

Instituto de Ciências Exatas Departamento de Ciência da Computação

### Programação Genética para Classificação Interpretável e de Alto Desempenho de Sinais EEG

Ícaro Marcelino Miranda

Dissertação apresentada como requisito parcial para conclusão do Mestrado Profissional em Computação Aplicada

Orientador Prof. Dr. Luís Paulo Faina Garcia

Coorientador Prof. Dr. Claus de Castro Aranha

> Brasília 2025

#### Ficha catalográfica elaborada automaticamente, com os dados fornecidos pelo(a) autor(a)

Miranda, Ícaro Marcelino

Programação Genética para Classificação Interpretável e de Alto Desempenho de Sinais EEG / Ícaro Marcelino Miranda; orientador Luís Paulo Faina Garcia; co-orientador Claus de Castro Aranha. Brasília, 2025.

109 p.

Dissertação(Mestrado Profissional em Computação Aplicada) Universidade de Brasília, 2025.

1. Aprendizado de Máquina. 2. Classificação. 3. Sinais de EEG. 4. Programação Genética. 5. Séries Temporais. I. Faina Garcia, Luís Paulo, orient. II. de Castro Aranha, Claus, co-orient. III. Título.

M672p



Instituto de Ciências Exatas Departamento de Ciência da Computação

### Programação Genética para Classificação Interpretável e de Alto Desempenho de Sinais EEG

Ícaro Marcelino Miranda

Dissertação apresentada como requisito parcial para conclusão do Mestrado Profissional em Computação Aplicada

Prof. Dr. Luís Paulo Faina Garcia (Orientador) CIC/UnB

Prof. Dr. Claus de Castro Aranha (Coorientador) University of Tsukuba Prof.a Dr.a Ana Carolina Lorena Instituto Tecnológico de Aeronáutica

Prof. Dr. Thiago de Paulo Faleiros CIC/UnB

Prof.a Dr.a Edna Dias Canedo Coordenador do Programa de Pós-graduação em Computação Aplicada

Brasília, 03 de abril de 2025

# Dedicatória

Dedico este trabalho à minha mãe, Ediane; ao meu irmão, Érick; e à minha companheira, Joyce.

### Agradecimentos

A conclusão desta dissertação marca o encerramento de uma etapa fundamental e desafiadora em minha trajetória acadêmica e profissional. Esse percurso foi difícil, mas extremamente enriquecedor. E não teria sido possível sem o apoio e a colaboração de diversas pessoas, às quais expresso minha mais sincera gratidão.

Agradeço primeiramente ao meu orientador, Luís Garcia, por sua dedicação, companheirismo, (muita) paciência e incentivo ao longo desta jornada. Sua orientação criteriosa e vasto conhecimento foram essenciais para o desenvolvimento deste trabalho. Agradeço também ao meu coorientador, Claus Aranha, por suas contribuições valiosas e pelo apoio contínuo.

Tive o prazer e a sorte de conhecer o professor Claus e o professor Marcelo Ladeira durante minha graduação em 2015. Graças ao incentivo e orientação deles nesses últimos quase 10 anos pude aprender e crescer como pesquisador e profissional de *Machine Learning*, que hoje é minha área de atuação.

Em especial, deixo um grande agradecimento ao professor Ladeira que me acompanhou e mentorou todos esses anos. Obrigado pela persistência, pois sei que nem sempre fui um aluno fácil!

Ao Yuri Lavinas e colegas do LARA, que compartilharam comigo aprendizados e desafios, criando um ambiente de colaboração e crescimento mútuo. Suas sugestões e discussões foram fundamentais para o aprimoramento deste trabalho.

Sou imensamente grato à minha família, cujo amor e apoio foram uma grande fonte de força e motivação. Obrigado por acreditarem em mim e por estarem ao meu lado em todos os momentos.

Por fim, agradeço à UnB e ao PPCA pelo suporte e compreensão durante o meu mestrado, período em que tive algumas idas e vindas.

Deixo a todos vocês meu mais sincero obrigado!

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES), por meio do Acesso ao Portal de Periódicos.

### Resumo

Sinais de Eletroencefalograma (EEG) são essenciais para diagnosticar condições neurológicas, como epilepsia e distúrbios do sono, pois representam graficamente a atividade cerebral de um indivíduo em tempo real. Contudo, a complexidade desses sinais, caracterizada pela alta variabilidade inter e intra-sujeitos, covariância espacial e temporal, além de sua natureza não estacionária, dificulta a extração de atributos discriminativos e compromete a generalização dos modelos de Aprendizado de Máquina (AM). Neste estudo, propomos o Genetic Programming-Based Feature Selection and Construction (GP-FeSCo), um framework baseado em Programação Genética (PG) para a engenharia automatizada de atributos em sinais de EEG. O GP-FeSCo permite a construção e seleção evolutiva de atributos, adaptando representações complexas ao contexto específico de cada tarefa, melhorando o desempenho e a interpretabilidade dos modelos de classificação tradicionais. Para garantir avaliações robustas e alinhadas com cenários do mundo real, adotamos a validação cruzada Leave-One-Subject-Out (LOSO), que simula a aplicação prática ao testar os modelos exclusivamente em sujeitos não vistos anteriormente, reduzindo vieses associados à segmentação tradicional dos dados. Realizamos experimentos em três problemas de classificação de sinais de EEG: Identificação de Fusos do Sono, Detecção de Alcoolismo e Classificação de Imagética Motora, utilizando cinco conjuntos de atributos amplamente citados na literatura. O desempenho dos atributos gerados pela PG foi avaliado, assim como a convergência da função fitnees das populações e a complexidade dos modelos gerados. Os resultados mostram que os atributos construídos pelo GP-FeSCo não apenas melhoraram o desempenho dos classificadores em F1-score, mas também possibilitaram a redução no número de atributos necessários para a classificação, sem aumento de overfitting. Além disso, a análise da frequência de seleção de atributos evidenciou os canais de EEG mais relevantes para cada tarefa, contribuindo para otimizar futuras aquisições de sinais, tornando-as mais eficientes e acessíveis.

Palavras-chave: Aprendizado de Máquina, Classificação, EEG, Programação Genética, Séries Temporais, Sobreajuste

### Abstract

Electroencephalography (EEG) signals are essential for diagnosing neurological conditions such as epilepsy and sleep disorders, as they graphically represent an individual's brain activity in real-time. However, the complexity of these signals, characterized by high inter- and intra-subject variability, spatial and temporal covariance, and their nonstationary nature, makes it challenging to extract discriminative features and compromises the generalization of Machine Learning (ML) models. In this study, we propose Genetic Programming-Based Feature Selection and Construction (GP-FeSCo), a framework based on Genetic Programming (GP) for the automated engineering of EEG features. GP-FeSCo enables the evolutionary construction and selection of features, adapting complex representations to the specific context of each task and enhancing both the performance and interpretability of traditional classification models. To ensure robust evaluations aligned with real-world scenarios, we adopt Leave-One-Subject-Out (LOSO) cross-validation, which simulates practical applications by testing models exclusively on previously unseen subjects, mitigating biases associated with conventional data segmentation methods. We conducted experiments on three EEG classification problems: Sleep Spindle Identification, Alcoholism Detection, and Motor Imagery Classification, using five widely referenced feature sets. The performance of the GP-generated features was assessed in addition to analyzing the convergence of the population's fitness function and the complexity of the evolved models. Results show that the features constructed by GP-FeSCo not only improved classifier performance in terms of F1-score but also enabled a reduction in the number of features required for classification without increasing overfitting. Furthermore, the analysis of feature selection frequency revealed the most relevant EEG channels for each task, contributing to optimizing future signal acquisitions, making them more efficient and accessible.

**Keywords:** Classification, EEG, Genetic Programming, Machine Learning, Overfitting, Time Series

# Sumário

1	Intr	rodução	1
	1.1	Justificativa	3
	1.2	Objetivos	3
	1.3	Hipóteses de Pesquisa	4
		1.3.1 Hipótese Principal	4
		1.3.2 Hipóteses Secundárias	4
	1.4	Organização do Documento	5
<b>2</b>	Fun	damentação Teórica	6
	2.1	Sinais EEG	6
	2.2	Programação Genética	9
	•	2.2.1 Tipos de Programação Genética	2
		2.2.2 Representação	4
		2.2.3 Função de Aptidão (fitness)	4
		2.2.4 Seleção, recombinação e mutação	5
		2.2.5 Fluxo tradicional da Programação Genética	6
	:	2.2.6 Inchaço	7
		2.2.7 Programação Genética para Construção de Atributos	7
	2.3	Validação Cruzada Orientada a Paciente	9
	2.4	Outras aplicações de Aprendizado de Máquina para problemas de EEG $$ $$ 2	3
	2.5	Conjuntos de Dados de EEG	4
		2.5.1 Imagética Motora	5
	:	2.5.2 Fusos do Sono - Sleep Spindles	6
	:	2.5.3 Detecção de Alcoolismo	7
3	Solı	ução Proposta 2	9
	3.1	Estrutura e Funcionamento do GP-FeSCo	9
	;	3.1.1 Melhorias de Eficiência e Redução de Redundância	3
		3.1.2 Aplicação de LOSO	5

	3.1.3 Processo de Extração de Atributos	35
4	Metodologia	41
	4.1 Pré-processamento e Extração de Atributos	41
	4.1.1 Segmentação dos Sinais	41
	4.1.2 Descrição dos Atributos	42
	4.2 Aplicação do GP-FeSCo	47
	4.2.1 Classificadores Utilizados	47
	4.2.2 Parâmetros de Programação Genética	48
	4.3 Validação Cruzada por Paciente	49
	4.4 Análise dos Resultados	50
	4.5 Visão Geral	
5	Resultados	<b>52</b>
	5.1 Análise Comparativa por Conjunto de Dados	52
	5.1.1 Sleep Spindles	53
	5.1.2 Detecção de Alcoolismo	63
	5.1.3 Imagética Motora	68
	5.2 Discussão Geral dos Resultados	75
6	Conclusão	79
	6.1 Limitações e Trabalhos Futuros	81
$\mathbf{R}$	Referências	83

