07% para 14,18% (+11,1 p.p), enquanto Direita→Esquerda reduziu de 7,28% para 1,92% (−5,4 p.p). Este padrão revela que o balanceamento tornou o modelo mais propenso a classificar amostras de Direita como Repouso, mas simultaneamente mais capaz de distinguir melhor Direita de Esquerda.

A análise das curvas ROC e de precisão versus sensibilidade feita pela Tab. 6.8 revela padrões complementares. Para ROC-AUC, a classe Esquerda apresentou o maior ganho (+9,4 p.p, de 0,715 para 0,809), confirmando que o balanceamento melhora a capacidade discriminativa para esta classe em toda a faixa de limiares de decisão. A classe Direita também melhorou (+4,7 p.p), enquanto Repouso apresentou ligeira redução (−1,5 p.p). A melhoria macro de 0,833 para 0,875 indica ganho na capacidade discriminativa global.

Tabela 6.8 – Áreas sob as curvas ROC e PR por classe — MLP sem e com balanceamento

Classe	ROC-AUC			PR-AUC		
	Sem Bal.	Random Over	Δ	Sem Bal.	Random Over	Δ
Repouso	0,921	0,906	−0,015	0,815	0,839	+0,024
Esquerda	0,715	0,809	+0,094	0,594	0,781	+0,187
Direita	0,863	0,910	+0,047	0,715	0,833	+0,118
Macro	0,833	0,875	+0,042	0,708	0,818	+0,110

Nota: Δ = Random Oversampling — Sem Balanceamento.

Para PR-AUC, os ganhos foram ainda mais expressivos, especialmente para a classe Esquerda (+18,7 p.p, de 0,594 para 0,781). A PR-AUC apresenta maior sensibilidade ao desempenho em classes minoritárias (SAITO; REHMSMEIER, 2015), e o ganho indica que o modelo balanceado consegue manter alta precisão em níveis mais elevados de sensibilidade para a classe Esquerda. A melhoria macro de 0,708 para 0,818 (+11,0 p.p) representa avanço considerável na qualidade das previsões probabilísticas.

A combinação de ganhos em ROC-AUC e PR-AUC para a classe Esquerda, junto com a melhoria da precisão observada na análise por classe, indica que o Random Oversampling não apenas desloca o ponto de operação do classificador, mas melhora a separabilidade das classes, permitindo ao modelo encontrar fronteiras de decisão mais discriminativas.

A Tab. 6.9 compara as métricas de calibração entre as configurações. Por ela tem-se que o ECE de Repouso melhorou de 0,181 para 0,140 (−22,7%), e o MCE reduziu de 0,310 para 0,181 (−41,6%), indicando probabilidades mais confiáveis para esta classe. O ECE de Direita também melhorou ligeiramente (0,127 para 0,117), mas o MCE deteriorou de 0,238 para 0,412, com a faixa de maior descalibração deslocando-se de [0,2–0,3] para [0,5–0,6].

Tabela 6.9 – Métricas de calibração por classe — MLP sem e com balanceamento

Classe	ECE		MCE		Bin com MCE	
	Sem Bal.	Random Over	Sem Bal.	Random Over	Sem Bal.	Random Over
Repouso	0,181	0,140	0,310	0,181	[0,5–0,6]	[0,6–0,7]
Esquerda	0,062	0,101	0,332	0,329	[0,5–0,6]	[0,3–0,4]
Direita	0,127	0,117	0,238	0,412	[0,2–0,3]	[0,5–0,6]

Para a classe Esquerda, o ECE deteriorou de 0,062 para 0,101, embora o MCE tenha permanecido estável (0,332 versus 0,329). Esta deterioração na calibração média da classe que mais se beneficiou do balanceamento sugere que as amostras sintéticas geradas pelo Random Oversampling, embora efetivas para melhorar a discriminação, introduzem alguma incerteza na estimativa de probabilidades. O modelo torna-se mais decisivo (predições mais concentradas), mas não necessariamente mais calibrado para a classe balanceada.

6.1.8 Síntese Comparativa

A Tab. 6.10 sintetiza os principais resultados das três configurações analisadas. A configuração cascata sem balanceamento apresentou o melhor desempenho em métricas baseadas em predições discretas (acurácia = 0,793, F1-macro = 0,787, MCC = 0,696) e em métricas baseadas em probabilidades (ROC-AUC macro = 0,907, PR-AUC macro = 0,854). A configuração plana com Random Oversampling obteve resultados intermediários, superiores à configuração plana sem balanceamento em todas as métricas.

Tabela 6.10 – Síntese comparativa das configurações do MLP

Métrica	Plano	Plano+RO	Cascata
Acurácia	0,702	0,782	0,793
F1-macro	0,684	0,774	0,787
MCC	0,557	0,676	0,696
ROC-AUC macro	0,833	0,875	0,907
PR-AUC macro	0,708	0,818	0,854
Brier Score	0,151	0,130	0,113

Nota: RO = Random Oversampling. Valores em negrito indicam melhor desempenho para cada métrica.

A adequação do modelo ao problema depende dos requisitos da aplicação. Para sistemas que priorizam a identificação correta de comandos de movimento com mínima perda de

intenções, a configuração cascata oferece o melhor equilíbrio, com F1-macro de 0,787, sensibilidade acima de 0,70 para todas as classes, e sensibilidade de 0,955 para detecção de movimento na primeira etapa. Para aplicações que requerem estimativas probabilísticas confiáveis, como sistemas de controle adaptativo baseados em confiança da predição, a configuração cascata também é preferível, apresentando ROC-AUC de 0,907 e Brier Score de 0,113, embora a calibração apresente heterogeneidade entre classes que requerem atenção.

A configuração plana sem balanceamento não se mostrou adequada para este problema específico, dado o desempenho marcadamente inferior da classe Esquerda ($F1 = 0,509$, sensibilidade = 0,451). O desbalanceamento moderado das classes (razão máxima de 1,2:1) é amplificado pela maior dificuldade intrínseca de discriminação dos padrões de imaginação motora da mão esquerda, resultando em viés sistemático do classificador em favor da classe Direita.

Em síntese, o MLP na configuração cascata sem balanceamento constitui a abordagem mais adequada para o problema de classificação de imaginação motora avaliado, apresentando superioridade estatisticamente significativa sobre a configuração plana (teste de McNemar, $p < 10^{-7}$) e melhor equilíbrio entre as métricas de desempenho por classe. A decomposição hierárquica do problema mostrou-se efetiva para mitigar a dificuldade de discriminação da classe Esquerda, com ganhos de 25,3 p.p em sensibilidade em comparação com a configuração plana.

6.2 Extra Trees

Na configuração plana multiclasse sem balanceamento, a configuração selecionada utilizou 200 estimadores, profundidade máxima de 5 níveis, número mínimo de amostras por folha igual a 30, número mínimo de amostras para divisão igual a 10, número de atributos considerados por divisão igual a $\log_2(n)$ e ponderação de classes por subamostra. A seleção univariada de atributos com função de pontuação ANOVA F reteve 29 atributos. O maior escore médio em validação cruzada foi $F1_{\text{macro}} = 0,562$, valor acima do desempenho esperado ao acaso (aproximadamente 0,333 para três classes equiprováveis), o que delimita o patamar inicial de desempenho observado no conjunto de teste.

Na arquitetura em cascata sem balanceamento, a primeira etapa (Movimento versus Repouso) utilizou 100 estimadores, profundidade máxima de 5 níveis, número mínimo de amostras por folha de 30, número mínimo de amostras para divisão de 30, número de atributos conside-

rados por divisão igual a $\sqrt{n}$, ponderação de classes por subamostra e seleção de 23 atributos. O escore de validação cruzada desta etapa foi $F1_{\text{macro}} = 0,678$. A otimização do limiar de decisão em validação cruzada estratificada selecionou 0,54, resultando em $F1_{\text{macro}} = 0,722$ em comparação com 0,720 no limiar padrão (0,50), correspondendo a variação relativa de 0,34%.

6.2.1 Desempenho Global sem Balanceamento

Conforme apresentado na Tab. 6.11, as métricas globais foram estimadas com intervalos de confiança de 95% obtidos por *bootstrap* estratificado com 2000 reamostras. As estimativas indicam proximidade entre as configurações plana e em cascata: a diferença de acurácia foi de $-0,005$ (de 0,757 para 0,752) e a diferença em $F1_{\text{macro}}$ foi de $-0,005$ (de 0,750 para 0,745). Métricas baseadas em ordenação de probabilidades apresentaram variações do mesmo nível de magnitude, com redução de 0,016 em ROC-AUC macro e de 0,025 em PR-AUC macro na configuração em cascata.

Tabela 6.11 – Métricas de desempenho global do Extra Trees nas configurações plana e em cascata sem balanceamento, com intervalos de confiança *bootstrap* de 95%

Métrica	Plano		Cascata		Δ
	Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Acurácia	0,757 $\pm$ 0,015	[0,728; 0,786]	0,752 $\pm$ 0,014	[0,723; 0,780]	$-0,005$
Acurácia Balanceada	0,755 $\pm$ 0,015	[0,726; 0,784]	0,756 $\pm$ 0,014	[0,727; 0,784]	$+0,001$
F1-macro	0,750 $\pm$ 0,015	[0,719; 0,781]	0,745 $\pm$ 0,015	[0,714; 0,774]	$-0,005$
F1-weighted	0,754 $\pm$ 0,015	[0,723; 0,784]	0,746 $\pm$ 0,015	[0,715; 0,776]	$-0,008$
Precisão (macro)	0,781 $\pm$ 0,013	[0,757; 0,805]	0,776 $\pm$ 0,013	[0,751; 0,800]	$-0,006$
Recall (macro)	0,755 $\pm$ 0,015	[0,726; 0,784]	0,756 $\pm$ 0,014	[0,727; 0,784]	$+0,001$
MCC	0,651 $\pm$ 0,021	[0,610; 0,692]	0,645 $\pm$ 0,020	[0,605; 0,685]	$-0,006$
ROC-AUC (macro)	0,929 $\pm$ 0,007	[0,915; 0,942]	0,914 $\pm$ 0,008	[0,897; 0,929]	$-0,016$
PR-AUC (macro)	0,887 $\pm$ 0,010	[0,866; 0,907]	0,862 $\pm$ 0,013	[0,835; 0,887]	$-0,025$
Brier Score	0,109 $\pm$ 0,006	[0,098; 0,120]	0,115 $\pm$ 0,006	[0,104; 0,127]	$+0,006$

Nota: $\Delta = \text{Cascata} - \text{Plano}$

Adotou-se como referência operacional a largura de 0,10 para o IC 95%. Na configuração plana, excederam este valor: sensibilidade da classe 1 (Esquerda), com largura de 0,120; sensibilidade da classe 0 (Repouso), com largura de 0,114; e precisão da classe 1, com largura de 0,101. Na configuração em cascata, excederam este valor: sensibilidade da classe 0 (Repouso), com largura de 0,117; e sensibilidade da classe 1 (Esquerda), com largura de 0,112.

6.2.2 Desempenho por Classe sem Balanceamento

Conforme detalhado na Tab. 6.12, as métricas por classe permitem localizar quais classes concentram as diferenças entre as duas arquiteturas. Na configuração plana, a classe Direita apresentou sensibilidade de 0,969, enquanto Repouso e Esquerda apresentaram sensibilidade de 0,656 e 0,640, respectivamente. Na configuração em cascata, a classe Esquerda apresentou aumento de sensibilidade (de 0,640 para 0,726), ao passo que a classe Repouso apresentou redução (de 0,656 para 0,575); para Direita, a sensibilidade permaneceu próxima (0,969 no plano e 0,966 no cascata).

Tabela 6.12 – Métricas de desempenho do Extra Trees por classe nas configurações plana e em cascata sem balanceamento

Métrica		Plano		Cascata		Δ
		Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Repouso (Classe 0)	F1	0,785 $\pm$ 0,021	[0,743; 0,826]	0,714 $\pm$ 0,024	[0,664; 0,759]	-0,071
	Precisão	0,979 $\pm$ 0,010	[0,957; 0,995]	0,942 $\pm$ 0,017	[0,906; 0,974]	-0,037
	Recall	0,656 $\pm$ 0,028	[0,599; 0,713]	0,575 $\pm$ 0,029	[0,514; 0,631]	-0,081
Esquerda (Classe 1)	F1	0,655 $\pm$ 0,025	[0,605; 0,701]	0,686 $\pm$ 0,021	[0,646; 0,728]	+0,031
	Precisão	0,672 $\pm$ 0,026	[0,622; 0,723]	0,651 $\pm$ 0,023	[0,607; 0,696]	-0,021
	Recall	0,640 $\pm$ 0,032	[0,579; 0,700]	0,726 $\pm$ 0,029	[0,670; 0,781]	+0,086
Direita (Classe 2)	F1	0,809 $\pm$ 0,013	[0,784; 0,834]	0,834 $\pm$ 0,013	[0,807; 0,860]	+0,025
	Precisão	0,694 $\pm$ 0,017	[0,661; 0,730]	0,733 $\pm$ 0,019	[0,698; 0,771]	+0,039
	Recall	0,969 $\pm$ 0,011	[0,946; 0,989]	0,966 $\pm$ 0,011	[0,943; 0,985]	-0,004

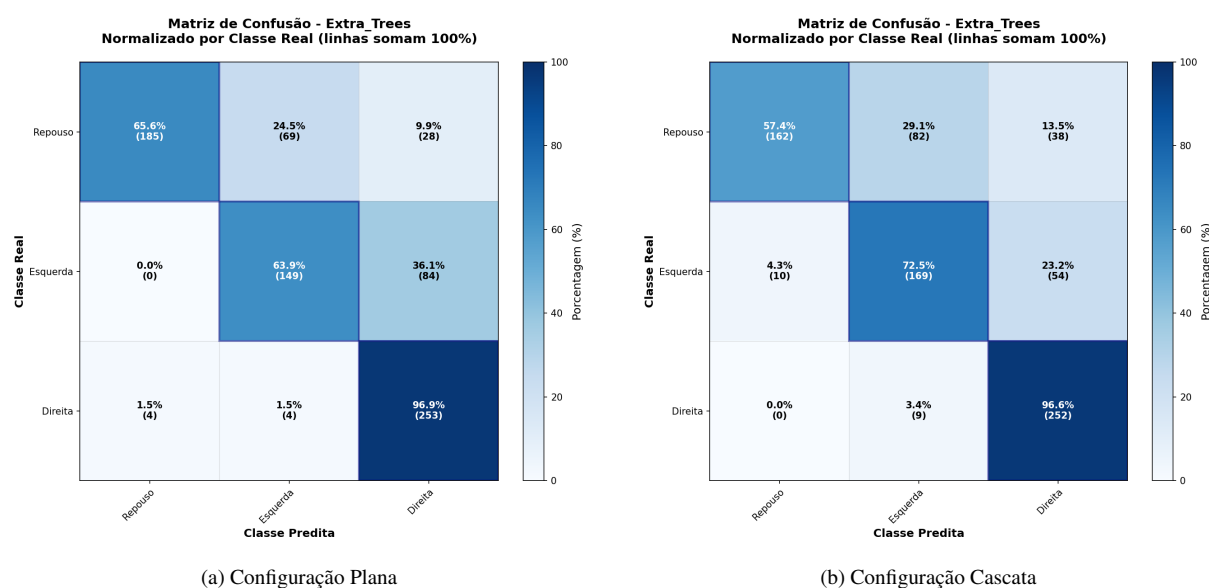
Nota: Δ = Cascata – Plano.

A variação observada entre as classes é compatível com o mecanismo operacional da configuração em cascata: a primeira etapa foi configurada com elevada sensibilidade para Movimento (sensibilidade de 0,980), o que eleva a taxa de falsos positivos na detecção de movimento. Nesta configuração, 42,55% das amostras de Repouso foram encaminhadas à segunda etapa, o que reduz a sensibilidade final de Repouso ao redistribuir parte destas amostras entre Esquerda e Direita. Em contrapartida, ao restringir a discriminação Esquerda versus Direita ao subconjunto encaminhado como movimento, parte da confusão Esquerda→Direita é mitigada, refletindo-se na elevação da sensibilidade da classe Esquerda no resultado final.

6.2.3 Matriz de Confusão sem Balanceamento

Conforme apresentado na Fig. 6.3, as matrizes de confusão percentuais permitem comparar diretamente os padrões de erro entre as configurações plana e em cascata. Na configuração plana (Figura 6.3a), a classe Repouso apresentou 65,60% de acertos, com confusão majoritariamente direcionada a Esquerda (24,47%) e, em menor proporção, a Direita (9,93%). A classe Esquerda apresentou 63,95% de acertos, com confusão exclusivamente para Direita (36,05%) e ausência de confusão para Repouso (0,00%). A classe Direita apresentou 96,93% de acertos.

Figura 6.3 – Matrizes de confusão percentuais do Extra Trees nas configurações plana (a) e em cascata (b) sem balanceamento. Os valores representam a proporção de cada classe verdadeira distribuída entre as classes preditas (normalização por linha).



Na configuração em cascata (Figura 6.3b), a classe Repouso apresentou 57,45% de acertos, com aumento da redistribuição para Esquerda (29,08%) e Direita (13,48%), em consonância com a taxa de falsos positivos na primeira etapa. A classe Esquerda apresentou 72,53% de acertos, com redução da confusão para Direita (23,18%) e surgimento de confusão para Repouso (4,29%), associada aos falsos negativos da etapa Movimento versus Repouso. A classe Direita apresentou 96,55% de acertos.

6.2.4 Análise por Etapa da Configuração em Cascata sem Balanceamento

A Tab. 6.13 apresenta as métricas de desempenho para cada etapa da configuração em cascata, incluindo a comparação entre os cenários condicionado (primeira etapa real) e ideal (primeira etapa perfeita, ou oráculo) para a segunda etapa.

Tabela 6.13 – Métricas por etapa da configuração em cascata — Extra Trees

Métrica	Etapa 1		E2 Condicionada		E2 Ideal	
	Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%
Acurácia	0,832 $\pm$ 0,013	[0,805; 0,860]	0,870 $\pm$ 0,015	[0,839; 0,899]	0,873 $\pm$ 0,015	[0,842; 0,903]
Acurácia Bal.	0,777 $\pm$ 0,015	[0,748; 0,806]	0,861 $\pm$ 0,016	[0,832; 0,892]	0,867 $\pm$ 0,015	[0,837; 0,896]
F1	0,881 $\pm$ 0,010	[0,861; 0,902]	0,889 $\pm$ 0,014	[0,860; 0,916]	0,889 $\pm$ 0,014	[0,859; 0,916]
Precisão	0,801 $\pm$ 0,016	[0,769; 0,833]	0,824 $\pm$ 0,022	[0,780; 0,867]	0,823 $\pm$ 0,022	[0,778; 0,866]
Sensibilidade	0,980 $\pm$ 0,006	[0,967; 0,992]	0,965 $\pm$ 0,011	[0,942; 0,985]	0,965 $\pm$ 0,012	[0,942; 0,986]
MCC	0,641 $\pm$ 0,026	[0,592; 0,691]	0,747 $\pm$ 0,028	[0,693; 0,802]	0,754 $\pm$ 0,028	[0,699; 0,809]
ROC-AUC	0,946 $\pm$ 0,009	[0,928; 0,962]	0,939 $\pm$ 0,011	[0,915; 0,959]	0,942 $\pm$ 0,011	[0,920; 0,961]
PR-AUC	0,961 $\pm$ 0,009	[0,943; 0,976]	0,933 $\pm$ 0,016	[0,900; 0,961]	0,933 $\pm$ 0,016	[0,899; 0,962]
Brier Score	0,116 $\pm$ 0,007	[0,102; 0,130]	0,116 $\pm$ 0,013	[0,091; 0,141]	0,113 $\pm$ 0,013	[0,089; 0,139]

A primeira etapa apresentou elevada capacidade de detecção de movimento, com sensibilidade de 0,980 e ROC-AUC de 0,946. A análise identificou o limiar de 0,54, resultando em F1-macro de 0,722 contra 0,720 do limiar padrão, representando melhoria relativa de 0,34%.

A comparação entre os cenários E2 Ideal e E2 Condicionada permite quantificar o impacto da propagação de erros da primeira etapa. A degradação observada foi pequena: $\Delta_{F1} = 0,000$, $\Delta_{\text{acurácia balanceada}} = -0,005$, $\Delta_{\text{precisão}} = 0,000$, e $\Delta_{\text{sensibilidade}} = 0,000$. Esta resiliência pode ser atribuída a dois fatores: (i) a baixa taxa de falsos negativos da primeira etapa em termos absolutos (10 de 494 amostras de Movimento, ou seja, 2,0% das amostras de Movimento); e (ii) a possibilidade de que as características espectrais das amostras de Repouso incorretamente encaminhadas à segunda etapa ocupem regiões do espaço de atributos que são razoavelmente bem separadas pelas fronteiras de decisão da segunda etapa. A análise de decomposição de erro revelou que 67,4% do erro total do sistema em cascata é atribuível à primeira etapa, com 120 falsos positivos (42,6% das amostras de Repouso) e apenas 10 falsos negativos (2,0% das amostras de Movimento). O erro com primeira etapa perfeita seria de 63 amostras, enquanto o erro real observado foi de 193 amostras.

6.2.5 Análise das Curvas ROC e PR sem Balanceamento

Conforme indicado na Tab. 6.14, o modelo plano apresentou ROC-AUC macro de 0,929, enquanto o cascata apresentou 0,913. Para PR-AUC, os valores foram 0,887 (plano) e 0,861 (cascata). A classe Esquerda apresentou PR-AUC praticamente inalterada (0,804 no plano e 0,806 no cascata), enquanto Direita apresentou a maior redução em PR-AUC (de 0,912 para 0,855). Uma interpretação compatível com esta variação consiste no efeito de composição de probabilidades na configuração em cascata, no qual a etapa final utiliza probabilidades condicionadas ao encaminhamento por E1, alterando a distribuição de escores utilizados na construção das curvas.

Tabela 6.14 – Áreas sob as curvas ROC e PR por classe e configuração — Extra Trees sem balanceamento

Classe	ROC-AUC			PR-AUC		
	Plano	Cascata	Δ	Plano	Cascata	Δ
Repouso	0,957	0,946	−0,011	0,944	0,922	−0,022
Esquerda	0,872	0,850	−0,023	0,804	0,806	+0,002
Direita	0,958	0,944	−0,014	0,912	0,855	−0,057
Macro	0,929	0,913	−0,016	0,887	0,861	−0,026

Nota: Δ = Cascata – Plano.

6.2.6 Análise de Calibração sem Balanceamento

Conforme apresentado na Tab. 6.15, a configuração plana apresentou ECE de 0,098 (Repouso), 0,083 (Esquerda) e 0,123 (Direita), enquanto a configuração em cascata apresentou ECE de 0,127, 0,100 e 0,131, respectivamente. O MCE apresentou máximos concentrados na faixa [0,5–0,6] em diversas classes, indicando regiões específicas em que a discrepância entre probabilidade predita e frequência observada atinge o maior valor.

Tabela 6.15 – Métricas de calibração por classe e configuração — Extra Trees sem balanceamento

Classe	ECE		MCE		Intervalo com MCE	
	Plano	Cascata	Plano	Cascata	Plano	Cascata
Repouso	0,098	0,127	0,478	0,559	[0,5–0,6]	[0,4–0,5]
Esquerda	0,083	0,100	0,251	0,300	[0,5–0,6]	[0,5–0,6]
Direita	0,123	0,131	0,576	0,563	[0,5–0,6]	[0,5–0,6]

Na configuração plana, o maior MCE ocorreu para Direita (0,576) no intervalo [0,5–0,6], enquanto para Repouso o MCE foi 0,478 também nesta faixa. Na configuração em cascata, observou-se aumento do ECE em todas as classes, com MCE de 0,559 (Repouso), 0,300 (Esquerda) e 0,563 (Direita). Estes resultados descrevem que as probabilidades fornecidas pelo modelo em cascata apresentam, em média, maior desalinhamento com frequências empíricas observadas no conjunto avaliado, sobretudo em regiões intermediárias da escala de probabilidade.

6.2.7 Efeito do ADASYN na Configuração Plana

Na configuração plana multiclasse com reamostragem ADASYN, a configuração selecionada para o classificador Extra Trees empregou uma floresta composta por 300 estimadores, profundidade máxima limitada a 12 níveis e ponderação de classes por subamostra. A estratégia de aleatorização utilizou fração de atributos de 0,3, com divisão mínima de 10 amostras por nó interno e 10 amostras por folha. A seleção de atributos manteve 37 variáveis.

Conforme apresentado na Tab. 6.16, a aplicação do ADASYN elevou a acurácia de 0,757 para 0,799 (+4,2 p.p), o F1-macro de 0,750 para 0,793 (+4,3 p.p) e o MCC de 0,651 para 0,707 (+5,6 p.p). As métricas baseadas em probabilidades permaneceram próximas entre as configurações: a ROC-AUC macro passou de 0,929 para 0,934 (+0,5 ponto percentual), enquanto a PR-AUC macro variou de 0,887 para 0,883 (−0,4 ponto percentual). O Brier Score reduziu de 0,109 para 0,096, indicando redução do erro quadrático médio associado às probabilidades preditas.

Tabela 6.16 – Métricas de desempenho global do Extra Trees na configuração plana sem balanceamento e com ADASYN, com intervalos de confiança *bootstrap* de 95%

Métrica	Sem Balanceamento		ADASYN		Δ
	Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Acurácia	0,757 $\pm$ 0,015	[0,728; 0,786]	0,799 $\pm$ 0,014	[0,771; 0,827]	+0,042
Acurácia Balanceada	0,755 $\pm$ 0,015	[0,726; 0,784]	0,795 $\pm$ 0,014	[0,766; 0,823]	+0,040
F1-macro	0,750 $\pm$ 0,015	[0,719; 0,781]	0,793 $\pm$ 0,015	[0,763; 0,822]	+0,043
F1-weighted	0,754 $\pm$ 0,015	[0,723; 0,784]	0,797 $\pm$ 0,014	[0,768; 0,825]	+0,043
Precisão (macro)	0,781 $\pm$ 0,013	[0,757; 0,805]	0,809 $\pm$ 0,014	[0,781; 0,836]	+0,028
Recall (macro)	0,755 $\pm$ 0,015	[0,726; 0,784]	0,795 $\pm$ 0,014	[0,766; 0,823]	+0,040
MCC	0,651 $\pm$ 0,021	[0,610; 0,692]	0,707 $\pm$ 0,020	[0,665; 0,747]	+0,056
ROC-AUC (macro)	0,929 $\pm$ 0,007	[0,915; 0,942]	0,934 $\pm$ 0,007	[0,919; 0,947]	+0,005
PR-AUC (macro)	0,887 $\pm$ 0,010	[0,866; 0,907]	0,883 $\pm$ 0,013	[0,858; 0,907]	−0,004
Brier Score	0,109 $\pm$ 0,006	[0,098; 0,120]	0,096 $\pm$ 0,005	[0,086; 0,106]	−0,013

Nota: Δ = ADASYN – Sem Balanceamento.

O detalhamento por classe, apresentado na Tab. 6.17, evidencia que a variação do desempenho não foi uniforme entre classes. Na classe Repouso, a sensibilidade aumentou de 0,656 para 0,766 (+11,0 p.p) e o F1-score passou de 0,785 para 0,838 (+5,3 p.p), acompanhado de redução na precisão de 0,979 para 0,926 (−5,3 p.p). Na classe Esquerda, a precisão aumentou de 0,672 para 0,771 (+9,9 p.p), enquanto a sensibilidade variou de 0,640 para 0,665 (+2,5 p.p). Na classe Direita, observou-se redução na sensibilidade de 0,969 para 0,954 (−1,5 ponto percentual), com aumento na precisão de 0,694 para 0,729 (+3,5 p.p).

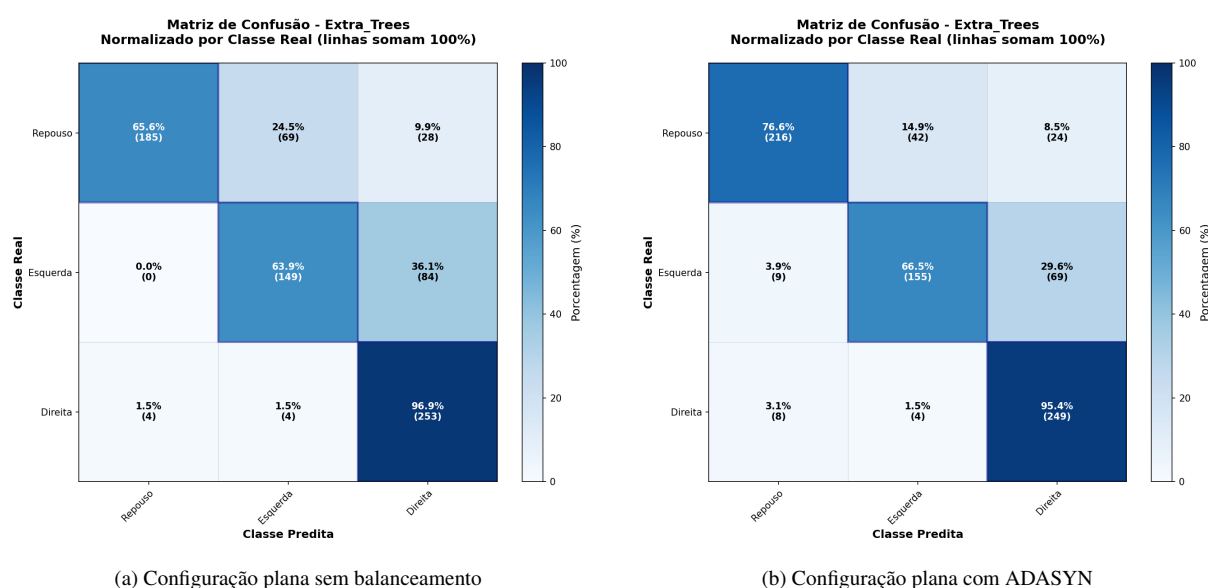
Tabela 6.17 – Métricas de desempenho do Extra Trees por classe na configuração plana sem balanceamento e com ADASYN

Métrica		Sem Balanceamento		ADASYN		Δ
		Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Repouso (Classe 0)	F1	0,785 $\pm$ 0,021	[0,743; 0,826]	0,838 $\pm$ 0,018	[0,802; 0,872]	+0,053
	Precisão	0,979 $\pm$ 0,010	[0,957; 0,995]	0,926 $\pm$ 0,016	[0,896; 0,957]	−0,053
	Sensibilidade	0,656 $\pm$ 0,028	[0,599; 0,713]	0,766 $\pm$ 0,025	[0,716; 0,816]	+0,110
Esquerda (Classe 1)	F1	0,655 $\pm$ 0,025	[0,605; 0,701]	0,714 $\pm$ 0,024	[0,665; 0,760]	+0,059
	Precisão	0,672 $\pm$ 0,026	[0,622; 0,723]	0,771 $\pm$ 0,026	[0,721; 0,824]	+0,099
	Sensibilidade	0,640 $\pm$ 0,032	[0,579; 0,700]	0,665 $\pm$ 0,031	[0,601; 0,725]	+0,025
Direita (Classe 2)	F1	0,809 $\pm$ 0,013	[0,784; 0,834]	0,826 $\pm$ 0,014	[0,799; 0,853]	+0,017
	Precisão	0,694 $\pm$ 0,017	[0,661; 0,730]	0,729 $\pm$ 0,019	[0,693; 0,764]	+0,035
	Sensibilidade	0,969 $\pm$ 0,011	[0,946; 0,989]	0,954 $\pm$ 0,013	[0,927; 0,977]	−0,015

Nota: Δ = ADASYN – Sem Balanceamento.

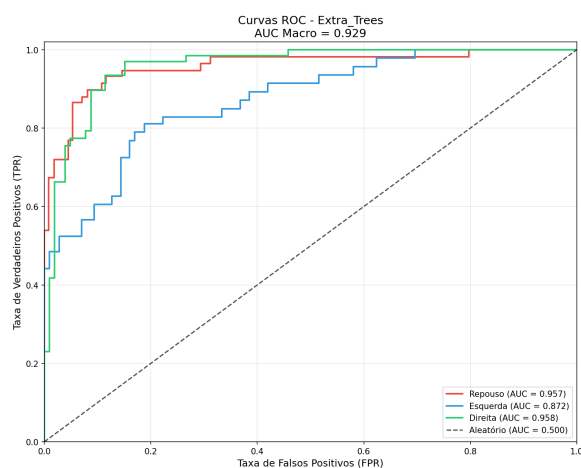
A redistribuição dos acertos entre classes pode ser visualizada nas matrizes de confusão apresentadas na Fig. 6.4. Em particular, a matriz com ADASYN (Figura 6.4b) indica aumento da taxa de acerto da classe Repouso para 76,60% (contra 65,60% sem balanceamento), com redução da confusão com Esquerda de 24,47% para 14,89%. Na classe Esquerda, a taxa de acerto passou de 63,95% para 66,52%, acompanhada de confusão com Repouso de 3,86%, ausente na configuração sem balanceamento. Na classe Direita, a taxa de acerto permaneceu em 95,40%.

Figura 6.4 – Matrizes de confusão percentuais do Extra Trees na configuração plana: (a) sem balanceamento e (b) com ADASYN. Os valores representam a proporção de cada classe verdadeira distribuída entre as classes preditas (normalização por linha).

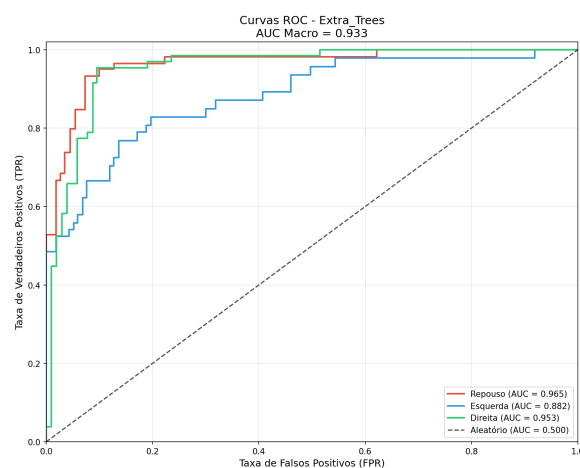


As curvas ROC e PR, apresentadas nas Figs. 6.5 e 6.6, complementam a avaliação ao explicitar o comportamento do classificador em diferentes limiares de decisão. Os valores de área sob as curvas, consolidados na Tab. 6.18, indicam aumento da ROC-AUC para as classes Repouso (+0,8 ponto percentual) e Esquerda (+1,0 ponto percentual), com redução para Direita (−0,5 ponto percentual). Para PR-AUC, observa-se aumento para Esquerda (+2,1 p.p) e redução para Direita (−3,5 p.p), enquanto Repouso variou de 0,944 para 0,946 (+0,2 ponto percentual).

Figura 6.5 – Curvas ROC do Extra Trees na configuração plana: (a) sem balanceamento e (b) com ADASYN, com áreas sob a curva por classe e média macro.

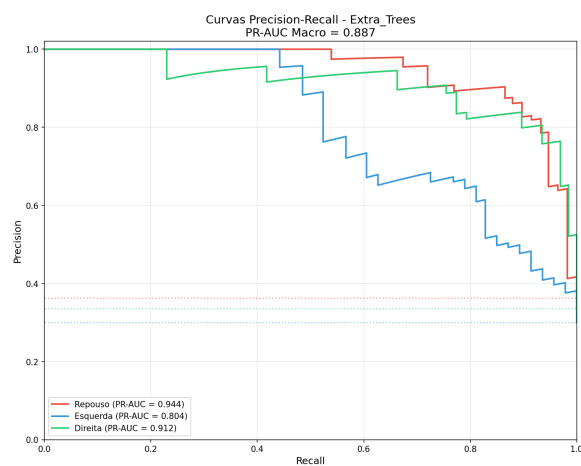


(a) Configuração plana sem balanceamento

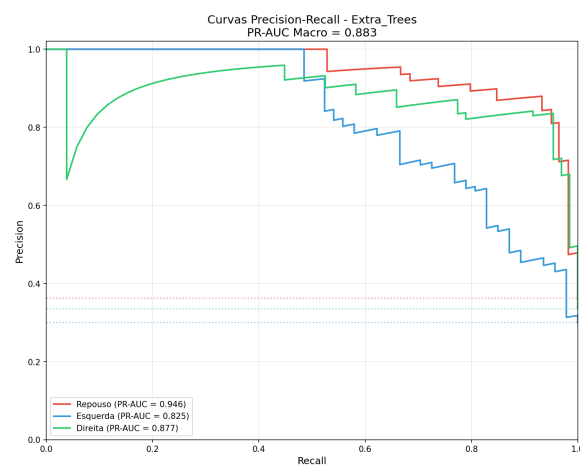


(b) Configuração plana com ADASYN

Figura 6.6 – Curvas Precision–Recall do Extra Trees na configuração plana: (a) sem balanceamento e (b) com ADASYN, com áreas sob a curva por classe e média macro.



(a) Configuração plana sem balanceamento



(b) Configuração plana com ADASYN

Tabela 6.18 – Áreas sob as curvas ROC e PR por classe — Extra Trees (plana): sem balanceamento versus ADASYN

Classe	ROC-AUC			PR-AUC		
	Sem Bal.	ADASYN	Δ	Sem Bal.	ADASYN	Δ
Repouso	0,957	0,965	+0,008	0,944	0,946	+0,002
Esquerda	0,872	0,882	+0,010	0,804	0,825	+0,021
Direita	0,958	0,953	−0,005	0,912	0,877	−0,035
Macro	0,929	0,934	+0,005	0,887	0,883	−0,004

Nota: Δ = ADASYN – Sem Balanceamento.

A calibração foi avaliada pelas métricas ECE e MCE, cujos valores comparativos constam na Tab. 6.19. Observa-se redução do ECE em todas as classes (Repouso: 0,098→0,083; Esquerda: 0,083→0,052; Direita: 0,123→0,084). Para o MCE, a classe Repouso apresentou redução de 0,478 para 0,349, enquanto a classe Direita manteve valor próximo (0,576 para 0,566). A mudança do intervalo associado ao MCE em Repouso e Esquerda para [0,6–0,7] indica deslocamento da região de maior discrepância máxima em relação à configuração sem balanceamento.

Tabela 6.19 – Métricas de calibração por classe — Extra Trees (plana): sem balanceamento versus ADASYN

Classe	ECE		MCE		Intervalo com MCE	
	Sem Bal.	ADASYN	Sem Bal.	ADASYN	Sem Bal.	ADASYN
Repouso	0,098	0,083	0,478	0,349	[0,5–0,6]	[0,6–0,7]
Esquerda	0,083	0,052	0,251	0,271	[0,5–0,6]	[0,6–0,7]
Direita	0,123	0,084	0,576	0,566	[0,5–0,6]	[0,5–0,6]

6.2.8 Efeito do SMOTE na Configuração Cascata

Na arquitetura em cascata com reamostragem SMOTE, a primeira etapa binária (Movimento versus Repouso) foi modelada por um classificador Extra Trees com 300 estimadores, profundidade máxima de 10 níveis e ponderação por subamostra. O particionamento utilizou fração de atributos igual a $\sqrt{p}$, com mínimo de 20 amostras para divisão interna e 10 amostras por folha, além da seleção de 33 atributos. O escore médio de validação cruzada obtido nesta etapa foi $F1_{\text{macro}} = 0,626$. O limiar de decisão otimizado foi 0,62, resultando em $F1_{\text{macro}} = 0,788$ contra 0,768 do limiar padrão, correspondendo a variação relativa de 2,53%.

Na segunda etapa binária (Esquerda versus Direita), treinada sob o cenário com classificador perfeito na primeira etapa, o modelo utilizou 300 estimadores, profundidade máxima de 12 níveis, fração de atributos de 0,3 e seleção de 37 atributos. O escore médio de validação cruzada atingiu $F1_{\text{macro}} = 0,993$, caracterizando o limite superior teórico associado ao desempenho da segunda etapa quando não há propagação de erro da etapa anterior.