# Lista de Figuras

2.1	Trecho de exame de Polissonografia (PSG) ilustrando múltiplos sinais fisi-	
	ológicos registrados simultaneamente. Os canais incluem: EEG (Fp1, Cz,	
	O1), EOG, EMG e ECG	7
2.2	Posicionamento dos eletrodos Fp1, C3, Cz, O1 e O2 conforme o sistema in-	
	ternacional 10-20, com destaque para a visualização lateral esquerda. Esses	
	canais são frequentemente utilizados em estudos de sono	8
2.3	Exemplo de um programa gerado por PG baseada em árvore. Os nós	
	internos correspondem a operadores matemáticos $(+, \times, \log)$ , enquanto os	
	nós terminais representam variáveis de entrada (A, B) e constantes	10
2.4	Exemplo de crossover em PG). Duas árvores representando indivíduos pais	
	trocam subárvores em um ponto selecionado aleatoriamente ( $One\ Point$	
	Crossover), gerando dois novos indivíduos filhos	15
2.5	Exemplo de mutação em PG. Neste caso, um nó da árvore representando	
	a função original foi modificado, substituindo a operação log por sin	16
2.6	Fluxo tradicional de execução da PG	17
2.7	Exemplo do fenômeno de $bloat$ (inchaço) em PG. O indivíduo original con-	
	tém expressões redundantes e desnecessárias que não contribuem para a	
	funcionalidade do programa	18
2.8	Comparação entre os esquemas de Validação Cruzada K-Fold (à esquerda)	
	e Validação Cruzada por Paciente (LOSO, à direita). Cada linha repre-	
	senta os dados de um paciente, e cada bloco representa um subconjunto de	
	amostras ao longo das iterações de validação. As amostras de treino estão	
	em verde e as de teste em roxo	21
3.1	Fluxograma do framework GP-FesCo para análise de dados de EEG	30
3.2	Ilustração de uma árvore de construção de atributos no framework GP-	
	FesCo.	31

3.3	Evolução do melhor indivíduo ao longo das gerações no framework GP-FeSCo. À esquerda, a árvore gerada na geração inicial (#0), composta por operadores simples e estruturas rasas, resulta em baixo desempenho preditivo (F1-score = 0.49). À direita, após 99 gerações, observa-se o aumento de complexidade da árvores, com combinações mais elaboradas de	
3.4	atributos, elevando significativamente o desempenho (F1-score = 0.87) Fluxo de decomposição e extração de atributos a partir de sinais EEG. O sinal bruto do canal FP1 é segmentado em cinco faixas de frequência, e cada componente é submetido à extração de atributos em quatro grupos distintos. Os mapas de calor representam os valores normalizados dos atributos extraídos (linhas: faixas de frequência; colunas: atributos), permitindo visualizar a distribuição e intensidade de cada característica em diferentes bandas	344
4.1	Fluxo detalhado da metodologia aplicada no GP-FeSCo	51
5.1	Desempenho médio de F1-Score sobre o conjunto de teste para o problema de detecção de fusos do sono (Sleep Spindles) nos cenários Baseline (à	
5.2	esquerda) e com a utilização do GP-FeSCo (à direita)	54
5.3	querda) e GP-FeSCo (à direita)	54
5.4	a diferença para um paciente (0 a 7)	54
5.5	avaliado sobre os diferentes pacientes	56
5.6	atributos no conjunto <i>Sleep Spindles</i>	57
	como entrada	58

5.7	Frequência dos 20 atributos mais selecionados pelo melhor indivíduo para os classificadores no conjunto de dados <i>Sleep Spindles</i> ao usar todos os	
	conjuntos de atributos como entrada	59
5.8	Melhor indivíduo gerado pelo GP-FeSCo para o problema de detecção de	
	Sleep Spindles, utilizando o classificador DT sobre o conjunto de atributos	
	catch22. A árvore representa a expressão evoluída pelo algoritmo, desta-	
	cando os novos atributos construídos (F1, F2 e F3) que contribuem para a	
	tomada de decisão do modelo	60
5.9	Melhor indivíduo gerado pelo GP-FeSCo para o problema de detecção de	
	Sleep Spindles, utilizando o classificador RF sobre o conjunto de atributos	
	all. A árvore representa a expressão evoluída pelo algoritmo, destacando	
	os novos atributos construídos (F1,, F9) que contribuem para a tomada	00
F 10	de decisão do modelo	62
5.10		
	de detecção de alcoolismo nos cenários <i>Baseline</i> (à esquerda) e com a uti-	c o
E 11	lização do GP-FeSCo (à direita)	63
0.11	Desempenho médio do F1-Score para o classificador SVC aplicado ao con-	
	junto de dados Detecção de Alcoolismo, comparando os cenários <i>Baseline</i>	64
5 19	(à esquerda) e GP-FeSCo (à direita)	04
0.12	cada classificador no conjunto de dados Detecção e Alcoolismo, comparando	
	os cenários <i>Baseline</i> (à esquerda) e GP-FeSCo (à direita). Cada barra	
	representa a diferença para um grupo de pacientes (0 a 9)	64
5.13	Convergência da função <i>fitness</i> ao longo das gerações para diferentes clas-	01
0.10	sificadores no conjunto de dados Detecção de Alcoolismo. Cada gráfico	
	representa a evolução do F1-Score populacional médio para um classifica-	
	dor específico avaliado sobre os diferentes pacientes	65
5.14	Evolução do número de atributos criados (esquerda) e da altura média	
	dos indivíduos (direita) ao longo das gerações para diferentes grupos de	
	atributos no conjunto Detecção de Alcoolismo	66
5.15	Distribuição dos atributos mais selecionados pelo melhor indivíduo por clas-	
	sificador no conjunto de dados Detecção de Alcoolismo sobre o conjunto de	
	atributos complexity	67
5.16	Desempenho médio de $F1$ - $Score$ sobre o conjunto de teste para o problema	
	de Imagética Motora nos cenários Baseline (à esquerda) e com a utilização	
	do GP-FeSCo (à direita)	68

5.17	Desempenho médio do F1-Score por classificador aplicado ao conjunto de	
	dados Imagética Motora, comparando os cenários $Baseline$ (à esquerda) e	
	GP-FeSCo (à direita)	69
5.18	Diferença média do $F1$ - $Score$ entre os conjuntos de validação e teste para	
	cada classificador no conjunto de dados Imagética Motora, comparando	
	os cenários Baseline (à esquerda) e GP-FeSCo (à direita). Cada barra	
	representa a diferença para um grupo de pacientes (0 a 9)	69
5.19	Convergência da função <i>fitness</i> ao longo das gerações para diferentes classi-	
	ficadores no conjunto de dados Imagética Motora. Cada gráfico representa	
	a evolução do $\emph{F1-Score}$ populacional médio para um classificador específico	
	avaliado sobre os diferentes pacientes	71
5.20	Evolução do número de atributos criados (esquerda) e da altura média	
	dos indivíduos (direita) ao longo das gerações para diferentes grupos de	
	atributos no conjunto Imagética Motora	72
5.21	Distribuição dos atributos mais selecionados pelo melhor indivíduo por clas-	
	sificador no conjunto de dados Imagética Motora sobre os conjunto de atri-	
	butos catch22 e spectral	73
5.22	Melhor indivíduo gerado pelo GP-FeSCo para o problema de detecção de	
	Imagética Motora, utilizando o classificador SVC sobre o conjunto de atri-	
	butos catch22. A árvore representa a expressão evoluída pelo algoritmo,	
	destacando os novos atributos construídos (F1,, F11) que contribuem	
	para a tomada de decisão do modelo	74

# Lista de Tabelas

2.1	Impacto da corrupção de rótulos nos métodos de validação K-fold e LOSO-	
	CV no estudo de Kunjan et al. [1] $\ \ldots \ \ldots \ \ldots \ \ldots \ \ldots$	22
4.1	Características das Bases de Dados	42
4.2	Características dos Grupos de Atributos	42
4.3	Atributos Estatísticos Extraídos	43
4.4	Atributos Espectrais Extraídos	44
4.5	Atributos de Complexidade Extraídos	45
4.6	Atributos do Grupo catch22 [2]	46
4.7	Parâmetros configurados para a Programação Genética no GP-FeSCo	48
5.1	Comparação de Resultados da Literatura	76

# Lista de Abreviaturas e Siglas

AM Aprendizado de Máquina.

**AUC** Area Under the Curve.

AutoML Aprendizado de Máquina Automatizado.

BCI Brain Computer Interface.

catch22 CAnonical Time-series CHaracteristics.

**CNN** Rede Neural Convolucional - Convolutional Neural Network.

**DAG** Grafo Acíclico Direcionado.

**DCG** Grafo Direcionado Cíclico.

DCGs Grafos Direcionados Cíclicos.

**DL** Deep Learning.

**DT** Árvore de decisão - *Decision tree*.

**ECG** Eletrocardiograma.

**EEG** Eletroencefalograma.

EMG Eletromiografia.

**EOG** Eletrooculografia.

GP-FeSCo Genetic Programming-Based Feature Selection and Construction.

**HCTSA** Highly comparative time-series analysis.

**HVGE** Horizontal Visibility Graph Entropy.

**IBGSA** Improved Binary Gravitational Search Algorithm.

ICM Interface Cérebro-Máquina.

KNN K Vizinhos mais Próximos - K-Nearest Neighbors.

LOSO Leave-One-Subject-Out.

LOSO-CV Leave-One-Subject-Out Cross Validation.

MI Imagética Motora.

NB Naive Bayes.

NCA Análise de Componentes de Vizinhança - Neighbourhood Components Analysis.

PG Programação Genética.

PGA Programação Genética baseada em Árvore.

PGC Programação Genética Cartesiana.

PGG Programação Genética baseada em Gramática.

PGL Programação Genética Linear.

PGP Programação Genética baseada em Pilhas.

PSG Polissonografia.

**RF** Random Forest.

RMS Raiz do Valor Quadrático Médio - Root Mean Square.

sCV Subject-Wise Cross Validation.

STD Desvio padrão - Standard deviation.

SVC Classificador de Vetores de Suporte - Support Vector Classifier.

SVM Máquina de Vetores de Suporte - Support Vector Machine.

**TPOT** Tree-Based Pipeline Optimization Tool.

# Capítulo 1

# Introdução

A aprendizagem é um processo inerente à inteligência humana, que possibilita a aquisição, organização e representação de novos conhecimentos. Inspirado por esse processo, o Aprendizado de Máquina (AM) surge como um campo computacional que modela a aprendizagem de diversas maneiras, desenvolvendo programas capazes de identificar padrões e realizar previsões sobre dados inéditos [3, 4]. Esse campo tem ganhado crescente popularidade em setores como saúde, finanças e indústria, à medida que o armazenamento de dados se torna mais acessível, o que aumenta a demanda por soluções analíticas automatizadas [5, 6].

A capacidade do AM de lidar com dados complexos tem impulsionado sua adoção em aplicações críticas, como diagnósticos médicos e previsão de eventos financeiros [7, 8]. Entretanto, a alta dimensionalidade dos dados representa um obstáculo, já que o grande volume de variáveis pode causar sobreajuste (overfitting), altos custos computacionais e dificuldades na interpretação dos modelos [9, 10]. O conceito de overfitting refere-se ao fenômeno em que um modelo de AM aprende características específicas do conjunto de treinamento, mas apresenta desempenho inferior em conjuntos novos devido à falta de generalização. Para superar essas limitações, são frequentemente aplicadas técnicas de engenharia de atributos e redução de dimensionalidade, que simplificam os modelos sem comprometer seu desempenho preditivo [11, 12].

A engenharia de atributos é uma etapa crucial no processo de AM. Consiste em um conjunto de técnicas que transformam dados brutos em representações mais adequadas para a análise preditiva [13, 14]. Essa etapa é fundamental para mitigar problemas de dimensionalidade, aprimorando a capacidade do modelo de identificar padrões significativos nos dados por meio de tarefas como tratamento, seleção e construção de atributos [15, 16].

No tratamento de dados temporais, como os sinais de Eletroencefalograma (EEG), os desafios são intensificados pela complexidade dos sinais, que são ruidosos, não estacionários e apresentam dependência temporal e posicional entre os eletrodos [17]. Os sinais de

EEG, utilizados para monitorar a atividade cerebral e diagnosticar distúrbios neurológicos, como epilepsia e distúrbios do sono, demandam análises precisas e interpretáveis, frequentemente realizadas por especialistas humanos. Entretanto, esse processo pode ser moroso e sujeito a vieses, como o estado emocional e a experiência do analista, o que dificulta a reprodutibilidade dos diagnósticos e gera inconsistências entre as avaliações [18, 19].

Diante dos desafios apresentados, a automação de tarefas no contexto do AM emerge como uma estratégia promissora, especialmente para problemas que envolvem dados complexos e alta dimensionalidade, como os sinais de EEG. A construção de *pipelines* automatizados — que inclui etapas como seleção de atributos, transformação de dados e ajuste de modelos — torna-se cada vez mais essencial para reduzir o esforço manual e melhorar a reprodutibilidade dos experimentos. Por outro lado, encontrar a combinação ideal de algoritmos, transformações e hiperparâmetros continua sendo um desafio, devido ao vasto número de possibilidades envolvidas [20, 21].

Nesse cenário, a Programação Genética (PG) se destaca por sua capacidade de gerar soluções altamente customizáveis e com elementos que proporcionam interpretabilidade [22, 23]. A PG é uma técnica inspirada na evolução natural, que busca otimizar soluções por meio da combinação de operadores e da construção de novas expressões computacionais [24]. Essa flexibilidade possibilita a construção automática de atributos relevantes e a criação de representações compreensíveis, essenciais em aplicações clínicas que envolvem sinais fisiológicos, onde a interpretabilidade das decisões do modelo é crucial.

As variações específicas de cada paciente representam um desafio significativo na classificação de sinais de EEG, tornando essencial o uso de métodos de validação que garantam uma generalização eficaz para novos sujeitos ainda não observados. A validação cruzada por paciente - Leave-One-Subject-Out Cross Validation (LOSO-CV) - emerge como uma técnica robusta nesse contexto [1, 25]. Esse método utiliza iterativamente os dados de um único sujeito exclusivamente para teste, enquanto o modelo é treinado com os dados dos demais pacientes. O processo é repetido para cada sujeito no conjunto de dados, permitindo uma avaliação abrangente da capacidade do modelo em generalizar para diferentes indivíduos. Ao assegurar que os dados de teste consistam inteiramente de sujeitos não vistos anteriormente, o LOSO-CV fornece uma estimativa mais realista do desempenho do modelo, crucial para aplicações em classificações de EEG, onde a variabilidade entre os pacientes é elevada [26, 27].

Neste trabalho, propomos um framework baseado em PG para a seleção e construção de atributos aplicados à análise de dados de EEG denominado Genetic Programming-Based Feature Selection and Construction (GP-FeSCo). Utilizamos validação cruzada LOSO para evitar o vazamento de dados entre os conjuntos de treino e teste. A abor-

dagem visa aprimorar a precisão e a interpretabilidade dos modelos, representando um avanço significativo no diagnóstico e na caracterização de fenômenos neurológicos. Como contribuição adicional, realizamos uma caracterização detalhada dos conjuntos de dados de EEG em múltiplos grupos de atributos, analisando quais deles mais contribuem para cada problema específico e identificando os canais de EEG mais relevantes em cada contexto. Além disso, conduzimos uma revisão da literatura recente para fundamentar as decisões metodológicas e comparar os avanços propostos com o estado da arte.

#### 1.1 Justificativa

A análise de sinais de EEG é desafiadora devido à alta dimensionalidade, à natureza não estacionária e à dependência temporal. Essas características dificultam sua aplicação em tarefas de AM. Além disso, a maioria das soluções propostas na literatura para problemas semelhantes é desenvolvida de forma empírica e manual, o que pode introduzir vieses e limitar a generalização dos modelos.

Combinar técnicas de pré-processamento com algoritmos de AM resulta, geralmente, em desempenho superior em comparação a abordagens isoladas, pois aprimora a capacidade de aprendizado dos modelos. No entanto, identificar as melhores combinações de técnicas e hiperparâmetros é uma tarefa computacionalmente exigente.

Nesse contexto, a PG é uma meta-heurística eficiente que automatiza etapas críticas do desenvolvimento de modelos de AM. A PG possibilita a criação de soluções personalizadas e interpretáveis, reduzindo o viés humano e promovendo maior eficiência no processamento de dados complexos, como sinais de EEG.

Portanto, a proposta de um *framework* que integra técnicas avançadas de seleção e construção de atributos, juntamente com PG e validação cruzada por paciente (LOSO-CV) atende à necessidade de desenvolver modelos mais precisos, interpretáveis e generalizáveis para a análise de EEG.

### 1.2 Objetivos

O objetivo principal deste trabalho é estudar a automação da etapa de engenharia de atributos em problemas de classificação de sinais EEG utilizando um framework baseado em PG. O estudo incluirá uma avaliação adequada para problemas clínicos por meio de LOSO-CV, além de uma comparação detalhada com o estado da arte, com o uso de três bases de dados públicas de EEG.

Os objetivos específicos são:

- 1. Estudo comparativo em bases de dados públicas: Realizar uma análise com três bases de dados públicas de EEG, rotuladas por especialistas, para avaliar a eficácia do framework proposto.
- 2. Análise da qualidade dos atributos gerados: Examinar os atributos selecionados e construídos pela PG, avaliando sua relevância e contribuição para os modelos preditivos.
- 3. Interpretação da estrutura dos indivíduos gerados: Avaliar a estrutura dos indivíduos gerados pela PG, focando em sua interpretabilidade e eficiência em termos computacionais.
- 4. Identificação dos canais mais relevantes: Determinar os canais de EEG mais importantes para diferentes tarefas de classificação, otimizando a análise para cada base de dados.
- 5. Validação com o estado da arte: Comparar os resultados do *framework* com as abordagens existentes na literatura, utilizando métricas reconhecidas no campo.

Essa abordagem visa validar a eficácia do framework e aprofundar o entendimento sobre os atributos e canais relevantes para a classificação de sinais EEG, contribuindo para avanços na área.

#### 1.3 Hipóteses de Pesquisa

#### 1.3.1 Hipótese Principal

A automação da engenharia de atributos, por meio de PG, melhora o desempenho dos modelos de classificação de sinais EEG, garantindo precisão, interpretabilidade e generalização, conforme validado pelo método LOSO-CV.

#### 1.3.2 Hipóteses Secundárias

- Eficácia dos atributos gerados: Os atributos selecionados e construídos pela PG
  possuem alta relevância e contribuem significativamente para o aumento da precisão
  dos modelos preditivos, mesmo em bases de dados com características distintas.
- 2. Estrutura interpretável dos indivíduos gerados: A estrutura dos indivíduos gerados pela PG é interpretável, permitindo *insights* sobre os padrões extraídos dos sinais EEG e promovendo aplicações práticas, como diagnósticos médicos.

- 3. Importância de canais específicos de EEG: É possível identificar um subconjunto de canais de EEG mais relevantes para cada tarefa de classificação, permitindo a otimização do uso de dados e a redução de dimensionalidade sem comprometer o desempenho.
- 4. Generalização do framework proposto: O framework baseado em PG, validado por meio de LOSO-CV, apresenta desempenho competitivo ou superior ao estado da arte, demonstrando capacidade de generalização para novos sujeitos e diferentes bases de dados.

#### 1.4 Organização do Documento

Estruturado da seguinte maneira, o restante deste texto abrange o Capítulo 2, que revisa o estado da arte e fundamenta a pesquisa, apresentando os formalismos utilizados nos capítulos subsequentes. O Capítulo 3 propõe a pesquisa, enquanto o Capítulo 4 descreve a metodologia adotada. O Capítulo 5 relata e analisa os resultados dos experimentos. Por fim, o Capítulo 6 discute e conclui sobre os resultados obtidos.

# Capítulo 2

# Fundamentação Teórica

Neste capítulo, apresentamos os principais conceitos, desafios e avanços relacionados à classificação de sinais de EEG e ao uso de técnicas de AM para abordar os problemas associados. Na Seção 2.1, exploramos as características e complexidades dos sinais de EEG, destacando sua natureza não-estacionária, alta dimensionalidade e susceptibilidade a ruídos, bem como os desafios envolvidos na análise. Em seguida, na Seção 2.2, discutimos abordagens modernas com foco na PG, enfatizando a construção de atributos e a capacidade dessa técnica em lidar com a variabilidade dos sinais. Na Seção 2.3, apresentamos estratégias de validação cruzada, com destaque para o paradigma orientado a pacientes - LOSO -, essencial para avaliar a generalização dos modelos. Por fim, na Seção 2.4, abordamos aplicações específicas de AM em EEG, como Interface Cérebro-Máquina (ICM), análise de sono e diagnóstico de distúrbios neurológicos. Com essa estrutura, buscamos fornecer uma visão abrangente, desde os fundamentos dos sinais de EEG até as inovações mais recentes, criando uma base sólida para compreender a relevância e o potencial das técnicas descritas para futuras pesquisas e aplicações.