Conforme apresentado na Tab. 6.20, a aplicação do SMOTE na configuração em cascata elevou a acurácia de 0,752 para 0,799 (+4,7 p.p), o F1-macro de 0,745 para 0,796 (+5,1 p.p) e o MCC de 0,645 para 0,703 (+5,8 p.p). As métricas probabilísticas também variaram: a ROC-

AUC macro passou de 0,914 para 0,921 (+0,7 ponto percentual) e a PR-AUC macro de 0,862 para 0,882 (+2,0 p.p). O Brier Score reduziu de 0,115 para 0,106.

Tabela 6.20 – Métricas de desempenho global do Extra Trees na configuração em cascata sem balanceamento e com SMOTE, com intervalos de confiança *bootstrap* de 95%

Métrica	Sem Balanceamento		SMOTE		Δ
	Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Acurácia	0,752 $\pm$ 0,014	[0,723; 0,780]	0,799 $\pm$ 0,014	[0,772; 0,826]	+0,047
Acurácia Balanceada	0,756 $\pm$ 0,014	[0,727; 0,784]	0,798 $\pm$ 0,014	[0,769; 0,825]	+0,042
F1-macro	0,745 $\pm$ 0,015	[0,714; 0,774]	0,796 $\pm$ 0,015	[0,768; 0,823]	+0,051
F1-weighted	0,746 $\pm$ 0,015	[0,715; 0,776]	0,799 $\pm$ 0,014	[0,771; 0,826]	+0,053
Precisão (macro)	0,776 $\pm$ 0,013	[0,751; 0,800]	0,803 $\pm$ 0,013	[0,776; 0,830]	+0,027
Recall (macro)	0,756 $\pm$ 0,014	[0,727; 0,784]	0,798 $\pm$ 0,014	[0,769; 0,825]	+0,042
MCC	0,645 $\pm$ 0,020	[0,605; 0,685]	0,703 $\pm$ 0,020	[0,662; 0,744]	+0,058
ROC-AUC (macro)	0,914 $\pm$ 0,008	[0,897; 0,929]	0,921 $\pm$ 0,008	[0,906; 0,936]	+0,007
PR-AUC (macro)	0,862 $\pm$ 0,013	[0,835; 0,887]	0,882 $\pm$ 0,012	[0,860; 0,905]	+0,020
Brier Score	0,115 $\pm$ 0,006	[0,104; 0,127]	0,106 $\pm$ 0,005	[0,094; 0,117]	−0,009

Nota: Δ = SMOTE – Sem Balanceamento.

A Tab. 6.21 apresenta as métricas por classe para o arranjo em cascata. Na classe Repouso, a sensibilidade aumentou de 0,575 para 0,753 (+17,8 p.p) e o F1-score passou de 0,714 para 0,822 (+10,8 p.p), acompanhados de redução na precisão de 0,942 para 0,906 (−3,6 p.p). Essa alteração ocorre em conjunto com o ajuste do limiar de decisão na primeira etapa para 0,62, associado à redução da taxa de falsos positivos de 42,55% para 24,82% (70 de 282 amostras de Repouso erroneamente classificadas como Movimento). Na classe Esquerda, a precisão aumentou de 0,651 para 0,739 (+8,8 p.p), com sensibilidade praticamente inalterada em 0,725. Na classe Direita, a sensibilidade reduziu de 0,966 para 0,916 (−5,0 p.p), enquanto a precisão aumentou de 0,733 para 0,764 (+3,1 p.p).

Tabela 6.21 – Métricas de desempenho do Extra Trees por classe na configuração em cascata sem balanceamento e com SMOTE

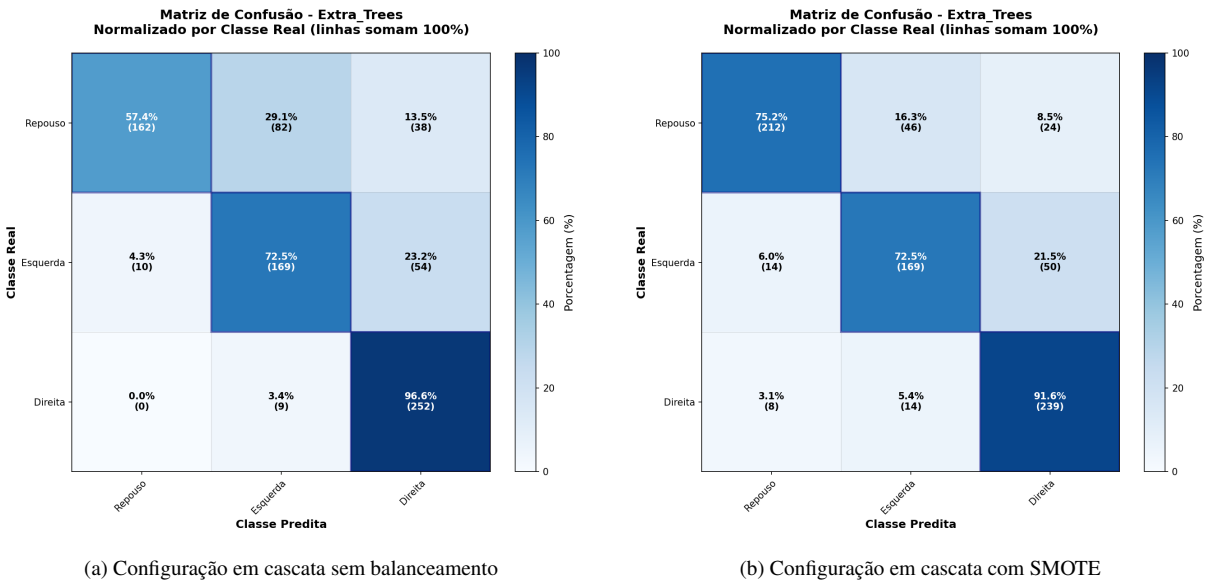
Métrica		Sem Balanceamento		SMOTE		Δ
		Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Repouso (Classe 0)	F1	0,714 $\pm$ 0,024	[0,664; 0,759]	0,822 $\pm$ 0,018	[0,783; 0,856]	+0,108
	Precisão	0,942 $\pm$ 0,017	[0,906; 0,974]	0,906 $\pm$ 0,016	[0,872; 0,940]	−0,036
	Sensibilidade	0,575 $\pm$ 0,029	[0,514; 0,631]	0,753 $\pm$ 0,026	[0,699; 0,801]	+0,178
Esquerda (Classe 1)	F1	0,686 $\pm$ 0,021	[0,646; 0,728]	0,732 $\pm$ 0,021	[0,689; 0,773]	+0,046
	Precisão	0,651 $\pm$ 0,023	[0,607; 0,696]	0,739 $\pm$ 0,024	[0,691; 0,786]	+0,088
	Sensibilidade	0,726 $\pm$ 0,029	[0,670; 0,781]	0,725 $\pm$ 0,029	[0,670; 0,785]	−0,001
Direita (Classe 2)	F1	0,834 $\pm$ 0,013	[0,807; 0,860]	0,833 $\pm$ 0,015	[0,804; 0,861]	−0,001
	Precisão	0,733 $\pm$ 0,019	[0,698; 0,771]	0,764 $\pm$ 0,019	[0,727; 0,803]	+0,031
	Sensibilidade	0,966 $\pm$ 0,011	[0,943; 0,985]	0,916 $\pm$ 0,016	[0,881; 0,946]	−0,050

Nota: Δ = SMOTE – Sem Balanceamento.

A decomposição de erros do modelo em cascata com SMOTE indica erro total de 156 amostras (20,10%) contra 193 (24,87%) sem balanceamento, correspondendo a uma redução de 37 erros. Os falsos positivos da primeira etapa reduziram de 120 para 70 amostras, enquanto os falsos negativos aumentaram de 10 para 22 amostras. O erro propagado da primeira etapa reduziu de 130 para 92 amostras, representando 59,0% do erro total contra 67,4% sem balanceamento.

As matrizes de confusão apresentadas na Fig. 6.7 permitem identificar como essa redução se distribui entre classes. Na matriz com SMOTE (Figura 6.7b), a classe Repouso apresenta 75,18% de acertos (contra 57,45% sem balanceamento), com redução da confusão com Esquerda de 29,08% para 16,31%. A classe Esquerda mantém 72,53% de acertos, com aumento da confusão com Repouso de 4,29% para 6,01%. A classe Direita apresenta 91,57% de acertos (contra 96,55%), com aumento da confusão com Esquerda de 3,45% para 5,36%.

Figura 6.7 – Matrizes de confusão percentuais do Extra Trees na configuração em cascata: (a) sem balanceamento e (b) com SMOTE. Os valores representam a proporção de cada classe verdadeira distribuída entre as classes preditas (normalização por linha).



As curvas ROC e PR comparativas são apresentadas nas Figs. 6.8 e 6.9. As áreas sob as curvas, consolidadas na Tab. 6.22, indicam aumento da ROC-AUC macro de 0,913 para 0,921 (+0,8 ponto percentual), com variações positivas em todas as classes. A PR-AUC macro variou de 0,861 para 0,882 (+2,1 p.p), com destaque para a classe Direita (0,855→0,903; +4,8 p.p).

Figura 6.8 – Curvas ROC do Extra Trees na configuração em cascata: (a) sem balanceamento e (b) com SMOTE, com áreas sob a curva por classe e média macro.

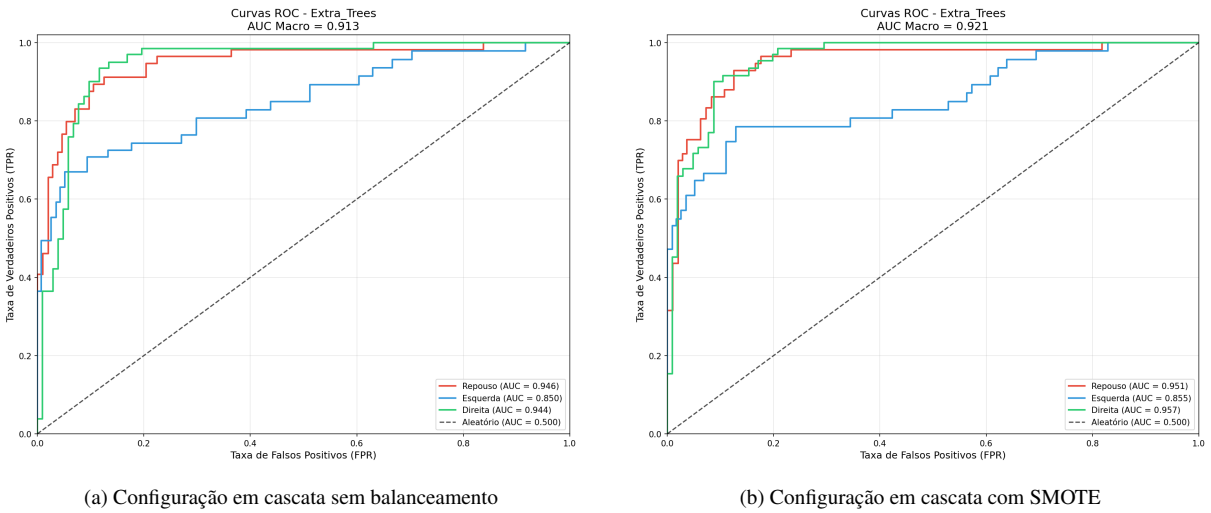
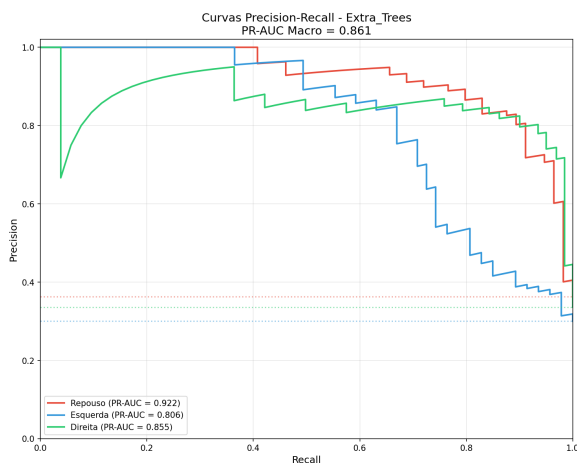
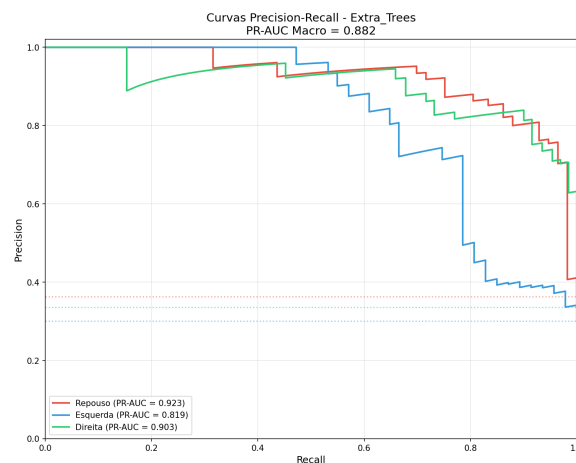


Figura 6.9 – Curvas Precision–Recall do Extra Trees na configuração em cascata: (a) sem balanceamento e (b) com SMOTE, com áreas sob a curva por classe e média macro.



(a) Configuração em cascata sem balanceamento



(b) Configuração em cascata com SMOTE

Tabela 6.22 – Áreas sob as curvas ROC e PR por classe — Extra Trees (cascata): sem balanceamento versus SMOTE

Classe	ROC-AUC			PR-AUC		
	Sem Bal.	SMOTE	Δ	Sem Bal.	SMOTE	Δ
Repouso	0,946	0,951	+0,005	0,922	0,923	+0,001
Esquerda	0,850	0,855	+0,005	0,806	0,819	+0,013
Direita	0,944	0,957	+0,013	0,855	0,903	+0,048
Macro	0,913	0,921	+0,008	0,861	0,882	+0,021

Nota: Δ = SMOTE – Sem Balanceamento.

A calibração é sintetizada na Tab. 6.23. Observa-se redução do ECE nas classes Repouso (0,127→0,105) e Direita (0,131→0,113), com variação mínima em Esquerda (0,100→0,099). Para o MCE, há redução em Repouso (0,559→0,453) e Esquerda (0,300→0,201), enquanto a classe Direita apresenta aumento (0,563→0,641). A variação do intervalo associado ao MCE em Esquerda para [0,2–0,3] indica que a maior discrepância máxima nessa classe ocorre em uma faixa de menor probabilidade predita quando o SMOTE é aplicado.

Tabela 6.23 – Métricas de calibração por classe — Extra Trees (cascata): sem balanceamento versus SMOTE

Classe	ECE		MCE		Intervalo com MCE	
	Sem Bal.	SMOTE	Sem Bal.	SMOTE	Sem Bal.	SMOTE
Repouso	0,127	0,105	0,559	0,453	[0,4–0,5]	[0,5–0,6]
Esquerda	0,100	0,099	0,300	0,201	[0,5–0,6]	[0,2–0,3]
Direita	0,131	0,113	0,563	0,641	[0,5–0,6]	[0,6–0,7]

6.2.9 Síntese Comparativa

A Tab. 6.24 reúne as métricas globais e sensibilidades por classe para as quatro configurações avaliadas (plana sem balanceamento, plana com ADASYN, cascata sem balanceamento e cascata com SMOTE), permitindo comparação direta entre arquiteturas e estratégias de balanceamento.

Tabela 6.24 – Síntese comparativa das configurações do Extra Trees

Métrica	Plana		Cascata	
	Sem Bal.	ADASYN	Sem Bal.	SMOTE
Acurácia	0,757	0,799	0,752	0,799
F1-macro	0,750	0,793	0,745	0,796
MCC	0,651	0,707	0,645	0,703
ROC-AUC (macro)	0,929	0,934	0,914	0,921
PR-AUC (macro)	0,887	0,883	0,862	0,882
Brier Score	0,109	0,096	0,115	0,106
Sens. Repouso	0,656	0,766	0,575	0,753
Sens. Esquerda	0,640	0,665	0,726	0,725
Sens. Direita	0,969	0,954	0,966	0,916

Nota: Valores em negrito indicam o melhor desempenho em cada métrica. Sens. = Sensibilidade.

A síntese indica que as técnicas de reamostragem elevaram as métricas globais em ambas as arquiteturas. Na configuração plana, o ADASYN elevou a acurácia em 4,2 p.p e o F1-macro em 4,3 p.p em relação à configuração sem balanceamento. Na configuração em cascata, o SMOTE elevou a acurácia em 4,7 p.p e o F1-macro em 5,1 p.p. Esses incrementos são compatíveis com a interpretação de que o desbalanceamento de classes limita o desempenho do Extra Trees no problema analisado.

A comparação entre arquiteturas, mantendo o mesmo regime de balanceamento, indica que a configuração plana apresenta valores maiores de ROC-AUC macro tanto sem balancea-

mento (0,929 versus 0,914) quanto com balanceamento (0,934 versus 0,921). Nas métricas de decisão, a acurácia atinge 0,799 tanto no plano com ADASYN quanto no cascata com SMOTE, enquanto o F1-macro máximo entre as quatro configurações ocorre no cascata com SMOTE (0,796), seguido pelo plano com ADASYN (0,793). Para PR-AUC macro, o maior valor ocorre na configuração plana sem balanceamento (0,887), com variações de pequena magnitude nas configurações balanceadas (0,883 no plano com ADASYN e 0,882 no cascata com SMOTE).

A interpretação das métricas por classe evidencia que a reamostragem alterou o compromisso entre precisão e sensibilidade, em especial na classe Repouso. No plano com ADASYN, a sensibilidade de Repouso aumentou para 0,766, enquanto a precisão reduziu (Tab. 6.17); no cascata com SMOTE, a sensibilidade de Repouso aumentou para 0,753, com redução de precisão e influência explícita do ajuste de limiar na etapa 1 (Tab. 6.21). Para a classe Esquerda, a sensibilidade manteve valores próximos entre as configurações balanceadas (0,665 no plano com ADASYN e 0,725 no cascata com SMOTE), enquanto a classe Direita apresentou redução de sensibilidade mais pronunciada no cascata com SMOTE (0,916), apesar do aumento de PR-AUC nessa classe (Tab. 6.22).

Quanto à análise de adequação no contexto do problema de classificação de imaginação motora, os valores observados de F1-macro (0,750 a 0,796) situam o desempenho acima do desempenho esperado ao acaso para três classes equiprováveis (aproximadamente 0,333). A ROC-AUC macro (0,913 a 0,934) caracteriza capacidade de ordenação das probabilidades preditas ao longo de diferentes limiares. O Brier Score (0,096 a 0,115) resume o erro quadrático médio associado às probabilidades, sendo menor na configuração plana com ADASYN (0,096).

Ainda assim, a sensibilidade da classe Esquerda permanece limitada a 0,665 na configuração plana com ADASYN e a 0,725 no cascata com SMOTE, o que implica proporção não desprezível de intenções de movimento para a esquerda classificadas incorretamente. Além disso, a presença de intervalos de confiança com largura superior a 0,10 para sensibilidades de classes individuais, reportada na análise sem balanceamento desta seção, indica que a incerteza das estimativas pontuais deve ser considerada na comparação fina entre configurações.

Em aplicações de controle motor em tempo real, uma taxa de erro da ordem de 20% (associada a acurácia próxima de 0,80) pode demandar mecanismos adicionais, como confirmação de comandos ou estratégias de correção de erro, dependendo do custo relativo entre falsos positivos e falsos negativos. Por fim, a calibração imperfeita das probabilidades, evidenciada por

valores de MCE superiores a 0,30 em múltiplas classes e configurações (Tabs. 6.15, 6.19 e 6.23), indica que o uso direto das probabilidades como estimativas de confiança requer cautela, sobretudo em faixas de probabilidade associadas aos maiores desvios máximos.

6.3 Random Forest

Na configuração plana multiclasse sem balanceamento, a configuração selecionada empregou 200 árvores, profundidade máxima de 12 níveis, mínimo de 30 amostras por folha, mínimo de 10 amostras para divisão, amostragem de 90% dos dados por estimador, seleção de atributos por $\log_2$ do número total, ponderação de classes balanceada por subamostragem e seleção de 31 atributos via seleção univariada com função de pontuação ANOVA F. O melhor escore médio em validação cruzada foi $F1_{\text{macro}} = 0,517$, valor acima do desempenho esperado ao acaso (aproximadamente 0,333 para três classes equiprováveis), compatível com o patamar observado no conjunto de teste.

Na arquitetura em cascata sem balanceamento, a primeira etapa (Movimento versus Repouso) utilizou 300 árvores, profundidade máxima de 8 níveis, mínimo de 10 amostras por folha, mínimo de 30 amostras para divisão, amostragem de 70% dos dados, seleção de atributos por raiz quadrada e seleção de 10 atributos. O escore médio em validação cruzada desta etapa foi $F1_{\text{macro}} = 0,648$. A segunda etapa (Esquerda versus Direita), treinada apenas com amostras de movimento, empregou 200 árvores, profundidade máxima de 5 níveis, mínimo de 10 amostras por folha, mínimo de 30 amostras para divisão, amostragem de 90% dos dados, seleção de atributos por raiz quadrada e seleção de 35 atributos, resultando em $F1_{\text{macro}} = 0,720$ em validação cruzada no cenário com oráculo.

6.3.1 Desempenho Global sem Balanceamento

Conforme apresentado na Tab. 6.25 observa-se que a arquitetura em cascata apresentou valores maiores nas métricas baseadas em predições discretas: a acurácia variou de 0,763 para 0,783 (+0,020) e o $F1_{\text{macro}}$ variou de 0,756 para 0,777 (+0,021). O coeficiente de correlação de Matthews também aumentou de 0,654 para 0,678 (+0,024), em consonância com o aumento das métricas de decisão.

Tabela 6.25 – Métricas de desempenho global do Random Forest nas configurações plana e cascata sem balanceamento, com intervalos de confiança *bootstrap* de 95%

Métrica	Plano		Cascata		Δ
	Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Acurácia	0,763 $\pm$ 0,015	[0,735; 0,793]	0,783 $\pm$ 0,014	[0,755; 0,811]	+0,020
Acurácia Balanceada	0,761 $\pm$ 0,015	[0,732; 0,791]	0,779 $\pm$ 0,014	[0,751; 0,807]	+0,018
F1-macro	0,756 $\pm$ 0,015	[0,726; 0,787]	0,777 $\pm$ 0,015	[0,747; 0,805]	+0,021
F1-weighted	0,759 $\pm$ 0,015	[0,730; 0,790]	0,781 $\pm$ 0,014	[0,752; 0,809]	+0,022
Precisão (macro)	0,772 $\pm$ 0,014	[0,745; 0,801]	0,783 $\pm$ 0,014	[0,754; 0,811]	+0,011
Recall (macro)	0,761 $\pm$ 0,015	[0,732; 0,791]	0,779 $\pm$ 0,014	[0,751; 0,807]	+0,018
MCC	0,654 $\pm$ 0,021	[0,613; 0,696]	0,678 $\pm$ 0,021	[0,636; 0,719]	+0,024
ROC-AUC (macro)	0,919 $\pm$ 0,008	[0,903; 0,933]	0,901 $\pm$ 0,009	[0,882; 0,918]	−0,018
PR-AUC (macro)	0,868 $\pm$ 0,012	[0,844; 0,891]	0,830 $\pm$ 0,015	[0,800; 0,860]	−0,038
Brier Score	0,111 $\pm$ 0,006	[0,099; 0,122]	0,120 $\pm$ 0,006	[0,109; 0,132]	+0,009

Nota: Δ = Cascata – Plano. IC = Intervalo de Confiança.

A largura dos intervalos de confiança foi utilizada como indicador de incerteza amostral, tomando-se 0,10 como referência operacional. Na configuração plana, excederam esse valor: sensibilidade da classe Esquerda (largura 0,124), sensibilidade da classe Repouso (largura 0,106) e precisão da classe Esquerda (largura 0,103). Na configuração em cascata, excederam esse valor: sensibilidade da classe Esquerda (0,120) e precisão da classe Esquerda (0,104).

Embora as métricas de decisão tenham apresentado aumento na arquitetura em cascata, observa-se, na Tab. 6.25, redução nas métricas baseadas em probabilidades: a ROC-AUC macro variou de 0,919 para 0,901 (−0,018) e a PR-AUC macro variou de 0,868 para 0,830 (−0,038). Essa diferença é compatível com o mecanismo de composição de probabilidades na arquitetura em cascata, em que $P(\text{classe})$ é obtida a partir da combinação das saídas de $E1$ e $E2$, o que pode alterar a distribuição de escores utilizada em métricas de ordenação.

6.3.2 Desempenho por Classe sem Balanceamento

Conforme detalhado na Tab. 6.26, a arquitetura em cascata alterou o compromisso entre precisão e sensibilidade de forma heterogênea entre classes. Para Repouso, a sensibilidade variou de 0,685 para 0,770 (+0,085) e o F1 variou de 0,778 para 0,812 (+0,034), enquanto a precisão variou de 0,901 para 0,861 (−0,040). Para Esquerda, o F1 variou de 0,671 para 0,691 (+0,020) e a precisão variou de 0,696 para 0,735 (+0,039), com sensibilidade praticamente

inalterada (0,648 para 0,652). Para Direita, a sensibilidade variou de 0,950 para 0,916 (−0,034), com aumento de precisão (0,720 para 0,755).

Tabela 6.26 – Métricas de desempenho do Random Forest por classe nas configurações plana e cascata sem balanceamento

Métrica		Plano		Cascata		Δ
		Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Repouso (Classe 0)	F1	0,778 $\pm$ 0,020	[0,737; 0,818]	0,812 $\pm$ 0,018	[0,776; 0,846]	+0,034
	Precisão	0,901 $\pm$ 0,019	[0,864; 0,938]	0,861 $\pm$ 0,019	[0,823; 0,900]	−0,040
	Recall	0,685 $\pm$ 0,028	[0,631; 0,738]	0,770 $\pm$ 0,025	[0,720; 0,819]	+0,085
Esquerda (Classe 1)	F1	0,671 $\pm$ 0,024	[0,623; 0,718]	0,691 $\pm$ 0,024	[0,643; 0,736]	+0,020
	Precisão	0,696 $\pm$ 0,026	[0,646; 0,749]	0,735 $\pm$ 0,026	[0,683; 0,787]	+0,039
	Recall	0,648 $\pm$ 0,031	[0,584; 0,708]	0,652 $\pm$ 0,031	[0,592; 0,712]	+0,004
Direita (Classe 2)	F1	0,819 $\pm$ 0,014	[0,792; 0,846]	0,827 $\pm$ 0,015	[0,798; 0,855]	+0,008
	Precisão	0,720 $\pm$ 0,019	[0,683; 0,757]	0,755 $\pm$ 0,019	[0,717; 0,793]	+0,035
	Recall	0,950 $\pm$ 0,013	[0,923; 0,973]	0,916 $\pm$ 0,017	[0,881; 0,946]	−0,034

Nota: Δ = Cascata − Plano.

6.3.3 Análise por Etapa da Configuração em Cascata sem Balanceamento

Conforme apresentado na Tab. 6.27, a avaliação por etapa descreve o desempenho da etapa 1 (Movimento versus Repouso) e da etapa 2 condicionada ao encaminhamento pela etapa 1. Na etapa 1, a sensibilidade para Movimento foi 0,929 e a ROC-AUC foi 0,944. A otimização do limiar foi conduzida em validação cruzada estratificada, usando $F1_{\text{macro}}$ como critério; o limiar selecionado foi 0,64, resultando em $F1_{\text{macro}} = 0,714$ em comparação com 0,679 no limiar padrão (0,50), correspondendo a variação relativa de 5,16%.

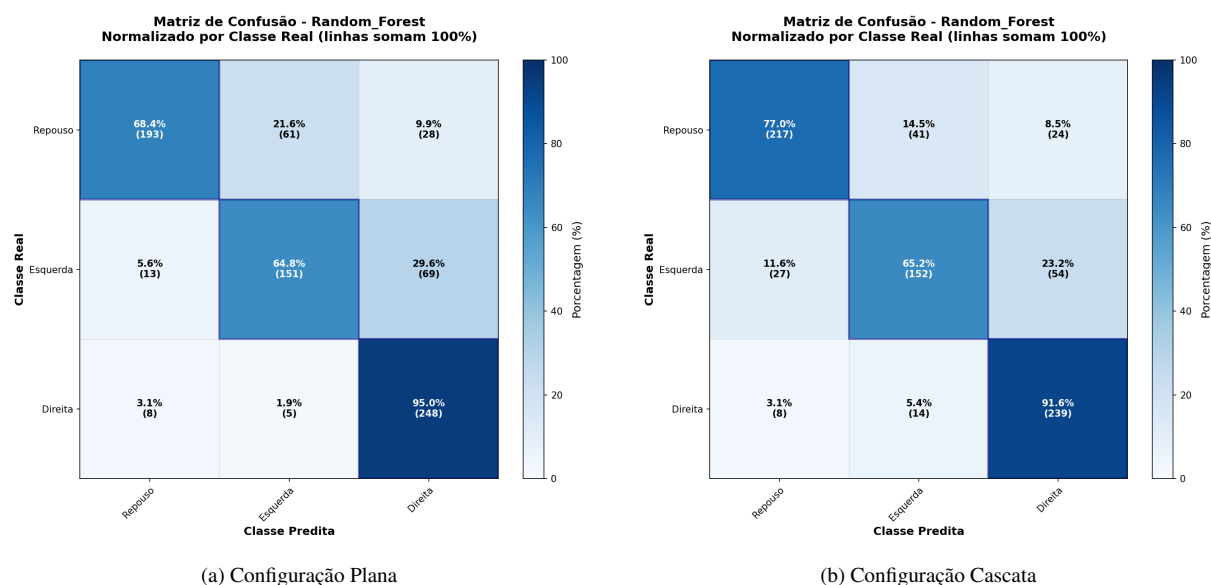
Tabela 6.27 – Métricas por etapa da configuração em cascata – Random Forest

Métrica	Etapa 1		E2 Condicionada	
	Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%
Acurácia	0,871 $\pm$ 0,012	[0,848; 0,894]	0,852 $\pm$ 0,017	[0,819; 0,885]
Acurácia Bal.	0,849 $\pm$ 0,013	[0,823; 0,875]	0,841 $\pm$ 0,017	[0,806; 0,873]
F1	0,902 $\pm$ 0,010	[0,882; 0,920]	0,875 $\pm$ 0,015	[0,843; 0,904]
Precisão	0,876 $\pm$ 0,014	[0,848; 0,903]	0,816 $\pm$ 0,023	[0,769; 0,859]
Recall	0,929 $\pm$ 0,011	[0,907; 0,951]	0,945 $\pm$ 0,014	[0,915; 0,970]
MCC	0,718 $\pm$ 0,025	[0,667; 0,766]	0,707 $\pm$ 0,032	[0,642; 0,767]
ROC-AUC	0,944 $\pm$ 0,008	[0,927; 0,960]	0,944 $\pm$ 0,010	[0,923; 0,962]
PR-AUC	0,969 $\pm$ 0,006	[0,956; 0,979]	0,951 $\pm$ 0,010	[0,928; 0,969]
Brier Score	0,121 $\pm$ 0,006	[0,109; 0,133]	0,121 $\pm$ 0,014	[0,095; 0,148]

6.3.4 Matriz de Confusão sem Balanceamento

Conforme apresentado na Fig. 6.10, as matrizes de confusão percentuais com normalização por linha permitem comparar a redistribuição de erros entre as configurações plana e em cascata. Na configuração plana (Figura 6.10a), a classe Esquerda apresentou 64,81% de acertos, com confusão para Direita (29,61%) e para Repouso (5,58%). A classe Direita apresentou 95,02% de acertos, com confusão para Repouso (3,07%) e para Esquerda (1,92%). A classe Repouso apresentou 68,44% de acertos, com confusão para Esquerda (21,63%) e para Direita (9,93%).

Figura 6.10 – Matrizes de confusão percentuais do Random Forest nas configurações plana (a) e cascata (b) sem balanceamento. Os valores representam a proporção de cada classe verdadeira distribuída entre as classes preditas (normalização por linha).



Na configuração em cascata (Figura 6.10b), a classe Repouso apresentou 76,95% de acertos, com confusão para Esquerda (14,54%) e para Direita (8,51%). A classe Esquerda apresentou 65,24% de acertos, com confusão para Direita (23,18%) e para Repouso (11,59%). A classe Direita apresentou 91,57% de acertos, com confusão para Esquerda (5,36%) e para Repouso (3,07%).

6.3.5 Análise das Curvas ROC e PR sem Balanceamento

Conforme indicado na Tab. 6.28, a configuração plana apresentou ROC-AUC macro de 0,918 e PR-AUC macro de 0,868, enquanto a configuração em cascata apresentou 0,901

e 0,830, respectivamente. A redução mais pronunciada foi observada na PR-AUC da classe Direita (0,886 para 0,794), apesar de a sensibilidade desta classe permanecer acima de 0,90 em ambas as configurações (Tab. 6.26).

Tabela 6.28 – Áreas sob as curvas ROC e PR por classe e configuração — Random Forest sem balanceamento

Classe	ROC-AUC			PR-AUC		
	Plano	Cascata	Δ	Plano	Cascata	Δ
Repouso	0,944	0,944	+0,000	0,917	0,893	−0,024
Esquerda	0,864	0,841	−0,023	0,800	0,802	+0,002
Direita	0,947	0,917	−0,030	0,886	0,794	−0,092
Macro	0,918	0,901	−0,018	0,868	0,830	−0,038

Nota: Δ = Cascata − Plano.

6.3.6 Análise de Calibração sem Balanceamento

Conforme apresentado na Tab. 6.29, na configuração plana os valores de ECE foram 0,097 (Repouso), 0,056 (Esquerda) e 0,092 (Direita). Na configuração em cascata, os valores foram 0,163, 0,109 e 0,120, respectivamente, indicando aumento do desalinhamento médio entre probabilidades preditas e frequências observadas.

Tabela 6.29 – Métricas de calibração por classe e configuração — Random Forest sem balanceamento

Classe	ECE		MCE		Bins com MCE	
	Plano	Cascata	Plano	Cascata	Plano	Cascata
Repouso	0,097	0,163	0,460	0,472	[0,5–0,6]	[0,3–0,4]
Esquerda	0,056	0,109	0,550	0,340	[0,5–0,6]	[0,5–0,6]
Direita	0,092	0,120	0,577	0,504	[0,4–0,5]	[0,6–0,7]

Observa-se, na Tab. 6.29, que os valores de MCE permaneceram acima de 0,30 em ambas as configurações, o que indica a existência de faixas específicas de probabilidade nas quais a discrepância máxima entre confiança predita e frequência observada é acentuada. Na arquitetura em cascata, o aumento do ECE em todas as classes é compatível com o mecanismo de composição de probabilidades, em que a etapa 2 é treinada em um subconjunto (movimento) e é posteriormente aplicada a um conjunto condicionado que inclui amostras encaminhadas incorretamente pela etapa 1.

6.3.7 Efeito do SMOTE na Configuração Plana

Na configuração plana multiclasse com reamostragem SMOTE, a configuração selecionada utilizou 100 árvores, profundidade máxima de 12 níveis, ponderação de classes balanceada por subamostragem, amostragem de 80% das observações por estimador, seleção de atributos $\log_2(p)$, mínimo de 30 amostras para divisão interna e 10 amostras por folha, além da seleção de 34 atributos. O melhor escore médio em validação cruzada foi $F1_{\text{macro}} = 0,648$, acima do obtido sem balanceamento (0,517).

Conforme apresentado na Tab. 6.30, o SMOTE elevou a acurácia de 0,763 para 0,775 (+0,012), o $F1_{\text{macro}}$ de 0,756 para 0,769 (+0,013) e o MCC de 0,654 para 0,670 (+0,016). As métricas baseadas em probabilidades também variaram: a ROC-AUC macro aumentou de 0,919 para 0,922 (+0,003) e a PR-AUC macro aumentou de 0,868 para 0,873 (+0,005). O Brier Score reduziu de 0,111 para 0,105.

Tabela 6.30 – Métricas de desempenho global do Random Forest na configuração plana sem balanceamento e com SMOTE, com intervalos de confiança *bootstrap* de 95%

Métrica	Sem Balanceamento		SMOTE		Δ
	Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Acurácia	0,763 $\pm$ 0,015	[0,735; 0,793]	0,775 $\pm$ 0,014	[0,745; 0,803]	+0,012
Acurácia Balanceada	0,761 $\pm$ 0,015	[0,732; 0,791]	0,772 $\pm$ 0,015	[0,743; 0,801]	+0,011
F1-macro	0,756 $\pm$ 0,015	[0,726; 0,787]	0,769 $\pm$ 0,015	[0,739; 0,798]	+0,013
F1-weighted	0,759 $\pm$ 0,015	[0,730; 0,790]	0,773 $\pm$ 0,015	[0,743; 0,802]	+0,014
Precisão (macro)	0,772 $\pm$ 0,014	[0,745; 0,801]	0,784 $\pm$ 0,014	[0,757; 0,811]	+0,012
Recall (macro)	0,761 $\pm$ 0,015	[0,732; 0,791]	0,772 $\pm$ 0,015	[0,743; 0,801]	+0,011
MCC	0,654 $\pm$ 0,021	[0,613; 0,696]	0,670 $\pm$ 0,021	[0,629; 0,711]	+0,016
ROC-AUC (macro)	0,919 $\pm$ 0,008	[0,903; 0,933]	0,922 $\pm$ 0,008	[0,907; 0,937]	+0,003
PR-AUC (macro)	0,868 $\pm$ 0,012	[0,844; 0,891]	0,873 $\pm$ 0,012	[0,850; 0,896]	+0,005
Brier Score	0,111 $\pm$ 0,006	[0,099; 0,122]	0,105 $\pm$ 0,005	[0,094; 0,115]	−0,006

Nota: Δ = SMOTE – Sem Balanceamento.

O detalhamento por classe, apresentado na Tab. 6.31, indica que as maiores variações ocorreram em Repouso: sensibilidade de 0,685 para 0,717 (+0,032), precisão de 0,901 para 0,922 (+0,021) e F1 de 0,778 para 0,806 (+0,028). Para Esquerda, observou-se aumento em sensibilidade (0,648 para 0,665) e em precisão (0,696 para 0,705). Para Direita, observou-se redução em sensibilidade (0,950 para 0,935).

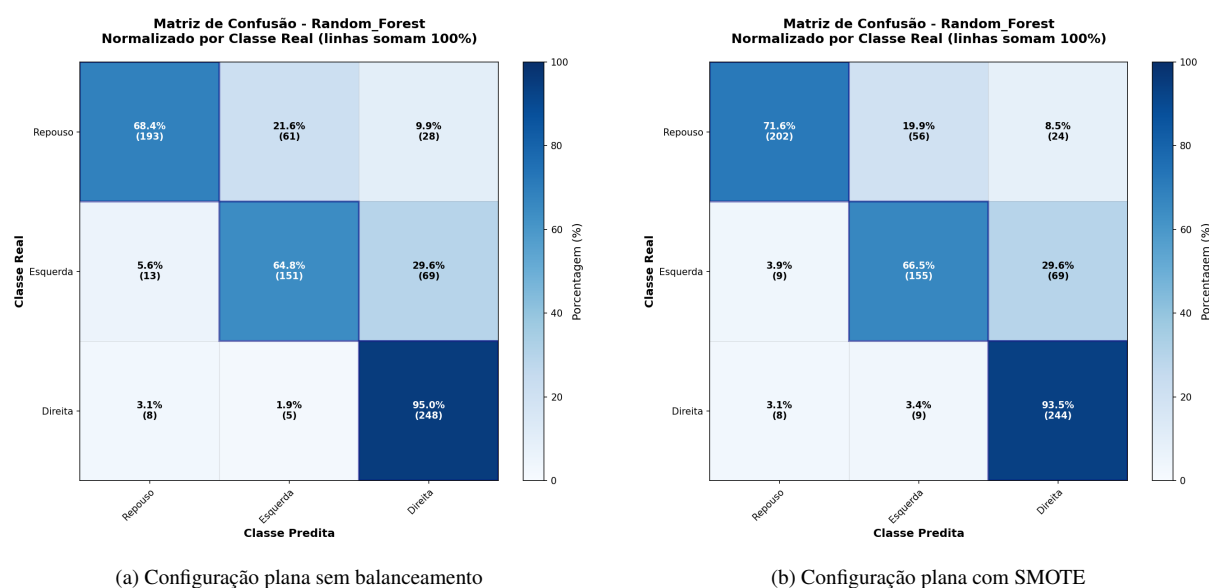
Tabela 6.31 – Métricas de desempenho do Random Forest por classe na configuração plana sem balanceamento e com SMOTE

Métrica		Sem Balanceamento		SMOTE		Δ
		Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Repouso (Classe 0)	F1	0,778 $\pm$ 0,020	[0,737; 0,818]	0,806 $\pm$ 0,019	[0,767; 0,844]	+0,028
	Precisão	0,901 $\pm$ 0,019	[0,864; 0,938]	0,922 $\pm$ 0,017	[0,888; 0,954]	+0,021
	Recall	0,685 $\pm$ 0,028	[0,631; 0,738]	0,717 $\pm$ 0,027	[0,663; 0,770]	+0,032
Esquerda (Classe 1)	F1	0,671 $\pm$ 0,024	[0,623; 0,718]	0,684 $\pm$ 0,024	[0,636; 0,729]	+0,013
	Precisão	0,696 $\pm$ 0,026	[0,646; 0,749]	0,705 $\pm$ 0,025	[0,655; 0,756]	+0,009
	Recall	0,648 $\pm$ 0,031	[0,584; 0,708]	0,665 $\pm$ 0,031	[0,601; 0,725]	+0,017
Direita (Classe 2)	F1	0,819 $\pm$ 0,014	[0,792; 0,846]	0,817 $\pm$ 0,014	[0,789; 0,843]	−0,002
	Precisão	0,720 $\pm$ 0,019	[0,683; 0,757]	0,725 $\pm$ 0,019	[0,688; 0,761]	+0,005
	Recall	0,950 $\pm$ 0,013	[0,923; 0,973]	0,935 $\pm$ 0,015	[0,904; 0,966]	−0,015

Nota: Δ = SMOTE – Sem Balanceamento.