#### 2.1 Sinais EEG

Os sinais de EEG são amplamente utilizados em pesquisas médicas e neurológicas para estudar a atividade cerebral [28]. Capturados do couro cabeludo, refletem variações elétricas geradas pela atividade neuronal, fornecendo *insights* informações fundamentais sobre as funções cerebrais. Entretanto, a classificação desses sinais apresenta desafios significativos devido à complexidade dos dados [29]. Caracterizam-se por não serem estacionários, apresentarem alta dimensionalidade e serem suscetíveis a ruídos e artefatos provenientes de fontes externas ou de condições fisiológicas não diretamente relacionadas à atividade cerebral. Essas características ressaltam a necessidade de desenvolvimento de modelos

robustos e precisos de classificação para diversas aplicações, incluindo o diagnóstico de distúrbios neurológicos, *Brain Computer Interface* (BCI) e estudos do sono [30].

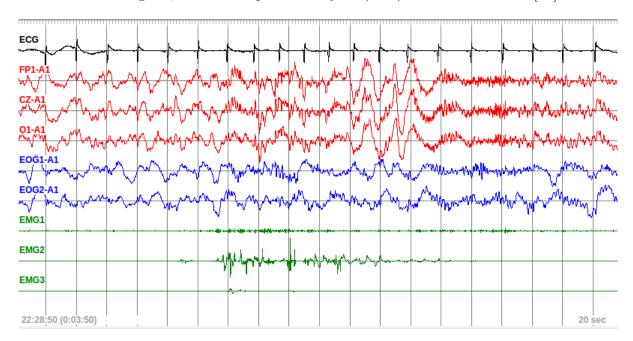


Figura 2.1: Trecho de exame de Polissonografia (PSG) ilustrando múltiplos sinais fisiológicos registrados simultaneamente. Os canais incluem: EEG (Fp1, Cz, O1), EOG, EMG e ECG.

A Figura 2.1 apresenta um exame típico de Polissonografia (PSG), uma técnica amplamente utilizada na análise do sono. Neste gráfico, múltiplos sinais fisiológicos essenciais são registrados simultaneamente para caracterizar os fenômenos do sono. Os canais de EEG — Fp1, Cz e O1 — estão posicionados de acordo com o sistema internacional 10-20 [31], conforme ilustrado na Figura 2.2.

Além dos sinais de EEG, a PSG registra sinais de Eletrooculografia (EOG), utilizados para detectar movimentos oculares rápidos, essenciais na identificação do sono REM. Os canais Eletromiografia (EMG) refletem a atividade muscular. Adicionalmente, existem canais de Eletrocardiograma (ECG) que monitoram a atividade cardíaca. Exames completos incluem também sensores de fluxo respiratório, esforço torácico e abdominal, saturação de oxigênio e posição corporal também são incluídos. A integração desses sinais possibilita uma análise abrangente do sono, permitindo a detecção de eventos como apneias, microdespertares, espasmos musculares e transições entre estágios. Nesse contexto, o estudo de biossinais é fundamental para o desenvolvimento de tecnologias diagnósticas mais precisas, interpretáveis e automatizadas, promovendo avanços significativos na medicina personalizada e na compreensão de fenômenos fisiológicos complexos [32].

A classificação de EEG envolve a extração de atributos relevantes dos dados brutos e a aplicação de algoritmos de AM para categorizar os sinais em classes predefinidas.

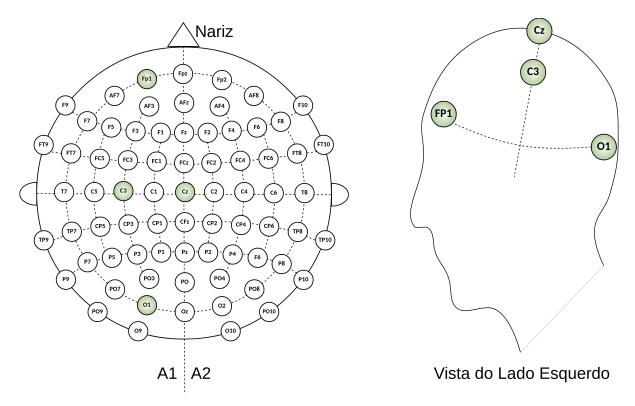


Figura 2.2: Posicionamento dos eletrodos Fp1, C3, Cz, O1 e O2 conforme o sistema internacional 10-20, com destaque para a visualização lateral esquerda. Esses canais são frequentemente utilizados em estudos de sono.

Os atributos extraídos podem pertencer a diferentes domínios — tempo, frequência ou tempo-frequência — cada um oferecendo perspectivas complementares sobre as dinâmicas neurais subjacentes. Os atributos comumente utilizados incluem medidas estatísticas (média, variância), métricas baseadas em frequência (densidade espectral de potência, coeficientes de wavelet) e dinâmicas não lineares (entropia, dimensões fractais). A escolha da representação dos atributos impacta diretamente o desempenho do modelo de classificação [33, 34, 35].

Apesar de seu potencial, a classificação de sinais de EEG enfrenta vários desafios. Um dos mais significativos é a alta suscetibilidade a ruídos e artefatos. Sinais EEG são frequentemente contaminados por atividades não neuronais, como movimentos musculares (EMG), piscar de olhos (EOG) e interferências elétricas externas. Esses ruídos podem sobrepor ou mascarar os sinais neurais de interesse, dificultando a identificação de padrões consistentes e impactando negativamente a acurácia dos modelos [36]. Técnicas de pré-processamento e detecção automáticas de artefatos, embora úteis, nem sempre são suficientes para eliminar completamente tais interferências [8, 37].

Outro aspecto crítico é a variabilidade inter e intra-sujeitos. Diferenças fisiológicas, estado emocional, fadiga, uso de medicamentos e a posição dos eletrodos entre as sessões

podem alterar significativamente a morfologia dos sinais. Essa variabilidade apresenta desafios substanciais para a generalização dos modelos de AM, que tendem a aprender padrões específicos dos sujeitos presentes no conjunto de treinamento [38, 39]. Em contextos clínicos, isso pode comprometer a aplicabilidade prática de soluções baseadas em AM, uma vez que o desempenho observado em dados conhecidos nem sempre se mantém em dados de novos pacientes.

Além disso, a natureza não estacionária dos sinais de EEG impõe limitações adicionais. A distribuição estatística dos sinais pode mudar ao longo do tempo, mesmo dentro da mesma sessão, tornando a extração de atributos eficazes ainda mais desafiadora. Conforme discutido por Rashid et al. [8], essa não estacionaridade pode degradar o desempenho dos modelos ao longo do tempo, exigindo abordagens dinâmicas ou adaptativas na modelagem.

Nesse cenário, a escolha de técnicas de validação torna-se fundamental. Métodos tradicionais de validação cruzada aleatória podem superestimar a performance dos modelos, pois permitem que dados de um mesmo sujeito apareçam tanto no treinamento quanto no teste. Como alternativa, a validação cruzada por sujeito (discutida na Seção 2.3) é amplamente recomendada para avaliar a capacidade de generalização dos modelos em tarefas envolvendo EEG [25]. Essa estratégia assegura que os dados de teste pertençam exclusivamente a indivíduos não observados durante o treinamento, refletindo de forma mais precisa os desafios do uso prático do modelo em novos pacientes.

Além disso, os conjuntos de dados frequentemente apresentam desbalanceamentos entre classes e tamanhos reduzidos, limitando a utilização de técnicas mais sofisticadas, como DL, que requerem grandes volumes de dados anotados para atingir desempenho competitivo [40]. Em muitos casos, isso resulta em abordagens híbridas, nas quais algoritmos tradicionais são combinados com técnicas avançadas de engenharia de atributos, como as baseadas em PG [41]. Essas técnicas são mais adequadas à escassez de dados e oferecem soluções com maior potencial para interpretabilidade.

#### 2.2 Programação Genética

Na análise de EEG, o AM tem se mostrado uma ferramenta poderosa para a compreensão dos complexos sinais cerebrais [42]. Técnicas tradicionais frequentemente enfrentam dificuldades na alta dimensionalidade e na natureza ruidosa dos padrões de EEG. Em contrapartida, algoritmos avançados de AM, como os baseados em meta-heurísticas e hiper-heurísticas, oferecem capacidades de reconhecimento automático de padrões, aprimorando a tomada de decisões clínicas. Esses métodos são particularmente eficazes na exploração de espaços de soluções complexos, frequentemente caracterizados por restrições intrínsecas e grande variabilidade.

Meta-heurísticas, em sua definição original, são métodos projetados para orquestrar a interação entre procedimentos de melhoria local e estratégias de nível superior, criando processos capazes de escapar de ótimos locais e realizar buscas otimizadas em espaços de solução [43]. Essas técnicas têm sido amplamente aplicadas para resolver problemas complexos em diversas disciplinas, como logística e bioinformática. Entretanto, ao serem utilizadas no contexto da análise de EEG, o desafio envolve não apenas a exploração do espaço de soluções, mas também a gestão da dinâmica e variabilidade inerentes aos dados. Nesse sentido, as hiper-heurísticas oferecem uma abordagem inovadora. Em vez de operar diretamente no espaço de soluções, as hiper-heurísticas exploram um espaço de heurísticas, identificando estratégias adaptativas e eficientes para resolver classes de problemas ou instâncias específicas [43, 44]. No domínio de sinais cerebrais, essas técnicas são essenciais para abordar a complexidade dos dados e garantir maior qualidade nos processos de análise.