Conforme apresentado na Fig. 6.11, as matrizes de confusão percentuais permitem localizar as variações associadas ao SMOTE no padrão de erros por classe.

Figura 6.11 – Matrizes de confusão percentuais do Random Forest na configuração plana: (a) sem balanceamento e (b) com SMOTE. Os valores representam a proporção de cada classe verdadeira distribuída entre as classes preditas (normalização por linha).



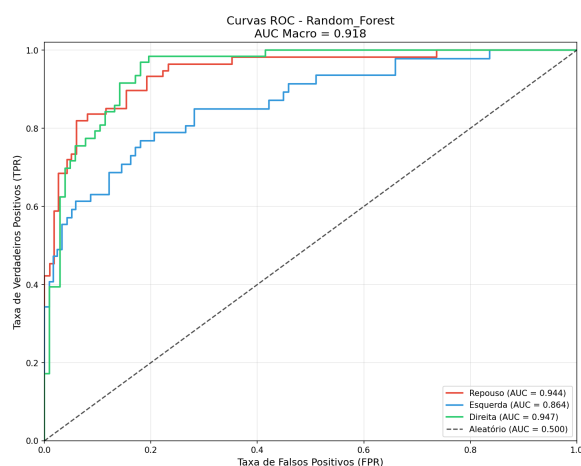
A comparação de áreas sob as curvas ROC e PR é apresentada na Tab. 6.32, em correspondência com as curvas das Figs. 6.12 e 6.13. Observa-se aumento em ROC-AUC e PR-AUC para Repouso e Esquerda, com variações negativas de pequena magnitude para Direita.

Tabela 6.32 – Áreas sob as curvas ROC e PR por classe — Random Forest (plana): sem balanceamento versus SMOTE

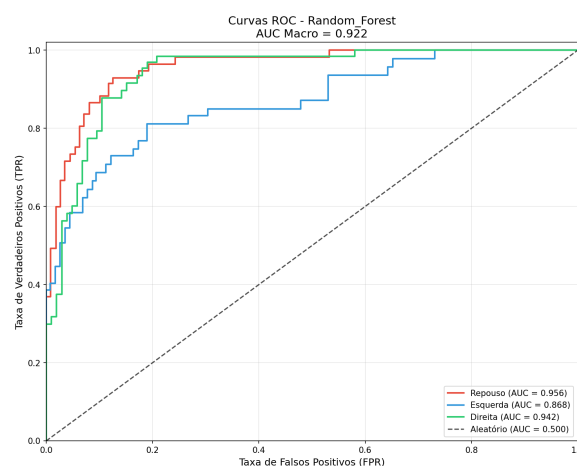
Classe	ROC-AUC			PR-AUC		
	Sem Bal.	SMOTE	Δ	Sem Bal.	SMOTE	Δ
Repouso	0,944	0,956	+0,012	0,917	0,928	+0,011
Esquerda	0,864	0,868	+0,004	0,800	0,808	+0,008
Direita	0,947	0,942	−0,005	0,886	0,882	−0,004
Macro	0,918	0,922	+0,004	0,868	0,873	+0,005

Nota: Δ = SMOTE – Sem Balanceamento.

Figura 6.12 – Curvas ROC do Random Forest na configuração plana: (a) sem balanceamento e (b) com SMOTE, com áreas sob a curva por classe e média macro.

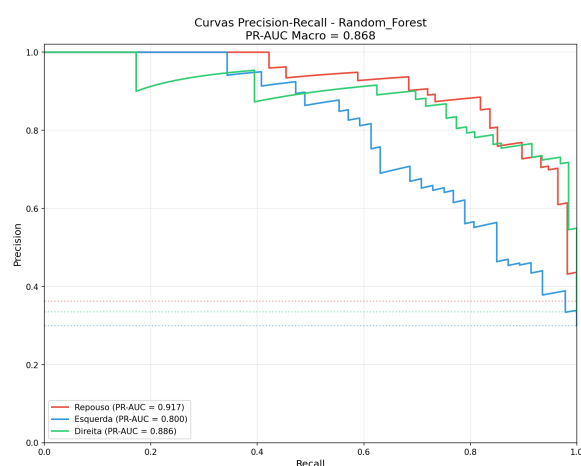


(a) Configuração plana sem balanceamento

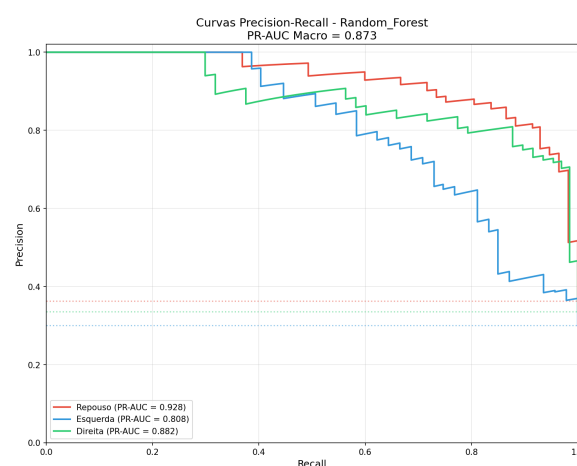


(b) Configuração plana com SMOTE

Figura 6.13 – Curvas Precision–Recall do Random Forest na configuração plana: (a) sem balanceamento e (b) com SMOTE, com áreas sob a curva por classe e média macro.



(a) Configuração plana sem balanceamento



(b) Configuração plana com SMOTE

A calibração comparativa é apresentada na Tab. 6.33, por ela observa-se redução do ECE em Repouso (0,097 para 0,093) e em Direita (0,092 para 0,073), enquanto Esquerda apresentou variação de 0,056 para 0,058. Em todas as classes, observa-se redução do MCE, com variações no intervalo associado ao erro máximo.

Tabela 6.33 – Métricas de calibração por classe — Random Forest (plana): sem balanceamento versus SMOTE

Classe	ECE		MCE		Bin com MCE	
	Sem Bal.	SMOTE	Sem Bal.	SMOTE	Sem Bal.	SMOTE
Repouso	0,097	0,093	0,460	0,453	[0,5–0,6]	[0,3–0,4]
Esquerda	0,056	0,058	0,550	0,234	[0,5–0,6]	[0,3–0,4]
Direita	0,092	0,073	0,577	0,552	[0,4–0,5]	[0,4–0,5]

6.3.8 Síntese Comparativa

A Tab. 6.34 sintetiza as três configurações reportadas (plana sem balanceamento, plana com SMOTE e cascata sem balanceamento). Observa-se que a configuração em cascata apresentou os maiores valores nas métricas de decisão (acurácia, $F1_{\text{macro}}$ e MCC), enquanto a configuração plana com SMOTE apresentou os maiores valores nas métricas baseadas em probabilidades (ROC-AUC macro e PR-AUC macro), além de menor Brier Score.

Tabela 6.34 – Síntese comparativa das configurações do Random Forest

Métrica	Plano	Plano+SMOTE	Cascata
Acurácia	0,763	0,775	0,783
F1-macro	0,756	0,769	0,777
MCC	0,654	0,670	0,678
ROC-AUC macro	0,919	0,922	0,901
PR-AUC macro	0,868	0,873	0,830
Brier Score	0,111	0,105	0,120

Nota: Valores em negrito indicam melhor desempenho para cada métrica.

Na configuração plana, a reamostragem SMOTE elevou as métricas globais e os índices por classe (Tabs. 6.30 e 6.31), com alterações no padrão de confusão descritas pela Fig. 6.11. O aumento de ROC-AUC e PR-AUC macro (Tab. 6.32) ocorreu em conjunto com redução do Brier Score (Tab. 6.30), descrevendo melhora nas métricas baseadas em probabilidade para a configuração plana com SMOTE.

A classe Esquerda apresentou valores de F1 entre 0,671 e 0,691 e sensibilidade entre 0,648 e 0,665 (Tab. 6.26 e Tab. 6.31).

6.4 Gradient Boosting

Na configuração plana multiclasse, a configuração selecionada apresentou profundidade máxima de árvore igual a 3, taxa de aprendizado de $3,28 \times 10^{-2}$, regularização L2 com $\alpha = 89,48$, máximo de 200 iterações com parada antecipada (*early stopping*) após 30 iterações sem melhoria, fração de validação de 20%, fração de atributos por divisão de 90%, mínimo de 50 amostras por folha e seleção de 30 atributos com função de pontuação ANOVA F. O melhor *score* médio em validação cruzada foi $F1_{\text{macro}} = 0,501$, permanecendo acima do desempenho esperado ao acaso ($\approx 0,333$ para três classes equiprováveis) e antecipando o patamar observado na avaliação final.

Na arquitetura em cascata, a primeira etapa (Movimento vs. Repouso) empregou profundidade máxima de 3, regularização L2 $\alpha = 44,01$, taxa de aprendizado de $2,21 \times 10^{-2}$, máximo de 300 iterações com parada antecipada após 20 iterações sem melhoria, fração de validação de 15%, fração de atributos de 50%, mínimo de 20 amostras por folha e seleção de 9 atributos. O *score* médio de validação cruzada da E1 foi $F1_{\text{macro}} = 0,637$. A segunda etapa utilizou profundidade máxima de 4, regularização L2 $\alpha = 9,92$, taxa de aprendizado de $2,99 \times 10^{-2}$, máximo de 300 iterações, fração de validação de 20%, fração de atributos de 90%, mínimo de 20 amostras por folha e seleção de 9 atributos, obtendo $F1_{\text{macro}} = 0,723$ em validação cruzada no cenário ideal.

6.4.1 Desempenho Global sem Balanceamento

Conforme apresentado na Tab. 6.35 a configuração plana apresentou valores superiores aos da configuração em cascata nas métricas de decisão e nas métricas probabilísticas: a acurácia foi 0,776 no plano e 0,758 no cascata ($\Delta = -0,018$), enquanto o $F1_{\text{macro}}$ foi 0,772 no plano e 0,753 no cascata ($\Delta = -0,019$). A redução também se estende a ROC-AUC macro (0,932 $\rightarrow$ 0,892) e PR-AUC macro (0,884 $\rightarrow$ 0,811), com aumento do Brier Score no cascata (0,107 $\rightarrow$ 0,125).

Tabela 6.35 – Métricas de desempenho global do Gradient Boosting nas configurações plana e cascata, com intervalos de confiança bootstrap de 95%

Métrica	Plano		Cascata		Δ
	Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Acurácia	0,776 $\pm$ 0,014	[0,747; 0,804]	0,758 $\pm$ 0,015	[0,728; 0,786]	−0,018
Acurácia Balanceada	0,779 $\pm$ 0,014	[0,750; 0,806]	0,756 $\pm$ 0,015	[0,726; 0,785]	−0,023
F1-macro	0,772 $\pm$ 0,015	[0,743; 0,801]	0,753 $\pm$ 0,016	[0,722; 0,781]	−0,019
F1-weighted	0,774 $\pm$ 0,015	[0,745; 0,803]	0,756 $\pm$ 0,015	[0,726; 0,784]	−0,018
Precisão (macro)	0,791 $\pm$ 0,013	[0,766; 0,815]	0,761 $\pm$ 0,015	[0,731; 0,789]	−0,030
Recall (macro)	0,779 $\pm$ 0,014	[0,750; 0,806]	0,756 $\pm$ 0,015	[0,726; 0,785]	−0,023
MCC	0,676 $\pm$ 0,020	[0,635; 0,714]	0,642 $\pm$ 0,022	[0,598; 0,683]	−0,034
ROC-AUC (macro)	0,932 $\pm$ 0,007	[0,918; 0,945]	0,892 $\pm$ 0,010	[0,873; 0,910]	−0,040
PR-AUC (macro)	0,884 $\pm$ 0,012	[0,860; 0,907]	0,811 $\pm$ 0,017	[0,778; 0,844]	−0,073
Brier Score	0,107 $\pm$ 0,004	[0,099; 0,115]	0,125 $\pm$ 0,005	[0,115; 0,136]	+0,018

Nota: Δ = Cascata – Plano.

A diferença entre as predições foi avaliada pelo teste de McNemar e observou-se assimetria na tabela de contingência, com 57 amostras corretamente classificadas exclusivamente pelo plano e 43 exclusivamente pelo cascata, além de 545 acertos comuns e 131 erros comuns (total de 776 amostras). O teste de McNemar com correção de continuidade não indicou diferença estatisticamente detectável ($\chi^2 = 1,69$; $p = 0,194$), não permitindo rejeitar a hipótese nula de equivalência entre as predições das duas configurações ao nível $\alpha = 0,05$.

A análise por *bootstrap* indicou que a maioria das métricas apresentou intervalos de confiança com largura inferior a 0,10. Na configuração plana, excederam esse limiar os intervalos de confiança do *recall* da classe 0 (Repouso), com largura 0,114, e do *recall* da classe 1 (Esquerda), com largura 0,112. na configuração em cascata, excederam esse limiar o *recall* da classe 1 (largura 0,125) e o *recall* da classe 0 (largura 0,106). Esse resultado indica que a decomposição hierárquica não reduziu o número de métricas com IC 95% acima do limiar adotado para este classificador.

6.4.2 Desempenho por Classe sem Balanceamento

Conforme detalhado na Tab. 6.36, a comparação por classe explicita a direção das variações observadas no desempenho global. A classe Esquerda apresentou, na configuração plana, *recall* de 0,756 e F1 de 0,716, enquanto no cascata os valores foram 0,670 e 0,674, respectivamente. Na classe Direita, o *recall* permaneceu acima de 0,90 em ambas as configurações (0,931

no plano e 0,900 no cascata), embora com redução de F1 (0,834 $\rightarrow$ 0,812). Na classe Repouso, o cascata aumentou o *recall* (0,650 $\rightarrow$ 0,698), acompanhado de redução de precisão (0,938 $\rightarrow$ 0,864), o que caracteriza aumento de falsos positivos na atribuição de Repouso.

Tabela 6.36 – Métricas de desempenho do Gradient Boosting por classe nas configurações plana e cascata

Métrica		Plano		Cascata		Δ
		Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Repouso (Classe 0)	F1	0,767 $\pm$ 0,022	[0,724; 0,808]	0,772 $\pm$ 0,020	[0,730; 0,809]	+0,005
	Precisão	0,938 $\pm$ 0,016	[0,906; 0,969]	0,864 $\pm$ 0,021	[0,821; 0,903]	−0,074
	Recall	0,650 $\pm$ 0,029	[0,592; 0,706]	0,698 $\pm$ 0,027	[0,645; 0,752]	+0,048
Esquerda (Classe 1)	F1	0,716 $\pm$ 0,020	[0,675; 0,755]	0,674 $\pm$ 0,024	[0,626; 0,720]	−0,042
	Precisão	0,681 $\pm$ 0,023	[0,635; 0,726]	0,678 $\pm$ 0,025	[0,629; 0,729]	−0,003
	Recall	0,756 $\pm$ 0,028	[0,700; 0,811]	0,670 $\pm$ 0,031	[0,605; 0,730]	−0,086
Direita (Classe 2)	F1	0,834 $\pm$ 0,014	[0,806; 0,860]	0,812 $\pm$ 0,015	[0,782; 0,841]	−0,022
	Precisão	0,756 $\pm$ 0,019	[0,719; 0,793]	0,740 $\pm$ 0,020	[0,701; 0,780]	−0,016
	Recall	0,931 $\pm$ 0,016	[0,900; 0,962]	0,900 $\pm$ 0,019	[0,862; 0,935]	−0,031

Nota: DP = desvio padrão bootstrap. Δ = Cascata – Plano.

A redução de desempenho na classe Esquerda, observada no cascata, é consistente com a redução de ROC-AUC e PR-AUC nessa classe apresentada posteriormente na Tab. 6.38. Esse comportamento indica que, neste caso, a decomposição em duas etapas não favoreceu a separação da classe Esquerda em relação às demais, mesmo com ajuste específico para Movimento vs. Repouso na E1.

6.4.3 Análise por Etapa da Configuração em Cascata sem Balanceamento

A Tab. 6.37 apresenta as métricas por etapa da configuração em cascata, incluindo a avaliação da segunda etapa tanto no cenário condicionado (E1 real) quanto no cenário ideal (*oracle*). Na E1, observa-se *recall* de 0,937 (IC 95%: [0,914; 0,957]) para a classe Movimento, com ROC-AUC de 0,935. O limiar de decisão foi otimizado em validação cruzada estratificada, utilizando $F1_{\text{macro}}$ como critério; o limiar selecionado foi 0,62, resultando em $F1_{\text{macro}} = 0,714$, contra 0,656 no limiar padrão, correspondendo a melhoria relativa de 8,84%.

Tabela 6.37 – Métricas por etapa da configuração em cascata – Gradient Boosting

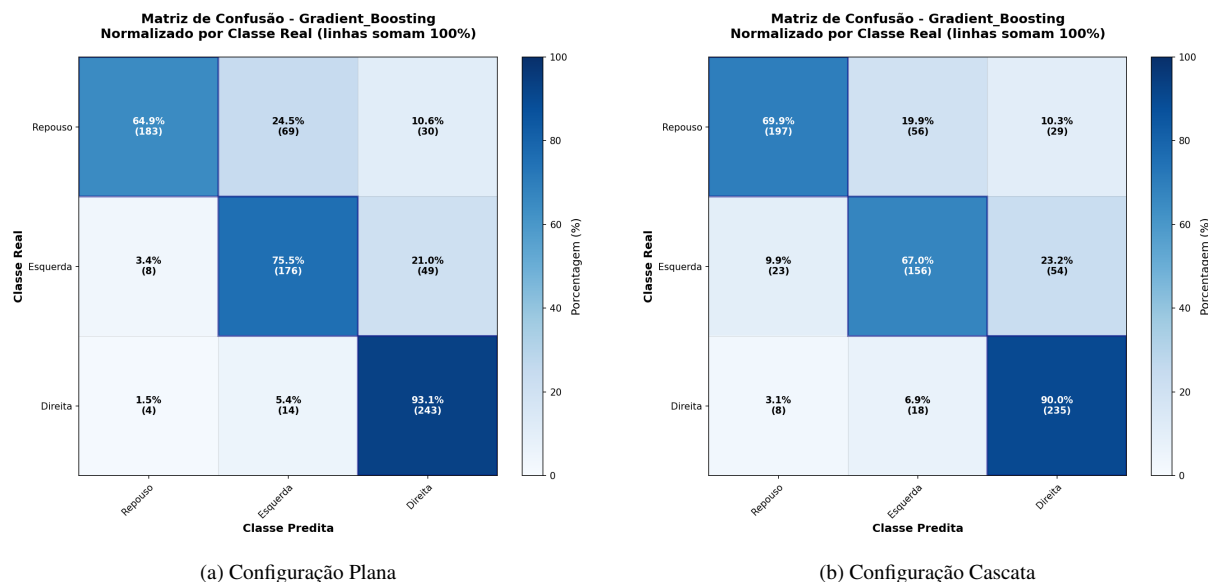
Métrica	Etapa 1		E2 Condicionada		E2 Ideal	
	Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%
Acurácia	0,850 $\pm$ 0,013	[0,826; 0,875]	0,844 $\pm$ 0,017	[0,810; 0,877]	0,838 $\pm$ 0,017	[0,804; 0,868]
Acurácia Bal.	0,818 $\pm$ 0,015	[0,790; 0,847]	0,836 $\pm$ 0,017	[0,801; 0,869]	0,834 $\pm$ 0,017	[0,800; 0,865]
F1	0,888 $\pm$ 0,010	[0,869; 0,907]	0,867 $\pm$ 0,016	[0,834; 0,896]	0,857 $\pm$ 0,016	[0,824; 0,886]
Precisão	0,845 $\pm$ 0,015	[0,816; 0,875]	0,813 $\pm$ 0,023	[0,764; 0,856]	0,804 $\pm$ 0,023	[0,757; 0,848]
Recall	0,937 $\pm$ 0,011	[0,914; 0,957]	0,929 $\pm$ 0,016	[0,896; 0,958]	0,917 $\pm$ 0,017	[0,880; 0,949]
MCC	0,671 $\pm$ 0,027	[0,619; 0,724]	0,691 $\pm$ 0,033	[0,626; 0,754]	0,681 $\pm$ 0,032	[0,615; 0,740]
ROC-AUC	0,935 $\pm$ 0,009	[0,917; 0,952]	0,914 $\pm$ 0,013	[0,886; 0,939]	0,915 $\pm$ 0,013	[0,888; 0,939]
PR-AUC	0,965 $\pm$ 0,006	[0,953; 0,976]	0,923 $\pm$ 0,014	[0,894; 0,949]	0,919 $\pm$ 0,014	[0,888; 0,945]
Brier Score	0,127 $\pm$ 0,007	[0,114; 0,141]	0,124 $\pm$ 0,012	[0,100; 0,149]	0,124 $\pm$ 0,012	[0,102; 0,148]

A comparação entre E2 Ideal e E2 Condicionada indica que a E2 Condicionada apresentou valores ligeiramente superiores em algumas métricas, com $\Delta_{F1} = +0,010$, $\Delta_{\text{acurácia}} = +0,006$, $\Delta_{\text{precisão}} = +0,009$ e $\Delta_{\text{recall}} = +0,012$. Esse resultado pode estar associado ao fato de que os 85 falsos positivos de Repouso encaminhados à E2 ocupem regiões do espaço de atributos nas quais o modelo da E2 atribui rótulos com baixa ambiguidade, não resultando em perda sistemática na média das métricas reportadas para a etapa condicionada.

6.4.4 Matriz de Confusão sem Balanceamento

Ao avaliar a Fig. 6.14 tem-se que na configuração plana (Figura 6.14a) 24,47% das amostras de Repouso foram classificadas como Esquerda, enquanto 10,64% foram classificadas como Direita. Para a classe Esquerda, a taxa de acerto foi 75,54%, com confusão predominante com Direita (21,03%). Para a classe Direita, a taxa de acerto foi 93,10%.

Figura 6.14 – Matrizes de confusão percentuais do Gradient Boosting nas configurações plana (a) e cascata (b). Os valores representam a proporção de cada classe verdadeira distribuída entre as classes preditas (normalização por linha).



Na configuração em cascata (Figura 6.14b), a taxa de acerto de Repouso aumentou de 64,89% para 69,86%, enquanto a taxa de acerto de Esquerda reduziu de 75,54% para 66,95%. A confusão Esquerda→Direita passou de 21,03% para 23,18%, e a confusão Esquerda→Repouso aumentou de 3,43% para 9,87%. Essa última fração corresponde, no cascata, aos falsos negativos da E1 que atribuem Repouso a amostras verdadeiras de Esquerda (23 de 233 amostras). A taxa de acerto de Direita foi 90,04%.

6.4.5 Análise das Curvas ROC e PR sem Balanceamento

Conforme apresentado na Tab. 6.38, a configuração em cascata apresentou redução de ROC-AUC e PR-AUC em todas as classes. A maior redução ocorreu na classe Esquerda, com variação de $-0,072$ em ROC-AUC e $-0,097$ em PR-AUC, em alinhamento com a redução de *recall* e F1 dessa classe na Tab. 6.36. No presente estudo, a distribuição de classes (Repouso = 282, Esquerda = 233, Direita = 261) indica desbalanceamento moderado (razão máxima $\approx 1,2 : 1$), e a redução de PR-AUC macro de 0,884 para 0,811 descreve maior perda de desempenho ao longo de limiares quando a composição em cascata é aplicada.

Tabela 6.38 – Áreas sob as curvas ROC e PR por classe e configuração — Gradient Boosting

Classe	ROC-AUC			PR-AUC		
	Plano	Cascata	Δ	Plano	Cascata	Δ
Repouso	0,945	0,935	−0,010	0,896	0,865	−0,031
Esquerda	0,898	0,826	−0,072	0,846	0,749	−0,097
Direita	0,953	0,916	−0,037	0,910	0,819	−0,091
Macro	0,932	0,892	−0,040	0,884	0,811	−0,073

Nota: Δ = Cascata − Plano.

6.4.6 Análise de Calibração sem Balanceamento

Conforme a Tab. 6.39, na configuração plana, os valores de ECE foram 0,118 (Repouso), 0,084 (Esquerda) e 0,092 (Direita), enquanto o maior MCE ocorreu em Direita (0,575) no bin [0,5–0,6]. Na configuração em cascata, o ECE de Esquerda reduziu de 0,084 para 0,076, enquanto Repouso e Direita apresentaram aumento de ECE. O MCE reduziu em Esquerda (0,466 → 0,348) e em Direita (0,575 → 0,386), com deslocamento do bin associado ao erro máximo em Direita para [0,7–0,8]. Em ambas as configurações, os valores de MCE (0,348 a 0,575) indicam discrepâncias máximas concentradas em faixas específicas de probabilidade.

Tabela 6.39 – Métricas de calibração por classe e configuração — Gradient Boosting

Classe	ECE		MCE		Bins com MCE	
	Plano	Cascata	Plano	Cascata	Plano	Cascata
Repouso	0,118	0,130	0,405	0,466	[0,5–0,6]	[0,3–0,4]
Esquerda	0,084	0,076	0,466	0,348	[0,4–0,5]	[0,3–0,4]
Direita	0,092	0,121	0,575	0,386	[0,5–0,6]	[0,7–0,8]

6.5 Máquinas de Vetores de Suporte

Na configuração plana multiclasse sem balanceamento, a configuração selecionada empregou função de base radial, parâmetro de regularização $C = 1,0$, coeficiente $\gamma = 10^{-4}$, ponderação de classes balanceada e seleção de 17 atributos por seleção univariada com função de pontuação ANOVA F. O maior escore médio em validação cruzada foi $F1_{\text{macro}} = 0,588$, permanecendo acima do desempenho esperado ao acaso.

Na arquitetura em cascata sem balanceamento, a primeira etapa (Movimento versus Repouso) utilizou $C = 0,1$, $\gamma = 10^{-3}$, ponderação balanceada e seleção de 13 atributos, com escore

médio de validação cruzada $F1_{\text{macro}} = 0,709$. A segunda etapa (Esquerda versus Direita), treinada exclusivamente com amostras de movimento, utilizou $C = 10,0$, $\gamma = 10^{-5}$ e seleção de 16 atributos, obtendo $F1_{\text{macro}} = 0,764$ em validação cruzada no cenário com oráculo, no qual a primeira etapa é assumida perfeita e, portanto, não há propagação de erros para a segunda etapa.

6.5.1 Desempenho Global sem Balanceamento

Conforme apresentado na Tab. 6.40 a configuração plana apresentou valores mais elevados do que a configuração em cascata nas métricas baseadas em predição discreta. A acurácia foi 0,802 no arranjo plano e 0,676 no cascata ($\Delta = -0,126$), enquanto o $F1_{\text{macro}}$ foi 0,796 no plano e 0,649 no cascata ($\Delta = -0,147$). O MCC também reduziu de 0,707 para 0,558 ($\Delta = -0,149$). Em contraste, as diferenças em métricas de área sob curva foram menores, com ROC-AUC macro de 0,917 no plano e 0,909 no cascata, e PR-AUC macro de 0,867 no plano e 0,843 no cascata.

Tabela 6.40 – Métricas de desempenho global do SVM nas configurações plana e cascata sem balanceamento, com intervalos de confiança bootstrap de 95%

Métrica	Plano		Cascata		Δ
	Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Acurácia	0,802 $\pm$ 0,014	[0,775; 0,829]	0,676 $\pm$ 0,013	[0,650; 0,702]	-0,126
Acurácia Balanceada	0,799 $\pm$ 0,014	[0,771; 0,825]	0,688 $\pm$ 0,013	[0,662; 0,714]	-0,111
F1-macro	0,796 $\pm$ 0,014	[0,767; 0,824]	0,649 $\pm$ 0,016	[0,618; 0,682]	-0,147
F1-weighted	0,800 $\pm$ 0,014	[0,772; 0,827]	0,645 $\pm$ 0,017	[0,613; 0,679]	-0,155
Precisão (macro)	0,806 $\pm$ 0,014	[0,779; 0,832]	0,751 $\pm$ 0,009	[0,734; 0,768]	-0,055
Recall (macro)	0,799 $\pm$ 0,014	[0,771; 0,825]	0,688 $\pm$ 0,013	[0,662; 0,714]	-0,111
MCC	0,707 $\pm$ 0,020	[0,667; 0,747]	0,558 $\pm$ 0,019	[0,522; 0,595]	-0,149
ROC-AUC (macro)	0,917 $\pm$ 0,007	[0,902; 0,930]	0,909 $\pm$ 0,008	[0,892; 0,924]	-0,008
PR-AUC (macro)	0,867 $\pm$ 0,011	[0,844; 0,889]	0,843 $\pm$ 0,014	[0,815; 0,870]	-0,024
Brier Score	0,115 $\pm$ 0,005	[0,106; 0,125]	0,127 $\pm$ 0,005	[0,117; 0,138]	+0,012

Nota: $\Delta = \text{Cascata} - \text{Plano}$.

A inspeção das larguras dos intervalos de confiança via *bootstrap* indicou que, na configuração plana, apenas a sensibilidade (*recall*) da classe Esquerda apresentou IC 95% com largura superior a 0,10 (largura 0,116). No arranjo em cascata, três métricas excederam esse limiar: F1 da classe Repouso (0,124), sensibilidade de Repouso (0,110) e sensibilidade de Esquerda (0,107). Em conjunto, esse resultado descreve maior variabilidade das estimativas no cascata, quando comparado ao plano, para o conjunto de amostras avaliado.

6.5.2 Desempenho por Classe sem Balanceamento

Conforme apresentado na Tab. 6.41, a análise por classe explicita a origem das diferenças observadas no desempenho global. No arranjo plano, a classe Repouso apresentou precisão de 0,915 e sensibilidade de 0,766, resultando em F1 de 0,834. No cascata, a precisão para Repouso foi 1,000, enquanto a sensibilidade foi 0,331, conduzindo a F1 de 0,496. Essa combinação decorre do limiar de decisão da primeira etapa, ajustado para 0,40 com o objetivo de maximizar a detecção de Movimento; nessa parametrização, a sensibilidade de Movimento torna-se 1,000 e, como efeito associado, 67,0% das amostras de Repouso são classificadas como Movimento e encaminhadas à segunda etapa.

Tabela 6.41 – Métricas de desempenho do SVM por classe nas configurações plana e cascata sem balanceamento

Métrica		Plano		Cascata		Δ
		Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Repouso (Classe 0)	F1	0,834 $\pm$ 0,018	[0,800; 0,867]	0,496 $\pm$ 0,032	[0,433; 0,558]	−0,338
	Precisão	0,915 $\pm$ 0,017	[0,883; 0,948]	1,000 $\pm$ 0,000	[1,000; 1,000]	+0,085
	Recall	0,766 $\pm$ 0,025	[0,716; 0,816]	0,331 $\pm$ 0,028	[0,277; 0,387]	−0,435
Esquerda (Classe 1)	F1	0,715 $\pm$ 0,023	[0,670; 0,758]	0,651 $\pm$ 0,020	[0,612; 0,689]	−0,064
	Precisão	0,742 $\pm$ 0,025	[0,694; 0,792]	0,577 $\pm$ 0,019	[0,540; 0,617]	−0,165
	Recall	0,691 $\pm$ 0,030	[0,631; 0,747]	0,747 $\pm$ 0,028	[0,691; 0,798]	+0,056
Direita (Classe 2)	F1	0,839 $\pm$ 0,014	[0,812; 0,865]	0,801 $\pm$ 0,013	[0,777; 0,826]	−0,038
	Precisão	0,759 $\pm$ 0,019	[0,722; 0,795]	0,675 $\pm$ 0,017	[0,643; 0,710]	−0,084
	Recall	0,939 $\pm$ 0,015	[0,908; 0,966]	0,985 $\pm$ 0,008	[0,966; 0,996]	+0,046

Nota: DP = desvio padrão bootstrap. Δ = Cascata – Plano.

A classe Direita apresentou sensibilidade elevada em ambas as configurações (0,939 no plano e 0,985 no cascata). Ainda assim, a principal contribuição para a redução das métricas globais no cascata permanece associada à classe Repouso, conforme indicado pela diferença de F1 de −0,338 nessa classe (Tab. 6.41).

6.5.3 Análise por Etapa da Configuração em Cascata sem Balanceamento

A Tab. 6.42 mostra que a primeira etapa apresentou sensibilidade perfeita de 1,000, indicando que todas as amostras de movimento foram corretamente identificadas e encaminhadas à segunda etapa. O ROC-AUC de 0,939 evidencia boa capacidade discriminativa do classificador.

A análise identificou o limiar de 0,40 (inferior ao padrão de 0,50), resultando em F1-macro de 0,738 contra 0,701 do limiar padrão, representando melhoria relativa de 5,35%.

Tabela 6.42 – Métricas por etapa da configuração em cascata — SVM

Métrica	Etapa 1		E2 Condicionada		E2 Ideal	
	Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%
Acurácia	0,757 $\pm$ 0,015	[0,727; 0,786]	0,872 $\pm$ 0,015	[0,842; 0,899]	0,872 $\pm$ 0,015	[0,842; 0,899]
Acurácia Bal.	0,665 $\pm$ 0,014	[0,638; 0,692]	0,865 $\pm$ 0,015	[0,836; 0,893]	0,865 $\pm$ 0,015	[0,836; 0,893]
F1	0,839 $\pm$ 0,011	[0,816; 0,861]	0,890 $\pm$ 0,014	[0,860; 0,915]	0,890 $\pm$ 0,014	[0,860; 0,915]
Precisão	0,724 $\pm$ 0,017	[0,690; 0,755]	0,813 $\pm$ 0,022	[0,767; 0,852]	0,813 $\pm$ 0,022	[0,767; 0,852]
Sensibilidade	1,000 $\pm$ 0,000	[1,000; 1,000]	0,985 $\pm$ 0,008	[0,968; 0,996]	0,985 $\pm$ 0,008	[0,968; 0,996]
MCC	0,489 $\pm$ 0,024	[0,440; 0,535]	0,760 $\pm$ 0,026	[0,708; 0,808]	0,760 $\pm$ 0,026	[0,708; 0,808]
ROC-AUC	0,939 $\pm$ 0,009	[0,921; 0,956]	0,943 $\pm$ 0,010	[0,924; 0,961]	0,943 $\pm$ 0,010	[0,924; 0,961]
PR-AUC	0,959 $\pm$ 0,008	[0,942; 0,974]	0,947 $\pm$ 0,010	[0,926; 0,964]	0,947 $\pm$ 0,010	[0,926; 0,964]
Brier Score	0,142 $\pm$ 0,008	[0,126; 0,157]	0,110 $\pm$ 0,012	[0,088; 0,134]	0,110 $\pm$ 0,012	[0,088; 0,134]

A precisão de 0,724 na primeira etapa indica que 27,6% das amostras classificadas como Movimento correspondem, na realidade, a amostras de Repouso. O Brier Score de 0,142 na primeira etapa, comparado a 0,110 na segunda etapa, indica que as probabilidades preditas apresentam maior discrepância em relação às frequências observadas na tarefa de discriminação Movimento versus Repouso.

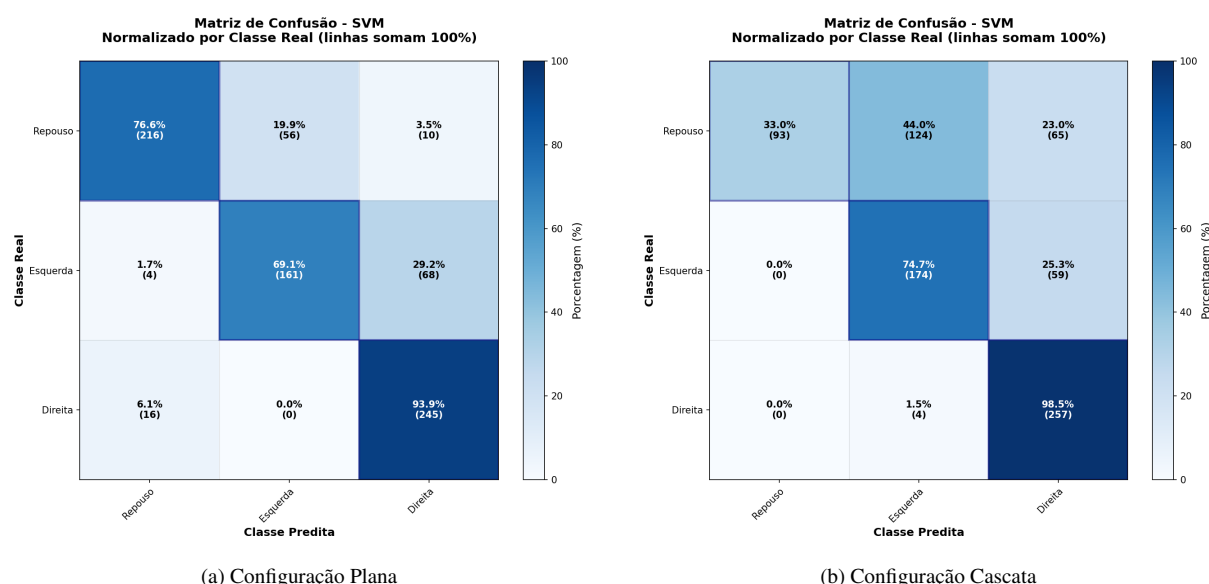
A comparação entre os cenários E2 Ideal e E2 Condicionada revela um resultado singular: não houve degradação de desempenho ($\Delta = 0$ para todas as métricas). Esta ausência de degradação decorre diretamente da sensibilidade perfeita da primeira etapa: com taxa de falsos negativos nula, todas as 494 amostras de movimento foram corretamente encaminhadas à segunda etapa. Os 189 falsos positivos da primeira etapa (67,0% das 282 amostras de Repouso) não afetaram as métricas da segunda etapa porque esta avalia exclusivamente a discriminação entre Esquerda e Direita nas amostras efetivamente de movimento.

Embora a igualdade entre os cenários ideal e condicionado possa parecer favorável, a elevada taxa de falsos positivos da primeira etapa (67,0%) representa uma limitação operacional significativa. Cada amostra de Repouso incorretamente classificada como Movimento é encaminhada à segunda etapa, onde será classificada como Esquerda ou Direita, resultando em comandos espúrios para a interface cérebro-computador. Este comportamento explica por que 75,0% do erro total do sistema cascata (189 de 252 erros) origina-se na primeira etapa.

6.5.4 Matriz de Confusão sem Balanceamento

A Fig. 6.15 apresenta as matrizes de confusão percentuais das configurações plana e em cascata. Na configuração plana (Figura 6.15a), a classe Repouso apresentou 76,60% de acertos, com confusão principalmente com Esquerda (19,86%) e menor parcela com Direita (3,55%). A classe Esquerda apresentou 69,10% de acertos, com confusão com Direita (29,18%) e com Repouso (1,72%). A classe Direita apresentou 93,87% de acertos.

Figura 6.15 – Matrizes de confusão percentuais do SVM nas configurações plana (a) e cascata (b) sem balanceamento. Os valores representam a proporção de cada classe verdadeira distribuída entre as classes preditas (normalização por linha).



Na configuração em cascata (Figura 6.15b), a classe Repouso apresentou 32,98% de acertos, com redistribuição de 67,02% das amostras entre as classes de movimento (43,97% em Esquerda e 23,05% em Direita). Esse padrão é consistente com a parametrização da primeira etapa descrita anteriormente: amostras de Repouso classificadas como Movimento são necessariamente encaminhadas à segunda etapa e rotuladas como Esquerda ou Direita. As classes de movimento não apresentaram confusão com Repouso no cascata, em decorrência da sensibilidade 1,000 para Movimento na primeira etapa. Nesse arranjo, Esquerda atingiu 74,68% de acertos e Direita 98,47%.

6.5.5 Análise das Curvas ROC e PR sem Balanceamento

A Tab. 6.43 indica ROC-AUC macro de 0,917 no plano e 0,908 no cascata ($\Delta = -0,009$), além de PR-AUC macro de 0,867 no plano e 0,842 no cascata ($\Delta = -0,025$). Na classe Direita, a PR-AUC apresentou redução de 0,925 para 0,846 ($\Delta = -0,079$), enquanto na classe Esquerda a configuração em cascata apresentou aumento de ROC-AUC (+0,008) e PR-AUC (+0,044), compatível com a especialização da segunda etapa para discriminar apenas entre Esquerda e Direita.

Tabela 6.43 – Áreas sob as curvas ROC e PR por classe e configuração — SVM sem balanceamento

Classe	ROC-AUC			PR-AUC		
	Plano	Cascata	Δ	Plano	Cascata	Δ
Repouso	0,959	0,939	-0,020	0,944	0,905	-0,039
Esquerda	0,834	0,842	+0,008	0,731	0,775	+0,044
Direita	0,957	0,944	-0,013	0,925	0,846	-0,079
Macro	0,917	0,908	-0,009	0,867	0,842	-0,025

Nota: $\Delta = \text{Cascata} - \text{Plano}$.

6.5.6 Análise de Calibração sem Balanceamento

A Tab. 6.44 mostra que na configuração plana, os valores de ECE foram 0,136 (Repouso), 0,080 (Esquerda) e 0,142 (Direita), enquanto os MCE variaram entre 0,375 e 0,481, com ocorrência em faixas de probabilidade intermediárias. Na configuração em cascata, o ECE de Repouso aumentou para 0,162, em alinhamento com a redução de sensibilidade e F1 para essa classe descrita na Tab. 6.41. Os valores de MCE permaneceram elevados nas duas configurações (0,353 a 0,545), indicando discrepâncias máximas concentradas em faixas específicas de probabilidade predita.

Tabela 6.44 – Métricas de calibração por classe e configuração — SVM sem balanceamento

Classe	ECE		MCE		Bins com MCE	
	Plano	Cascata	Plano	Cascata	Plano	Cascata
Repouso	0,136	0,162	0,481	0,545	[0,2–0,3]	[0,3–0,4]
Esquerda	0,080	0,116	0,413	0,353	[0,5–0,6]	[0,3–0,4]
Direita	0,142	0,119	0,375	0,432	[0,3–0,4]	[0,4–0,5]

6.5.7 Efeito do SMOTEENN na Configuração Plana

A aplicação da técnica SMOTEENN na configuração plana multiclasse resultou em configuração de hiperparâmetros distinta: $C = 0,1$ (regularização mais intensa comparada ao cenário sem balanceamento), $\gamma = 10^{-4}$, ponderação balanceada e seleção de 13 atributos. O escore de validação cruzada foi $F1_{\text{macro}} = 0,390$, valor inferior ao obtido sem balanceamento (0,588), indicando que o aumento artificial da variabilidade dos dados de treinamento dificulta a generalização durante a validação cruzada.

Conforme pode ser observado na Tab. 6.45 a aplicação do SMOTEENN resultou em ganhos nas métricas baseadas em predições discretas: acurácia aumentou de 0,802 para 0,826 (+2,4 p.p.), F1-macro de 0,796 para 0,822 (+2,6 p.p.) e MCC de 0,707 para 0,748 (+4,1 p.p.). Entretanto, as métricas baseadas em probabilidades preditas apresentaram degradação: ROC-AUC macro diminuiu de 0,917 para 0,893 (−2,4 p.p.) e PR-AUC macro apresentou redução de 0,867 para 0,772 (−9,5 p.p.). Esta divergência entre métricas de decisão e métricas probabilísticas sugere que o balanceamento melhora a fronteira de decisão no limiar padrão, porém compromete a ordenação relativa das probabilidades preditas.

Tabela 6.45 – Métricas de desempenho global do SVM na configuração plana sem balanceamento e com SMOTEENN, com intervalos de confiança bootstrap de 95%

Métrica	Sem Balanceamento		SMOTEENN		Δ
	Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Acurácia	0,802 $\pm$ 0,014	[0,775; 0,829]	0,826 $\pm$ 0,013	[0,800; 0,852]	+0,024
Acurácia Balanceada	0,799 $\pm$ 0,014	[0,771; 0,825]	0,825 $\pm$ 0,013	[0,798; 0,851]	+0,026
F1-macro	0,796 $\pm$ 0,014	[0,767; 0,824]	0,822 $\pm$ 0,014	[0,794; 0,848]	+0,026
F1-weighted	0,800 $\pm$ 0,014	[0,772; 0,827]	0,825 $\pm$ 0,014	[0,798; 0,851]	+0,025
Precisão (macro)	0,806 $\pm$ 0,014	[0,779; 0,832]	0,836 $\pm$ 0,012	[0,812; 0,859]	+0,030
Recall (macro)	0,799 $\pm$ 0,014	[0,771; 0,825]	0,825 $\pm$ 0,013	[0,798; 0,851]	+0,026
MCC	0,707 $\pm$ 0,020	[0,667; 0,747]	0,748 $\pm$ 0,019	[0,710; 0,785]	+0,041
ROC-AUC (macro)	0,917 $\pm$ 0,007	[0,902; 0,930]	0,893 $\pm$ 0,009	[0,875; 0,911]	−0,024
PR-AUC (macro)	0,867 $\pm$ 0,011	[0,844; 0,889]	0,772 $\pm$ 0,017	[0,739; 0,806]	−0,095
Brier Score	0,115 $\pm$ 0,005	[0,106; 0,125]	0,112 $\pm$ 0,005	[0,103; 0,122]	−0,003

Nota: Δ = SMOTEENN – Sem Balanceamento.

A Tab. 6.46 detalha o desempenho por classe, por ela conclui-se que o SMOTEENN produziu aumento na precisão da classe Repouso de 0,915 para 0,982 (+6,7 p.p.), com leve redução na sensibilidade de 0,766 para 0,748 (−1,8 p.p.). A classe Esquerda apresentou melhoria

na sensibilidade de 0,691 para 0,743 (+5,2 p.p.) sem alteração na precisão. A classe Direita apresentou ganhos em ambas as métricas, com sensibilidade atingindo 0,985.

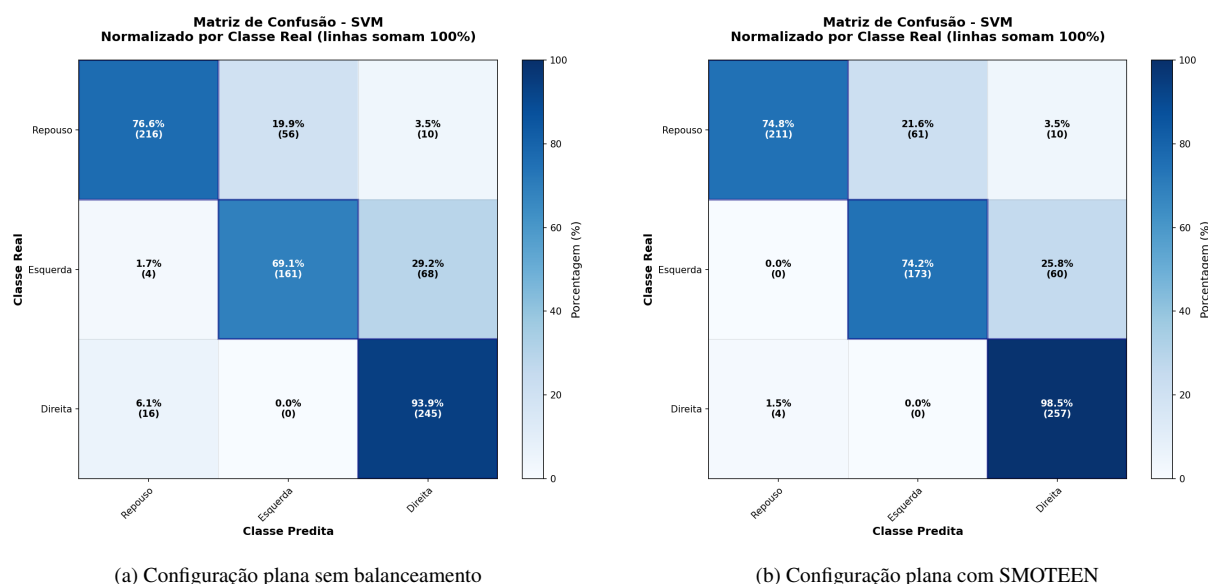
Tabela 6.46 – Métricas de desempenho do SVM por classe na configuração plana sem balanceamento e com SMOTEENN

Métrica		Sem Balanceamento		SMOTEENN		Δ
		Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Repouso (Classe 0)	F1	0,834 $\pm$ 0,018	[0,800; 0,867]	0,849 $\pm$ 0,018	[0,815; 0,882]	+0,015
	Precisão	0,915 $\pm$ 0,017	[0,883; 0,948]	0,982 $\pm$ 0,009	[0,961; 0,996]	+0,067
	Recall	0,766 $\pm$ 0,025	[0,716; 0,816]	0,748 $\pm$ 0,026	[0,699; 0,801]	−0,018
Esquerda (Classe 1)	F1	0,715 $\pm$ 0,023	[0,670; 0,758]	0,741 $\pm$ 0,021	[0,697; 0,783]	+0,026
	Precisão	0,742 $\pm$ 0,025	[0,694; 0,792]	0,740 $\pm$ 0,024	[0,694; 0,788]	−0,002
	Recall	0,691 $\pm$ 0,030	[0,631; 0,747]	0,743 $\pm$ 0,029	[0,687; 0,798]	+0,052
Direita (Classe 2)	F1	0,839 $\pm$ 0,014	[0,812; 0,865]	0,875 $\pm$ 0,012	[0,850; 0,897]	+0,036
	Precisão	0,759 $\pm$ 0,019	[0,722; 0,795]	0,787 $\pm$ 0,018	[0,750; 0,822]	+0,028
	Recall	0,939 $\pm$ 0,015	[0,908; 0,966]	0,985 $\pm$ 0,008	[0,966; 0,996]	+0,046

Nota: DP = desvio padrão bootstrap. Δ = SMOTEENN – Sem Balanceamento.

A Fig. 6.16 apresenta as matrizes de confusão percentuais para a configuração plana, comparando os cenários sem balanceamento e com SMOTEENN. A normalização por linha permite avaliar como cada classe verdadeira é distribuída entre as classes preditas, facilitando a identificação de padrões de confusão específicos.

Figura 6.16 – Matrizes de confusão percentuais do SVM na configuração plana: (a) sem balanceamento e (b) com SMOTEEN. Os valores representam a proporção de cada classe verdadeira distribuída entre as classes preditas (normalização por linha).

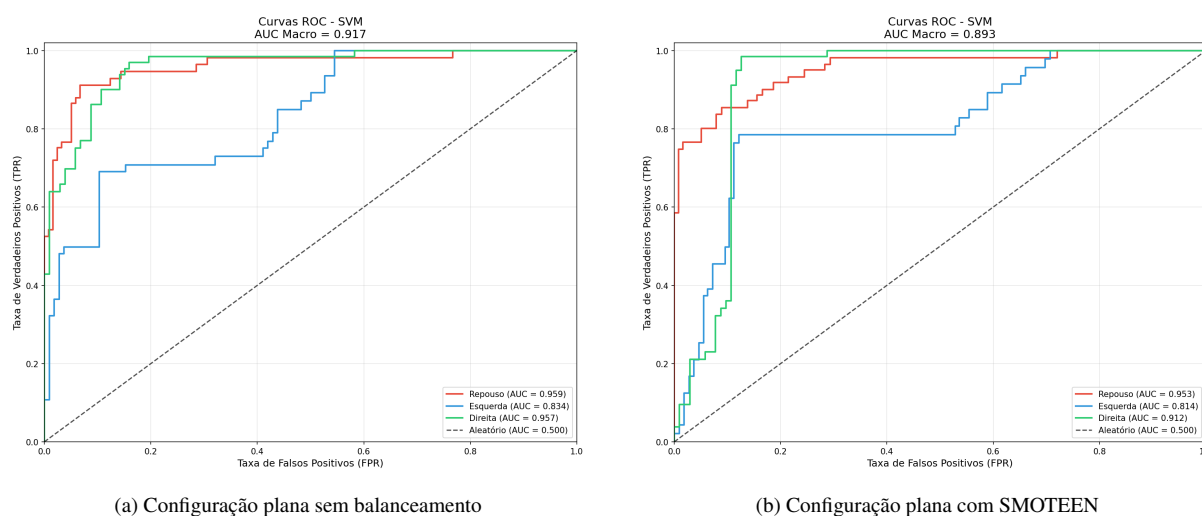


Na configuração plana sem balanceamento (Figura 6.16a), observa-se que a classe Repouso apresenta confusão predominante com Esquerda (19,86%), enquanto a confusão com Direita é de 3,55%. A classe Esquerda apresenta confusão com Direita (29,18%) e baixa confusão com Repouso (1,72%). A classe Direita apresenta taxa de acerto de 93,87%, com erros distribuídos para Esquerda (6,13%) e ausência de confusão com Repouso.

Com a aplicação do SMOTEENN (Figura 6.16b), a classe Repouso mantém padrão similar de confusão (74,82% de acertos, 21,63% confundidos com Esquerda). A classe Esquerda apresenta melhoria na taxa de acertos (74,25% contra 69,10% sem balanceamento), com redução da confusão com Direita (25,75% contra 29,18%). A classe Direita atinge 98,47% de acertos, com eliminação da confusão com Repouso.

As curvas ROC da configuração plana são apresentadas na Fig. 6.17, permitindo avaliar a capacidade discriminativa do classificador em diferentes pontos de operação. Na configuração sem balanceamento (Figura 6.17a), a ROC-AUC macro de 0,917 indica capacidade discriminativa elevada. A classe Repouso apresenta ROC-AUC de 0,959, seguida por Direita (0,957) e Esquerda (0,834). A menor ROC-AUC para Esquerda é consistente com a maior taxa de confusão observada nas matrizes de confusão.

Figura 6.17 – Curvas ROC do SVM na configuração plana: (a) sem balanceamento e (b) com SMOTEENN, com áreas sob a curva por classe e média macro.

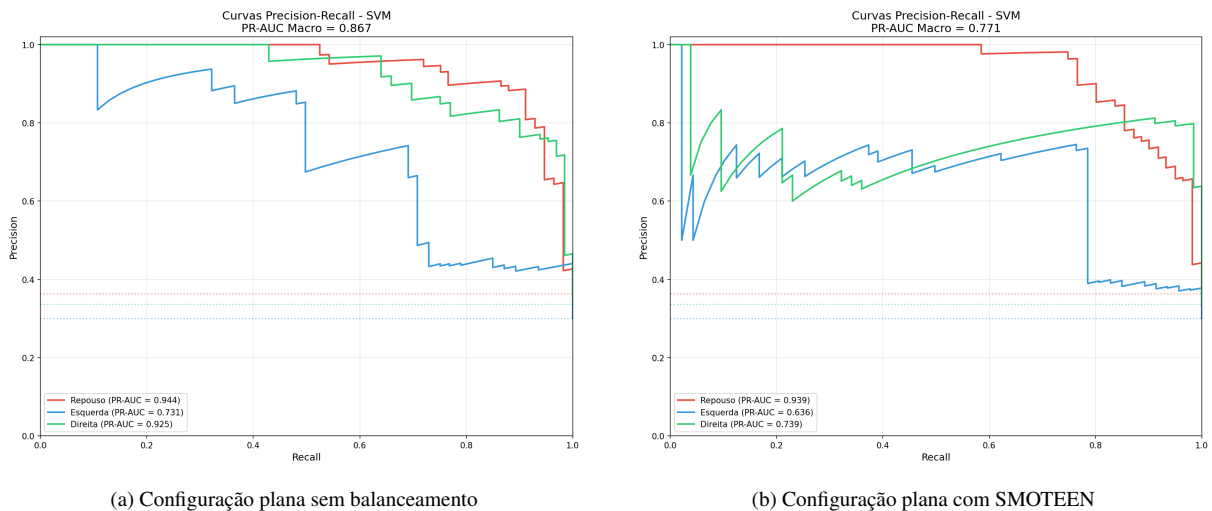


Com SMOTEENN (Figura 6.17b), observa-se redução da ROC-AUC macro para 0,893. A classe Repouso mantém ROC-AUC de 0,959, Direita apresenta 0,899 e Esquerda 0,822. A

redução nas curvas de Direita e Esquerda indica que o balanceamento, embora melhore métricas pontuais no limiar padrão, compromete a ordenação relativa das probabilidades preditas.

Na configuração sem balanceamento (Figura 6.18a), a PR-AUC macro de 0,867 reflete o desempenho geral do modelo. A classe Repouso apresenta PR-AUC de 0,944, Direita 0,925 e Esquerda 0,731. A menor PR-AUC para Esquerda é consistente com a menor prevalência e maior dificuldade de discriminação dessa classe.

Figura 6.18 – Curvas Precision–Recall do SVM na configuração plana: (a) sem balanceamento e (b) com SMOTEEN, com áreas sob a curva por classe e média macro.



Com SMOTEENN (Figura 6.18b), a PR-AUC macro reduz para 0,772, com quedas em todas as classes: Repouso (0,862), Direita (0,752) e Esquerda (0,703). A redução da PR-AUC indica que o SMOTEENN, apesar de melhorar métricas de decisão no limiar padrão, prejudica a qualidade das estimativas probabilísticas em diferentes pontos de operação.

6.5.8 Efeito do SMOTEENN na Configuração Cascata

Na arquitetura em cascata com SMOTEENN, a primeira etapa utilizou $C = 10,0$, γ calculado automaticamente com base no número de atributos, ponderação balanceada e seleção de 23 atributos, obtendo $F1_{\text{macro}} = 0,535$ em validação cruzada. O limiar de decisão otimizado foi 0,58 (superior ao padrão de 0,50), representando configuração mais conservadora para classificação como Movimento comparada ao cenário sem balanceamento, onde o limiar foi 0,40.

A Tab. 6.47 evidencia como o comportamento do SMOTEENN na configuração cascata difere do observado na plana. As métricas de acurácia permaneceram estáveis ou apresentaram leve degradação, enquanto F1-macro aumentou de 0,649 para 0,674 (+2,5 p.p.) e F1-weighted de 0,645 para 0,679 (+3,4 p.p.). Entretanto, a precisão macro diminuiu de 0,751 para 0,699 (−5,2 p.p.), o MCC reduziu de 0,558 para 0,521 (−3,7 p.p.) e o Brier Score aumentou de 0,127 para 0,153 (+2,6 p.p.), indicando degradação na qualidade das estimativas probabilísticas.

Tabela 6.47 – Métricas de desempenho global do SVM na configuração em cascata sem balanceamento e com SMOTEENN, com intervalos de confiança bootstrap de 95%

Métrica	Sem Balanceamento		SMOTEENN		Δ
	Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Acurácia	0,676 $\pm$ 0,013	[0,650; 0,702]	0,671 $\pm$ 0,017	[0,638; 0,705]	−0,005
Acurácia Balanceada	0,688 $\pm$ 0,013	[0,662; 0,714]	0,674 $\pm$ 0,017	[0,641; 0,707]	−0,014
F1-macro	0,649 $\pm$ 0,016	[0,618; 0,682]	0,674 $\pm$ 0,016	[0,642; 0,707]	+0,025
F1-weighted	0,645 $\pm$ 0,017	[0,613; 0,679]	0,679 $\pm$ 0,016	[0,647; 0,712]	+0,034
Precisão (macro)	0,751 $\pm$ 0,009	[0,734; 0,768]	0,699 $\pm$ 0,015	[0,669; 0,729]	−0,052
Recall (macro)	0,688 $\pm$ 0,013	[0,662; 0,714]	0,674 $\pm$ 0,017	[0,641; 0,707]	−0,014
MCC	0,558 $\pm$ 0,019	[0,522; 0,595]	0,521 $\pm$ 0,024	[0,473; 0,568]	−0,037
ROC-AUC (macro)	0,909 $\pm$ 0,008	[0,892; 0,924]	0,853 $\pm$ 0,011	[0,830; 0,875]	−0,056
PR-AUC (macro)	0,843 $\pm$ 0,014	[0,815; 0,870]	0,767 $\pm$ 0,017	[0,733; 0,800]	−0,076
Brier Score	0,127 $\pm$ 0,005	[0,117; 0,138]	0,153 $\pm$ 0,004	[0,145; 0,161]	+0,026

Nota: Δ = SMOTEENN – Sem Balanceamento.

Como pode ser observado na Tab. 6.48 a aplicação do SMOTEENN na cascata produziu redistribuição do desempenho entre classes. A classe Repouso apresentou melhoria sendo que o F1 aumentou de 0,496 para 0,747 (+25,1 p.p.) e sensibilidade de 0,331 para 0,664 (+33,3 p.p.), embora com redução na precisão de 1,000 para 0,853. Esta alteração decorre do ajuste do limiar de decisão na primeira etapa para 0,58, que reduz a taxa de falsos positivos de 67,0% para 33,7% (95 de 282 amostras de Repouso erroneamente classificadas como Movimento).

Tabela 6.48 – Métricas de desempenho do SVM por classe na configuração em cascata sem balanceamento e com SMOTEENN

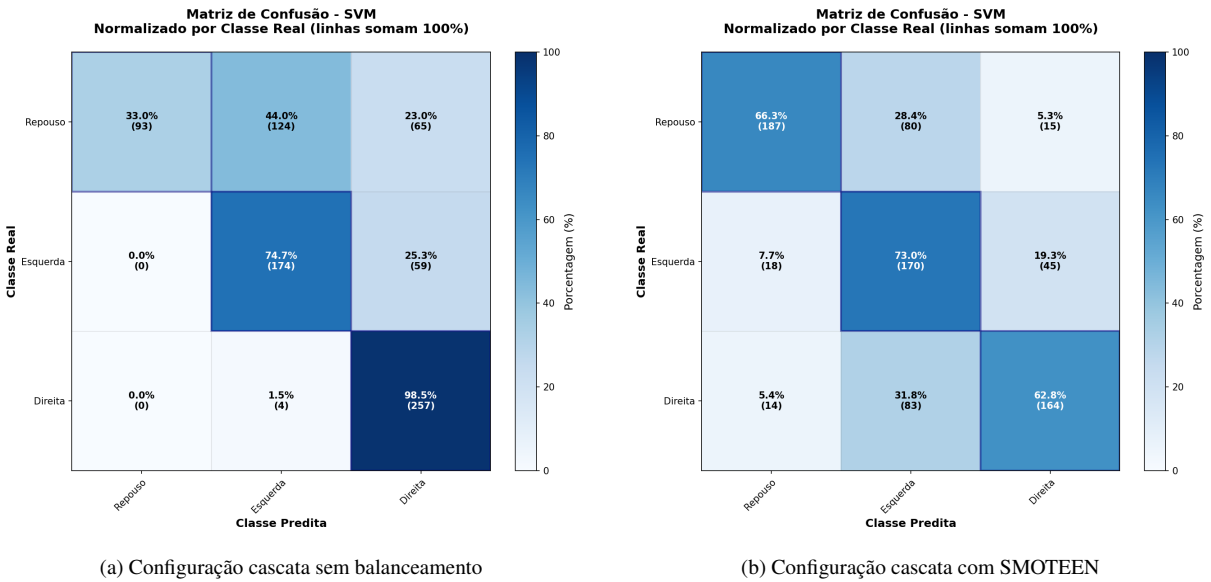
Métrica		Sem Balanceamento		SMOTEENN		Δ
		Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Repouso (Classe 0)	F1	0,496 $\pm$ 0,032	[0,433; 0,558]	0,747 $\pm$ 0,021	[0,706; 0,788]	+0,251
	Precisão	1,000 $\pm$ 0,000	[1,000; 1,000]	0,853 $\pm$ 0,022	[0,810; 0,896]	−0,147
	Recall	0,331 $\pm$ 0,028	[0,277; 0,387]	0,664 $\pm$ 0,028	[0,610; 0,716]	+0,333
Esquerda (Classe 1)	F1	0,651 $\pm$ 0,020	[0,612; 0,689]	0,601 $\pm$ 0,020	[0,562; 0,639]	−0,050
	Precisão	0,577 $\pm$ 0,019	[0,540; 0,617]	0,511 $\pm$ 0,019	[0,473; 0,549]	−0,066
	Recall	0,747 $\pm$ 0,028	[0,691; 0,798]	0,730 $\pm$ 0,028	[0,674; 0,785]	−0,017
Direita (Classe 2)	F1	0,801 $\pm$ 0,013	[0,777; 0,826]	0,675 $\pm$ 0,024	[0,626; 0,721]	−0,126
	Precisão	0,675 $\pm$ 0,017	[0,643; 0,710]	0,733 $\pm$ 0,025	[0,686; 0,782]	+0,058
	Recall	0,985 $\pm$ 0,008	[0,966; 0,996]	0,627 $\pm$ 0,031	[0,563; 0,686]	−0,358

Nota: DP = desvio padrão bootstrap. Δ = SMOTEENN – Sem Balanceamento.

Entretanto, a classe Direita sofreu degradação, a sensibilidade caiu de 0,985 para 0,627 (−35,8 p.p.), representando a maior queda absoluta entre todas as métricas. A decomposição de erros do cascata com SMOTEENN revela 83 erros Direita para Esquerda contra 45 erros Esquerda para Direita, invertendo a assimetria observada sem balanceamento (4 versus 59). Esta inversão sugere que a segunda etapa treinada com dados balanceados perdeu a capacidade de distinguir adequadamente os padrões da classe Direita.

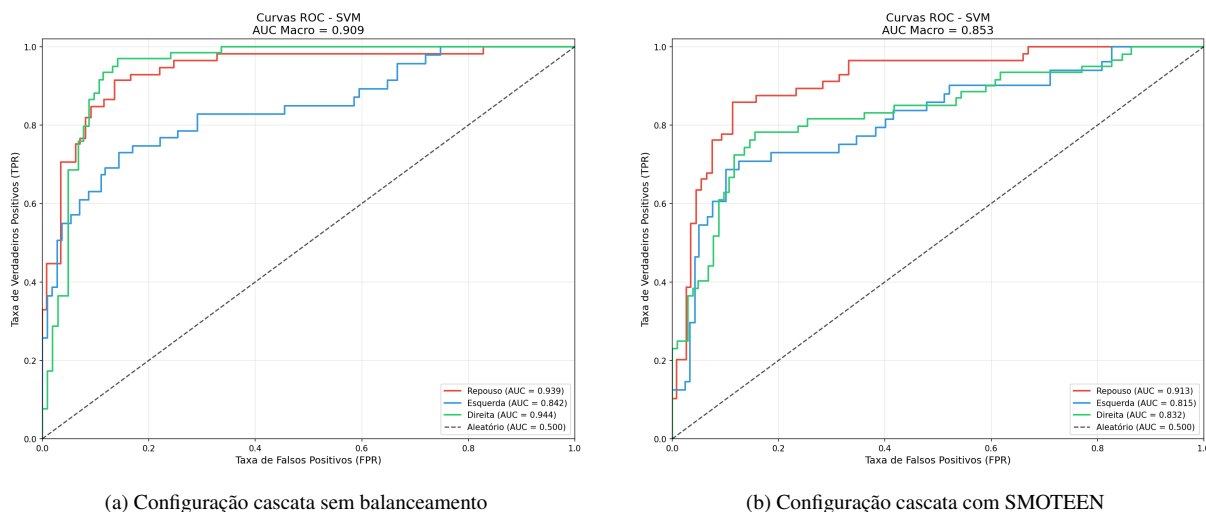
Como pode ser observado na Fig. 6.19 a matriz de confusão do cascata com SMOTEENN (Repouso: 66,31% corretos; Esquerda: 72,96% corretos; Direita: 62,84% corretos) apresenta distribuição de erros mais equilibrada entre classes comparada ao cenário sem balanceamento, porém com desempenho global inferior para as classes de movimento.

Figura 6.19 – Matrizes de confusão percentuais do SVM na configuração cascata: (a) sem balanceamento e (b) com SMOTEEN. Os valores representam a proporção de cada classe verdadeira distribuída entre as classes preditas (normalização por linha).



A Tab. 6.49 apresenta as áreas sob as curvas ROC e PR para a configuração cascata sem e com SMOTEENN. A ROC-AUC macro diminuiu de 0,908 para 0,853 (−5,5 p.p.) e a PR-AUC macro de 0,842 para 0,767 (−7,5 p.p.). A classe Direita apresentou a maior degradação em ROC-AUC (−11,2 p.p.), consistente com a queda na sensibilidade observada nas métricas pontuais. Todas as classes apresentaram redução em ambas as métricas de área sob curva, indicando que o SMOTEENN compromete a capacidade discriminativa do modelo cascata em múltiplos pontos de operação.

Figura 6.20 – Curvas ROC do SVM na configuração cascata: (a) sem balanceamento e (b) com SMOTEEN, com áreas sob a curva por classe e média macro.

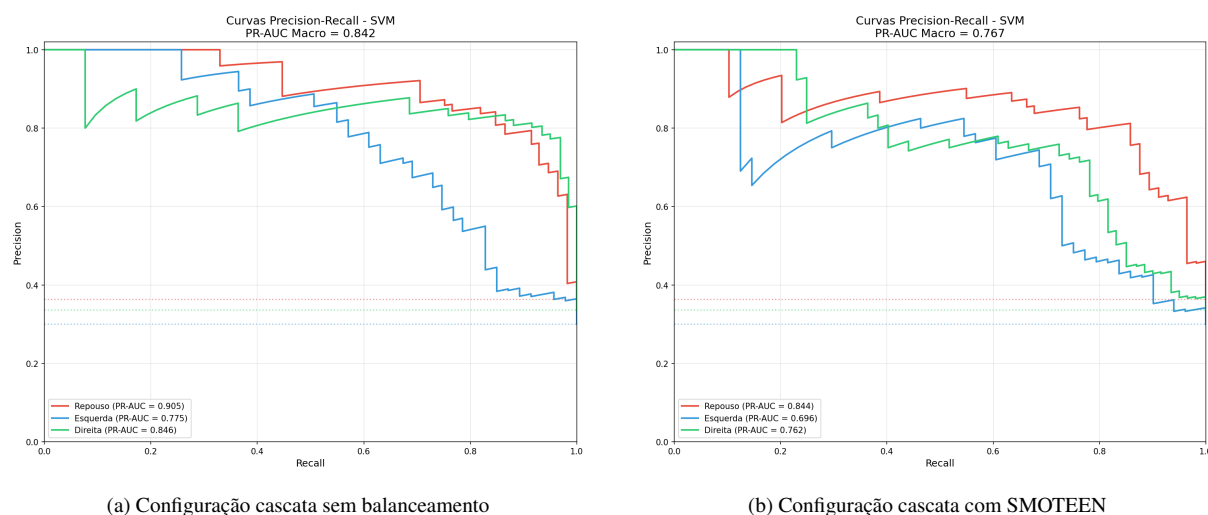


Na configuração sem balanceamento (Figura 6.20a), a ROC-AUC macro de 0,908 é comparável à configuração plana (0,917). A classe Repouso apresenta ROC-AUC de 0,939, Direita 0,944 e Esquerda 0,842. A manutenção de valores elevados de ROC-AUC indica que a capacidade discriminativa do classificador é preservada em diferentes limiares, apesar da degradação observada nas métricas pontuais.

Com SMOTEENN (Figura 6.20b), a ROC-AUC macro reduz para 0,853. A classe Direita apresenta a maior queda (de 0,944 para 0,832), consistente com a redução da sensibilidade de 0,985 para 0,627 observada nas métricas pontuais. As classes Repouso (0,913) e Esquerda (0,815) também apresentam redução em relação ao cenário sem balanceamento.

As curvas Precision-Recall da configuração em cascata são apresentadas na Fig. 6.21, complementando a análise das curvas ROC ao evidenciar o compromisso entre precisão e recall em diferentes pontos de operação. Na configuração sem balanceamento (Figura 6.21a), a PR-AUC macro de 0,842 é ligeiramente inferior à configuração plana (0,867). A classe Repouso apresenta PR-AUC de 0,905, Direita 0,846 e Esquerda 0,775. A redução da PR-AUC de Direita em comparação com a configuração plana (0,925) decorre da propagação de falsos positivos da primeira etapa.

Figura 6.21 – Curvas Precision–Recall do SVM na configuração cascata: (a) sem balanceamento e (b) com SMOTEEN, com áreas sob a curva por classe e média macro.



Com SMOTEENN (Figura 6.21b), a PR-AUC macro reduz para 0,767. Todas as classes apresentam queda: Repouso (0,844), Direita (0,762) e Esquerda (0,696). A redução de 8,4 p.p na PR-AUC de Direita em relação ao cenário sem balanceamento é consistente com a inversão da assimetria de erros na segunda etapa.

Tabela 6.49 – Áreas sob as curvas ROC e PR por classe — SVM cascata sem balanceamento e com SMOTEENN

Classe	ROC-AUC			PR-AUC		
	Sem Bal.	SMOTEENN	Δ	Sem Bal.	SMOTEENN	Δ
Repouso	0,939	0,913	−0,026	0,905	0,844	−0,061
Esquerda	0,842	0,815	−0,027	0,775	0,696	−0,079
Direita	0,944	0,832	−0,112	0,846	0,762	−0,084
Macro	0,908	0,853	−0,055	0,842	0,767	−0,075

Nota: Δ = SMOTEENN – Sem Balanceamento.

A Tab. 6.50 mostra que o SMOTEENN reduziu o MCE de Repouso de 0,545 para 0,345 e de Direita de 0,432 para 0,321, indicando melhoria na calibração em faixas específicas. Entretanto, o ECE de Esquerda aumentou de 0,116 para 0,136, e os valores de ECE permanecem elevados para todas as classes.

Tabela 6.50 – Métricas de calibração por classe — SVM cascata sem balanceamento e com SMOTEENN

Classe	ECE		MCE		Bins com MCE	
	Sem Bal.	SMOTEENN	Sem Bal.	SMOTEENN	Sem Bal.	SMOTEENN
Repouso	0,162	0,158	0,545	0,345	[0,3–0,4]	[0,5–0,6]
Esquerda	0,116	0,136	0,353	0,339	[0,3–0,4]	[0,4–0,5]
Direita	0,119	0,121	0,432	0,321	[0,4–0,5]	[0,4–0,5]

Nota: ECE = Erro de Calibração Esperado. MCE = Erro de Calibração Máximo.

A análise da calibração revela que o SMOTEENN produz efeitos distintos por classe. Para Repouso, o ECE permanece similar (0,162 para 0,158), mas o MCE reduz de 0,545 para 0,345, indicando melhoria na calibração das faixas de probabilidade com maior erro. Para Esquerda, o ECE aumenta de 0,116 para 0,136, enquanto o MCE permanece similar. Para Direita, o ECE permanece estável e o MCE reduz de 0,432 para 0,321. A faixa de probabilidade com maior erro de calibração desloca-se para valores intermediários com SMOTEENN, indicando redistribuição das discrepâncias entre probabilidades preditas e frequências observadas.

6.5.9 Síntese Comparativa

A Tab. 6.51 sintetiza os principais resultados das quatro configurações analisadas. A configuração plana com SMOTEENN apresentou o melhor desempenho em métricas baseadas em predições discretas (acurácia = 0,826, F1-macro = 0,822, MCC = 0,748). A configuração plana sem balanceamento obteve os melhores resultados em métricas baseadas em probabilidades (ROC-AUC macro = 0,917, PR-AUC macro = 0,867). As configurações cascata apresentaram desempenho consistentemente inferior às configurações planas em todas as métricas globais.

Tabela 6.51 – Síntese comparativa das configurações do SVM

Métrica	Plano	Plano+SM	Cascata	Cascata+SM
Acurácia	0,802	0,826	0,676	0,671
F1-macro	0,796	0,822	0,649	0,674
MCC	0,707	0,748	0,558	0,521
ROC-AUC macro	0,917	0,893	0,908	0,853
PR-AUC macro	0,867	0,772	0,842	0,767
Brier Score	0,115	0,112	0,127	0,153

Nota: SM = SMOTEENN. Valores em negrito indicam melhor desempenho para cada métrica.

A decomposição hierárquica em duas etapas não proporcionou ganho de desempenho para o classificador baseado em máquinas de vetores de suporte neste conjunto de dados. A otimização do limiar de decisão na primeira etapa para maximizar a sensibilidade da classe Movimento introduz viés sistemático que se propaga para a classificação final, comprometendo especialmente a discriminação da classe Repouso.

O efeito do SMOTEENN apresentou comportamento divergente entre as arquiteturas: na configuração plana, melhorou métricas de decisão ao custo de métricas probabilísticas; na configuração cascata, redistribuiu o desempenho entre classes sem ganho global, e com degradação da classe Direita (sensibilidade caiu de 0,985 para 0,627).

A adequação do modelo ao problema depende dos requisitos da aplicação. Para sistemas que priorizam a identificação correta de comandos de movimento com mínima perda de intenções, a configuração plana com SMOTEENN oferece o melhor equilíbrio, com F1-macro de 0,822 e sensibilidade acima de 0,74 para todas as classes. Para aplicações que requerem estimativas probabilísticas confiáveis, como sistemas de controle adaptativo baseados em confiança da predição, a configuração plana sem balanceamento é preferível, apresentando ROC-AUC de 0,917 e calibração mais adequada. A configuração cascata não se mostrou adequada para este problema específico, independentemente do uso de balanceamento, dado que a decomposição hierárquica introduz propagação de erros que compromete o desempenho global do sistema.

6.6 XGBoost

O XGBoost na configuração plana multiclasse sem balanceamento, utilizou profundidade máxima de 5 níveis, 75 estimadores, taxa de aprendizado de $5,45 \times 10^{-2}$, parâmetro de poda mínima $\gamma = 0,5$, peso mínimo por folha igual a 5, subamostragem de 57,1% das instâncias por árvore, amostragem de 52,1% das colunas por árvore e 61,2% por nível, regularização L1 com $\alpha = 6,63$ e L2 com $\lambda = 1,62$, e seleção de 30 atributos.

Na arquitetura em cascata sem balanceamento, a primeira etapa, destinada à discriminação entre Movimento e Repouso, empregou profundidade máxima de 5 níveis, 100 estimadores, taxa de aprendizado de $1,69 \times 10^{-2}$, $\gamma = 0,1$, peso mínimo por folha de 20, subamostragem de 68,7%, amostragem de 66,2% das colunas por árvore e 68,1% por nível, regularização L1 com $\alpha = 2,45$ e L2 com $\lambda = 31,32$, e seleção de 10 atributos.