A PG emerge como uma das abordagens mais promissoras no campo, sendo considerada uma hiper-heurística poderosa [45, 44]. Inspirada pelo processo de seleção natural, a PG é uma técnica de computação evolutiva amplamente adotada em áreas como classificação, regressão e construção de atributos [46]. No contexto da análise de EEG, a PG destaca-se pela flexibilidade e eficácia na descoberta de padrões. Por meio da evolução de programas de computador ou expressões matemáticas, a PG otimiza funções-objetivo específicas, permitindo a geração de soluções adaptadas às necessidades do problema [41]. Diferentemente dos algoritmos tradicionais de AM, que dependem de arquiteturas de modelo fixas, a PG explora um processo iterativo de variação, seleção e reprodução, facilitando a busca por soluções otimizadas em cenários complexos [47].

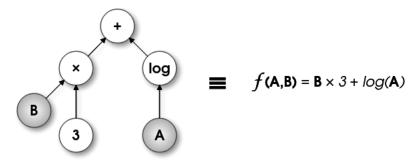


Figura 2.3: Exemplo de um programa gerado por PG baseada em árvore. Os nós internos correspondem a operadores matemáticos  $(+, \times, \log)$ , enquanto os nós terminais representam variáveis de entrada (A, B) e constantes.

Uma parte importante de PG é o conceito de representar soluções candidatas como estruturas de árvore - como ilustrado na Figura 2.3 -, onde cada nó representa uma opera-

ção (como funções matemáticas ou transformações de atributos) e as folhas representam variáveis de entrada ou valores constantes [22]. O algoritmo de PG começa gerando aleatoriamente uma população inicial dessas árvores, cada uma correspondendo a uma solução potencial para o problema de classificação. Ao longo de gerações sucessivas, essas árvores evoluem por meio de operações genéticas, como o cruzamento (em que subárvores de duas soluções parentais são trocadas) e a mutação (onde partes de uma árvore são alteradas aleatoriamente). A aptidão, ou *fitness*, de cada solução é avaliada com base em seu desempenho em uma tarefa predefinida, como maximizar a precisão da classificação ou minimizar o erro em uma tarefa de regressão. As soluções de melhor desempenho são selecionadas para transmitir suas características à próxima geração [23].

Uma das principais vantagens da PG na classificação de EEG é sua capacidade de construir e otimizar automaticamente atributos extraídos. Como os sinais de EEG são altamente complexos e não-lineares, a extração de atributos desempenha um papel fundamental na classificação bem-sucedida. A PG pode evoluir novas características ao combinar atributos existentes de maneiras inovadoras, potencialmente revelando relações ocultas nos dados que não são facilmente capturadas por métodos tradicionais de preparação de dados [48]. Essa capacidade de construção de atributos é especialmente benéfica em conjuntos de dados de alta dimensionalidade onde a seleção de características relevantes pode reduzir significativamente a carga computacional e melhorar o desempenho do modelo [49].

Além disso, a PG é interpretável em comparação com muitas técnicas de AM, como o DL. Os programas ou expressões evoluídas podem ser analisados e compreendidos pelos pesquisadores, fornecendo *insights* sobre os padrões subjacentes que orientam as decisões de classificação. Essa interpretabilidade é particularmente valiosa em aplicações médicas, onde compreender a lógica por trás das previsões de um modelo é crucial para obter a confiança de clínicos e garantir a confiabilidade das ferramentas de diagnóstico [50].

A flexibilidade da PG se estende à sua capacidade de incorporar conhecimentos específicos do domínio no processo evolutivo. Projetando operadores e conjuntos terminais apropriados que refletem a natureza dos sinais de EEG, os pesquisadores podem direcionar o processo de PG para soluções mais significativas [51, 52, 53]. Por exemplo, a incorporação de conhecimento sobre bandas de frequência, características no domínio do tempo ou dinâmicas não lineares no framework da PG pode aprimorar a capacidade do algoritmo de capturar aspectos relevantes dos dados [54].

Em resumo, a PG oferece uma estrutura poderosa e adaptável para a classificação de sinais de EEG. Suas forças estão na capacidade de evoluir modelos e representações de características de forma dinâmica, tornando-a especialmente adequada para lidar com a complexidade e variabilidade dos dados de EEG. Além disso, sua interpretabilidade e

personalização para o domínio a tornam uma escolha atrativa para aplicações em neurociência e diagnósticos clínicos, onde precisão e transparência são fundamentais [55, 41].

A estrutura das subseções a seguir foi elaborada para fornecer uma visão clara e sequencial dos principais aspectos relacionados à PG. Na subseção 2.2.1, apresentamos as variantes mais comuns de PG e suas aplicações. Em seguida, na subseção 2.2.2, discutimos como as soluções em PG são modeladas como estruturas hierárquicas em árvore, destacando o papel dos componentes primitivos e terminais. A subseção 2.2.3 aborda o mecanismo de avaliação das soluções, enfatizando sua importância para orientar o processo evolutivo. Na subseção 2.2.4 exploramos os operadores genéticos fundamentais para a criação de novas soluções, detalhando o funcionamento de cada etapa. Posteriormente, a subseção 2.2.5 apresenta uma visão geral do processo iterativo característico da PG, incluindo as configurações iniciais e o ciclo de execução. A subseção 2.2.6 analisa o problema do crescimento desnecessário das soluções e as estratégias para sua mitigação. Por fim, a subseção 2.2.7 discute estratégias de construção de atributos através de PG.

#### 2.2.1 Tipos de Programação Genética

A PG é uma técnica poderosa e flexível de computação evolutiva, composta por variantes adaptáveis a diferentes necessidades e desafios. Essas variantes incluem Programação Genética baseada em Árvore (PGA), Programação Genética Linear (PGL), Programação Genética Cartesiana (PGC), Programação Genética baseada em Gramática (PGG) e Programação Genética baseada em Pilhas (PGP). Cada abordagem reflete escolhas de design que influenciam a expressividade, a eficiência e a aplicabilidade em diversos domínios [56, 57, 44].

- Programação Genética baseada em Árvore: representa a forma mais clássica e amplamente utilizada, por meio de árvores de derivação. Essa representação facilita a recombinação de subárvores, permitindo a geração de soluções complexas e bem adaptadas. No entanto, um desafio comum é o inchaço das árvores, que resulta em programas excessivamente grandes e difíceis de interpretar [58]. Estratégias como a introdução de restrições estruturais são frequentemente empregadas para mitigar tais problemas [57].
- Programação Genética Linear: representa programas como sequências de instruções semelhantes a linguagens de máquina, operando em registros que armazenam valores intermediários. Essa abordagem proporciona maior controle sobre os recursos computacionais e pode ser mais eficiente em ambientes restritos. Contudo, seu design linear limita a expressividade em comparação a formas mais estruturadas,

como grafos ou árvores [59]. Estudos comparativos indicam que a PGL pode superar a PGA em tarefas com alta dependência sequencial, mas pode ser menos eficaz para problemas altamente hierárquicos [44].

- Programação Genética Cartesiana: utiliza grafos dirigidos acíclicos para representar soluções. Essa abordagem permite a reutilização de subestruturas, aumentando a eficiência em aplicações como a otimização de circuitos digitais. Além disso, a PGC pode incorporar restrições específicas, como o parâmetro de níveis para trás, que controla a conectividade dos nós no grafo [60]. Essa flexibilidade torna a metodologia adequada para problemas de controle automático e classificação complexa, como observado em benchmarks nas áreas médica e de regressão [57].
- Programação Genética baseada em Gramática: combina a força da PG com gramáticas formais, construindo soluções que respeitam restrições estruturais ou sintáticas específicas [61]. Essa variante é ideal para domínios onde a conformidade com regras predefinidas é essencial, como na geração de código ou na modelagem genética. No entanto, a necessária dependência de gramáticas bem projetadas pode limitar a generalidade e a flexibilidade da abordagem [57].
- Programação Genética baseada em Pilhas: simplifica a execução de programas ao empregar pilhas para gerenciar dados e operações. Essa abordagem reduz a complexidade estrutural e oferece suporte eficiente para diversos tipos de dados [62]. Entretanto, sua aplicação é mais restrita a problemas que exigem baixa interconexão entre os elementos do programa [44].

Avanços recentes exploraram representações mais flexíveis, como os Grafos Direcionados Cíclicos (DCGs) [56]. Diferentemente dos DAGs utilizados na PG cartesiana, os DCGs permitem a formação de ciclos, aumentando a capacidade de modelar *feedbacks* e interdependências complexas. Estudos comparativos demonstram que essa liberdade adicional pode melhorar o desempenho em tarefas de classificação, mas também dificultar a convergência em problemas de regressão, devido à maior complexidade estrutural [56].

Cada tipo de PG apresenta vantagens e desvantagens que devem ser consideradas de acordo com os requisitos do problema. A escolha da variante ideal depende de uma análise cuidadosa de fatores como a natureza do espaço de busca, a necessidade de precisão ou eficiência computacional e as restrições do domínio de aplicação. Ao explorar essas variantes e suas interações, a PG continua a expandir suas fronteiras como uma ferramenta poderosa e adaptável no campo de AM e na computação evolutiva [56, 57, 44].

Neste trabalho, adotamos a PGA como base do GP-FeSCo devido à sua capacidade de gerar soluções complexas, bem adaptadas e interpretáveis. Essa escolha está alinhada ao objetivo de construir atributos com estruturas mais compreensíveis, favorecendo a

interpretabilidade das soluções. Nas seções a seguir, aprofundaremos a representação e as particularidades dessa abordagem.

#### 2.2.2 Representação

Na PGA, os indivíduos são estruturas hierárquicas em forma de árvore, compostas por funções primitivas e valores terminais definidos pelo usuário para um domínio específico. A seleção desses parâmetros é condicionada ao contexto da aplicação e aos tipos de dados envolvidos.

O conjunto de funções primitivas pode incluir, por exemplo, operações aritméticas, funções matemáticas, operações lógicas e condicionais, além de funções definidas pelo usuário. O conjunto de terminais pode ser formado por constantes numéricas e entradas específicas do problema [23].

Por exemplo, a Figura 2.3 ilustra um indivíduo no qual o programa representado possui como terminais os atributos A, B e a constante de valor 3. Adicionalmente, são empregadas as funções primitivas de adição +, multiplicação  $\times$  e logaritmo natural log. Por fim, pode ser reescrito com a expressão  $B \times 3 + \log(A)$ . Este é um exemplo puramente numérico. No entanto, a técnica é flexível para qualquer tipo de dado de entrada, desde que as funções primitivas e as entradas e terminais sejam compatíveis.

A PG é tradicionalmente iniciada com uma população formada por indivíduos gerados aleatoriamente, criados por combinações de terminais e funções primitivas. Essa criação da população inicial aleatória representa uma busca cega no espaço de busca do problema [23, 63], onde cada indivíduo tem seu valor da função *fitness* calculado. Embora os indivíduos mais aptos nesse estágio não resolvam o problema de maneira satisfatória, podem apresentar trechos relevantes para uma boa solução.

#### 2.2.3 Função de Aptidão (fitness)

A cada geração da PG, uma nova população é criada com base na tentativa de reproduzir o princípio da seleção natural, onde os indivíduos mais aptos sobrevivem e possuem mais chances de se reproduzir. Isso se resume a uma avaliação da qualidade do indivíduo, calculada através da função de aptidão chamada *fitness* e sua definição varia para cada problema, sendo um passo crucial na preparação do algoritmo [22]. Em um problema de regressão simbólica, por exemplo, uma função *fitness* adequada é o cálculo do erro médio quadrático entre os dados reais e a representação obtida pelo indivíduo. Em aplicações de problemas de classificação normalmente são utilizadas funções de custo de otimização ou métricas de avaliação como AUC, *F1-score* ou acurácia balanceada, pois refletem a qualidade do *pipeline* obtido.

#### 2.2.4 Seleção, recombinação e mutação

A função fitness permite avaliar e comparar indivíduos entre si, ou seja, identificar quais programas resolvem o problema analisado de maneira mais satisfatória. Então, para a manutenção de partes relevantes dos melhores programas gerados, o operador de **seleção** utiliza valores de fitness obtidos para cada indivíduo e assim escolher quais deles serão utilizados na geração de nova população de programas-filhos. Esse mecanismo pode ser implementado, por exemplo, através do sorteio de indivíduos da população com probabilidade de escolha proporcional ao seus valores de fitness, conhecido como sorteio por roleta [22]. Uma outra maneira é através do método de Torneio onde um subconjunto de indivíduos é selecionado aleatoriamente da população atual. O tamanho desse subconjunto é definido pelo parâmetro K, chamado de tamanho do torneio. Dentre os indivíduos escolhidos, aquele que possui o melhor valor fitness é selecionado como um dos pais para a próxima geração [64]. O tamanho do torneio influencia a pressão seletiva; valores maiores de K favorecem indivíduos com fitness elevado, reduzindo a diversidade genética, enquanto valores menores promovem maior diversidade.

A partir da seleção, novos indivíduos são gerados com informações obtidas por indivíduos selecionados da geração anterior. Isso ocorre devido ao mecanismo de recombinação, o *crossover* que consiste em formar programas-filhos a partir de dois programas-pais. O novo indivíduo é uma combinação de trechos dos indivíduos-pais, podendo receber partes relevantes para o problema e pode resultar em uma nova solução com valor de *fitness* mais alto. Normalmente, o *crossover* tem uma probabilidade de ocorrência associada, o que aumenta a variabilidade do algoritmo. Na Figura 2.4, foi escolhido um nó ao acaso para cada indivíduo-pai. As sub-árvores que possuem esse nó como raiz são trocadas entre os indivíduos, gerando dois novos (chamado *One Point Crossover*).

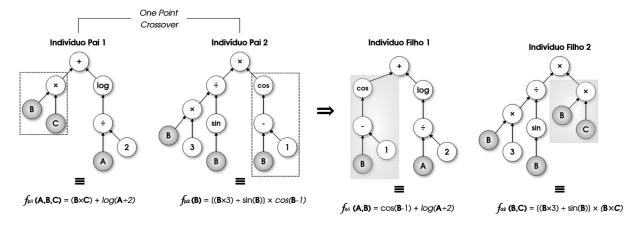


Figura 2.4: Exemplo de crossover em PG). Duas árvores representando indivíduos pais trocam subárvores em um ponto selecionado aleatoriamente (*One Point Crossover*), gerando dois novos indivíduos filhos.

Novos indivíduos também podem ser gerados através do mecanismo de **mutação** causando pequenas perturbações em um indivíduo. A mutação contribui com o aumento da diversidade na população, evitando que as soluções encontradas permaneçam em ótimos locais. Também pode aumentar a precisão de um bom indivíduo, já que, em geral, não provocam alterações drásticas na estrutura do programa. Na Figura 2.5, um indivíduo tem um nó selecionado ao acaso e seu operador substituído por outro. Um outro exemplo é substituir a sub-árvore que tem o nó selecionado como raiz por outra gerada aleatoriamente.

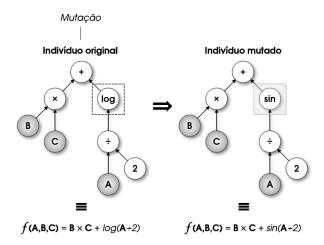


Figura 2.5: Exemplo de mutação em PG. Neste caso, um nó da árvore representando a função original foi modificado, substituindo a operação log por sin.

#### 2.2.5 Fluxo tradicional da Programação Genética

Em PG, existem cinco configurações preparatórias principais selecionadas pelo programador [64]: definir conjunto de terminais (variáveis de entrada e constantes); conjunto de operadores; função de *fitness*; parâmetros genéticos (*crossover*, mutação, seleção) e critério de parada.

Então, com os parâmetros definidos, o fluxo de execução é iniciado e possui os seguintes passos (Figura 2.6):

- 1. Inicialização de uma população aleatória
- 2. Cálculo da fitness dos indivíduos
- 3. Verificação do critério de parada comumente, o algoritmo é limitado a um número de gerações, tempo de execução ou quando um valor definido de *fitness* é atingido.
- 4. Aplicação de *crossover* e mutação.

- 5. Nova população é criada: nessa etapa, toda a população anterior pode ser substituída pela nova. Uma abordagem comum é utilizar elitismo, onde o melhor indivíduo da geração anterior é passado para a próxima, garantindo que as próximas populações tenham sempre um programa melhor ou igual à anterior.
- 6. A execução volta ao passo 2.



Figura 2.6: Fluxo tradicional de execução da PG

#### 2.2.6 Inchaço

Inchaço (bloat) é um fenômeno que ocorre durante a execução da PG. Com o passar das gerações, programas mais complexos tendem a surgir sem melhorias em seu valor fitness [58]. Isso se deve à presença de subárvores irrelevantes para o problema ou a trechos que podem ser simplificados. Esse crescimento exige mais processamento para avaliar os programas, o que diminui o desempenho do algoritmo e aumenta a ocorrência de overfitting [65, 66].

Na Figura 2.7, o programa representado inicialmente requer 6 operações para ser concluído. Entretanto, simplificando as operações, o mesmo resultado pode ser alcançado com apenas uma única instrução.

Existem várias maneiras de se evitar o bloat [67]. Porém, a mais simples é estabelecer uma profundidade máxima para as árvores geradas. Caso algum dos indivíduos exceda o limite estabelecido após uma operação de crossover ou mutação, será substituído por um de seus pais. A desvantagem dessa estratégia é a permanência de programas na população que estão próximos de ultrapassar o limite de profundidade.

#### 2.2.7 Programação Genética para Construção de Atributos

A construção de atributos é uma etapa crucial na classificação de sinais de EEG, pois os dados brutos de EEG frequentemente contêm informações de alta dimensionalidade, complexas e ruidosas. Essas informações devem ser transformadas em representações significativas para o treinamento eficaz de modelos. Métodos tradicionais de extração de

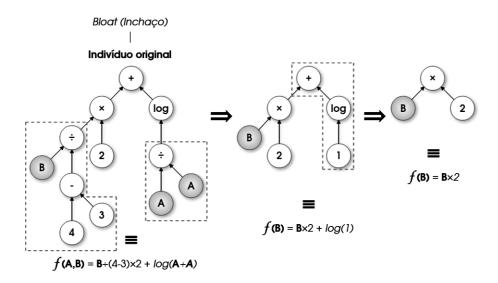


Figura 2.7: Exemplo do fenômeno de *bloat* (inchaço) em PG. O indivíduo original contém expressões redundantes e desnecessárias que não contribuem para a funcionalidade do programa.

atributos dependem de características projetadas manualmente, com base em análises no domínio do tempo, domínio da frequência ou tempo-frequência, como a densidade espectral de potência (PSD) ou transformadas wavelet [68, 18]. No entanto, essas abordagens podem falhar em capturar relações intrincadas e não-lineares nos sinais de EEG, especialmente quando os sinais são influenciados por diversos fatores cognitivos e fisiológicos. A PG oferece uma solução poderosa para esse problema, pois permite construir automaticamente novos atributos a partir dos dados. Isso possibilita a descoberta de conjuntos de atributos mais relevantes e informativos, aprimorando o desempenho da classificação [41].

A PG para construção de atributos evolui expressões matemáticas ou funções simbólicas que combinam os atributos originais de maneiras inovadoras. Esse processo gera novos atributos que podem não ser intuitivamente evidentes, mas são altamente eficazes em capturar a estrutura subjacente dos dados. A PG começa com uma população inicial de expressões geradas aleatoriamente, representando diferentes combinações dos atributos originais [69, 70]. Por meio de operações evolutivas como cruzamento, mutação e seleção, essas expressões são refinadas iterativamente ao longo de várias gerações para produzir soluções mais otimizadas. A qualidade de cada construção de atributos é avaliada com base em sua capacidade de melhorar a precisão da classificação, tornando a PG uma abordagem orientada por dados para a engenharia de atributos [41, 53].

Uma das principais vantagens da PG na construção de atributos é sua capacidade de descobrir interações não-lineares entre atributos que métodos lineares tradicionais podem ignorar [71]. Os sinais de EEG frequentemente exibem dinâmicas temporais e espaciais complexas, e as relações entre diferentes bandas de frequência ou canais de sinal podem ser

altamente não-lineares. Ao construir atributos que capturam essas não-linearidades, a PG permite que o classificador distinga melhor entre diferentes estados cerebrais ou condições cognitivas, levando a um desempenho aprimorado em tarefas como detecção de convulsões, classificação de estágios do sono e reconhecimento de imaginação motora [72, 73].