6.6.1 Desempenho Global sem Balanceamento

Conforme apresentado na Tab. 6.52 tem-se que a configuração plana apresentou valores maiores do que a arquitetura em cascata em todas as métricas listadas. A acurácia foi 0,776 no plano e 0,733 no cascata ($\Delta = -0,042$), enquanto o $F1_{\text{macro}}$ foi 0,770 no plano e 0,730 no cascata ($\Delta = -0,041$). O MCC reduziu de 0,672 para 0,610 ($\Delta = -0,063$), indicando decréscimo na concordância global entre rótulos verdadeiros e preditos quando a decomposição hierárquica é empregada. Nas métricas baseadas em probabilidades, ROC-AUC macro reduziu de 0,936 para 0,894, PR-AUC macro reduziu de 0,890 para 0,826, e o Brier Score aumentou de 0,103 para 0,127, descrevendo maior erro quadrático médio das probabilidades preditas no cascata.

Tabela 6.52 – Métricas de desempenho global do XGBoost nas configurações plana e cascata sem balanceamento, com intervalos de confiança bootstrap de 95%

Métrica	Plano		Cascata		Δ
	Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Acurácia	0,776 $\pm$ 0,015	[0,747; 0,804]	0,733 $\pm$ 0,015	[0,702; 0,763]	-0,042
Acurácia Balanceada	0,773 $\pm$ 0,015	[0,745; 0,803]	0,733 $\pm$ 0,016	[0,703; 0,763]	-0,040
F1-macro	0,770 $\pm$ 0,015	[0,741; 0,800]	0,730 $\pm$ 0,016	[0,698; 0,760]	-0,041
F1-weighted	0,774 $\pm$ 0,015	[0,744; 0,803]	0,732 $\pm$ 0,016	[0,700; 0,763]	-0,042
Precisão (macro)	0,786 $\pm$ 0,014	[0,759; 0,814]	0,748 $\pm$ 0,015	[0,717; 0,777]	-0,039
Recall (macro)	0,773 $\pm$ 0,015	[0,745; 0,803]	0,733 $\pm$ 0,016	[0,703; 0,763]	-0,040
MCC	0,672 $\pm$ 0,021	[0,631; 0,714]	0,610 $\pm$ 0,023	[0,565; 0,653]	-0,063
ROC-AUC (macro)	0,936 $\pm$ 0,007	[0,922; 0,949]	0,894 $\pm$ 0,010	[0,874; 0,912]	-0,042
PR-AUC (macro)	0,890 $\pm$ 0,011	[0,867; 0,911]	0,826 $\pm$ 0,015	[0,796; 0,856]	-0,063
Brier Score	0,103 $\pm$ 0,005	[0,093; 0,113]	0,127 $\pm$ 0,006	[0,116; 0,139]	+0,024

Nota: $\Delta = \text{Cascata} - \text{Plano}$.

A análise por *bootstrap* indicou largura moderada dos intervalos de confiança para a maioria das métricas. Na configuração plana, excederam o limiar operacional de 0,10 as larguras do IC 95% do *recall* da classe Esquerda (0,116), do *recall* da classe Repouso (0,106) e da precisão da classe Esquerda (0,104). No cascata, observou-se padrão análogo, com larguras de 0,120, 0,114 e 0,103 para as mesmas métricas, respectivamente.

6.6.2 Desempenho por Classe sem Balanceamento

Conforme detalhado na Tab. 6.53, a comparação por classe explicita como as diferenças globais se distribuem entre as categorias. Para Repouso, o *recall* foi 0,716 no plano e 0,649 no cascata, com redução concomitante de F1 (0,799 $\rightarrow$ 0,745). Para Direita, a configuração plana

manteve *recall* de 0,935 e F1 de 0,810, enquanto no cascata os valores foram 0,882 e 0,763, respectivamente. Para Esquerda, o *recall* permaneceu em 0,669 em ambas as configurações, ao passo que a precisão foi menor no cascata (0,740 $\rightarrow$ 0,694), refletindo aumento de falsos positivos associados a essa classe.

Tabela 6.53 – Métricas de desempenho do XGBoost por classe nas configurações plana e cascata sem balanceamento

Métrica		Plano		Cascata		Δ
		Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Repouso (Classe 0)	F1	0,799 $\pm$ 0,020	[0,760; 0,837]	0,745 $\pm$ 0,022	[0,700; 0,786]	−0,054
	Precisão	0,905 $\pm$ 0,018	[0,869; 0,940]	0,875 $\pm$ 0,021	[0,833; 0,915]	−0,030
	Recall	0,716 $\pm$ 0,027	[0,663; 0,770]	0,649 $\pm$ 0,029	[0,592; 0,706]	−0,067
Esquerda (Classe 1)	F1	0,702 $\pm$ 0,024	[0,656; 0,747]	0,681 $\pm$ 0,024	[0,633; 0,726]	−0,021
	Precisão	0,740 $\pm$ 0,026	[0,688; 0,791]	0,694 $\pm$ 0,026	[0,642; 0,745]	−0,045
	Recall	0,669 $\pm$ 0,031	[0,609; 0,725]	0,669 $\pm$ 0,031	[0,609; 0,730]	−0,001
Direita (Classe 2)	F1	0,810 $\pm$ 0,014	[0,783; 0,836]	0,763 $\pm$ 0,016	[0,732; 0,793]	−0,047
	Precisão	0,714 $\pm$ 0,019	[0,678; 0,749]	0,673 $\pm$ 0,019	[0,637; 0,712]	−0,041
	Recall	0,935 $\pm$ 0,015	[0,904; 0,962]	0,882 $\pm$ 0,021	[0,843; 0,920]	−0,053

Nota: DP = desvio padrão bootstrap. Δ = Cascata – Plano.

6.6.3 Análise por Etapa da Configuração em Cascata sem Balanceamento

A Tab. 6.54 apresenta as métricas de desempenho para cada etapa da configuração em cascata, incluindo a comparação entre os cenários condicionado (primeira etapa real) e ideal (primeira etapa perfeita, ou oráculo) para a segunda etapa.

Tabela 6.54 – Métricas por etapa da configuração em cascata — XGBoost

Métrica	Etapa 1		E2 Condicionada		E2 Ideal	
	Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%
Acurácia	0,839 $\pm$ 0,013	[0,813; 0,863]	0,825 $\pm$ 0,018	[0,789; 0,859]	0,834 $\pm$ 0,017	[0,800; 0,866]
Acurácia Bal.	0,798 $\pm$ 0,015	[0,768; 0,827]	0,817 $\pm$ 0,018	[0,783; 0,851]	0,830 $\pm$ 0,017	[0,796; 0,861]
F1	0,882 $\pm$ 0,010	[0,861; 0,902]	0,848 $\pm$ 0,017	[0,813; 0,879]	0,853 $\pm$ 0,017	[0,819; 0,884]
Precisão	0,825 $\pm$ 0,016	[0,794; 0,857]	0,796 $\pm$ 0,024	[0,746; 0,842]	0,801 $\pm$ 0,024	[0,752; 0,846]
Sensibilidade	0,947 $\pm$ 0,010	[0,927; 0,965]	0,909 $\pm$ 0,018	[0,871; 0,942]	0,912 $\pm$ 0,018	[0,876; 0,946]
MCC	0,646 $\pm$ 0,028	[0,591; 0,699]	0,650 $\pm$ 0,035	[0,583; 0,717]	0,672 $\pm$ 0,034	[0,605; 0,736]
ROC-AUC	0,938 $\pm$ 0,008	[0,922; 0,954]	0,928 $\pm$ 0,011	[0,905; 0,949]	0,931 $\pm$ 0,011	[0,908; 0,951]
PR-AUC	0,965 $\pm$ 0,006	[0,953; 0,975]	0,935 $\pm$ 0,012	[0,911; 0,955]	0,934 $\pm$ 0,012	[0,908; 0,955]
Brier Score	0,128 $\pm$ 0,007	[0,114; 0,141]	0,129 $\pm$ 0,014	[0,103; 0,157]	0,122 $\pm$ 0,013	[0,098; 0,149]

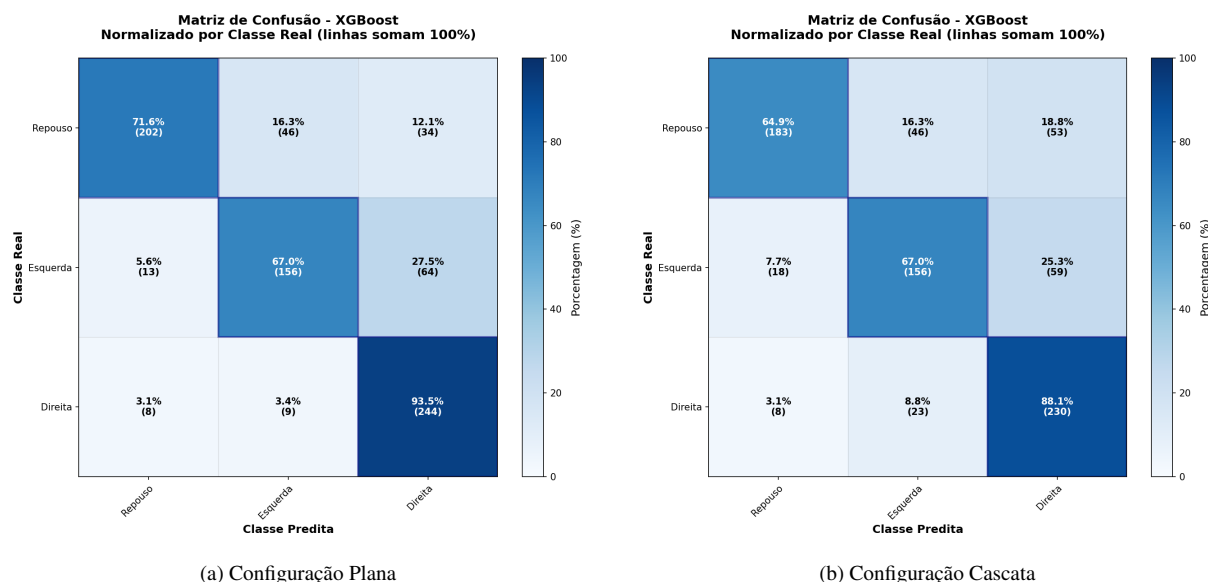
A primeira etapa apresentou capacidade de detecção de movimento satisfatória, com sensibilidade de 0,947 e ROC-AUC de 0,938. A análise identificou o limiar ótimo de 0,62, resultando em F1-macro de 0,721 contra 0,672 do limiar padrão, representando melhoria relativa de 7,31%.

A comparação entre os cenários E2 Ideal e E2 Condicionada permite quantificar o impacto da propagação de erros da primeira etapa. A degradação observada foi moderada: $\Delta_{F1} = -0,004$, $\Delta_{\text{acurácia balanceada}} = -0,012$, $\Delta_{\text{precisão}} = -0,006$, e $\Delta_{\text{sensibilidade}} = -0,003$. Esta resiliência pode ser atribuída a dois fatores: (i) a taxa de falsos positivos da primeira etapa em termos absolutos (99 de 282 amostras de Repouso, ou seja, 35,1% das amostras de Repouso, mas apenas 12,8% do total de amostras); e (ii) a possibilidade de que as características espectrais das amostras de Repouso incorretamente encaminhadas à segunda etapa ocupem regiões do espaço de atributos que são razoavelmente bem separadas pelas fronteiras de decisão da segunda etapa.

6.6.4 Matriz de Confusão sem Balanceamento

Conforme ilustrado na Fig. 6.22, as matrizes de confusão percentuais das configurações plana e cascata descrevem, por normalização em linha, a distribuição das classes preditas para cada classe verdadeira. Observa-se que a maior concentração na diagonal ocorre na classe Direita, em consonância com os valores de *recall* apresentados na Tab. 6.53 (0,935 no plano e 0,882 no cascata). Para Repouso e Esquerda, a diagonal é menos concentrada, compatível com os menores valores de *recall* observados nas duas configurações.

Figura 6.22 – Matrizes de confusão percentuais do XGBoost nas configurações plana (a) e cascata (b) sem balanceamento. Os valores representam a proporção de cada classe verdadeira distribuída entre as classes preditas (normalização por linha).



6.6.5 Análise das Curvas ROC e PR sem Balanceamento

Conforme apresentado na Tab. 6.55, a configuração plana apresentou valores maiores de ROC-AUC e PR-AUC em todas as classes. A maior diferença absoluta ocorreu na PR-AUC da classe Direita, que reduziu de 0,901 (plano) para 0,789 (cascata), indicando perda de precisão ao longo de diferentes níveis de sensibilidade quando a arquitetura em cascata é aplicada.

Tabela 6.55 – Áreas sob as curvas ROC e PR por classe e configuração — XGBoost sem balanceamento

Classe	ROC-AUC			PR-AUC		
	Plano	Cascata	Δ	Plano	Cascata	Δ
Repouso	0,954	0,939	−0,015	0,929	0,891	−0,038
Esquerda	0,898	0,842	−0,056	0,839	0,797	−0,042
Direita	0,955	0,901	−0,054	0,901	0,789	−0,112
Macro	0,936	0,894	−0,042	0,889	0,826	−0,063

Nota: Δ = Cascata – Plano.

6.6.6 Análise de Calibração sem Balanceamento

Conforme apresentado na Tab. 6.56, a calibração das probabilidades preditas foi avaliada por ECE e MCE por classe, utilizando binagem uniforme com 10 intervalos. Na configu-

ração plana, os valores de ECE foram 0,108 (Repouso), 0,071 (Esquerda) e 0,089 (Direita). Na configuração em cascata, os valores aumentaram para 0,133, 0,128 e 0,124, respectivamente, indicando maior discrepância média entre probabilidade predita e frequência empírica por classe. Esse comportamento pode estar associado ao mecanismo de composição de probabilidades no cascata, no qual a combinação entre etapas pode alterar a distribuição final de confiança, especialmente quando há propagação de erros a partir da primeira etapa.

Tabela 6.56 – Métricas de calibração por classe e configuração — XGBoost sem balanceamento

Classe	ECE		MCE		Intervalo com MCE	
	Plano	Cascata	Plano	Cascata	Plano	Cascata
Repouso	0,108	0,133	0,521	0,380	[0,3–0,4]	[0,3–0,4]
Esquerda	0,071	0,128	0,359	0,663	[0,5–0,6]	[0,3–0,4]
Direita	0,089	0,124	0,329	0,637	[0,5–0,6]	[0,6–0,7]

6.6.7 Efeito do SMOTEENN na Configuração Plana

A aplicação de SMOTEENN na configuração plana multiclasse alterou a combinação de hiperparâmetros selecionada na otimização: 75 estimadores, profundidade máxima de 5, taxa de aprendizado de 0,078, parâmetro de poda $\gamma = 0,3$, subamostragem de 73,4%, amostragem de 64,9% das colunas por árvore e 76,4% por nível, regularização L1 com $\alpha = 3,20$ e L2 com $\lambda = 15,89$, e seleção de 37 atributos. Em validação cruzada, obteve-se $F1_{\text{macro}} = 0,290$, valor inferior ao observado na condição sem balanceamento.

Conforme apresentado na Tab. 6.57, a aplicação de SMOTEENN elevou as métricas de decisão no limiar padrão, com Δ de +0,013 em acurácia e de +0,016 em $F1$ -macro. Em contrapartida, as métricas baseadas em probabilidades reduziram, com ROC-AUC macro passando de 0,936 para 0,927 e PR-AUC macro de 0,890 para 0,868. O Brier Score aumentou de 0,103 para 0,106, indicando aumento do erro quadrático médio das probabilidades previstas.

Tabela 6.57 – Comparação do XGBoost plano sem balanceamento e com SMOTEENN

Métrica	Sem Balanceamento		SMOTEENN		Δ
	Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Acurácia	0,776 $\pm$ 0,015	[0,747; 0,804]	0,789 $\pm$ 0,015	[0,759; 0,817]	+0,013
Acurácia Balanceada	0,773 $\pm$ 0,015	[0,745; 0,803]	0,786 $\pm$ 0,015	[0,755; 0,815]	+0,013
F1-macro	0,770 $\pm$ 0,015	[0,741; 0,800]	0,786 $\pm$ 0,015	[0,755; 0,814]	+0,016
F1-weighted	0,774 $\pm$ 0,015	[0,744; 0,803]	0,790 $\pm$ 0,015	[0,759; 0,818]	+0,016
Precisão (macro)	0,786 $\pm$ 0,014	[0,759; 0,814]	0,792 $\pm$ 0,014	[0,762; 0,820]	+0,006
Recall (macro)	0,773 $\pm$ 0,015	[0,745; 0,803]	0,786 $\pm$ 0,015	[0,755; 0,815]	+0,013
MCC	0,672 $\pm$ 0,021	[0,631; 0,714]	0,686 $\pm$ 0,022	[0,641; 0,728]	+0,014
ROC-AUC (macro)	0,936 $\pm$ 0,007	[0,922; 0,949]	0,927 $\pm$ 0,007	[0,911; 0,941]	-0,009
PR-AUC (macro)	0,890 $\pm$ 0,011	[0,867; 0,911]	0,868 $\pm$ 0,013	[0,842; 0,892]	-0,022
Brier Score	0,103 $\pm$ 0,005	[0,093; 0,113]	0,106 $\pm$ 0,004	[0,097; 0,115]	+0,003

Nota: Δ = SMOTEENN – Sem Balanceamento.

A comparação por classe, apresentada na Tab. 6.58, indica redistribuição do desempenho entre as classes. Em Repouso, o recall aumentou de 0,716 para 0,784 (+0,068) e o $F1$ de 0,799 para 0,843. Em Esquerda, o recall aumentou de 0,669 para 0,708 (+0,039), acompanhado de redução de 0,019 em precisão. Em Direita, observou-se redução de 0,069 no recall (de 0,935 para 0,866), com aumento de 0,028 em precisão.

Tabela 6.58 – Desempenho por classe do XGBoost plano sem balanceamento e com SMOTEENN

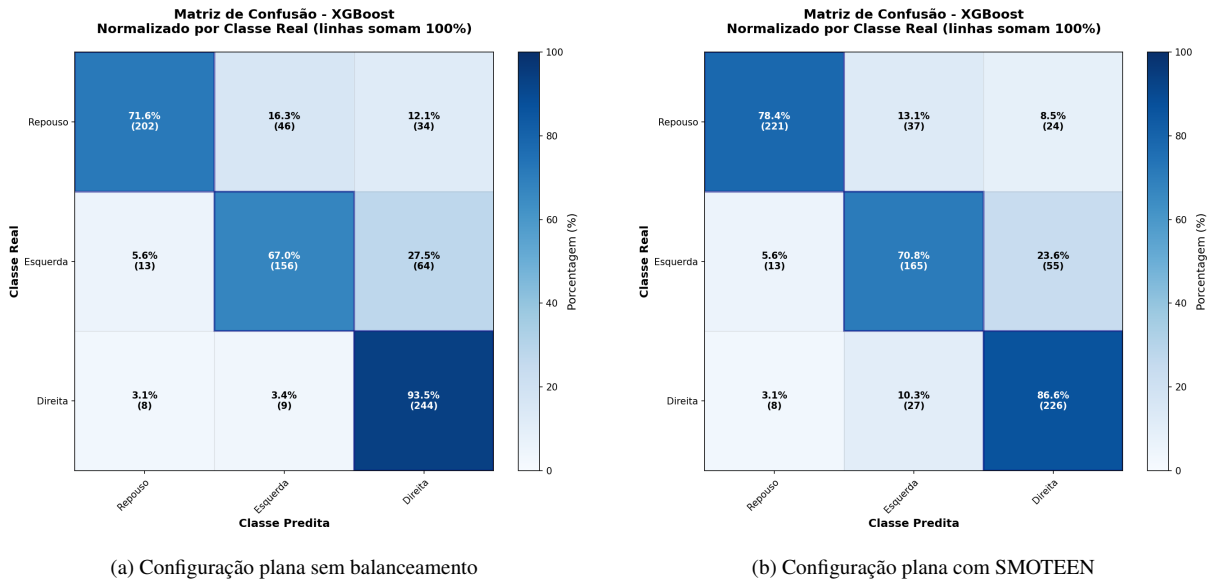
Métrica		Sem Balanceamento		SMOTEENN		Δ
		Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Repouso (Classe 0)	F1	0,799 $\pm$ 0,020	[0,760; 0,837]	0,843 $\pm$ 0,017	[0,809; 0,876]	+0,044
	Precisão	0,905 $\pm$ 0,018	[0,869; 0,940]	0,913 $\pm$ 0,017	[0,879; 0,946]	+0,008
	Recall	0,716 $\pm$ 0,027	[0,663; 0,770]	0,784 $\pm$ 0,025	[0,734; 0,830]	+0,068
Esquerda (Classe 1)	F1	0,702 $\pm$ 0,024	[0,656; 0,747]	0,715 $\pm$ 0,023	[0,668; 0,758]	+0,013
	Precisão	0,740 $\pm$ 0,026	[0,688; 0,791]	0,721 $\pm$ 0,026	[0,671; 0,773]	-0,019
	Recall	0,669 $\pm$ 0,031	[0,609; 0,725]	0,708 $\pm$ 0,030	[0,648; 0,764]	+0,039
Direita (Classe 2)	F1	0,810 $\pm$ 0,014	[0,783; 0,836]	0,799 $\pm$ 0,017	[0,766; 0,829]	-0,011
	Precisão	0,714 $\pm$ 0,019	[0,678; 0,749]	0,742 $\pm$ 0,020	[0,700; 0,781]	+0,028
	Recall	0,935 $\pm$ 0,015	[0,904; 0,962]	0,866 $\pm$ 0,022	[0,824; 0,908]	-0,069

Nota: Δ = SMOTEENN – Sem Balanceamento.

Os resultados por classe são coerentes com as matrizes de confusão percentuais apresentadas na Fig. 6.23. Na configuração plana sem balanceamento, as sensibilidades foram 71,63% (Repouso), 66,95% (Esquerda) e 93,49% (Direita). Com SMOTEENN, os valores passaram

para 78,37%, 70,82% e 86,59%, respectivamente, indicando aumento de acerto em Repouso e Esquerda e redução em Direita.

Figura 6.23 – Matrizes de confusão percentuais do XGBoost na configuração plana: (a) sem balanceamento e (b) com SMOTEEN. Os valores representam a proporção de cada classe verdadeira distribuída entre as classes preditas (normalização por linha).



As curvas ROC e Precision–Recall correspondentes às duas condições são apresentadas nas Figs. 6.24 e 6.25. Em concordância com a Tab. 6.59, observa-se redução das áreas sob as curvas (média macro) após a aplicação de SMOTEENN, apesar do aumento nas métricas calculadas no limiar padrão.

Figura 6.24 – Curvas ROC do XGBoost na configuração plana: (a) sem balanceamento e (b) com SMOTEEN.

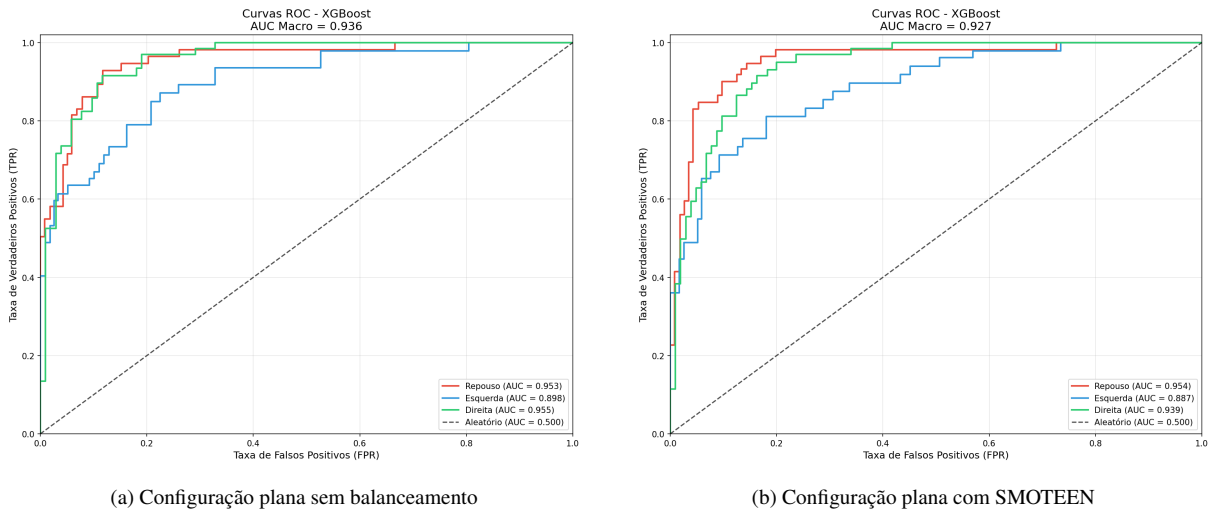
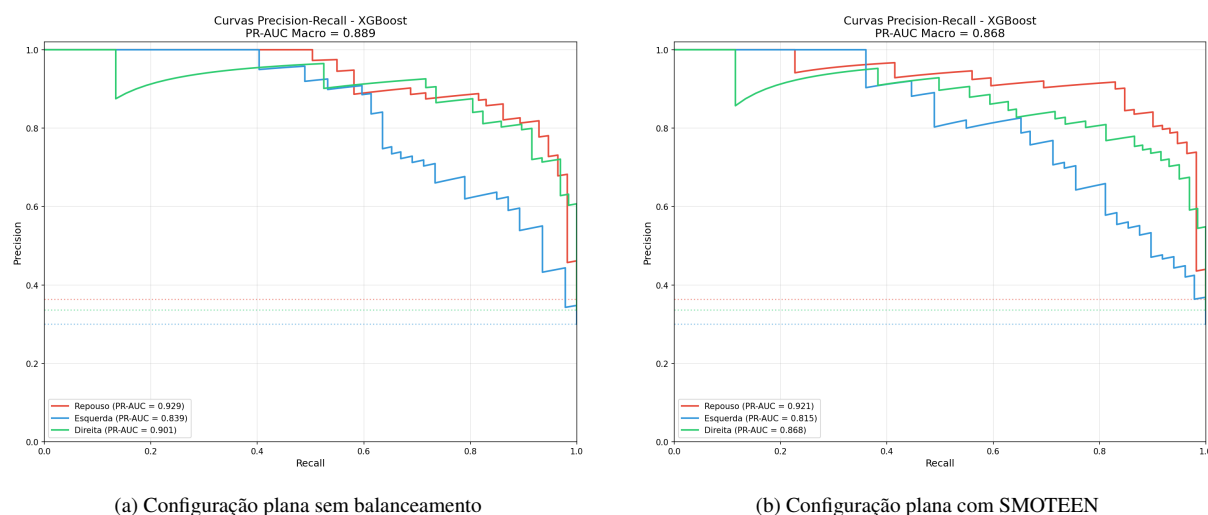


Figura 6.25 – Curvas Precision–Recall do XGBoost na configuração plana: (a) sem balanceamento e (b) com SMOTEEN.



O modelo XGBoost na configuração plana com SMOTEENN apresentou desempenho discriminativo elevado para as três classes avaliadas. Conforme os resultados da Tab. 6.59 a classe Repouso obteve os melhores resultados, com ROC-AUC de 0,954 e PR-AUC de 0,921, indicando excelente capacidade de distinção entre o estado de repouso e os demais estados motores, além de boa precisão ao longo de diferentes limiares de decisão.

Tabela 6.59 – Áreas sob as curvas ROC e PR por classe — XGBoost plano sem balanceamento e com SMOTEENN

Classe	ROC-AUC			PR-AUC		
	Sem Bal.	SMOTEENN	Δ	Sem Bal.	SMOTEENN	Δ
Repouso	0,939	0,954	+0,015	0,891	0,921	+0,030
Esquerda	0,842	0,887	+0,045	0,797	0,815	+0,018
Direita	0,901	0,939	+0,038	0,789	0,868	+0,079
Macro	0,894	0,927	+0,033	0,826	0,868	+0,042

Nota: Δ = SMOTEENN – Sem Balanceamento.

A classe Direita obteve ROC-AUC de 0,939 e PR-AUC de 0,868, sugerindo que o classificador consegue identificar adequadamente a imaginação de movimento da mão direita. A classe Esquerda, embora apresente valores inferiores às demais (ROC-AUC de 0,887 e PR-AUC de 0,815), ainda mantém um bom desempenho, sendo a classe de maior dificuldade de classificação.

Na média macro, o modelo alcançou ROC-AUC de 0,927 e PR-AUC de 0,868, valores que indicam boa generalização entre as classes. A diferença entre as duas métricas sugere que, apesar da aplicação do SMOTEENN, ainda existe algum impacto do desbalanceamento residual sobre a precisão em determinados pontos de operação.

6.6.8 Efeito do SMOTEENN na Configuração Cascata

Na arquitetura em cascata com SMOTEENN, a primeira etapa empregou 100 estimadores, profundidade máxima de 3, taxa de aprendizado de 0,084, peso mínimo por folha de 10, subamostragem de 79,5% e seleção de 6 atributos, obtendo $F1_{\text{macro}} = 0,431$ em validação cruzada. O limiar de decisão otimizado foi 0,66, acima do padrão (0,50), com incremento de 1,5% em relação ao limiar padrão.

Conforme apresentado na Tab. 6.60, o SMOTEENN elevou as métricas de decisão na arquitetura em cascata, com aumento de 0,032 em acurácia e de 0,035 em $F1$ -macro, além de incremento de 0,044 em MCC. Em contraste, as áreas sob curva reduziram ligeiramente (ROC-AUC macro: 0,894 para 0,892; PR-AUC macro: 0,826 para 0,815).

Tabela 6.60 – Comparação do XGBoost cascata sem balanceamento e com SMOTEENN

Métrica	Sem Balanceamento		SMOTEENN		Δ
	Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Acurácia	0,733 $\pm$ 0,015	[0,702; 0,763]	0,766 $\pm$ 0,015	[0,736; 0,795]	+0,032
Acurácia Balanceada	0,733 $\pm$ 0,016	[0,703; 0,763]	0,767 $\pm$ 0,015	[0,736; 0,797]	+0,034
$F1$ -macro	0,730 $\pm$ 0,016	[0,698; 0,760]	0,764 $\pm$ 0,015	[0,734; 0,794]	+0,035
$F1$ -weighted	0,732 $\pm$ 0,016	[0,700; 0,763]	0,767 $\pm$ 0,015	[0,737; 0,797]	+0,035
Precisão (macro)	0,748 $\pm$ 0,015	[0,717; 0,777]	0,771 $\pm$ 0,015	[0,742; 0,800]	+0,024
Recall (macro)	0,733 $\pm$ 0,016	[0,703; 0,763]	0,767 $\pm$ 0,015	[0,736; 0,797]	+0,034
MCC	0,610 $\pm$ 0,023	[0,565; 0,653]	0,653 $\pm$ 0,023	[0,609; 0,698]	+0,044
ROC-AUC (macro)	0,894 $\pm$ 0,010	[0,874; 0,912]	0,892 $\pm$ 0,010	[0,871; 0,910]	−0,002
PR-AUC (macro)	0,826 $\pm$ 0,015	[0,796; 0,856]	0,815 $\pm$ 0,016	[0,782; 0,846]	−0,012
Brier Score	0,127 $\pm$ 0,006	[0,116; 0,139]	0,128 $\pm$ 0,006	[0,117; 0,140]	+0,001

Nota: Δ = SMOTEENN – Sem Balanceamento.

A comparação por classe, apresentada na Tab. 6.61, evidencia mudanças assimétricas entre precisão e recall. Em Repouso, o recall aumentou de 0,649 para 0,702 (+0,053), com redução de 0,011 em precisão. Em Esquerda, o recall aumentou de 0,669 para 0,751 (+0,082), acompanhado de redução de 0,050 em precisão. Em Direita, a precisão aumentou de 0,673 para 0,805 (+0,132), enquanto o recall reduziu de 0,882 para 0,847 (−0,035).

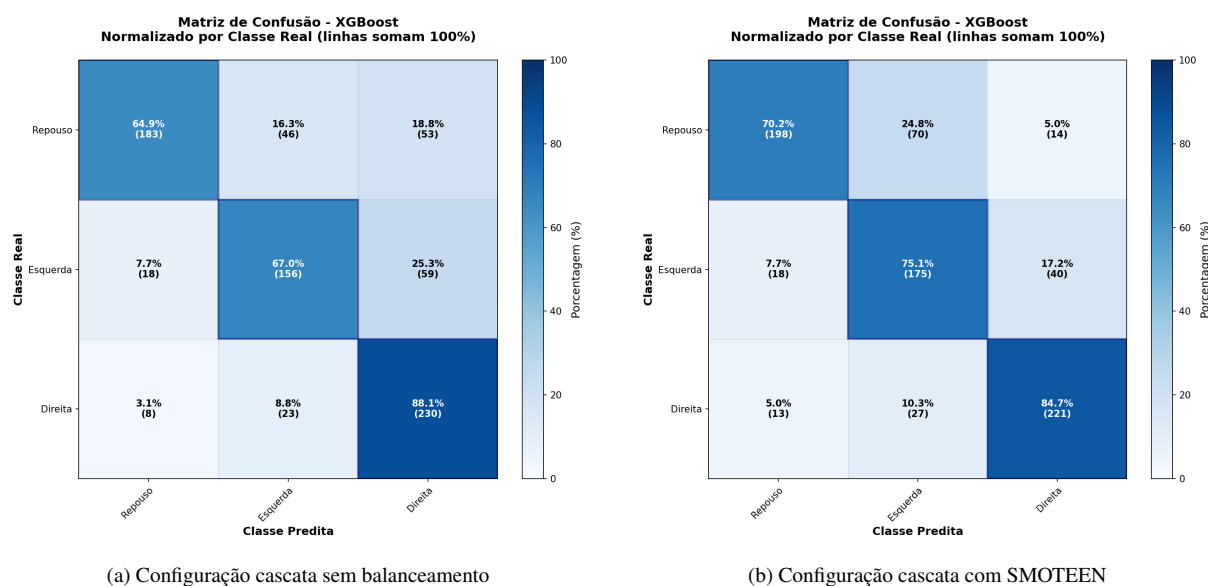
Tabela 6.61 – Desempenho por classe do XGBoost cascata sem balanceamento e com SMOTEENN

Métrica		Sem Balanceamento		SMOTEENN		Δ
		Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Repouso (Classe 0)	F1	0,745 $\pm$ 0,022	[0,700; 0,786]	0,775 $\pm$ 0,021	[0,733; 0,813]	+0,030
	Precisão	0,875 $\pm$ 0,021	[0,833; 0,915]	0,864 $\pm$ 0,021	[0,824; 0,904]	−0,011
	Recall	0,649 $\pm$ 0,029	[0,592; 0,706]	0,702 $\pm$ 0,028	[0,645; 0,755]	+0,053
Esquerda (Classe 1)	F1	0,681 $\pm$ 0,024	[0,633; 0,726]	0,693 $\pm$ 0,021	[0,651; 0,733]	+0,012
	Precisão	0,694 $\pm$ 0,026	[0,642; 0,745]	0,644 $\pm$ 0,023	[0,599; 0,689]	−0,050
	Recall	0,669 $\pm$ 0,031	[0,609; 0,730]	0,751 $\pm$ 0,028	[0,700; 0,807]	+0,082
Direita (Classe 2)	F1	0,763 $\pm$ 0,016	[0,732; 0,793]	0,825 $\pm$ 0,017	[0,792; 0,858]	+0,062
	Precisão	0,673 $\pm$ 0,019	[0,637; 0,712]	0,805 $\pm$ 0,020	[0,764; 0,844]	+0,132
	Recall	0,882 $\pm$ 0,021	[0,843; 0,920]	0,847 $\pm$ 0,023	[0,801; 0,889]	−0,035

Nota: Δ = SMOTEENN – Sem Balanceamento.

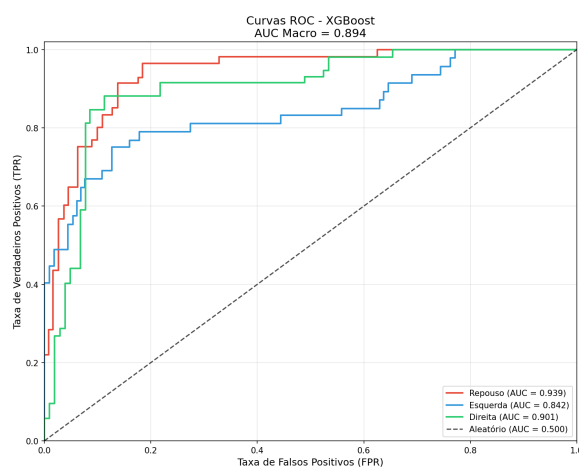
Os efeitos por classe refletem-se nas matrizes de confusão percentuais apresentadas na Fig. 6.26. Na configuração cascata sem balanceamento, as sensibilidades foram 64,88% (Repouso), 66,95% (Esquerda) e 88,16% (Direita). Com SMOTEENN, observaram-se 70,20%, 75,18% e 84,77%, respectivamente, com aumento de acerto em Repouso e Esquerda e redução em Direita.

Figura 6.26 – Matrizes de confusão percentuais do XGBoost na configuração cascata: (a) sem balanceamento e (b) com SMOTEENN.

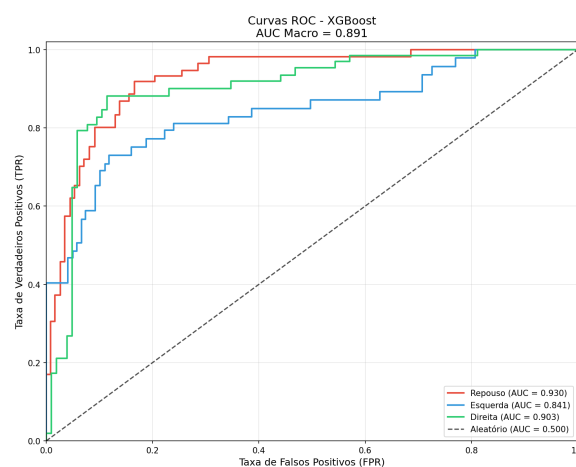


As curvas ROC e Precision–Recall para a arquitetura em cascata são apresentadas nas Figs. 6.27 e 6.28. A comparação visual é consistente com os valores médios macro da Tab. 6.62, que indicam alteração pequena em ROC-AUC e redução em PR-AUC após a reamostragem.

Figura 6.27 – Curvas ROC do XGBoost na configuração cascata: (a) sem balanceamento e (b) com SMOTEEN.

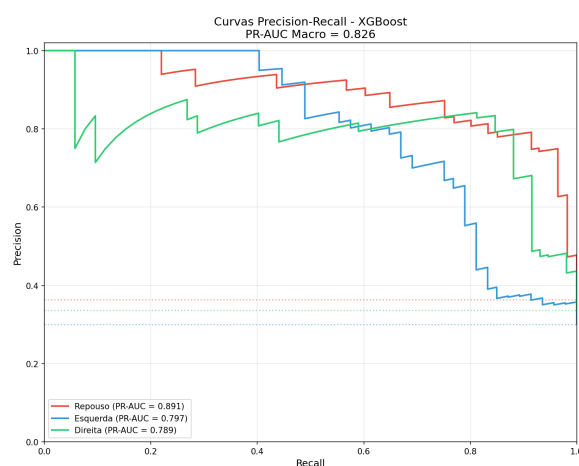


(a) Configuração cascata sem balanceamento

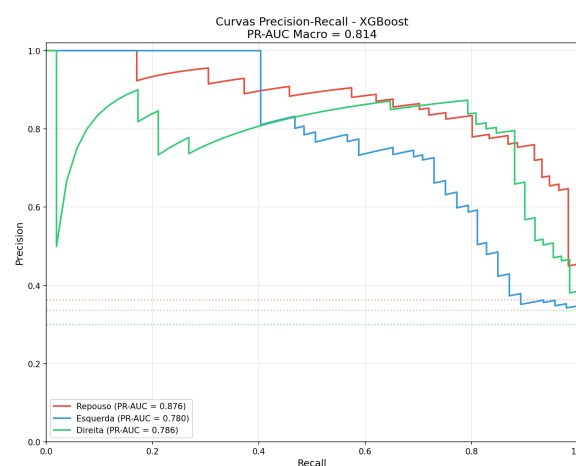


(b) Configuração cascata com SMOTEEN

Figura 6.28 – Curvas Precision–Recall do XGBoost na configuração cascata: (a) sem balanceamento e (b) com SMOTEEN.