Outro benefício importante da construção de atributos baseada em PG é sua capacidade de reduzir a dimensionalidade dos dados. Conjuntos de dados de alta dimensão, como aqueles gerados a partir de gravações de EEG com múltiplos canais e pontos de tempo, podem levar ao sobreajuste e aumentar a complexidade computacional [74]. A PG pode resolver isso ao selecionar e combinar apenas os atributos mais relevantes, reduzindo, assim, o conjunto geral de atributos, preservando as informações essenciais necessárias para a classificação. Isso não apenas melhora a generalização do modelo, mas também reduz a carga computacional, tornando a abordagem adequada para aplicações em tempo real onde a eficiência é crucial [75, 48].

Além disso, a PG oferece um alto grau de interpretabilidade, muitas vezes ausente em outros métodos de construção de atributos, particularmente aqueles baseados em aprendizado profundo. As expressões simbólicas geradas pela PG podem ser analisadas e compreendidas por especialistas da área, fornecendo *insights* sobre as relações entre diferentes atributos e sua relevância para a tarefa de classificação [76, 77]. Essa interpretabilidade é especialmente importante em aplicações médicas e clínicas, onde compreender como o modelo chega às suas decisões é essencial para conquistar a confiança dos profissionais e garantir a confiabilidade das ferramentas diagnósticas [78, 79].

Em resumo, a PG para construção de atributos oferece uma abordagem flexível, poderosa e interpretável para aprimorar a classificação de sinais de EEG. Ao descobrir interações não-lineares entre atributos e reduzir a dimensionalidade dos dados, a PG melhora o desempenho dos modelos de classificação, mantendo a eficiência computacional. Sua capacidade de gerar atributos significativos e interpretáveis a torna uma ferramenta valiosa em domínios onde a complexidade dos dados e a necessidade de transparência são cruciais [54, 41].

### 2.3 Validação Cruzada Orientada a Paciente

Como discutido anteriormente, a classificação de sinais de EEG apresenta desafios distintos devido à significativa variabilidade da atividade cerebral entre indivíduos e em diferentes sessões. Essa variabilidade torna essencial a adoção de estratégias de validação que assegurem a capacidade do modelo de generalizar de maneira eficaz para novos pacientes. Uma das técnicas mais robustas para esse propósito é a Validação Cruzada a

Paciente - Subject-Wise Cross Validation (sCV) [26, 27], também conhecida como Leave-One-Subject-Out Cross Validation (LOSO-CV) [1, 25], que aborda esses desafios ao fornecer uma avaliação mais precisa do desempenho do modelo.

Na classificação de EEG, três paradigmas principais de validação cruzada são frequentemente empregados [80, 81, 82]:

- 1. Paradigma de Paciente Comum (ou validação cruzada por registro): O conjunto de dados é dividido em conjuntos de treinamento e teste sem considerar os pacientes individuais. Essa abordagem pode levar a vazamento de dados, pois segmentos do mesmo sujeito podem aparecer em ambos os conjuntos, inflando as métricas de desempenho e superestimando a eficácia do modelo.
- 2. O segundo paradigma é o **Paradigma Específico de Paciente**, onde o modelo é treinado e testado usando dados de um único sujeito. Embora esse método possa produzir alta precisão para o sujeito específico, ele não generaliza bem para outros indivíduos, pois o modelo não encontra a variabilidade presente nos dados de diferentes pacientes.
- 3. O paradigma mais eficaz para a classificação de EEG é o Paradigma Leave-One-Subject-Out (LOSO) [1] ou Subject-Wise Cross Validation (sCV) (neste trabalho, daremos preferência ao termo LOSO). Nesse método, os dados de um paciente são usados exclusivamente para teste, enquanto os dados de todos os outros são usados para treinamento. Esse processo é repetido para cada paciente no conjunto de dados, com o desempenho do modelo sendo a média de todas as iterações. Ao garantir que os dados de teste consistam inteiramente de sujeitos não vistos anteriormente, o LOSO-CV oferece uma estimativa mais realista da capacidade de generalização do modelo entre diferentes indivíduos, o que é crítico para aplicações de EEG [25].

A Figura 2.8 ilustra a diferença entre dois paradigmas de validação amplamente utilizados na classificação de sinais fisiológicos, como EEG: a Validação Cruzada K-Fold (Paradigma de Paciente Comum) e a Validação Cruzada LOSO. No K-Fold, o conjunto de dados é dividido aleatoriamente em subconjuntos, o que pode resultar na presença de dados do mesmo paciente tanto no conjunto de treino quanto no de teste. Essa sobreposição pode levar a uma estimativa otimista do desempenho do modelo, especialmente em domínios com alta variabilidade entre sujeitos, como em sinais de EEG. Em contraste, a abordagem LOSO utiliza todos os dados de um paciente ou grupo de pacientes específico exclusivamente como conjunto de teste em cada iteração, enquanto os dados dos demais pacientes compõem o conjunto de treino.

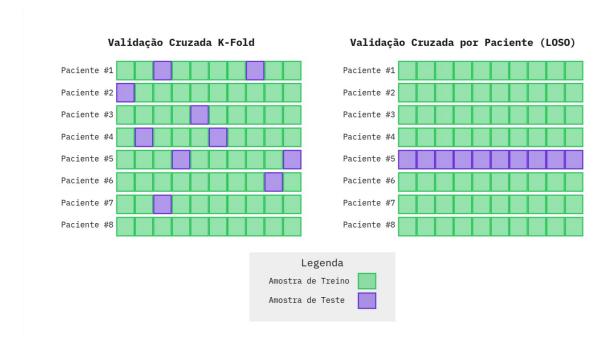


Figura 2.8: Comparação entre os esquemas de Validação Cruzada K-Fold (à esquerda) e Validação Cruzada por Paciente (LOSO, à direita). Cada linha representa os dados de um paciente, e cada bloco representa um subconjunto de amostras ao longo das iterações de validação. As amostras de treino estão em verde e as de teste em roxo.

Vários estudos ressaltam a importância do LOSO-CV devido à alta variabilidade nos sinais de EEG entre diferentes indivíduos. Por exemplo, Saeb et al. [83] compararam métodos de validação cruzada por paciente e por registro, mostrando que as abordagens por registro tendem a superestimar o desempenho do modelo, resultando em taxas de precisão excessivamente otimistas. Em contraste, a validação por sujeito oferece uma estimativa mais realista da capacidade de generalização do modelo para novos indivíduos, o que é essencial para a aplicação clínica.

O estudo de Kunjan et al. [1] demonstra o impacto da validação LOSO ao comparar sua eficácia com a validação tradicional K-fold em um experimento de classificação EEG para diagnóstico de esquizofrenia. Os autores realizaram diversas análises que evidenciam como a abordagem K-fold pode levar a resultados enganosos e superestimados, devido à falta de separação adequada entre os sujeitos nos conjuntos de treino e teste.

Os autores analisaram dados de EEG coletados de indivíduos com esquizofrenia e de um grupo de controle. A validação do modelo foi realizada por duas abordagens principais:

• K-Fold Cross-Validation: O modelo foi treinado e testado utilizando 10-fold cross-validation, permitindo que os dados de um mesmo paciente estivessem tanto no treinamento quanto no teste.

• LOSO-CV: Cada iteração removeu completamente um sujeito do treinamento e usou seus dados exclusivamente para teste, garantindo que o modelo fosse avaliado em um paciente totalmente novo a cada rodada.

Os resultados demonstraram uma diferença significativa entre as abordagens. Com K-fold, o modelo obteve uma acurácia de 98%, sugerindo um desempenho aparentemente excelente. Por outro lado, com LOSO, a acurácia reduziu para 70%, refletindo um valor mais realista para a classificação da doença. Essa redução de desempenho ao adotar LOSO indica que a validação K-fold permitiu que o modelo memorizasse características individuais dos sujeitos, em vez de aprender padrões generalizáveis da doença. Assim, ao testar em pacientes não vistos anteriormente, o modelo apresentou um desempenho mais condizente com a realidade.

Para fortalecer o argumento, os autores realizaram um experimento adicional em que corromperam os rótulos dos dados, tanto no nível do paciente quanto no das amostras. No nível do paciente, os rótulos de um indivíduo foram trocados aleatoriamente. Por exemplo, um sujeito originalmente rotulado como *controle* poderia ter todos os seus rótulos alterados para *esquizofrenia*, e vice-versa. Assim, todas as amostras de um mesmo sujeito ficaram incorretamente rotuladas, mas mantiveram a estrutura original dos sinais.

A nível de amostra, os rótulos foram randomizados independente do sujeito. Dentro de um mesmo paciente, algumas amostras mantiveram o rótulo correto, enquanto outras foram trocadas aleatoriamente. Esse método comprometeu qualquer relação entre os sinais EEG e a condição neurológica real dos pacientes. A acurácia da classificação pode ser vista na Tabela 2.1

Tabela 2.1: Impacto da corrupção de rótulos nos métodos de validação K-fold e LOSO-CV no estudo de Kunjan et al. [1]

Método de Validação	Estado dos Rótulos	Acurácia
LOSO	Dados originais	$70\% \pm 36$
K-fold	Dados originais	$98\% \pm 03$
LOSO	Corrupção por paciente	$30\% \pm 36$
K-fold	Corrupção por paciente	$98\% \pm 03$
K-fold	Corrupção por amostra	$37\% \pm 10$

Podemos interpretar os resultados da seguinte maneira:

Quando os rótulos foram corrompidos por paciente, a validação K-fold ainda apresentou 98% de acurácia, sugerindo que o modelo estava aprendendo características dos sujeitos e não da doença.

- Quando a mesma estratégia foi aplicada com LOSO, a acurácia despencou para 30%, demonstrando que o modelo já não conseguia se apoiar em informações espúrias para manter sua performance.
- Por fim, quando os rótulos foram completamente aleatorizados por amostra, a acurácia caiu para 37%, confirmando que a avaliação feita por *K-fold* era artificialmente inflada.

Esses experimentos demonstram que a validação K-fold pode ocultar a verdadeira capacidade do modelo, permitindo que ele memorize informações espúrias dos indivíduos. Por outro lado, a validação LOSO força o modelo a aprender padrões genuínos relacionados à doença, pois não tem acesso aos dados do mesmo sujeito durante o treinamento e o teste.