(a) Configuração cascata sem balanceamento



(b) Configuração cascata com SMOTEEN

Em relação à ROC-AUC, pela Tab. 6.62, observa-se que as diferenças entre as duas abordagens são mínimas. A classe Repouso apresentou redução de 0,009, enquanto a classe Esquerda permaneceu praticamente estável com variação de apenas 0,001. A classe Direita foi a única a registrar um pequeno incremento (+0,002), embora esse ganho seja marginal e pos-

sivelmente atribuível à variabilidade experimental. Na média macro, a diferença foi de apenas 0,003, indicando que ambas as configurações apresentam capacidade discriminativa equivalente entre classes positivas e negativas.

Tabela 6.62 – Áreas sob as curvas ROC e PR por classe — XGBoost cascata sem balanceamento e com SMOTEENN

Classe	ROC-AUC			PR-AUC		
	Sem Bal.	SMOTEENN	Δ	Sem Bal.	SMOTEENN	Δ
Repouso	0,939	0,930	−0,009	0,891	0,876	−0,015
Esquerda	0,842	0,841	−0,001	0,797	0,780	−0,017
Direita	0,901	0,903	+0,002	0,789	0,786	−0,003
Macro	0,894	0,891	−0,003	0,826	0,814	−0,012

Nota: Δ = SMOTEENN – Sem Balanceamento.

Quanto à PR-AUC, a classe Repouso sofreu queda de 0,015, a classe Esquerda de 0,017 e a classe Direita de 0,003. A média macro reduziu em 0,012, sugerindo que o SMOTEENN pode ter introduzido ruído ou removido instâncias informativas das regiões de fronteira entre classes.

6.6.9 Síntese Comparativa

Conforme sintetizado na Tab. 6.63, a configuração plana apresentou valores maiores do que a arquitetura em cascata, tanto sem balanceamento quanto com SMOTEENN, em todas as métricas de decisão apresentadas. No arranjo em cascata, erros da primeira etapa propagam-se para a decisão final; na condição sem balanceamento.

Tabela 6.63 – Síntese comparativa das configurações do XGBoost

Métrica	Plano	Plano + SMOTEENN	Cascata	Cascata + SMOTEENN
Acurácia	0,776	0,789	0,733	0,766
Acurácia Balanceada	0,773	0,786	0,733	0,767
F1-macro	0,770	0,786	0,730	0,764
MCC	0,672	0,686	0,610	0,653
ROC-AUC (macro)	0,936	0,927	0,894	0,892
PR-AUC (macro)	0,890	0,868	0,826	0,815
Brier Score	0,103	0,106	0,127	0,128

Nota: Valores em negrito indicam o melhor desempenho para cada métrica.

A aplicação de SMOTEENN aumentou as métricas calculadas no limiar padrão, com incrementos de 0,013 a 0,035 em acurácia e em $F1$ -macro, enquanto as métricas baseadas em probabilidades apresentaram redução, como indicado por ROC-AUC e PR-AUC. Esse comportamento caracteriza um deslocamento do ponto de operação de decisão sem melhoria proporcional da separação probabilística ao longo de limiares.

A configuração plana com SMOTEENN obteve o melhor desempenho em métricas de classificação pontual (acurácia = 0,789, $F1$ -macro = 0,786, MCC = 0,686), enquanto a configuração plana sem balanceamento apresentou as melhores métricas baseadas em probabilidade (ROC-AUC = 0,936, PR-AUC = 0,890, Brier Score = 0,103).

A seleção da configuração depende dos requisitos específicos da aplicação. Quando o critério de decisão é centrado no limiar padrão, a configuração plana com SMOTEENN apresenta os maiores valores em acurácia (0,789), $F1$ -macro (0,786) e MCC (0,686). Quando o foco recai sobre separação e calibração probabilística, a configuração plana sem balanceamento concentra os maiores valores de ROC-AUC (0,936) e PR-AUC (0,890), além do menor Brier Score (0,103), conforme a Tab. 6.63.

6.7 Comitê de Classificadores

Na configuração plana multiclasse sem balanceamento, seis modelos foram aprovados para compor o comitê: MLP, SVM, Random Forest, Extra Trees, Gradient Boosting e XGBoost. Os escores de validação cruzada ($F1_{\text{macro}}$) variaram de 0,541 (MLP) a 0,750 (Random Forest). O Extra Trees apresentou a menor diferença de generalização (0,003), enquanto o XGBoost apresentou a maior diferença dentro do critério de aprovação (0,229). O SVM apresentou diferença negativa de $-0,119$, refletindo desempenho em validação superior ao observado em treinamento, comportamento compatível com variabilidade amostral.

Para a arquitetura em cascata, a primeira etapa (Movimento versus Repouso) utilizou os mesmos seis classificadores, com métrica de otimização baseada em $F1$. Os escores de validação variaram de 0,716 (Random Forest) a 0,817 (SVM), com diferenças de generalização entre $-0,099$ (SVM) e 0,201 (Random Forest). Na segunda etapa (Esquerda versus Direita), treinada no cenário com oráculo que assume classificação perfeita na primeira etapa, os escores de validação variaram de 0,864 (Gradient Boosting) a 0,914 (SVM e XGBoost), indicando

menor dificuldade da tarefa binária quando avaliada de forma isolada do problema de detecção de repouso.

6.7.1 Desempenho Global sem Balanceamento

Conforme indicado na Tab. 6.64, o arranjo plano apresentou acurácia de 0,782 contra 0,764 na cascata, e *F1*-macro de 0,776 contra 0,761, correspondendo a diferenças de 0,018 e 0,015, respectivamente. O MCC passou de 0,656 (cascata) para 0,681 (plano). As métricas probabilísticas também apresentaram valores maiores no arranjo plano, com ROC-AUC macro de 0,934 versus 0,914 e PR-AUC macro de 0,894 versus 0,865. O Brier Score foi menor no plano (0,103) em comparação à cascata (0,114), indicando menor erro médio quadrático das probabilidades previstas.

Tabela 6.64 – Métricas de desempenho global do Comitê nas configurações plana e cascata, com intervalos de confiança de 95%

Métrica	Plano		Cascata		Δ
	Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Acurácia	0,782 $\pm$ 0,014	[0,755; 0,812]	0,764 $\pm$ 0,015	[0,733; 0,793]	−0,018
Acurácia Balanceada	0,780 $\pm$ 0,015	[0,752; 0,809]	0,765 $\pm$ 0,015	[0,734; 0,793]	−0,015
F1-macro	0,776 $\pm$ 0,015	[0,747; 0,806]	0,761 $\pm$ 0,015	[0,729; 0,789]	−0,015
F1-weighted	0,779 $\pm$ 0,015	[0,751; 0,810]	0,763 $\pm$ 0,015	[0,732; 0,792]	−0,016
Precisão (macro)	0,789 $\pm$ 0,014	[0,761; 0,817]	0,778 $\pm$ 0,014	[0,749; 0,804]	−0,011
Recall (macro)	0,780 $\pm$ 0,015	[0,752; 0,809]	0,765 $\pm$ 0,015	[0,734; 0,793]	−0,015
MCC	0,681 $\pm$ 0,021	[0,640; 0,723]	0,656 $\pm$ 0,021	[0,612; 0,697]	−0,025
ROC-AUC (macro)	0,934 $\pm$ 0,007	[0,920; 0,947]	0,914 $\pm$ 0,009	[0,897; 0,930]	−0,020
PR-AUC (macro)	0,894 $\pm$ 0,010	[0,872; 0,913]	0,865 $\pm$ 0,013	[0,839; 0,889]	−0,029
Brier Score	0,103 $\pm$ 0,004	[0,095; 0,112]	0,114 $\pm$ 0,005	[0,104; 0,125]	+0,011

Nota: Δ = Cascata – Plano.

A inspeção da largura dos intervalos de confiança indicou valores moderados para a maioria das métricas. Na configuração plana, três métricas apresentaram largura superior a 0,10: recall da classe Esquerda (largura = 0,116), recall da classe Repouso (largura = 0,103) e precisão da classe Esquerda (largura = 0,102). No arranjo em cascata, duas métricas excederam esse limiar: recall da classe Esquerda (largura = 0,116) e recall da classe Repouso (largura = 0,110).

6.7.2 Desempenho por Classe sem Balanceamento

A partir da Tab. 6.65, observa-se que a configuração em cascata apresentou valores maiores apenas para a classe Esquerda, com aumento de 0,035 em *recall* ($0,669 \rightarrow 0,704$) e de 0,011 em F1 ($0,695 \rightarrow 0,706$). Em contrapartida, para Direita e Repouso, o *recall* reduziu de 0,954 para 0,916 ($\Delta = -0,038$) e de 0,716 para 0,674 ($\Delta = -0,042$), respectivamente, acompanhando reduções em F1 nessas classes.

Tabela 6.65 – Métricas de desempenho do Comitê por classe nas configurações plana e cascata

Métrica		Plano		Cascata		Δ
		Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Repouso (Classe 0)	F1	$0,799 \pm 0,020$	[0,760; 0,837]	$0,775 \pm 0,021$	[0,733; 0,813]	-0,024
	Precisão	$0,905 \pm 0,018$	[0,868; 0,940]	$0,913 \pm 0,019$	[0,877; 0,949]	+0,008
	Recall	$0,716 \pm 0,027$	[0,663; 0,766]	$0,674 \pm 0,028$	[0,617; 0,727]	-0,042
Esquerda (Classe 1)	F1	$0,695 \pm 0,024$	[0,648; 0,739]	$0,706 \pm 0,022$	[0,662; 0,750]	+0,011
	Precisão	$0,722 \pm 0,026$	[0,673; 0,775]	$0,708 \pm 0,025$	[0,660; 0,756]	-0,014
	Recall	$0,669 \pm 0,031$	[0,609; 0,725]	$0,704 \pm 0,030$	[0,648; 0,764]	+0,035
Direita (Classe 2)	F1	$0,833 \pm 0,014$	[0,807; 0,860]	$0,801 \pm 0,015$	[0,771; 0,830]	-0,032
	Precisão	$0,740 \pm 0,019$	[0,703; 0,776]	$0,712 \pm 0,019$	[0,675; 0,750]	-0,028
	Recall	$0,954 \pm 0,013$	[0,927; 0,977]	$0,916 \pm 0,017$	[0,881; 0,946]	-0,038

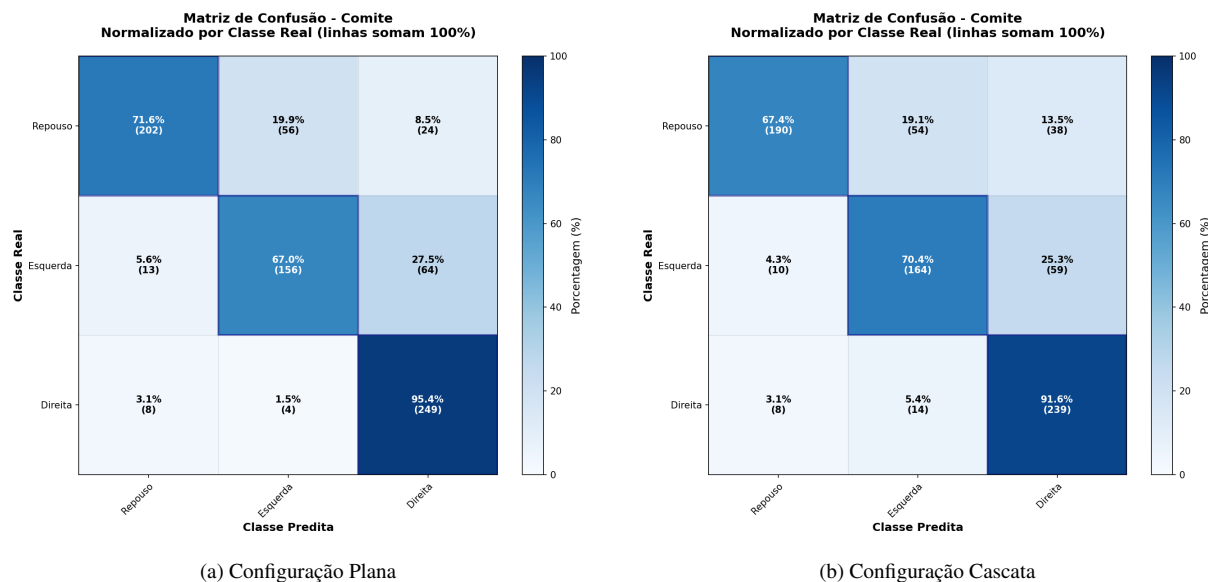
Nota: $\Delta = \text{Cascata} - \text{Plano}$.

Para Direita, os valores de *recall* (0,954 no plano e 0,916 no cascata) permaneceram acima dos observados nas demais classes. Para Repouso, a configuração em cascata aumentou a precisão ($0,905 \rightarrow 0,913$) e reduziu o *recall* ($0,716 \rightarrow 0,674$), comportamento coerente com a propagação de falsos positivos da etapa Movimento versus Repouso para a etapa subsequente, que redistribui amostras classificadas como Movimento entre Esquerda e Direita.

6.7.3 Matriz de Confusão sem Balanceamento

Na configuração plana (Figura 6.29a), a classe Esquerda apresentou taxa de acerto de 66,95%, com 27,47% das amostras atribuídas à classe Direita e 5,58% atribuídas à classe Repouso. Para Direita, observou-se 95,40% de acerto, com 1,53% de confusão com Esquerda e 3,07% com Repouso. Para Repouso, observou-se 71,63% de acerto, com confusão de 19,86% com Esquerda e 8,51% com Direita.

Figura 6.29 – Matrizes de confusão percentuais do Comitê nas configurações plana (a) e cascata (b). Os valores representam a proporção de cada classe verdadeira distribuída entre as classes preditas (normalização por linha).



Na configuração em cascata (Figura 6.29b), a taxa de acerto de Esquerda foi 70,39%, com 25,32% de confusão com Direita e 4,29% com Repouso. Para Direita, a taxa de acerto foi 91,57%, com 5,36% de confusão com Esquerda. Para Repouso, a taxa de acerto foi 67,38%, com confusão de 19,15% com Esquerda e 13,48% com Direita. Esse padrão é compatível com o mecanismo em cascata descrito na subseção anterior, no qual falsos positivos da Etapa 1 são redistribuídos entre as classes de movimento na Etapa 2.

6.7.4 Análise das Curvas ROC e PR sem Balanceamento

A Tab. 6.66 indica redução das áreas sob curva em todas as classes ao se adotar a arquitetura em cascata. A maior diferença absoluta ocorreu na PR-AUC da classe Direita ($\Delta = -0,061$), enquanto a PR-AUC de Esquerda permaneceu praticamente inalterada ($\Delta = -0,001$).

Tabela 6.66 – Áreas sob as curvas ROC e PR por classe e configuração — Comitê

Classe	ROC-AUC			PR-AUC		
	Plano	Cascata	Δ	Plano	Cascata	Δ
Repouso	0,960	0,951	−0,009	0,942	0,916	−0,026
Esquerda	0,884	0,858	−0,026	0,827	0,826	−0,001
Direita	0,958	0,932	−0,026	0,912	0,851	−0,061
Macro	0,934	0,914	−0,020	0,894	0,865	−0,029

Nota: Δ = Cascata − Plano.

6.7.5 Análise de Calibração sem Balanceamento

Conforme apresentado na Tab. 6.67, os valores de ECE foram menores no arranjo plano para todas as classes. Na configuração plana, os valores de ECE foram 0,118 (Repouso), 0,070 (Esquerda) e 0,096 (Direita). Na configuração em cascata, os valores foram 0,146, 0,134 e 0,133, respectivamente. Quanto ao MCE, Direita apresentou 0,555 na cascata no intervalo [0,5–0,6], enquanto Repouso e Esquerda apresentaram MCE de 0,411 e 0,384, respectivamente. Esse comportamento é compatível com a composição de probabilidades no arranjo em cascata, em que $P(\text{classe}) = P_{E1}(\text{Movimento}) \times P_{E2}(\text{classe}|\text{Movimento})$, alterando a distribuição final de confiança em comparação à calibração de cada etapa isoladamente.

Tabela 6.67 – Métricas de calibração por classe e configuração — Comitê

Classe	ECE		MCE		Intervalo com MCE	
	Plano	Cascata	Plano	Cascata	Plano	Cascata
Repouso	0,118	0,146	0,453	0,411	[0,3–0,4]	[0,4–0,5]
Esquerda	0,070	0,134	0,203	0,384	[0,4–0,5]	[0,3–0,4]
Direita	0,096	0,133	0,390	0,555	[0,5–0,6]	[0,5–0,6]

6.7.6 Efeito do ADASYN na Configuração Plana

Na configuração plana multiclasse com reamostragem ADASYN, o comitê foi composto por Extra Trees, Random Forest e Gradient Boosting. A redução do número de modelos aprovados (de seis para três) decorreu do critério de diferença máxima de generalização de 0,25, que reprovou MLP, SVM e XGBoost por apresentarem diferenças acima do limiar (44,0%, 43,9% e 34,1%, respectivamente).

O classificador Extra Trees que integrou o comitê foi configurado com 300 estimadores, profundidade máxima de 12 níveis, fração de atributos de 0,3, mínimo de 10 amostras por divisão e por folha, ponderação de classes balanceada por subamostra e seleção de 37 atributos, resultando em $F1_{\text{macro}} = 0,520$ em validação cruzada. O Random Forest foi configurado com 200 árvores, profundidade máxima de 12 níveis, amostragem de 80% das observações por estimador, fração de atributos igual à raiz quadrada do total, mínimo de 20 amostras por divisão interna e 10 por folha, com seleção de 36 atributos, obtendo $F1_{\text{macro}} = 0,523$. O Gradient Boosting utilizou penalização L2 de 1,22, taxa de aprendizado de 0,084, profundidade máxima de 2 níveis, até 500 iterações com parada antecipada, fração de validação interna de 20% e seleção de 18 atributos, resultando em $F1_{\text{macro}} = 0,497$.

Conforme apresentado na Tab. 6.68, a reamostragem por ADASYN elevou as métricas baseadas na decisão no limiar padrão, com aumento de 0,023 na acurácia, de 0,024 no $F1$ -macro e de 0,035 no MCC. Em contraste, observou-se redução nas métricas baseadas em probabilidades, com ROC-AUC macro reduzindo de 0,934 para 0,926 e PR-AUC macro reduzindo de 0,894 para 0,859, enquanto o Brier Score permaneceu em 0,103.

Tabela 6.68 – Métricas de desempenho global do Comitê na configuração plana sem balanceamento e com ADASYN, com intervalos de confiança de 95%

Métrica	Sem Balanceamento		ADASYN		Δ
	Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Acurácia	0,782 $\pm$ 0,014	[0,755; 0,812]	0,805 $\pm$ 0,014	[0,777; 0,832]	+0,023
Acurácia Balanceada	0,780 $\pm$ 0,015	[0,752; 0,809]	0,802 $\pm$ 0,014	[0,774; 0,830]	+0,022
F1-macro	0,776 $\pm$ 0,015	[0,747; 0,806]	0,800 $\pm$ 0,015	[0,771; 0,829]	+0,024
F1-weighted	0,779 $\pm$ 0,015	[0,751; 0,810]	0,804 $\pm$ 0,014	[0,775; 0,832]	+0,025
Precisão (macro)	0,789 $\pm$ 0,014	[0,761; 0,817]	0,814 $\pm$ 0,013	[0,788; 0,841]	+0,025
Recall (macro)	0,780 $\pm$ 0,015	[0,752; 0,809]	0,802 $\pm$ 0,014	[0,774; 0,830]	+0,022
MCC	0,681 $\pm$ 0,021	[0,640; 0,723]	0,716 $\pm$ 0,020	[0,675; 0,755]	+0,035
ROC-AUC (macro)	0,934 $\pm$ 0,007	[0,920; 0,947]	0,926 $\pm$ 0,008	[0,910; 0,941]	−0,008
PR-AUC (macro)	0,894 $\pm$ 0,010	[0,872; 0,913]	0,859 $\pm$ 0,014	[0,831; 0,886]	−0,035
Brier Score	0,103 $\pm$ 0,004	[0,095; 0,112]	0,103 $\pm$ 0,004	[0,095; 0,111]	+0,000

Nota: Δ = ADASYN – Sem Balanceamento.

Conforme detalhado na Tab. 6.69, o efeito do ADASYN concentrou-se nas classes Repouso e Esquerda, mantendo Direita inalterada nas métricas reportadas.

Tabela 6.69 – Métricas de desempenho do Comitê por classe na configuração plana sem balanceamento e com ADASYN

Métrica		Sem Balanceamento		ADASYN		Δ
		Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Repouso (Classe 0)	F1	0,799 $\pm$ 0,020	[0,760; 0,837]	0,838 $\pm$ 0,018	[0,802; 0,873]	+0,039
	Precisão	0,905 $\pm$ 0,018	[0,868; 0,940]	0,926 $\pm$ 0,016	[0,896; 0,957]	+0,021
	Recall	0,716 $\pm$ 0,027	[0,663; 0,766]	0,766 $\pm$ 0,025	[0,713; 0,816]	+0,050
Esquerda (Classe 1)	F1	0,695 $\pm$ 0,024	[0,648; 0,739]	0,729 $\pm$ 0,023	[0,682; 0,773]	+0,034
	Precisão	0,722 $\pm$ 0,026	[0,673; 0,775]	0,777 $\pm$ 0,026	[0,729; 0,828]	+0,055
	Recall	0,669 $\pm$ 0,031	[0,609; 0,725]	0,687 $\pm$ 0,030	[0,622; 0,742]	+0,018
Direita (Classe 2)	F1	0,833 $\pm$ 0,014	[0,807; 0,860]	0,833 $\pm$ 0,014	[0,807; 0,860]	+0,000
	Precisão	0,740 $\pm$ 0,019	[0,703; 0,776]	0,740 $\pm$ 0,019	[0,703; 0,776]	+0,000
	Recall	0,954 $\pm$ 0,013	[0,927; 0,977]	0,954 $\pm$ 0,013	[0,927; 0,977]	+0,000

Nota: Δ = ADASYN – Sem Balanceamento.

A partir da Tab. 6.69, observa-se aumento de 0,050 em *recall* para Repouso (0,716 $\rightarrow$ 0,766) e de 0,039 em F1 (0,799 $\rightarrow$ 0,838). Para Esquerda, a precisão aumentou de 0,722 para 0,777 (Δ = +0,055), enquanto o *recall* passou de 0,669 para 0,687 (Δ = +0,018). Para Direita, as métricas permaneceram inalteradas.

A Tab. 6.70 apresenta as áreas sob as curvas ROC e PR por classe para as configurações sem balanceamento e com ADASYN, permitindo avaliar o comportamento do classificador ao longo de diferentes limiares de decisão.

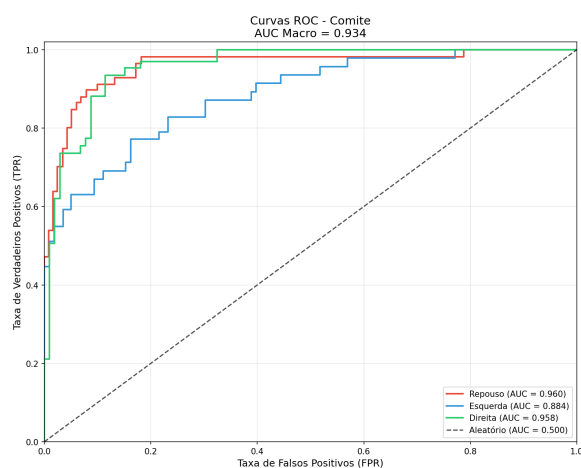
Tabela 6.70 – Áreas sob as curvas ROC e PR por classe — Comitê (plana): sem balanceamento versus ADASYN

Classe	ROC-AUC			PR-AUC		
	Sem Bal.	ADASYN	Δ	Sem Bal.	ADASYN	Δ
Repouso	0,960	0,961	+0,001	0,942	0,933	−0,009
Esquerda	0,884	0,872	−0,012	0,827	0,816	−0,011
Direita	0,958	0,944	−0,014	0,912	0,827	−0,085
Macro	0,934	0,926	−0,008	0,894	0,859	−0,035

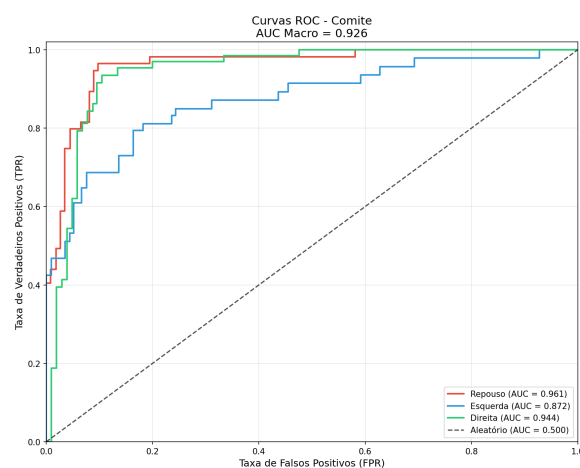
Nota: Δ = ADASYN – Sem Balanceamento.

Conforme apresentado na Tab. 6.70, a ROC-AUC macro reduziu de 0,934 para 0,926 (Δ = −0,008), enquanto a PR-AUC macro reduziu de 0,894 para 0,859 (Δ = −0,035). Entre as classes, a maior variação negativa ocorreu na PR-AUC de Direita (Δ = −0,085), ao passo que, para Repouso, a ROC-AUC permaneceu praticamente inalterada (Δ = +0,001).

Figura 6.30 – Curvas ROC do Comitê na configuração plana: (a) sem balanceamento e (b) com ADASYN, com áreas sob a curva por classe e média macro.

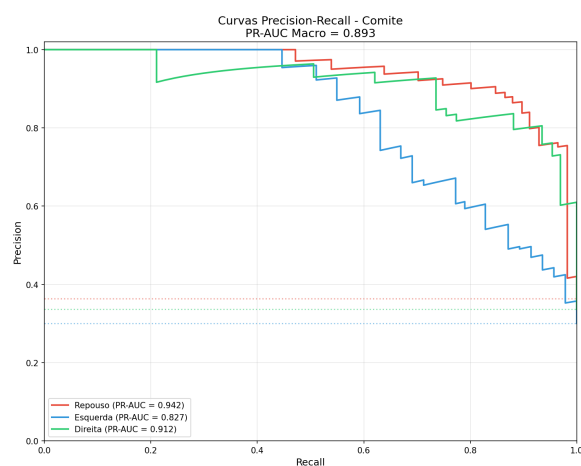


(a) Configuração plana sem balanceamento

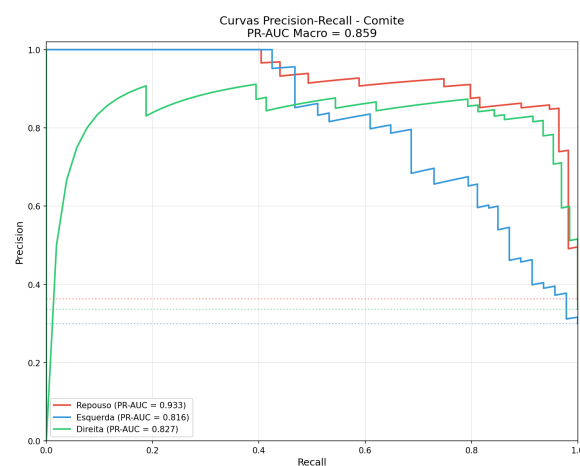


(b) Configuração plana com ADASYN

Figura 6.31 – Curvas Precision-Recall do Comitê na configuração plana: (a) sem balanceamento e (b) com ADASYN, com áreas sob a curva por classe e média macro.



(a) Configuração plana sem balanceamento



(b) Configuração plana com ADASYN

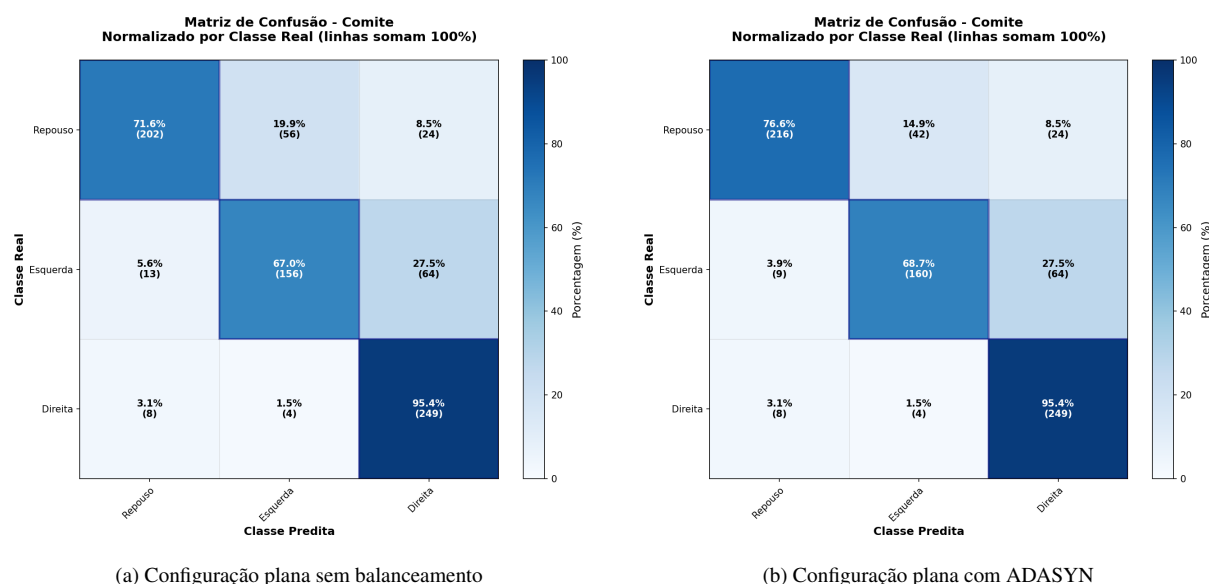
Conforme apresentado na Tab. 6.71, o ECE de Repouso aumentou de 0,118 para 0,139, enquanto o MCE reduziu de 0,453 para 0,388, mantendo o intervalo associado ao MCE em [0,3–0,4]. Para Esquerda, o ECE permaneceu em 0,070, com aumento do MCE de 0,203 para 0,253 e deslocamento do intervalo de maior erro de [0,4–0,5] para [0,7–0,8]. Para Direita, o ECE reduziu de 0,096 para 0,092, ao passo que o MCE aumentou de 0,390 para 0,445, com mudança do intervalo associado de [0,5–0,6] para [0,4–0,5].

Tabela 6.71 – Métricas de calibração por classe — Comitê (plana): sem balanceamento versus ADASYN

Classe	ECE		MCE		Intervalo com MCE	
	Sem Bal.	ADASYN	Sem Bal.	ADASYN	Sem Bal.	ADASYN
Repouso	0,118	0,139	0,453	0,388	[0,3–0,4]	[0,3–0,4]
Esquerda	0,070	0,070	0,203	0,253	[0,4–0,5]	[0,7–0,8]
Direita	0,096	0,092	0,390	0,445	[0,5–0,6]	[0,4–0,5]

Conforme ilustrado na Fig. 6.32, a taxa de acerto de Repouso passou de 71,63% (sem balanceamento) para 76,60% (ADASYN), acompanhada de redução da confusão com Esquerda de 19,86% para 14,89%. Para Esquerda, a taxa de acerto passou de 66,95% para 68,67%, enquanto Direita permaneceu com 95,40% em ambas as condições, de forma coerente com os valores de *recall* apresentados na Tab. 6.69.

Figura 6.32 – Matrizes de confusão percentuais do Comitê na configuração plana: (a) sem balanceamento e (b) com ADASYN. Os valores representam a proporção de cada classe verdadeira distribuída entre as classes preditas (normalização por linha).



6.7.7 Efeito do Random Oversampling na Configuração Cascata

Na arquitetura em cascata com reamostragem por *Random Oversampling*, a primeira etapa binária do comitê (Movimento versus Repouso) foi composta pelos classificadores Extra Trees, Random Forest e Gradient Boosting. Os modelos MLP, SVM e XGBoost foram excluídos por apresentarem diferença de generalização acima do limiar de 0,25 (31,4%, 29,1% e 26,5%, respectivamente).

O modelo Extra Trees da primeira etapa utilizou 100 estimadores, profundidade máxima de 12 níveis, fração de atributos igual ao logaritmo base 2 do total, mínimo de 10 amostras por divisão, 20 amostras por folha e seleção de 31 atributos, obtendo $F1_{\text{macro}} = 0,624$ em validação cruzada. O Random Forest foi configurado com 300 árvores, profundidade máxima de 5 níveis, amostragem de 90% das observações, fração de atributos de 0,3, mínimo de 10 amostras por divisão e por folha, além de seleção de 22 atributos, resultando em $F1_{\text{macro}} = 0,619$.

Na segunda etapa binária da arquitetura em cascata (Esquerda versus Direita), treinada sob o cenário ideal com oráculo, o comitê foi composto pelos modelos Extra Trees, Random Forest e XGBoost. O Extra Trees atingiu $F1_{\text{macro}} = 0,984$, o Random Forest alcançou $F1_{\text{macro}} = 0,980$ e o XGBoost obteve $F1_{\text{macro}} = 0,962$, descrevendo o desempenho da segunda etapa quando avaliada sem propagação de erros da primeira etapa.

A partir da Tab. 6.72, a acurácia passou de 0,764 para 0,767 ($\Delta = +0,003$) e o $F1$ -macro de 0,761 para 0,763 ($\Delta = +0,002$). Nas métricas baseadas em probabilidades, a ROC-AUC macro passou de 0,914 para 0,912 e a PR-AUC macro de 0,865 para 0,858, enquanto o Brier Score reduziu de 0,114 para 0,112.

Tabela 6.72 – Métricas de desempenho global do Comitê na configuração em cascata sem balanceamento e com Random Oversampling, com intervalos de confiança de 95%

Métrica	Sem Balanceamento		Random Oversampling		Δ
	Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Acurácia	0,764 $\pm$ 0,015	[0,733; 0,793]	0,767 $\pm$ 0,015	[0,736; 0,794]	+0,003
Acurácia Balanceada	0,765 $\pm$ 0,015	[0,734; 0,793]	0,770 $\pm$ 0,015	[0,739; 0,796]	+0,005
F1-macro	0,761 $\pm$ 0,015	[0,729; 0,789]	0,763 $\pm$ 0,015	[0,731; 0,791]	+0,002
F1-weighted	0,763 $\pm$ 0,015	[0,732; 0,792]	0,765 $\pm$ 0,015	[0,733; 0,792]	+0,002
Precisão (macro)	0,778 $\pm$ 0,014	[0,749; 0,804]	0,776 $\pm$ 0,014	[0,748; 0,802]	−0,002
Recall (macro)	0,765 $\pm$ 0,015	[0,734; 0,793]	0,770 $\pm$ 0,015	[0,739; 0,796]	+0,005
MCC	0,656 $\pm$ 0,021	[0,612; 0,697]	0,659 $\pm$ 0,021	[0,614; 0,698]	+0,003
ROC-AUC (macro)	0,914 $\pm$ 0,009	[0,897; 0,930]	0,912 $\pm$ 0,009	[0,895; 0,928]	−0,002
PR-AUC (macro)	0,865 $\pm$ 0,013	[0,839; 0,889]	0,858 $\pm$ 0,013	[0,831; 0,883]	−0,007
Brier Score	0,114 $\pm$ 0,005	[0,104; 0,125]	0,112 $\pm$ 0,005	[0,102; 0,122]	−0,002

Nota: Δ = Random Oversampling – Sem Balanceamento.

Conforme apresentado na Tab. 6.73, Repouso apresentou redução em F1 (0,775 $\rightarrow$ 0,749), precisão (0,913 $\rightarrow$ 0,892) e *recall* (0,674 $\rightarrow$ 0,646). Para Esquerda, o *recall* aumentou de 0,704 para 0,747, acompanhado de redução em precisão de 0,708 para 0,673. Para Direita, a precisão aumentou de 0,712 para 0,765, mantendo-se o *recall* em 0,916.

Tabela 6.73 – Métricas de desempenho do Comitê por classe na configuração em cascata sem balanceamento e com Random Oversampling

Métrica		Sem Balanceamento		Random Oversampling		Δ
		Média $\pm$ DP	IC 95%	Média $\pm$ DP	IC 95%	
Repouso (Classe 0)	F1	0,775 $\pm$ 0,021	[0,733; 0,813]	0,749 $\pm$ 0,022	[0,701; 0,790]	−0,026
	Precisão	0,913 $\pm$ 0,019	[0,877; 0,949]	0,892 $\pm$ 0,020	[0,852; 0,930]	−0,021
	Recall	0,674 $\pm$ 0,028	[0,617; 0,727]	0,646 $\pm$ 0,029	[0,589; 0,702]	−0,028
Esquerda (Classe 1)	F1	0,706 $\pm$ 0,022	[0,662; 0,750]	0,708 $\pm$ 0,021	[0,664; 0,746]	+0,002
	Precisão	0,708 $\pm$ 0,025	[0,660; 0,756]	0,673 $\pm$ 0,023	[0,628; 0,717]	−0,035
	Recall	0,704 $\pm$ 0,030	[0,648; 0,764]	0,747 $\pm$ 0,028	[0,691; 0,803]	+0,043
Direita (Classe 2)	F1	0,801 $\pm$ 0,015	[0,771; 0,830]	0,833 $\pm$ 0,015	[0,803; 0,861]	+0,032
	Precisão	0,712 $\pm$ 0,019	[0,675; 0,750]	0,765 $\pm$ 0,020	[0,724; 0,803]	+0,053
	Recall	0,916 $\pm$ 0,017	[0,881; 0,946]	0,916 $\pm$ 0,017	[0,881; 0,946]	+0,000

Nota: Δ = Random Oversampling – Sem Balanceamento.

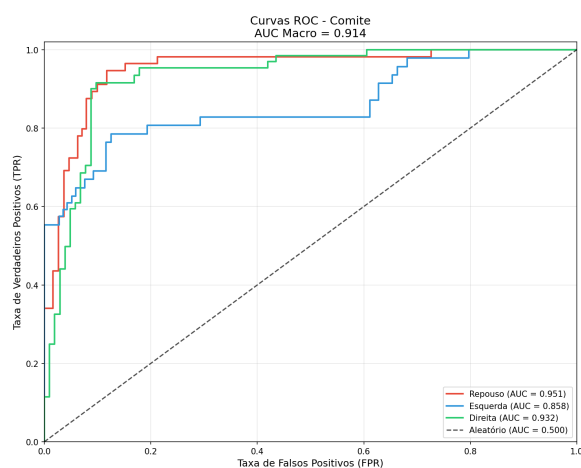
Conforme apresentado na Tab. 6.74, a ROC-AUC macro reduziu de 0,914 para 0,912 e a PR-AUC macro de 0,865 para 0,858. Entre as classes, Direita apresentou aumento em PR-AUC (0,851 $\rightarrow$ 0,860), enquanto Repouso e Esquerda apresentaram reduções em PR-AUC.

Tabela 6.74 – Áreas sob as curvas ROC e PR por classe — Comitê (cascata): sem balanceamento versus Random Oversampling

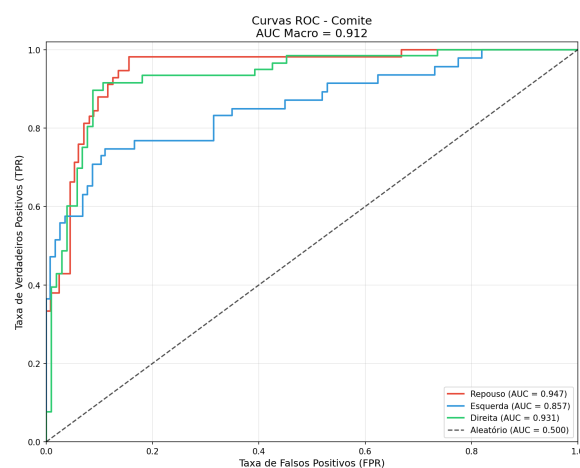
Classe	ROC-AUC			PR-AUC		
	Sem Bal.	Random OS	Δ	Sem Bal.	Random OS	Δ
Repouso	0,951	0,947	−0,004	0,916	0,904	−0,012
Esquerda	0,858	0,857	−0,001	0,826	0,809	−0,017
Direita	0,932	0,931	−0,001	0,851	0,860	+0,009
Macro	0,914	0,912	−0,002	0,865	0,858	−0,007

Nota: Δ = Random Oversampling – Sem Balanceamento.

Figura 6.33 – Curvas ROC do Comitê na configuração cascata: (a) sem balanceamento e (b) com Random Oversampling, com áreas sob a curva por classe e média macro.

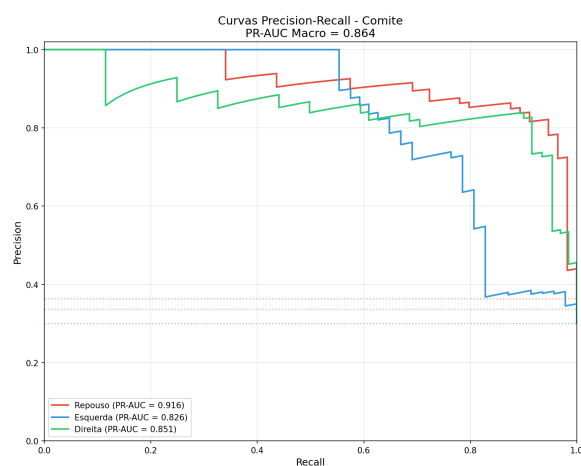


(a) Configuração cascata sem balanceamento

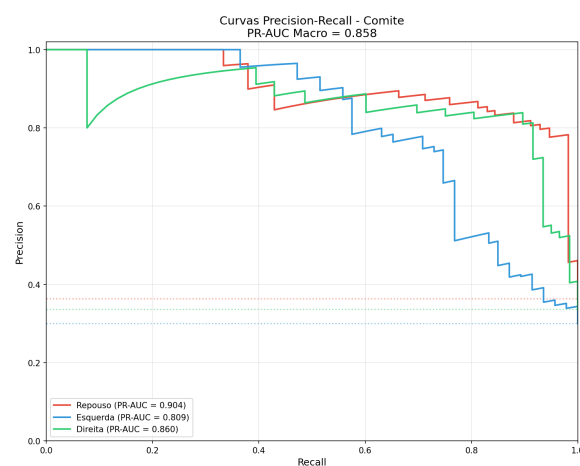


(b) Configuração cascata com Random Oversampling

Figura 6.34 – Curvas Precision–Recall do Comitê na configuração cascata: (a) sem balanceamento e (b) com Random Oversampling, com áreas sob a curva por classe e média macro.



(a) Configuração cascata sem balanceamento



(b) Configuração cascata com Random Oversampling

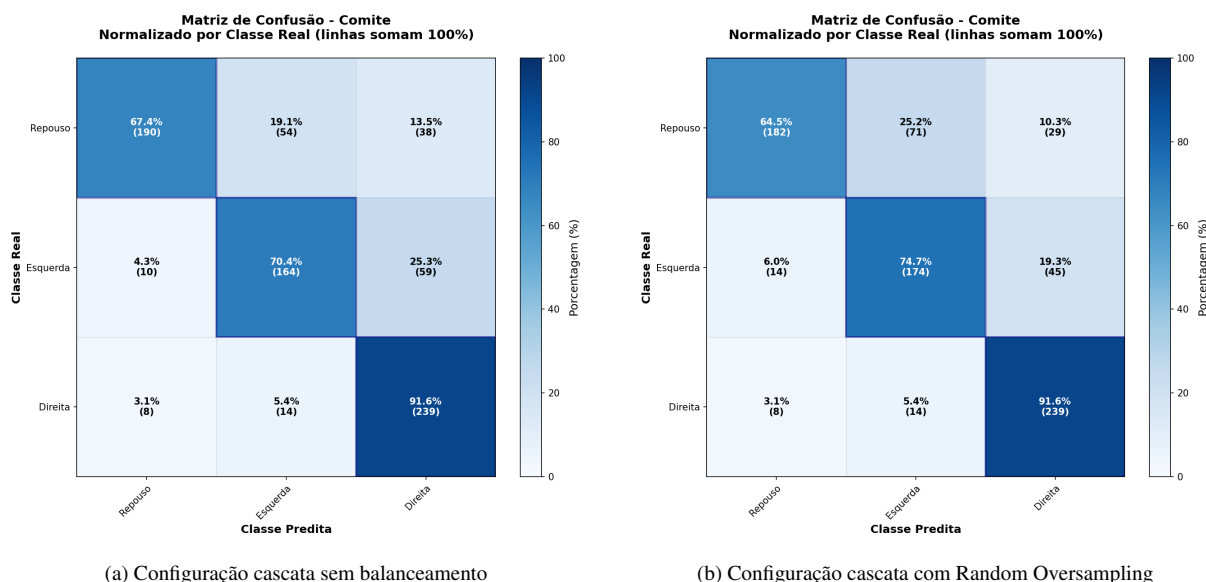
Conforme apresentado na Tab. 6.75, o ECE reduziu para as três classes (Repouso: 0,146 $\rightarrow$ 0,126; Esquerda: 0,134 $\rightarrow$ 0,063; Direita: 0,133 $\rightarrow$ 0,082). Para o MCE, observou-se aumento em Repouso (0,411 $\rightarrow$ 0,421) e em Esquerda (0,384 $\rightarrow$ 0,449), e redução em Direita (0,555 $\rightarrow$ 0,453), com deslocamento do intervalo com MCE para [0,4–0,5] nas três classes após a reamostragem.

Tabela 6.75 – Métricas de calibração por classe — Comitê (cascata): sem balanceamento versus Random Oversampling

Classe	ECE		MCE		Intervalo com MCE	
	Sem Bal.	Random OS	Sem Bal.	Random OS	Sem Bal.	Random OS
Repouso	0,146	0,126	0,411	0,421	[0,4–0,5]	[0,4–0,5]
Esquerda	0,134	0,063	0,384	0,449	[0,3–0,4]	[0,4–0,5]
Direita	0,133	0,082	0,555	0,453	[0,5–0,6]	[0,4–0,5]

Conforme ilustrado na Fig. 6.35, a taxa de acerto de Repouso passou de 67,38% para 64,54%, acompanhada de aumento da confusão com Esquerda de 19,15% para 25,18%. Para Esquerda, a taxa de acerto passou de 70,39% para 74,68%, com redução da confusão com Direita de 25,32% para 19,31%. Para Direita, a taxa de acerto permaneceu em 91,57%, em concordância com o *recall* apresentado na Tab. 6.73.

Figura 6.35 – Matrizes de confusão percentuais do Comitê na configuração cascata: (a) sem balanceamento e (b) com Random Oversampling. Os valores representam a proporção de cada classe verdadeira distribuída entre as classes preditas (normalização por linha).



6.7.8 Síntese Comparativa

De acordo com a Tab. 6.76, a configuração plana com ADASYN apresentou acurácia de 0,805, *F1*-macro de 0,800 e MCC de 0,716, enquanto a configuração plana sem balanceamento apresentou ROC-AUC macro de 0,934 e PR-AUC macro de 0,894. Em ambas as configurações

em cascata, as métricas globais permaneceram abaixo das correspondentes configurações planas para as métricas sintetizadas, independentemente do uso de reamostragem.

Tabela 6.76 – Síntese comparativa das configurações do Comitê

Métrica	Plano	Plano+ADASYN	Cascata	Cascata+ROS
Acurácia	0,782	0,805	0,764	0,767
F1-macro	0,776	0,800	0,761	0,763
MCC	0,681	0,716	0,656	0,659
ROC-AUC macro	0,934	0,926	0,914	0,912
PR-AUC macro	0,894	0,859	0,865	0,858
Brier Score	0,103	0,103	0,114	0,112

Nota: ROS = *Random Oversampling*. Valores em negrito indicam o maior valor por métrica; para o Brier Score, indicam o menor valor.

A análise por classe, reportada nas Tabs. 6.65, 6.69 e 6.73, indica que Direita manteve *recall* acima de 0,91 nas quatro configurações. Para Esquerda, o *recall* variou entre 0,669 e 0,747, enquanto, para Repouso, variou entre 0,646 e 0,766, refletindo redistribuição de erros.

O efeito das técnicas de balanceamento apresentou comportamento distinto entre as arquiteturas. Na configuração plana, a aplicação de ADASYN aumentou as métricas no limiar padrão (Tab. 6.68) e reduziu PR-AUC macro (Tab. 6.70). Na configuração em cascata, o *Random Oversampling* promoveu pequenas alterações no desempenho global (Tab. 6.72), acompanhadas de mudanças por classe (Tab. 6.73) sem aumento correspondente nas áreas sob curva (Tab. 6.74).

Em aplicações cuja decisão é tomada no limiar padrão e o objetivo é maximizar métricas como acurácia, F1-macro e MCC, a configuração plana com ADASYN apresenta os maiores valores entre as configurações avaliadas (Tab. 6.76). Em aplicações nas quais a separação ao longo de limiares e o comportamento probabilístico são centrais, a configuração plana sem balanceamento apresenta maiores valores de ROC-AUC e PR-AUC (Tab. 6.76) e menor ECE para as três classes (Tab. 6.67). A interpretação final deve considerar, adicionalmente, o *recall* de Esquerda, que permaneceu abaixo de 0,75 mesmo após reamostragem (Tab. 6.69), e as variações de calibração sintetizadas nas Tabs. 6.71 e 6.75.

6.8 Avaliação Comparativa dos Modelos Desenvolvidos

Conforme apresentado na Tab. 6.77, as métricas globais dos classificadores na configuração plana foram consolidadas a partir das combinações selecionadas, reportando média, desvio-padrão e intervalo de confiança de 95% obtido por *bootstrap*. Em cada linha, destaca-se o maior valor, com exceção do Brier Score, para o qual se considera o menor valor como referência de menor erro quadrático médio das probabilidades preditas.

Tabela 6.77 – Maiores valores por classificador na configuração plana (média, desvio-padrão e IC 95% *bootstrap*). Em cada linha, destaca-se o maior valor; para Brier Score, considera-se o menor valor.

Métrica	Comitê (ADASYN)	Extra Trees (ADASYN)	Gradient Boosting (sem bal.)	MLP (ROS)	Random Forest (SMOTE)	SVM (SMOTEENN)	XGBoost (SMOTEENN)
Acurácia	0,805 ± 0,014 [0,777; 0,832]	0,799 ± 0,014 [0,771; 0,827]	0,776 ± 0,014 [0,747; 0,804]	0,782 ± 0,014 [0,754; 0,811]	0,775 ± 0,014 [0,745; 0,803]	0,826 ± 0,013 [0,800; 0,852]	0,789 ± 0,015 [0,759; 0,817]
Acurácia Balanceada	0,802 ± 0,014 [0,774; 0,830]	0,795 ± 0,014 [0,766; 0,823]	0,779 ± 0,014 [0,750; 0,806]	0,773 ± 0,015 [0,745; 0,803]	0,772 ± 0,015 [0,743; 0,801]	0,825 ± 0,013 [0,798; 0,851]	0,786 ± 0,015 [0,755; 0,815]
F1-macro	0,800 ± 0,015 [0,771; 0,829]	0,793 ± 0,015 [0,763; 0,822]	0,772 ± 0,015 [0,743; 0,801]	0,774 ± 0,015 [0,745; 0,804]	0,769 ± 0,015 [0,739; 0,798]	0,822 ± 0,014 [0,794; 0,848]	0,786 ± 0,015 [0,755; 0,814]
F1-weighted	0,804 ± 0,014 [0,775; 0,832]	0,797 ± 0,014 [0,768; 0,825]	0,774 ± 0,015 [0,745; 0,803]	0,778 ± 0,015 [0,749; 0,808]	0,773 ± 0,015 [0,743; 0,802]	0,825 ± 0,014 [0,798; 0,851]	0,790 ± 0,015 [0,759; 0,818]
Precisão (macro)	0,814 ± 0,013 [0,788; 0,841]	0,809 ± 0,014 [0,781; 0,836]	0,791 ± 0,013 [0,766; 0,815]	0,791 ± 0,015 [0,762; 0,819]	0,784 ± 0,014 [0,757; 0,811]	0,836 ± 0,012 [0,812; 0,859]	0,792 ± 0,014 [0,762; 0,820]
MCC	0,716 ± 0,020 [0,675; 0,755]	0,707 ± 0,020 [0,665; 0,747]	0,676 ± 0,020 [0,635; 0,714]	0,676 ± 0,022 [0,634; 0,719]	0,670 ± 0,021 [0,629; 0,711]	0,748 ± 0,019 [0,710; 0,785]	0,686 ± 0,022 [0,641; 0,728]
ROC-AUC (macro)	0,926 ± 0,008 [0,910; 0,941]	0,934 ± 0,007 [0,919; 0,947]	0,932 ± 0,007 [0,918; 0,945]	0,875 ± 0,010 [0,854; 0,896]	0,922 ± 0,008 [0,907; 0,937]	0,893 ± 0,009 [0,875; 0,911]	0,927 ± 0,007 [0,911; 0,941]
PR-AUC (macro)	0,859 ± 0,014 [0,831; 0,886]	0,883 ± 0,013 [0,858; 0,907]	0,884 ± 0,012 [0,860; 0,907]	0,818 ± 0,014 [0,789; 0,846]	0,873 ± 0,012 [0,850; 0,896]	0,772 ± 0,017 [0,739; 0,806]	0,868 ± 0,013 [0,842; 0,892]
Brier Score	0,103 ± 0,004 [0,095; 0,111]	0,096 ± 0,005 [0,086; 0,106]	0,107 ± 0,004 [0,099; 0,115]	0,130 ± 0,004 [0,123; 0,138]	0,105 ± 0,005 [0,094; 0,115]	0,112 ± 0,005 [0,103; 0,122]	0,106 ± 0,004 [0,097; 0,115]

A partir da Tab. 6.77, observa-se que o SVM com SMOTEENN concentra os maiores valores nas métricas baseadas em rótulo, com acurácia de 0,826, acurácia balanceada de 0,825, *F1*-macro de 0,822, *F1*-weighted de 0,825, precisão (macro) de 0,836 e MCC de 0,748. Nas métricas probabilísticas, o Extra Trees com ADASYN apresenta ROC-AUC (macro) de 0,934 e Brier Score de 0,096, enquanto o Gradient Boosting sem balanceamento apresenta PR-AUC (macro) de 0,884. Os resultados do Comitê com ADASYN (0,800 em *F1*-macro) e do Extra Trees com ADASYN situam-se em faixas próximas em acurácia e *F1*-macro, mas diferenciam-se nas métricas probabilísticas, nas quais o Extra Trees apresenta ROC-AUC (macro) superior e menor Brier Score. O XGBoost com SMOTEENN (0,786 em *F1*-macro), o MLP com *Random OverSampling* (0,774 em *F1*-macro) e o Random Forest com SMOTE (0,769 em *F1*-macro) permanecem abaixo do SVM com SMOTEENN nas métricas de classificação reportadas na tabela.

Conforme apresentado na Tab. 6.78, o mesmo procedimento é aplicado à configuração em cascata, reunindo-se os resultados de todos os classificadores a partir das combinações selecionadas, incluindo versões com reamostragem externa para Comitê, Extra Trees, SVM e XGBoost, e versões sem balanceamento para os demais.

Tabela 6.78 – Desempenho dos classificadores na configuração em cascata (média, desvio-padrão e IC 95% *bootstrap*). Em cada linha, destaca-se o maior valor; para Brier Score, considera-se o menor valor.

Métrica	Comitê (ROS)	Extra Trees (SMOTE)	Gradient Boosting (sem bal.)	MLP (sem bal.)	Random Forest (sem bal.)	SVM (SMOTEENN)	XGBoost (SMOTEENN)
Acurácia	0,767 ± 0,015 [0,736; 0,794]	0,799 ± 0,014 [0,772; 0,826]	0,758 ± 0,015 [0,728; 0,786]	0,793 ± 0,014 [0,765; 0,820]	0,783 ± 0,014 [0,755; 0,811]	0,671 ± 0,017 [0,638; 0,705]	0,766 ± 0,015 [0,736; 0,795]
Acurácia Balanceada	0,770 ± 0,015 [0,739; 0,796]	0,798 ± 0,014 [0,769; 0,825]	0,756 ± 0,015 [0,726; 0,785]	0,791 ± 0,014 [0,764; 0,819]	0,779 ± 0,014 [0,751; 0,807]	0,674 ± 0,017 [0,641; 0,707]	0,767 ± 0,015 [0,736; 0,797]
F1-macro	0,763 ± 0,015 [0,731; 0,791]	0,796 ± 0,015 [0,768; 0,823]	0,753 ± 0,016 [0,722; 0,781]	0,787 ± 0,014 [0,760; 0,816]	0,777 ± 0,015 [0,747; 0,805]	0,674 ± 0,016 [0,642; 0,707]	0,764 ± 0,015 [0,734; 0,794]
F1-weighted	0,765 ± 0,015 [0,733; 0,792]	0,799 ± 0,014 [0,771; 0,826]	0,756 ± 0,015 [0,726; 0,784]	0,790 ± 0,014 [0,762; 0,818]	0,781 ± 0,014 [0,752; 0,809]	0,679 ± 0,016 [0,647; 0,712]	0,767 ± 0,015 [0,737; 0,797]
Precisão (macro)	0,776 ± 0,014 [0,748; 0,802]	0,803 ± 0,013 [0,776; 0,830]	0,761 ± 0,015 [0,731; 0,789]	0,799 ± 0,013 [0,773; 0,825]	0,783 ± 0,014 [0,754; 0,811]	0,699 ± 0,015 [0,669; 0,729]	0,771 ± 0,015 [0,742; 0,800]
MCC	0,659 ± 0,021 [0,614; 0,698]	0,703 ± 0,020 [0,662; 0,744]	0,642 ± 0,022 [0,598; 0,683]	0,696 ± 0,020 [0,657; 0,736]	0,678 ± 0,021 [0,636; 0,719]	0,521 ± 0,024 [0,473; 0,568]	0,653 ± 0,023 [0,609; 0,698]
ROC-AUC (macro)	0,912 ± 0,009 [0,895; 0,928]	0,921 ± 0,008 [0,906; 0,936]	0,892 ± 0,010 [0,873; 0,910]	0,907 ± 0,009 [0,889; 0,923]	0,901 ± 0,009 [0,882; 0,918]	0,853 ± 0,011 [0,830; 0,875]	0,892 ± 0,010 [0,871; 0,910]
PR-AUC (macro)	0,858 ± 0,013 [0,831; 0,883]	0,882 ± 0,012 [0,860; 0,905]	0,811 ± 0,017 [0,778; 0,844]	0,854 ± 0,013 [0,828; 0,879]	0,830 ± 0,015 [0,800; 0,860]	0,767 ± 0,017 [0,733; 0,800]	0,815 ± 0,016 [0,782; 0,846]
Brier Score	0,112 ± 0,005 [0,102; 0,122]	0,106 ± 0,005 [0,094; 0,117]	0,125 ± 0,005 [0,115; 0,136]	0,113 ± 0,006 [0,102; 0,124]	0,120 ± 0,006 [0,109; 0,132]	0,153 ± 0,004 [0,145; 0,161]	0,128 ± 0,006 [0,117; 0,140]

A partir da Tab. 6.78, observa-se que o Extra Trees com SMOTE apresenta os maiores valores em todas as métricas globais listadas, com acurácia de 0,799, acurácia balanceada de 0,798, $F1$ -macro de 0,796, MCC de 0,703, ROC-AUC (macro) de 0,921, PR-AUC (macro) de 0,882 e Brier Score de 0,106. O XGBoost com SMOTEENN apresenta $F1$ -macro de 0,764, enquanto o SVM com SMOTEENN apresenta $F1$ -macro de 0,674, permanecendo abaixo do Extra Trees nas métricas baseadas em rótulo reportadas. O Comitê com *Random OverSampling* apresenta $F1$ -macro de 0,763, em faixa próxima ao XGBoost com SMOTEENN para métricas baseadas em rótulo, mas com ROC-AUC (macro) e PR-AUC (macro) inferiores aos do Extra Trees com SMOTE. Entre os modelos sem balanceamento na cascata, o Gradient Boosting, o MLP e o Random Forest apresentam $F1$ -macro de 0,753, 0,787 e 0,777, respectivamente.

A comparação entre as soluções de referência das configurações plana e em cascata pode ser conduzida mantendo-se o mesmo conjunto de métricas globais e a incerteza associada às estimativas. Considerando-se o $F1$ -macro como critério de otimização, o maior valor na configuração plana é observado no SVM com SMOTEENN ($F1$ -macro de 0,822), enquanto o maior valor na configuração em cascata é observado no Extra Trees com SMOTE ($F1$ -macro de 0,796). Do ponto de vista probabilístico, a Tab. 6.77 indica que o Extra Trees com ADASYN apresenta ROC-AUC (macro) de 0,934 e Brier Score de 0,096, enquanto a Tab. 6.78 indica que o Extra Trees com SMOTE apresenta ROC-AUC (macro) de 0,921 e Brier Score de 0,106. Nas combinações reportadas, as médias das métricas probabilísticas e do $F1$ -macro permanecem mais elevadas na configuração plana, enquanto, na configuração em cascata, o Extra Trees com SMOTE concentra os maiores valores entre os classificadores avaliados nessa arquitetura.

6.9 Impacto computacional

Esta seção examina os custos computacionais associados às configurações plana e em cascata, distinguindo explicitamente os regimes de uso offline (treinamento) e online (inferência), conforme sumarizado nas Tabs. 6.79, 6.80 e 6.81. Em cenários offline, o tempo de treinamento é amortizado ao longo do ciclo de vida do modelo, tornando-se criticamente relevante apenas quando há re-treinamentos frequentes ou janelas curtas para atualização. Em cenários online, a viabilidade operacional é governada sobretudo pela latência e, principalmente, pelas caudas de latência (P95 e P99), pois estas refletem o risco de violações de prazo em condições desfavoráveis. No arranjo plano, o throughput máximo e o batch recomendado complementam a análise por caracterizarem capacidade de processamento em lote e eficiência de execução, aspectos relevantes em pontuações periódicas (*batch scoring*) e serviços de inferência com paralelismo.

6.9.1 Comparação entre configurações plana e cascata sem balanceamento

A Tab. 6.79 indica que não existe uma configuração globalmente superior em todas as métricas computacionais; o desempenho depende do modelo e do regime de uso.

Na configuração plana, o menor tempo de treinamento ocorre em XGBoost (49,66 s), resultado associado à eficiência do algoritmo de *gradient boosting* com regularização e à configuração compacta de 100 estimadores e profundidade máxima de 3 níveis, conforme descrito na Seção 6.6. Em contraste, MLP e SVM apresentam os maiores tempos de treinamento (207,69 s e 333,90 s, respectivamente). O tempo elevado do MLP decorre do processo iterativo de otimização por gradiente descendente estocástico, enquanto o SVM requer a construção e manipulação da matriz de kernel, cuja complexidade escala de forma superlinear com o número de amostras e atributos selecionados.

Apesar do custo de treinamento, o MLP apresenta a menor latência média (0,606 ms) e o maior throughput máximo (215.844 amostras/s), caracterizando-se como o modelo de maior eficiência para serviços online. Este comportamento resulta da natureza puramente matricial da inferência em redes neurais: uma vez treinados os pesos, a predição consiste em multiplicações matriciais e aplicações de funções de ativação, operações altamente otimizadas em bibliotecas

numéricas. A arquitetura compacta do MLP plano (16 e 8 neurônios nas camadas ocultas, conforme Tab. 6.1) contribui para a baixa latência observada.

O SVM, embora com throughput inferior ao MLP (3.990 amostras/s), apresenta caudas baixas ($P99 = 1,074$ ms), indicando maior previsibilidade temporal no pior caso. Esta característica decorre da natureza determinística da predição por vetores de suporte, sem dependência de estruturas arbóreas que possam induzir caminhos de execução de comprimento variável.

Em contraste, Extra Trees e Random Forest exibem latências médias elevadas na configuração plana (220,367 ms e 220,812 ms, respectivamente), com caudas igualmente altas. Conforme detalhado nas Seções 6.2 e 6.3, os modelos utilizam 200 e 100 estimadores, respectivamente, e a inferência requer a travessia de múltiplas árvores de decisão. A agregação dos votos de centenas de árvores, cada qual com profundidade de até 5 níveis, resulta em latência incompatível com restrições de tempo real, embora aceitável em configurações offline ou de processamento em lote.

Na configuração em cascata, observa-se aumento de latência média para MLP, SVM e XGBoost (+92,7%, +92,7% e +152,8%, respectivamente), conforme reportado na Tab. 6.79. Este comportamento é esperado, uma vez que a arquitetura em cascata requer a execução sequencial de dois classificadores: a primeira etapa (Movimento versus Repouso) e, condicionalmente, a segunda etapa (Esquerda versus Direita). A duplicação do número de modelos implica, aproximadamente, duplicação do tempo de inferência por amostra, justificando o aumento observado.

Para o MLP, a arquitetura cascata emprega camadas ocultas com 64 e 32 neurônios em cada etapa, configuração de maior capacidade do que a versão plana (16 e 8 neurônios), conforme descrito na Seção 6.1. Este aumento de complexidade, combinado com a execução sequencial, explica a elevação da latência média de 0,606 ms para 1,168 ms. Ainda assim, a latência absoluta permanece na ordem de milissegundos, mantendo viabilidade para aplicações com restrições moderadas de tempo.

Por outro lado, para modelos baseados em árvores, há redução marcante da latência média de Extra Trees (−81,6%) e de Gradient Boosting (−67,1%). Esta redução aparentemente contraditória pode ser explicada pela estrutura da cascata: na primeira etapa, apenas a distinção binária Movimento versus Repouso é requerida, permitindo modelos de menor complexidade do que o classificador multiclasse plano. Conforme reportado na Seção 6.2, a primeira etapa

do Extra Trees cascata utiliza 100 estimadores, metade do número empregado na configuração plana. A segunda etapa, embora também utilize 100 estimadores, é executada apenas para amostras classificadas como Movimento, reduzindo o número médio de travessias por amostra.

Entretanto, a redução da latência média no Random Forest não é acompanhada por melhoria nas caudas. Conforme a Tab. 6.79, o P95 aumenta em 96,2% e o P99 em 80,4%, evidenciando maior instabilidade temporal e risco operacional. Este comportamento sugere que, embora a média seja reduzida pela economia de processamento em amostras de Repouso, casos específicos requerem travessias mais extensas, elevando os percentis superiores da distribuição de latência.

Tabela 6.79 – Cenário sem balanceamento: comparação computacional entre as configurações plano e cascata para todos os modelos avaliados.

Métrica	MLP			SVM			XGBoost			Gradient Boosting			Extra Trees			Random Forest		
	Plano	Cascata	Δ (%)	Plano	Cascata	Δ (%)	Plano	Cascata	Δ (%)	Plano	Cascata	Δ (%)	Plano	Cascata	Δ (%)	Plano	Cascata	Δ (%)
Tempo treino (s)	207.69	255.54	23.0	333.90	371.53	11.3	49.66	79.50	60.1	93.89	78.21	-16.7	58.62	90.59	54.5	106.43	161.84	52.1
Latência média (ms)	0.606	1.168	92.7	0.843	1.625	92.7	1.033	2.612	152.8	46.379	15.278	-67.1	220.367	40.584	-81.6	220.812	116.928	-47.0
P95 (ms)	0.787	1.406	78.6	1.009	1.816	80.0	1.165	3.959	239.8	269.437	20.617	-92.3	229.956	51.990	-77.4	230.348	452.053	96.2
P99 (ms)	2.448	1.953	-20.2	1.074	1.902	77.1	1.217	4.478	268.0	379.204	25.549	-93.3	239.336	53.773	-77.5	259.422	467.915	80.4
Throughput máx. (amostras/s)	215,844	–	–	3,990	–	–	73,950	–	–	13,265	–	–	580	–	–	576	–	–
Batch recomendado	128	–	–	128	–	–	128	–	–	128	–	–	128	–	–	128	–	–

Em síntese, no cenário sem balanceamento, a cascata pode ser coerente para reduzir custo médio de inferência do Extra Trees e do Gradient Boosting, mas é criticamente desfavorável para Random Forest pela degradação das caudas. Para MLP, SVM e XGBoost, a cascata tende a ser desfavorável por elevar a latência média, embora os valores absolutos permaneçam em faixas operacionalmente aceitáveis para os dois primeiros modelos.

6.9.2 Impacto do balanceamento na configuração plana

A Tab. 6.80 sintetiza o impacto computacional do balanceamento na configuração plana, restrito aos pares modelo–técnica identificados como de melhor desempenho preditivo nas seções anteriores. Observa-se um cenário heterogêneo, no qual o balanceamento não atua como custo adicional uniforme; os efeitos dependem do modelo e da interação entre a técnica de reamostragem e os hiperparâmetros selecionados durante a otimização.

Tabela 6.80 – Configuração plana: impacto computacional do balanceamento restrito aos melhores pares modelo–técnica, comparados ao sem balanceamento do mesmo modelo.

Métrica	MLP (Random Oversampling)			Extra Trees (ADASYN)			Random Forest (SMOTE)			SVM (SMOTEENN)			XGBoost (SMOTEENN)		
	Sem bal.	Bal.	Δ (%)	Sem bal.	Bal.	Δ (%)	Sem bal.	Bal.	Δ (%)	Sem bal.	Bal.	Δ (%)	Sem bal.	Bal.	Δ (%)
Tempo treino (s)	207.69	251.74	21.2	58.62	34.96	-40.4	106.43	93.55	-12.1	333.90	525.26	57.3	49.66	14.20	-71.4
Latência média (ms)	0.606	0.599	-1.1	220.367	219.726	-0.3	220.812	204.529	-7.4	0.843	0.929	10.2	1.033	37.522	3530.8
P95 (ms)	0.787	0.656	-16.7	229.956	236.221	2.7	230.348	214.470	-6.9	1.009	0.933	-7.5	1.165	54.830	4606.3
P99 (ms)	2.448	0.797	-67.4	239.336	252.607	5.5	259.422	225.364	-13.1	1.074	1.023	-4.7	1.217	75.839	6131.2
Throughput máx. (amostras/s)	215,844	187,824	-13.0	580	1,079	86.1	576	617	7.2	3,990	3,245	-18.7	73,950	3,683	-95.0
Batch recomendado	128	128	–	128	128	–	128	128	–	128	64	–	128	128	–

Para MLP com Random Oversampling, há aumento do tempo de treinamento (+21,2%), de 207,69 s para 251,74 s. Este incremento é consistente com o mecanismo do Random Oversampling, que duplica aleatoriamente instâncias da classe minoritária (Esquerda), aumentando o conjunto de treinamento efetivo e, conseqüentemente, o número de iterações necessárias para convergência. Conforme reportado na Tab. 6.6, o ganho preditivo associado é de 7,99 p.p. em acurácia (de 0,702 para 0,782), configurando uma relação custo-benefício favorável quando o re-treinamento é infrequente.

A aplicação do Random Oversampling no MLP resulta em redução das caudas de latência (P99 de 2,448 ms para 0,797 ms, -67,4%). Este comportamento pode estar associado à estabilização das fronteiras de decisão: o aumento da representação da classe Esquerda durante o treinamento pode resultar em regiões de decisão mais bem definidas, reduzindo a variabilidade nos caminhos de execução durante a inferência. O throughput máximo diminui moderadamente (-13,0%), permanecendo na ordem de 10^5 amostras/s, o que mantém margem operacional para serviços online e processamento em lote.

Para Extra Trees com ADASYN, o tempo de treinamento reduz em 40,4% (de 58,62 s para 34,96 s). A latência média permanece praticamente inalterada (-0,3%) e o throughput máximo aumenta expressivamente (+86,1%), de 580 para 1.079 amostras/s. As caudas apresentam aumento moderado (P95 +2,7%; P99 +5,5%), o que sugere pequena deterioração do pior caso sem alteração estrutural de ordem de grandeza. Considerando o ganho preditivo de 4,25 p.p. em acurácia (Tab. 6.16), este é um resultado favorável do ponto de vista computacional, pois melhora custo offline e capacidade em lote, preservando o patamar de latência online.

Para Random Forest com SMOTE, ocorre redução de treinamento (-12,1%), de 106,43 s para 93,55 s, redução de latência média (-7,4%) e redução das caudas (P99 -13,1%), com leve aumento de throughput (+7,2%). Embora o SMOTE tipicamente aumente o conjunto efetivo de

treinamento pela geração de instâncias sintéticas, o resultado observado sugere que a configuração selecionada durante a otimização resultou em um modelo de menor complexidade efetiva.

Para SVM com SMOTEENN, o treinamento aumenta de forma acentuada (+57,3%), de 333,90 s para 525,26 s. O batch recomendado reduz de 128 para 64 amostras, acompanhado de queda de throughput (−18,7%). As caudas reduzem levemente (P95 −7,5%; P99 −4,7%), enquanto a latência média aumenta (+10,2%), mas permanece sub-milisegundo (0,929 ms). Considerando o ganho preditivo de 2,45 p.p. em acurácia (Tab. 6.45), este resultado caracteriza uma penalidade offline relevante e perda de eficiência em lote, porém com manutenção de viabilidade online por latência absoluta baixa. A aceitabilidade desse custo depende da frequência de re-treinamento e do volume processado.

Para XGBoost com SMOTEENN, o tempo de treinamento é reduzido em 71,4% (de 49,66 s para 14,20 s), resultado associado à convergência para uma configuração de menor complexidade (75 estimadores e profundidade máxima de 5, conforme descrito na Seção 6.6). Entretanto, há deterioração acentuada das métricas de inferência: a latência média aumenta de 1,033 ms para 37,522 ms (aumento de 3.530,8%) e o P99 eleva-se para 54,830 ms, acompanhado de colapso de throughput (−95,0%), de 73.950 para 3.683 amostras/s. Esta degradação pode estar associada a características do modelo treinado sob SMOTEENN que induzem caminhos de decisão mais complexos durante a inferência. Considerando o ganho preditivo de apenas 1,29 p.p. em acurácia (Tab. 6.57), o ganho offline é obtido à custa de inviabilidade prática para uso online com restrição de tempo, e mesmo em processamento em lote há perda operacional importante.

6.9.3 Impacto do balanceamento na configuração em cascata

A Tab. 6.81 apresenta o impacto do balanceamento na cascata para os pares definidos como de melhor desempenho preditivo. A análise revela comportamentos distintos que refletem a interação entre as técnicas de reamostragem e a estrutura hierárquica do classificador.

Tabela 6.81 – Configuração em cascata: impacto computacional do balanceamento restrito aos melhores pares modelo–técnica, comparados ao sem balanceamento do mesmo modelo.

Métrica	Extra Trees (SMOTE)			SVM (SMOTEENN)			XGBoost (SMOTEENN)		
	Sem bal.	Bal.	Δ (%)	Sem bal.	Bal.	Δ (%)	Sem bal.	Bal.	Δ (%)
Tempo treino (s)	90.59	74.34	-17.9	371.53	464.92	25.1	79.50	36.67	-53.9
Latência média (ms)	40.584	123.659	204.7	1.625	1.607	-1.1	2.612	2.242	-14.2
P95 (ms)	51.990	465.780	795.9	1.816	3.629	99.8	3.959	2.945	-25.6
P99 (ms)	53.773	484.659	801.3	1.902	4.092	115.2	4.478	3.555	-20.6

Extra Trees com SMOTE reduz o tempo de treinamento em 17,9% (de 90,59 s para 74,34 s), porém impõe penalidade severa em inferência: a latência média aumenta de 40,584 ms para 123,659 ms (+204,7%) e as caudas apresentam crescimento acentuado (P95 +795,9%, de 51,990 ms para 465,780 ms; P99 +801,3%, de 53,773 ms para 484,659 ms). Considerando que, conforme a Tab. 6.78, o Extra Trees com SMOTE apresenta o maior *F1*-macro (0,796) entre os classificadores em cascata, o ganho preditivo de 4,77 p.p. em acurácia (em relação ao Extra Trees cascata sem balanceamento) deve ser ponderado contra a degradação temporal. Em termos de criticidade para sistemas online sob restrição de tempo, esse comportamento é tipicamente incompatível, pois a degradação concentra-se precisamente nos percentis P95/P99, que definem o risco de violações de prazo.

SVM com SMOTEENN na cascata aumenta o tempo de treinamento em 25,1% (de 371,53 s para 464,92 s) e mantém a latência média praticamente constante (−1,1%), porém as caudas deterioram (P95 +99,8%, de 1,816 ms para 3,629 ms; P99 +115,2%, de 1,902 ms para 4,092 ms). Este padrão caracteriza maior imprevisibilidade temporal: embora o caso médio permaneça estável, o pior caso apresenta degradação relevante. Conforme reportado na Tab. 6.47, o SMOTEENN na cascata do SVM não melhora o desempenho preditivo global (*F1*-macro de 0,674 versus 0,649 sem balanceamento, com redução de acurácia de −0,39 p.p.), embora promova redistribuição do desempenho por classe (melhoria de Repouso às custas de Direita, conforme Tab. 6.48). Em sistemas online, este padrão é particularmente problemático, pois melhora marginal em média não compensa a deterioração do P99.

XGBoost com SMOTEENN na cascata reduz o tempo de treinamento em 53,9% (de 79,50 s para 36,67 s) e apresenta redução consistente em todas as métricas de inferência: latência média −14,2% (de 2,612 ms para 2,242 ms), P95 −25,6% (de 3,959 ms para 2,945 ms) e P99 −20,6% (de 4,478 ms para 3,555 ms). Esta melhoria simultânea em custos offline e online

é notável e pode estar associada à configuração de hiperparâmetros selecionada durante a otimização, que resultou em um modelo de menor complexidade efetiva. Conforme a Tab. 6.60, o ganho preditivo é de 3,22 p.p. em acurácia (de 0,733 para 0,766), configurando benefício simultâneo em desempenho e custo. Entre os pares analisados na cascata, este é o único que melhora simultaneamente métricas offline e online, tornando-se o candidato mais consistente quando a restrição de tempo é relevante.

6.9.4 Implicações práticas e recomendações por regime de uso

Em regime offline, no qual o tempo de treinamento é a métrica de maior relevância e a inferência ocorre em lote sem restrições de tempo real, os melhores resultados são obtidos por Extra Trees com ADASYN (plano) e XGBoost com SMOTEENN (cascata), que reduzem tempo de treinamento enquanto mantêm ou melhoram métricas de inferência em lote. O Random Forest com SMOTE (plano) também apresenta coerência técnica por melhorar simultaneamente todas as métricas computacionais.

Em regime online, no qual a criticidade recai sobre P95/P99 e sobre a ordem de grandeza da latência, o MLP com Random Oversampling (plano) permanece altamente viável por operar em sub-milissegundo (latência média de 0,599 ms) e reduzir caudas (P99 de 0,797 ms). O SVM com SMOTEENN (plano) também permanece viável pela latência absoluta baixa (0,929 ms), apesar da perda de throughput e redução do batch. Entre os modelos em cascata, o XGBoost com SMOTEENN apresenta latência média de 2,242 ms e P99 de 3,555 ms, valores compatíveis com aplicações de tempo moderado.

Em contraste, Extra Trees com SMOTE (cascata) e XGBoost com SMOTEENN (plano) tornam-se criticamente desfavoráveis para uso online, pois elevam P95/P99 para patamares de dezenas a centenas de milissegundos, configurando risco de violação de prazo e degradação operacional mesmo quando há ganho preditivo. A adoção dessas configurações deve ser restrita a cenários em que a latência não constitui restrição operacional.

A análise conjunta do desempenho preditivo (Tabs. 6.77 e 6.78) e computacional indica que, para aplicações de interface cérebro-computador com requisitos simultâneos de desempenho preditivo e baixa latência, o SVM com SMOTEENN na configuração plana oferece o melhor equilíbrio: maior acurácia (0,826) e $F1$ -macro (0,822) entre todos os classificadores avaliados, com latência absoluta sub-milissegundo. Para aplicações em que o custo de trei-

namento é crítico e a inferência em lote é aceitável, o Random Forest com SMOTE (plano) constitui alternativa dominante na dimensão computacional.

7 CONCLUSÃO

Este trabalho investigou a classificação de sinais de eletroencefalograma (EEG) associados à imaginação motora para aplicação em interfaces cérebro-computador voltadas ao controle de cadeiras de rodas. O estudo utilizou dados do Projeto Milo, coletados por meio da plataforma OpenBCI Cyton com frequência de amostragem de 250 Hz e quatro eletrodos posicionados sobre o córtex sensoriomotor (C1, C2, C3 e C4), contemplando três classes: imaginação de movimento da mão esquerda, imaginação de movimento da mão direita e repouso. A extração de características baseou-se em geometria riemanniana, com estimação de matrizes de covariância regularizadas e projeção no espaço tangente. Seis algoritmos foram avaliados individualmente (MLP, SVM, Random Forest, Extra Trees, Gradient Boosting e XGBoost), além de um comitê de classificadores por votação suave. A comparação abrangeu duas arquiteturas de decisão: a classificação plana multiclasse e a classificação hierárquica em cascata, que decompõe o problema em duas etapas binárias sequenciais. A otimização de hiperparâmetros foi conduzida por random search com validação cruzada estratificada de cinco partições, adotando-se o $F1_{\text{macro}}$ como critério de seleção. A estimativa de desempenho final incluiu métricas complementares e intervalos de confiança de 95% obtidos por 2000 reamostras bootstrap.

As técnicas de balanceamento de classes apresentaram efeito heterogêneo entre modelos e arquiteturas. O SMOTEENN beneficiou o SVM e o XGBoost em ambas as configurações, enquanto o ADASYN favoreceu o Extra Trees e o comitê na configuração plana. O Random Oversampling promoveu ganhos no MLP plano e no comitê cascata. Para os demais modelos e configurações, o cenário sem balanceamento produziu os melhores resultados, indicando que a aplicação indiscriminada de técnicas de reamostragem não garante melhoria de desempenho. A redistribuição do desempenho entre classes observada após balanceamento, com aumento de recall em uma classe às custas de outra, sugere que os ganhos globais podem mascarar perdas localizadas que afetam a funcionalidade do sistema de controle. Na configuração plana, o SVM com reamostragem SMOTEENN apresentou $F1_{\text{macro}}$ de 0,822 (IC 95%: 0,794 a 0,848), acurácia de 0,826 (IC 95%: 0,800 a 0,852) e MCC de 0,748 (IC 95%: 0,710 a 0,785). Na configuração em cascata, o Extra Trees com SMOTE obteve $F1_{\text{macro}}$ de 0,796 (IC 95%: 0,768 a 0,823), acurácia de 0,799 (IC 95%: 0,772 a 0,826) e MCC de 0,703 (IC 95%: 0,662 a 0,744). A configuração plana superou a cascata nas métricas globais consolidadas, e a diferença entre

as arquiteturas foi confirmada pelo teste de McNemar ($p < 10^{-7}$), mesmo após correção de Holm-Bonferroni para comparações múltiplas.

Os intervalos de confiança de 95% obtidos por bootstrap indicaram que as estimativas pontuais do SVM plano com SMOTEENN apresentaram largura de 0,054 para $F1_{\text{macro}}$ e 0,052 para acurácia, valores que correspondem a aproximadamente 6% da escala útil de variação. A sobreposição parcial entre os intervalos de confiança de diferentes modelos impede a afirmação de superioridade estatística em todas as comparações pareadas, porém a consistência do SVM nas métricas baseadas em rótulo e a estabilidade dos intervalos conferem maior confiança na replicabilidade do desempenho em conjuntos de dados semelhantes. A estabilidade temporal do processo de classificação também foi maior no SVM, cujos percentis de latência (P95 e P99) permaneceram abaixo de 2 ms, indicando baixa variabilidade no tempo de resposta individual.

A análise de custo computacional revelou que o MLP na configuração plana apresentou a menor latência média (0,606 ms) e o maior throughput (215.844 amostras por segundo), tornando-o adequado para serviços de inferência com alta demanda de processamento paralelo. O SVM plano com SMOTEENN, apesar de menor throughput (3.990 amostras por segundo), manteve latência média de 0,929 ms e P99 de 1,074 ms, valores compatíveis com ciclos de controle da ordem de milissegundos. Em contraste, o Extra Trees cascata com SMOTE, que concentra o melhor desempenho preditivo na arquitetura em cascata, apresentou degradação expressiva das caudas de latência, com P95 de 465,78 ms e P99 de 484,66 ms.

A escolha entre configurações depende, portanto, dos requisitos específicos da aplicação. Para sistemas BCI embarcados ou de controle em tempo real, nos quais a latência de inferência não deve ultrapassar dezenas de milissegundos, o SVM plano com SMOTEENN oferece equilíbrio entre desempenho preditivo e custo computacional. A latência sub-milissegundo permite ciclos de atualização compatíveis com a frequência de amostragem do EEG (250 Hz, correspondente a 4 ms por amostra), viabilizando classificação contínua sem acúmulo de atraso. Para aplicações offline ou em lote, nas quais o tempo de inferência não constitui restrição operacional, o Extra Trees com ADASYN (ROC-AUC de 0,934, Brier Score de 0,096) ou o Gradient Boosting sem balanceamento (PR-AUC de 0,884) podem ser preferíveis por suas propriedades probabilísticas. A calibração das probabilidades preditas, mensurada pelo Brier Score, torna-se útil quando o sistema de controle adota limiares adaptativos ou requer estimativas de incerteza para tomada de decisão.

Os resultados deste trabalho indicam que a classificação de sinais de EEG para controle de cadeiras de rodas é tecnicamente viável com os algoritmos e configurações avaliados, porém a taxa de erro residual (aproximadamente 17% no melhor caso) requer consideração na concepção do sistema de controle final. Estratégias complementares, como a adoção de limiares de confiança para rejeição de classificações incertas, a integração com sistemas de navegação assistida ou a incorporação de feedback multimodal, podem mitigar os efeitos de erros de classificação na experiência do usuário. Investigações futuras podem explorar a adaptação online dos modelos a cada usuário específico, a incorporação de características temporais de longo prazo por meio de arquiteturas recorrentes, e a validação em cenários de uso real com participantes pertencentes à população-alvo de pessoas com deficiência motora.

Como limitações, destacam-se a avaliação restrita ao contexto *offline* e ao protocolo de dados disponível, a ausência de validação em tempo real e a influência variabilidade intersessão e interindivíduo, típicas de sinais de EEG. Como trabalhos futuros, recomenda-se prioritariamente a realização da coleta própria dos sinais em condições controladas, reduzindo a dependência de bases de terceiros e permitindo o desenho experimental alinhado aos requisitos do sistema (classes, rotulagem, protocolo de repetição e registro de metadados). Adicionalmente, é fundamental estender a validação para um ambiente *online*, incorporando restrições de latência, estabilidade e segurança, bem como mecanismos de decisão em tempo real (p. ex., rejeição por incerteza e suavização temporal), de modo a verificar a robustez da inteligência quando integrado ao ciclo de comando e supervisão de uma cadeira de rodas em operação.

REFERÊNCIAS

- ABIRI, R. et al. A comprehensive review of EEG-based brain–computer interface paradigms. **Journal of Neural Engineering**, IOP Publishing, v. 16, n. 1, p. 011001, 2019.
- AHAMEZ; TEAM, M. P. **Milo: Middleware for Large Language Models**. 2024. GitHub repository. Disponível em: <<https://github.com/ahamez/milo>>. Acesso em: 14 abr. 2025.
- AHMADI, H.; MESIN, L. Enhancing motor imagery electroencephalography classification with a correlation-optimized weighted stacking ensemble model. **Electronics**, v. 13, n. 6, p. 1033, 2024.
- AL-QAZAZ, N. K. et al. Electroencephalography signal processing: A comprehensive review and analysis of methods and techniques. **Sensors**, MDPI, v. 23, n. 14, p. 6434, 2023.
- ALBERT, H.; SIMERAL, J. D.; THOMPSON, D. E. User-centered design of a powered wheelchair controller based on a non-invasive brain-computer interface. **Frontiers in Human Neuroscience**, Frontiers, v. 17, p. 1183854, 2023.
- ALOTAIBY, T. N. et al. EEG seizure detection and prediction algorithms: A survey. **EURASIP Journal on Advances in Signal Processing**, Springer, v. 2023, n. 1, p. 1–29, 2023.
- ALTAHERI, H. et al. Deep learning techniques for classification of electroencephalogram (EEG) motor imagery (MI) signals: a review. **Neural Computing and Applications**, Springer, v. 35, n. 20, p. 14681–14722, 2023.
- ANG, K. K. et al. Filter bank common spatial pattern algorithm on bci competition iv datasets 2a and 2b. In: **Frontiers in Neuroscience**. [S.l.]: Frontiers, 2012. v. 6, p. 39.
- ARONSZAJN, N. Theory of reproducing kernels. **Transactions of the American Mathematical Society**, American Mathematical Society, v. 68, n. 3, p. 337–404, 1950.
- ARSIGNY, V. et al. Geometric means in a novel vector space structure on symmetric positive-definite matrices. **SIAM Journal on Matrix Analysis and Applications**, SIAM, v. 29, n. 1, p. 328–347, 2007.
- BANJANOVIC, E. S.; OSBORNE, J. W. Confidence intervals for effect sizes: Applying bootstrap resampling. **Practical Assessment, Research, and Evaluation**, v. 21, n. 1, p. 5, 2016.
- BARACHANT, A. et al. Multiclass brain-computer interface classification by Riemannian geometry. **IEEE Transactions on Biomedical Engineering**, IEEE, v. 59, n. 4, p. 920–928, 2012.
- BARACHANT, A. et al. Classification of covariance matrices using a Riemannian-based kernel for BCI applications. **Neurocomputing**, Elsevier, v. 112, p. 172–178, 2013.
- BASHASHATI, A. et al. A survey of signal processing algorithms in brain-computer interfaces based on electrical brain signals. **Journal of Neural Engineering**, IOP Publishing, v. 4, n. 2, p. R32–R57, 2007.

BATISTA, G. E.; PRATI, R. C.; MONARD, M. C. A study of the behavior of several methods for balancing machine learning training data. **ACM SIGKDD Explorations Newsletter**, ACM New York, NY, USA, v. 6, n. 1, p. 20–29, 2004.

BEAR, M. F.; CONNORS, B. W.; PARADISO, M. A. **Neuroscience: Exploring the Brain**. 3rd. ed. Philadelphia, PA: Lippincott Williams & Wilkins, 2007. ISBN 978-0-7817-6003-4.

BERGSTRA, J.; BENGIO, Y. Random search for hyper-parameter optimization. **Journal of Machine Learning Research**, v. 13, n. Feb, p. 281–305, 2012.

BIAU, G.; DEVROYE, L.; LUGOSI, G. Consistency of random forests and other averaging classifiers. **Journal of Machine Learning Research**, v. 9, p. 2015–2033, 2008.

BIRBAUMER, N. Breaking the silence: brain–computer interfaces (bci) for communication and motor control. **Psychophysiology**, Wiley Online Library, v. 43, n. 6, p. 517–532, 2006.

BIRBAUMER, N.; COHEN, L. G. Brain-computer interfaces: communication and restoration of movement in paralysis. **The Journal of Physiology**, Wiley Online Library, v. 579, n. 3, p. 621–636, 2006.

BLANKERTZ, B. et al. Optimizing spatial filters for robust EEG single-trial analysis. **IEEE Signal Processing Magazine**, IEEE, v. 25, n. 1, p. 41–56, 2008.

BREIMAN, L. Bagging predictors. **Machine Learning**, Springer, v. 24, n. 2, p. 123–140, 1996.

BREIMAN, L. Random forests. **Machine Learning**, Springer, v. 45, n. 1, p. 5–32, 2001.

BREIMAN, L. et al. **Classification and Regression Trees**. Boca Raton, FL: Chapman and Hall/CRC, 1984. ISBN 978-0412048418.

BRIER, G. W. Verification of forecasts expressed in terms of probability. **Monthly Weather Review**, v. 78, n. 1, p. 1–3, 1950.

BRODERSEN, K. H. et al. The balanced accuracy and its posterior distribution. **IEEE**, p. 3121–3124, 2010.

BROOKSHIRE, G. et al. Data leakage in deep learning studies of translational EEG. **Frontiers in Neuroscience**, Frontiers, v. 18, p. 1373515, 2024.

BUHLMANN, P.; HOTHORN, T. Boosting algorithms: Regularization, prediction and model fitting. **Statistical Science**, Institute of Mathematical Statistics, v. 22, n. 4, p. 477–505, 2007.

BUYUKKECECI, M.; OKUR, M. C. A comprehensive review of feature selection and feature selection stability in machine learning. **Gazi University Journal of Science**, v. 36, n. 4, p. 1506–1520, 2023.

CHAPELLE, O. et al. Choosing multiple parameters for support vector machines. **Machine Learning**, Springer, v. 46, n. 1-3, p. 131–159, 2002.

CHAWLA, N. V. et al. SMOTE: Synthetic minority oversampling technique. **Journal of Artificial Intelligence Research**, v. 16, p. 321–357, 2002.

CHEN, T.; GUESTRIN, C. XGBoost: A scalable tree boosting system. In: **Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining**. [S.l.]: ACM, 2016. p. 785–794.

CHICCO, D.; JURMAN, G. The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. **BMC Genomics**, v. 21, n. 1, p. 6, 2020.

CLERC, M.; BOUGRAIN, L.; LOTTE, F. **Brain-computer interfaces 1: Methods and perspectives**. [S.l.]: John Wiley & Sons, 2016.

COHEN, M. X. **Analyzing neural time series data: Theory and practice**. Cambridge, MA: MIT Press, 2014. ISBN 978-0262019873.

COMBRISSE, E.; JERBI, K. The caveat of theoretical chance levels in brain signal classification and statistical assessment of decoding accuracy. **Journal of Neuroscience Methods**, Elsevier, v. 250, p. 126–136, 2015.

CONGEDO, M.; BARACHANT, A.; BHATIA, R. Riemannian geometry for EEG-based brain-computer interfaces; a primer and a review. **Brain-Computer Interfaces**, Taylor & Francis, v. 4, n. 3, p. 155–174, 2017.

COOK, A. M.; POLGAR, J. M. **Assistive Technologies: Principles and Practice**. [S.l.]: Elsevier Health Sciences, 2020.

CORTES, C.; VAPNIK, V. Support-vector networks. **Machine Learning**, v. 20, n. 3, p. 273–297, 1995.

COVER, T. M.; THOMAS, J. A. **Elements of Information Theory**. 2. ed. Hoboken, NJ: John Wiley & Sons, 2006.

CRAIK, A.; HE, Y.; CONTRERAS-VIDAL, J. L. Deep learning for electroencephalogram (EEG) classification tasks: A review. **Journal of Neural Engineering**, IOP Publishing, v. 16, n. 3, p. 031001, 2019.

CRISTIANINI, N.; SHAW-TAYLOR, J. **An Introduction to Support Vector Machines and Other Kernel-based Learning Methods**. Cambridge, UK: Cambridge University Press, 2000. ISBN 978-0521780193.

CYBENKO, G. Approximation by superpositions of a sigmoidal function. **Mathematics of Control, Signals and Systems**, Springer, v. 2, n. 4, p. 303–314, 1989.

DALY, J. J.; WOLPAW, J. R. Brain-computer interfaces in neurological rehabilitation. **The Lancet Neurology**, Elsevier, v. 7, n. 11, p. 1032–1043, 2008.

DAVISON, A. C.; HINKLEY, D. V. **Bootstrap Methods and Their Application**. Cambridge, UK: Cambridge University Press, 1997. ISBN 978-0521574716.

DELORME, A. et al. EEG is better left alone. **Scientific Reports**, Nature Publishing Group, v. 13, n. 1, p. 2372, 2023.

- DEMSAR, J. Statistical comparisons of classifiers over multiple data sets. **Journal of Machine Learning Research**, v. 7, p. 1–30, 2006.
- DENG, X. et al. A bayesian shared control approach for wheelchair robot with brain machine interface. **IEEE Transactions on Neural Systems and Rehabilitation Engineering**, v. 28, n. 1, p. 328–338, 2020.
- DICICCIO, T. J.; EFRON, B. Bootstrap confidence intervals. **Statistical Science**, Institute of Mathematical Statistics, v. 11, n. 3, p. 189–228, 1996.
- DIETTERICH, T. G. Ensemble methods in machine learning. Springer, Berlin, Heidelberg, v. 1857, p. 1–15, 2000.
- EFRON, B.; TIBSHIRANI, R. J. **An Introduction to the Bootstrap**. [S.l.]: Chapman & Hall/CRC, 1994.
- ESPINELLI, J. H. C. Trabalho de Conclusão de Curso, **Uso do Eletroencefalograma e Instrumentação Virtual para Análise do Perfil do Sono**. Sorocaba, SP: [s.n.], 2011.
- FANG, H. et al. Feature extraction method based on filter banks and Riemannian tangent space in motor-imagery BCI. **IEEE Journal of Biomedical and Health Informatics**, IEEE, v. 26, n. 6, p. 2504–2514, 2022.
- FAWCETT, T. An introduction to roc analysis. **Pattern Recognition Letters**, v. 27, n. 8, p. 861–874, 2006.
- FERNANDES, S. et al. A novelty detection system for EEG data quality assessment in brain-computer interfaces. **Computer Methods and Programs in Biomedicine**, Elsevier, v. 242, p. 107879, 2024.
- FERNÁNDEZ, A. et al. SMOTE for learning from imbalanced data: progress and challenges, marking the 15-year anniversary. **Journal of artificial intelligence research**, v. 61, p. 863–905, 2018.
- FRIEDMAN, J. H. Greedy function approximation: A gradient boosting machine. **Annals of Statistics**, Institute of Mathematical Statistics, v. 29, n. 5, p. 1189–1232, 2001.
- FRIEDMAN, M. The use of ranks to avoid the assumption of normality implicit in the analysis of variance. **Journal of the American Statistical Association**, v. 32, n. 200, p. 675–701, 1937.
- GALÁN, F. et al. A brain-actuated wheelchair: asynchronous and non-invasive brain-computer interfaces for continuous control of robots. **Clinical Neurophysiology**, Elsevier, v. 119, n. 9, p. 2159–2169, 2008.
- GAO, L. et al. EEG classification for motor imagery and resting state in brain-computer interface using multi-resolution analysis and multi-objective optimization. **Review of Scientific Instruments**, v. 87, n. 8, p. 085110, 2016.
- GARCÍA, S. et al. Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power. **Information Sciences**, v. 180, n. 10, p. 2044–2064, 2010.

GARSON, G. D. Interpreting neural-network connection weights. **AI expert**, Miller Freeman, Inc. Lawrence, KS, USA, v. 6, n. 4, p. 46–51, 1991.

GAUR, P. et al. A sliding window common spatial pattern for enhancing motor imagery classification in EEG-bci. **IEEE Transactions on Instrumentation and Measurement**, IEEE, v. 70, p. 1–9, 2021.

GEURTS, P.; ERNST, D.; WEHENKEL, L. Extremely randomized trees. **Machine Learning**, v. 63, n. 1, p. 3–42, 2006.

GHASEMZADEH, H.; HILLMAN, R. E.; MEHTA, D. D. Toward generalizable machine learning models in speech, language, and hearing sciences: Estimating sample size and reducing overfitting. **Journal of Speech, Language, and Hearing Research**, ASHA, v. 67, n. 3, p. 753–781, 2024.

GLOROT, X.; BORDES, A.; BENGIO, Y. Deep sparse rectifier neural networks. In: **Proceedings of the 14th International Conference on Artificial Intelligence and Statistics**. [S.l.: s.n.], 2011. (JMLR Workshop and Conference Proceedings, v. 15), p. 315–323.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. **Deep Learning**. Cambridge, MA: MIT Press, 2016. ISBN 978-0-262-03561-3.

GORODKIN, J. Comparing two k-category assignments by a k-category correlation coefficient. **Computational Biology and Chemistry**, v. 28, n. 5-6, p. 367–374, 2004.

GRANDINI, M.; BAGLI, E.; VISANI, G. Metrics for multi-class classification: an overview. **arXiv preprint**, 2020.

GUGER, C.; GRÜNWALD, J.; XU, R. Noninvasive and invasive bcis and hardware and software components for bcis. In: _____. **Handbook of Neuroengineering**. Singapore: Springer Nature Singapore, 2023. p. 1193–1224. ISBN 978-981-16-5540-1.

GUO, C. et al. On calibration of modern neural networks. **Proceedings of the 34th International Conference on Machine Learning (ICML)**, PMLR, v. 70, p. 1321–1330, 2017.

HAN, H.; WANG, W.-Y.; MAO, B.-H. Borderline-SMOTE: A new over-sampling method in imbalanced data sets learning. In: **International Conference on Intelligent Computing (ICIC 2005)**. [S.l.: Springer, 2005. (Lecture Notes in Computer Science, v. 3644), p. 878–887.

HAND, D. J.; TILL, R. J. A simple generalisation of the area under the roc curve for multiple class classification problems. **Machine Learning**, v. 45, n. 2, p. 171–186, 2001.

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. **The Elements of Statistical Learning: Data Mining, Inference, and Prediction**. 2. ed. New York: Springer, 2009. ISBN 978-0-387-84857-0.

HAYKIN, S. **Neural Networks and Learning Machines**. 3. ed. Upper Saddle River, NJ: Pearson, 2009. ISBN 978-0131471399.

- HE, H. et al. Adasyn: Adaptive synthetic sampling approach for imbalanced learning. In: **IEEE. 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)**. [S.l.], 2008. p. 1322–1328.
- HE, H.; GARCIA, E. A. Learning from imbalanced data. **IEEE Transactions on Knowledge and Data Engineering**, IEEE, v. 21, n. 9, p. 1263–1284, 2009.
- HETTIARACHCHI, I. T. et al. A fresh look at functional link neural network for motor imagery-based brain–computer interface. **Journal of Neuroscience Methods**, v. 305, p. 28–35, 2018.
- HOLM, S. A simple sequentially rejective multiple test procedure. **Scandinavian Journal of Statistics**, v. 6, n. 2, p. 65–70, 1979.
- HOU, Y. et al. A novel method for classification of multi-class motor imagery tasks based on feature fusion. **Neuroscience Research**, Elsevier, v. 176, p. 40–48, 2022.
- HSU, C.-W.; CHANG, C.-C.; LIN, C.-J. A practical guide to support vector classification. **Technical Report**, 2003.
- IBGE. **Pessoas com deficiência têm menor acesso à educação, ao trabalho e à renda**. 2022. Agência IBGE Notícias. Disponível em: <<https://agenciadenoticias.ibge.gov.br/agencia-noticias/2012-agencia-de-noticias/noticias/37317-pessoas-com-deficiencia-tem-menor-acesso-a-educacao-ao-trabalho-e-a-renda>>. Acesso em: 14 abr. 2025.
- JAS, M. et al. Autoreject: Automated artifact rejection for MEG and EEG data. **NeuroImage**, Elsevier, v. 159, p. 417–429, 2017.
- JIANG, X.; BIAN, G.-B.; TIAN, Z. Removal of artifacts from EEG signals: a review. **Sensors**, MDPI, v. 19, n. 5, p. 987, 2019.
- JR., C. N. S.; FREITAS, A. A. A survey of hierarchical classification across different application domains. **Data Mining and Knowledge Discovery**, v. 22, n. 1-2, p. 31–72, 2011.
- KE, G. et al. LightGBM: A highly efficient gradient boosting decision tree. In: **Advances in Neural Information Processing Systems**. [S.l.]: Curran Associates, 2017. v. 30, p. 3146–3154.
- KINGMA, D. P.; BA, J. Adam: A method for stochastic optimization. **arXiv preprint arXiv:1412.6980**, 2015.
- KRASKOV, A.; STÖGBAUER, H.; GRASSBERGER, P. Estimating mutual information. **Physical Review E**, American Physical Society, v. 69, n. 6, p. 066138, 2004.
- KUBLER, A.; HOLZ, E. M.; BOTREL, L. Brain-computer interfaces for communication in paralysis. In: **Clinical Systems Neuroscience**. [S.l.]: Springer, Tokyo, 2013. p. 151–168.
- KUHN, M.; JOHNSON, K. **Applied predictive modeling**. New York: Springer, 2013.
- LAUREYS, S. et al. The locked-in syndrome: what is it like to be conscious but paralyzed and voiceless? **Progress in brain research**, Elsevier, v. 150, p. 495–611, 2005.

- LAWHERN, V. J. et al. EEGNet: A compact convolutional neural network for EEG-based brain-computer interfaces. **Journal of Neural Engineering**, IOP Publishing, v. 15, n. 5, p. 056013, 2018.
- LEDOIT, O.; WOLF, M. A well-conditioned estimator for large-dimensional covariance matrices. **Journal of Multivariate Analysis**, Elsevier, v. 88, n. 2, p. 365–411, 2004.
- LEE, H.-T. et al. Risk of data leakage in estimating the diagnostic performance of a deep-learning-based computer-aided system for psychiatric disorders. **Scientific Reports**, Nature Publishing Group, v. 14, n. 1, p. 2210, 2024.
- LEE, J.; JEONG, J. Improving multi-class motor imagery EEG classification using overlapping sliding window and deep learning model. **Electronics**, MDPI, v. 12, n. 5, p. 1186, 2023.
- LEMAÎTRE, G.; NOGUEIRA, F.; ARIDAS, C. K. Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning. **Journal of Machine Learning Research**, v. 18, n. 17, p. 1–5, 2017.
- LEMM, S. et al. Introduction to machine learning for brain imaging. **NeuroImage**, v. 56, n. 2, p. 387–399, 2011.
- LONG, J. et al. A hybrid brain computer interface to control the direction and speed of a simulated or real wheelchair. **IEEE Transactions on Neural Systems and Rehabilitation Engineering**, v. 20, n. 5, p. 720–729, 2012.
- LOTTE, F. et al. A review of classification algorithms for EEG-based brain–computer interfaces: A 10 year update. **Journal of Neural Engineering**, v. 15, n. 3, p. 031005, 2018.
- LOTTE, F. et al. A review of classification algorithms for EEG-based brain-computer interfaces: a 10 year update. **Journal of Neural Engineering**, v. 15, n. 3, p. 031005, 2018.
- LOTTE, F. et al. A review of classification algorithms for EEG-based brain-computer interfaces. **Journal of Neural Engineering**, IOP Publishing, v. 4, n. 2, p. R1–R13, 2007.
- LOTTE, F.; GUAN, C. Regularizing common spatial patterns to improve BCI designs: unified theory and new algorithms. **IEEE Transactions on Biomedical Engineering**, IEEE, v. 58, n. 2, p. 355–362, 2011.
- LOUPPE, G. Understanding random forests: From theory to practice. **PhD Thesis, University of Liège**, 2014.
- LUO, J. et al. Motor imagery eeg classification based on ensemble support vector learning. **Computer Methods and Programs in Biomedicine**, v. 193, p. 105464, 2020.
- MAK, J. N.; WOLPAW, J. R. Clinical applications of brain-computer interfaces: current state and future prospects. **IEEE Reviews in Biomedical Engineering**, IEEE, v. 2, p. 187–199, 2009.
- MANI, I.; ZHANG, I. kNN approach to unbalanced data distributions: A case study involving information extraction. In: **Proceedings of the Workshop on Learning from Imbalanced Datasets (ICML 2003)**. [S.l.: s.n.], 2003. p. 1–7.

- MASON, L. et al. Boosting algorithms as gradient descent. In: **Advances in Neural Information Processing Systems**. [S.l.]: MIT Press, 1999. v. 12, p. 512–518.
- MILLAN, J. d. R. et al. Asynchronous non-invasive brain-actuated control of an intelligent wheelchair. In: IEEE. **2009 Annual International Conference of the IEEE Engineering in Medicine and Biology Society**. [S.l.], 2009. p. 3361–3364.
- MILLAN, J. d. R. et al. Combining brain-computer interfaces and assistive technologies: state-of-the-art and challenges. **Frontiers in Neuroscience**, Frontiers, v. 4, p. 161, 2010.
- MITRA, P.; BOKIL, H. **Observed brain dynamics**. New York: Oxford University Press, 2007. ISBN 978-0195178081.
- MOKHTAR, S. A.; EL-RAFAIE, H. M.; MAHMOUD, T. A. Confidence interval estimation for machine learning classification algorithms using bootstrap methods. **Journal of Computational and Applied Mathematics**, v. 418, p. 114639, 2023.
- MULLER, K.-R. et al. Machine learning techniques for brain-computer interfaces. **Biomedizinische Technik**, Walter de Gruyter, v. 49, n. 1, p. 11–22, 2008.
- MURPHY, A. H. A new vector partition of the probability score. **Journal of Applied Meteorology**, v. 12, n. 4, p. 595–600, 1973.
- NAEINI, M. P.; COOPER, G. F.; HAUSKRECHT, M. Obtaining well calibrated probabilities using Bayesian binning. In: **Proceedings of the AAAI Conference on Artificial Intelligence**. [S.l.: s.n.], 2015. v. 29, n. 1, p. 2901–2907.
- NAIR, V.; HINTON, G. E. Rectified linear units improve restricted Boltzmann machines. In: **Proceedings of the 27th International Conference on Machine Learning**. Madison, WI: Omnipress, 2010. p. 807–814.
- NATEKIN, A.; KNOLL, A. Gradient boosting machines, a tutorial. **Frontiers in Neurorobotics**, Frontiers, v. 7, p. 21, 2013.
- NEMENYI, P. **Distribution-free Multiple Comparisons**. Tese (Doutorado) — Princeton University, 1963.
- NEUPER, C.; PFURTSCHELLER, G. Event-related dynamics of cortical rhythms during motor imagery. **Clinical Neurophysiology**, v. 117, n. 5, p. 1091–1100, 2006.
- NEUPER, C. et al. Imagery of motor actions: Differential effects of kinesthetic and visual-motor mode of imagery in single-trial EEG. **Cognitive Brain Research**, Elsevier, v. 25, n. 3, p. 668–677, 2005.
- NG, A. Y. Feature selection, L1 vs. L2 regularization, and rotational invariance. **ACM**, p. 78, 2004.
- NICOLAS-ALONSO, L. F.; GOMEZ-GIL, J. Brain computer interfaces, a review. **Sensors**, Multidisciplinary Digital Publishing Institute, v. 12, n. 2, p. 1211–1279, 2012.

NICULESCU-MIZIL, A.; CARUANA, R. Predicting good probabilities with supervised learning. In: **Proceedings of the 22nd International Conference on Machine Learning**. [S.l.]: ACM, 2005. p. 625–632.

NIELSEN, D. **Tree boosting with XGBoost-why does XGBoost win "every" machine learning competition**. Dissertação (Mestrado) — NTNU, 2016.

NOLAN, H.; WHELAN, R.; REILLY, R. B. Faster: Fully automated statistical thresholding for EEG artifact rejection. **Journal of Neuroscience Methods**, Elsevier, v. 192, n. 1, p. 152–162, 2010.

OJALA, M.; GARRIGA, G. C. Permutation tests for studying classifier performance. **Journal of Machine Learning Research**, v. 11, p. 1833–1863, 2010.

ORGANIZATION, W. H.; BANK, T. W. **World Report on Disability**. Geneva: World Health Organization, 2011.

OSHIRO, T. M.; PEREZ, P. S.; BARANAUSKAS, J. A. How many trees in a random forest? Springer, p. 154–168, 2012.

PALUMBO, A. et al. Motor-imagery eeg-based bcis in wheelchair movement and control: A systematic literature review. **Sensors**, v. 21, n. 18, p. 6285, 2021.

PAN, L. et al. Riemannian geometric and ensemble learning for decoding cross-session motor imagery electroencephalography signals. **Journal of Neural Engineering**, v. 20, n. 6, 2023.

PEDREGOSA, F. et al. Scikit-learn: Machine learning in Python. **Journal of Machine Learning Research**, v. 12, p. 2825–2830, 2011.

PFURTSCHELLER, G. et al. Mu rhythm (de)synchronization and EEG single-trial classification of different motor imagery tasks. **NeuroImage**, Elsevier, v. 31, n. 1, p. 153–159, 2006.

PFURTSCHELLER, G.; NEUPER, C. Motor imagery and direct brain-computer communication. **Proceedings of the IEEE**, IEEE, v. 89, n. 7, p. 1123–1134, 2001.

PHIPSON, B.; SMYTH, G. K. Permutation p-values should never be zero: Calculating exact p-values when permutations are randomly drawn. **Statistical Applications in Genetics and Molecular Biology**, v. 9, n. 1, p. Article 39, 2010.

PLATT, J. C. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In: **Advances in Large Margin Classifiers**. [S.l.]: MIT Press, 1999. p. 61–74.

PRABHAKAR, S. K.; WON, D.-O. Ensemble fusion models using various strategies and machine learning for EEG classification. **Bioengineering**, MDPI, v. 11, n. 10, p. 986, 2024.

PRECHELT, L. Early stopping - but when? In: _____. **Neural Networks: Tricks of the Trade**. Berlin, Heidelberg: Springer Berlin Heidelberg, 1998. p. 55–69. ISBN 978-3-540-49430-0.

QIAN, X. et al. Brain-computer-interface-based intervention re-normalizes brain functional network topology in children with attention deficit/hyperactivity disorder. **Translational psychiatry**, Nature Publishing Group UK London, v. 8, n. 1, p. 149, 2018.

RAMOSER, H.; MÜLLER-GERKING, J.; PFURTSCHELLER, G. Optimal spatial filtering of single trial eeg during imagined hand movement. **IEEE Transactions on Rehabilitation Engineering**, v. 8, n. 4, p. 441–446, 2000.

RASCHKA, S. Creating confidence intervals for machine learning classifiers. **Sebastian Raschka Blog**, 2022.

RASHID, M. et al. Current status, challenges, and possible solutions of EEG-based brain-computer interface: a comprehensive review. **Frontiers in Neurorobotics**, Frontiers, v. 14, p. 25, 2020.

REFAELZADEH, P.; TANG, L.; LIU, H. Cross-validation. In: **Encyclopedia of Database Systems**. Boston, MA: Springer, 2009. p. 532–538.

RIFKIN, R.; KLAUTAU, A. In defense of one-vs-all classification. **Journal of Machine Learning Research**, v. 5, p. 101–141, 2004.

RODRIGUES, P. L. C.; JUTTEN, C.; CONGEDO, M. Riemannian Procrustes analysis: Transfer learning for brain-computer interfaces. **IEEE Transactions on Biomedical Engineering**, IEEE, v. 66, n. 8, p. 2390–2401, 2019.

RODRIGUES, T.; CARDOSO, J. S.; NICOLAU, H. Sample selection for EEG motor imagery classification. **arXiv preprint arXiv:2401.17707**, 2024.

ROUSSEEUW, P. J.; CROUX, C. **Alternatives to the Median Absolute Deviation**. [S.l.]: Taylor & Francis, 1993. v. 88. 1273–1283 p.

SAITO, T.; REHMSMEIER, M. The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets. **PLOS ONE**, v. 10, n. 3, p. e0118432, 2015.

SASAKI, Y. The truth of the F-measure. **Teach Tutor Mater**, v. 1, n. 5, p. 1–5, 2007.

SCHIRRMESTER, R. T. et al. Deep learning with convolutional neural networks for EEG decoding and visualization. **Human Brain Mapping**, Wiley Online Library, v. 38, n. 11, p. 5391–5420, 2017.

SCHÖLKOPF, B.; SMOLA, A. J. **Learning with kernels: support vector machines, regularization, optimization, and beyond**. Cambridge, MA: MIT Press, 2002. ISBN 978-0262194754.

SIMPSON, R. C.; LOPRESTI, E. F.; COOPER, R. A. How many people would benefit from a smart wheelchair? **Journal of Rehabilitation Research & Development**, v. 45, n. 1, p. 53–72, 2008.

SOKOLOVA, M.; LAPALME, G. A systematic analysis of performance measures for classification tasks. **Information Processing & Management**, v. 45, n. 4, p. 427–437, 2009.

STEYRL, D. et al. Random forests in non-invasive sensorimotor rhythm brain-computer interfaces: a practical and convenient non-linear classifier. **Biomedical Engineering / Biomedizinische Technik**, v. 61, n. 1, p. 77–86, 2016.

TANGERMANN, M. et al. Review of the bci competition iv. **Frontiers in Neuroscience**, Frontiers, v. 6, p. 55, 2012.

TERPIŁOWSKI, M. A. scikit-posthocs: Pairwise multiple comparison tests in python. **Journal of Open Source Software**, v. 4, n. 36, p. 1169, 2019.

THENG, D.; BHOYAR, K. K. Feature selection techniques for machine learning: A survey of more than two decades of research. **Knowledge and Information Systems**, Springer, v. 66, p. 1575–1637, 2024.

THENMOZHI, T.; HELEN, R. Feature selection using extreme gradient boosting bayesian optimization to upgrade the classification performance of motor imagery signals for bci. **Journal of Neuroscience Methods**, Elsevier, v. 366, p. 109425, 2022.

TIBSHIRANI, R. Regression shrinkage and selection via the lasso. **Journal of the Royal Statistical Society: Series B (Methodological)**, Wiley, v. 58, n. 1, p. 267–288, 1996.

TOMEK, I. Two modifications of CNN. **IEEE Transactions on Systems, Man, and Cybernetics**, v. 6, n. 11, p. 769–772, 1976.

TSUI, C. S. L.; GAN, J. Q.; HU, H. A self-paced motor imagery based brain-computer interface for robotic wheelchair control. **Clinical EEG and Neuroscience**, v. 42, n. 4, p. 225–229, 2011.

United Nations. **Sustainable Development Goals: Reduced Inequalities**. [S.l.]: United Nations, 2015. Disponível em: <<https://www.undp.org/sustainable-development-goals/reduced-inequalities>>. Acesso em: 13 maio 2025.

VAPNIK, V. N. **Statistical Learning Theory**. New York: Wiley-Interscience, 1998. ISBN 978-0471030034.

VAROQUAUX, G. et al. Assessing and tuning brain decoders: cross-validation, caveats, and guidelines. **NeuroImage**, Elsevier, v. 145, p. 166–179, 2017.

VIDAL, J. J. Toward direct brain-computer communication. **Annual review of Biophysics and Bioengineering**, Annual Reviews 4139 El Camino Way, PO Box 10139, Palo Alto, CA 94303-0139, USA, v. 2, n. 1, p. 157–180, 1973.

VIRTANEN, P. et al. SciPy 1.0: Fundamental algorithms for scientific computing in python. **Nature Methods**, v. 17, n. 3, p. 261–272, 2020.

WIDMANN, A.; SCHRÖGER, E.; MAESS, B. Digital filter design for electrophysiological data – a practical approach. **Journal of Neuroscience Methods**, Elsevier, v. 250, p. 34–46, 2015.

WILSON, D. L. Asymptotic properties of nearest neighbor rules using edited data. **IEEE Transactions on Systems, Man, and Cybernetics**, IEEE, v. 2, n. 3, p. 408–421, 1972.

WOLPAW, J. R. Brain–computer interfaces. In: **Handbook of clinical neurology**. [S.l.]: Elsevier, 2013. v. 110, p. 67–74.

WOLPAW, J. R. et al. Brain–computer interfaces for communication and control. **Clinical neurophysiology**, Elsevier, v. 113, n. 6, p. 767–791, 2002.

WOLPERT, D. H. Stacked generalization. **Neural Networks**, Elsevier, v. 5, n. 2, p. 241–259, 1992.

ZADROZNY, B.; ELKAN, C. Transforming classifier scores into accurate multiclass probability estimates. In: **Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining**. [S.l.]: ACM, 2002. p. 694–699.

ZHANG, J.; WANG, B. A survey on robots controlled by motor imagery brain–computer interfaces. **Cognitive Robotics**, v. 1, 2021.

ZHANG, X. et al. Hybrid brain-computer interface techniques for improved classification accuracy and increased number of commands: A review. **Frontiers in neurorobotics**, Frontiers Media SA, v. 15, p. 4, 2021.

ZHOU, Z.-H. **Ensemble Methods: Foundations and Algorithms**. Boca Raton, FL: Chapman and Hall/CRC, 2012.

ZOU, H.; HASTIE, T. Regularization and variable selection via the elastic net. **Journal of the Royal Statistical Society: Series B (Statistical Methodology)**, Wiley, v. 67, n. 2, p. 301–320, 2005.