Essa evidência reforça que qualquer estudo que utilize EEG para diagnóstico ou classificação deve obrigatoriamente empregar o LOSO-CV, a fim de assegurar que os modelos consigam realmente generalizar para novos pacientes, em vez de apenas memorizar características individuais.

# 2.4 Outras aplicações de Aprendizado de Máquina para problemas de EEG

Sinais de EEG são amplamente utilizados na medicina e na neurociência para investigar a atividade cerebral [84, 85]. Capturados de forma não invasiva por eletrodos posicionados no couro cabeludo, esses sinais refletem flutuações elétricas decorrentes da atividade neuronal. A análise e classificação desses sinais são essenciais para diversas aplicações, como diagnóstico de epilepsia, sistemas de ICM, estudos do sono e decodificação de comportamentos cerebrais [86].

A classificação de EEG enfrenta desafios significativos devido à alta dimensionalidade dos dados, à variabilidade entre indivíduos e à suscetibilidade a ruídos e artefatos [87]. Tradicionalmente, a interpretação desses sinais requer a expertise de profissionais especializados, tornando-se um processo demorado e suscetível a erros. Assim, há uma demanda crescente por métodos automatizados que extraiam informações relevantes e realizem classificações precisas [88].

O AM tem se mostrado fundamental nesse contexto, fornecendo ferramentas para a análise eficiente e automatizada de EEG [89, 90]. As técnicas de AM variam desde algoritmos tradicionais de classificação até abordagens mais avançadas, como DL, PG e Aprendizado de Máquina Automatizado (AutoML). Essas técnicas têm sido aplicadas com sucesso em diversas áreas:

- Diagnóstico Clínico: AM auxilia na detecção e classificação de diferentes tipos de crises epilépticas, melhorando o diagnóstico e tratamento [86].
- Interface Cérebro-Máquina: Sistemas de ICM interpretam a atividade cerebral para controlar dispositivos externos, permitindo interação sem movimento físico [91]. Avanços em AM têm aprimorado a precisão e velocidade dessas interfaces [89].
- Análise do Sono: Classificação automatizada de estágios do sono e detecção de distúrbios têm se beneficiado de técnicas de AM. Por exemplo, Mei et al. [92] utilizaram o framework TPOT, baseado em PG, para identificar fenômenos em EEG do sono, demonstrando a eficácia de métodos automatizados na otimização de pipelines de classificação.
- Prognóstico Pós-Parada Cardíaca: Yang et al. [93] analisaram EEG de pacientes que sofreram parada cardíaca, aplicando técnicas de AM para prognóstico neurológico, evidenciando o potencial dessas abordagens em contextos críticos de saúde.

Para lidar com a complexidade dos sinais de EEG, é fundamental extrair atributos significativos dos dados brutos. Esses atributos podem ser categorizados em quatro grupos principais: estatísticos, espectrais, tempo-frequência e não lineares [18]. A identificação dos atributos mais relevantes para a classificação, no entanto, apresenta desafios, especialmente devido ao elevado número de canais utilizados, que pode chegar a 256, e à possibilidade de redundância nos dados [94].

Abordagens avançadas de AM, como a PG, são utilizadas para automatizar a seleção e construção de atributos, aprimorando o desempenho dos classificadores [55]. Além disso, ferramentas de AutoML também são métodos aplicáveis que facilitam a criação de pipelines eficientes, automatizando a seleção de modelos e a otimização de hiperparâmetros [93].

Em resumo, a aplicação de técnicas de AM na classificação de sinais de EEG é fundamental para enfrentar os desafios inerentes a esses dados. O desenvolvimento contínuo de métodos que abordem a alta dimensionalidade, variabilidade e ruído dos sinais é crucial para avanços em aplicações clínicas e pesquisas neurocientíficas.

# 2.5 Conjuntos de Dados de EEG

O estudo de conjuntos de dados de EEG abrange diversas aplicações, incluindo análise de Imagética Motora (MI), detecção de fusos do sono e identificação de alcoolismo. Esses conjuntos evidenciam a versatilidade das aplicações de EEG em diferentes áreas. Cada conjunto, com suas configurações e focos específicos, fornece a base essencial para a utilização de metodologias de AM em tarefas de classificação. Por exemplo, o conjunto

de dados *Physionet EEG Motor Movement/Imagery* é fundamental para o avanço da pesquisa em ICM e na classificação de MI. De forma semelhante, o conjunto DREAMS oferece informações cruciais sobre eventos de fusos do sono, contribuindo para o desenvolvimento de algoritmos automatizados de detecção do sono. Por outro lado, o conjunto de dados *UCI Alcoholism Detection* auxilia na identificação de marcadores neurofisiológicos associados ao alcoolismo. Através de extração criteriosa de atributos, pré-processamento direcionado e uso de técnicas avançadas de AM, esses conjuntos tornam-se indispensáveis para o avanço das pesquisas baseadas em EEG, promovendo aplicações inovadoras em monitoramento de saúde, reabilitação e diagnósticos.

Esses três conjuntos de dados de EEG foram selecionados como objetos de estudo para a validação do GP-FeSCo. Eles serão definidos e discutidos nas subseções a seguir.

### 2.5.1 Imagética Motora

O conjunto de dados *Physionet EEG Motor Movement/Imagery* [95, 96] é amplamente utilizado na pesquisa de ICM, principalmente para a análise de MI e movimentos motores. Este recurso público contém gravações de EEG de 109 sujeitos realizando uma série de tarefas que incluem movimentos motores reais e imaginados. Os dados de EEG são coletados através de 64 canais a uma taxa de amostragem de 160 Hz, registrando a atividade cerebral em várias localizações do couro cabeludo durante tarefas relacionadas a movimentos. Cada paciente participa de diversos ensaios em seis condições diferentes: duas tarefas de base com os olhos abertos e fechados, duas tarefas envolvendo movimentos reais de punho e duas tarefas de MI, nas quais os sujeitos imaginam mover um ou ambos os punhos e os pés. Devido ao número extenso de pacientes e às diversas condições de tarefa, o conjunto *Physionet* fornece uma base sólida para o desenvolvimento e teste de modelos de classificação baseados em MI.

#### Resultados da Literatura

Em estudos recentes, diversas arquiteturas de AM e redes neurais foram propostas para classificar dados de EEG, utilizando o conjunto *Physionet* para estudar a detecção de MI em aplicações de BCI. Por exemplo, Wang et al. [97] propuseram um modelo baseado em EEGNet para classificar tarefas de MI usando os dados do *Physionet*. Este modelo, uma CNN, demonstrou desempenho competitivo em relação aos métodos de ponta, alcançando uma acurácia de classificação de 82,43% em tarefas de MI binárias, utilizando técnicas como redução temporal e seleção de canais para otimizar o uso de memória sem perda significativa de precisão. Essa abordagem inovadora suporta aplicações em dispositivos

portáteis e em tempo real, demonstrando como o EEGNet pode se adaptar a sistemas embarcados com recursos computacionais limitados.

Outro estudo, realizado por Tacsar et al. [98], destacou a importância da extração e seleção de atributos para melhorar a eficiência e precisão da classificação de MI. A partir do conjunto *Physionet*, foram extraídos 30 atributos estatísticos por canal, reduzidos posteriormente usando a Análise de Componentes de Vizinhança - *Neighbourhood Components Analysis* (NCA). Após a seleção de atributos, três classificadores foram comparados—NCA, NB e SVM—com o SVM alcançando a maior acurácia de classificação de 99,51%. Esse trabalho ressaltou a relevância de técnicas de seleção de atributos na otimização do desempenho de classificadores, especialmente ao lidar com dados de alta dimensionalidade.

Ambos os estudos demonstram a importância do pré-processamento de dados e da engenharia de atributos para lidar com a variabilidade inerente dos sinais de EEG. O conjunto *Physionet* apresenta desafios devido à variabilidade inter e intra-sujeito, ruídos fisiológicos e a complexidade das tarefas, exigindo técnicas sofisticadas de processamento. Métodos como filtragem, redução temporal e seleção de canais são essenciais para desenvolver modelos que generalizem bem em diferentes sujeitos e condições, demonstrando melhorias significativas na acurácia.

### 2.5.2 Fusos do Sono - Sleep Spindles

O conjunto DREAMS [99, 87], desenvolvido pelos laboratórios da Universidade de MONS-TCTS e Université Libre de Bruxelles, é um recurso valioso para a pesquisa do sono, especialmente na detecção de fusos. Este conjunto inclui trechos de EEG de 30 minutos de oito participantes, com idades entre 31 e 53 anos, capturados nos canais C3-A1 e CZ-A1, a taxas de amostragem de 50 Hz, 100 Hz e 200 Hz. Seu principal objetivo é apoiar o desenvolvimento e teste de algoritmos automatizados para a detecção de fusos do sono, fornecendo dados anotados por especialistas conforme os critérios de Rechtschaffen e Kales [100]. Os fusos são caracterizados como oscilações na faixa de 11 a 16 Hz, com duração de 0,5 a 3 segundos.

### Resultados da Literatura

Nos últimos anos, a detecção dos fusos do sono avançou significativamente com a aplicação de metodologias de AM e DL, utilizando dados anotados do conjunto DREAMS. Lachner-Piza et al. [101] introduziram o detector MUSSDET, que emprega uma SVM para classificar fusos com base em atributos como a potência espectral. Essa abordagem

demonstrou adaptabilidade a gravações heterogêneas, alcançando sensibilidade de 53% e especificidade de 96%.

Chen et al. [102] propuseram uma abordagem que combina atributos extraídos com métricas de entropia, permitindo a captura de características complexas dos fusos. Este modelo obteve um *F1-score* de 0,66 no conjunto DREAMS, evidenciando a eficácia da integração de DL com medidas estatísticas.

### 2.5.3 Detecção de Alcoolismo

O conjunto de dados *UCI Alcoholism Detection* [103], da Universidade da Califórnia, Irvine, é amplamente utilizado na pesquisa sobre detecção de alcoolismo. Este conjunto contém gravações de EEG de 122 sujeitos, divididos em dois grupos: indivíduos com histórico de alcoolismo e um grupo de controle. Cada participante realizou 120 ensaios com estímulos visuais, gerando sinais de EEG de alta resolução temporal, coletados a 256 Hz em 64 canais.

O conjunto de dados UCI Alcoholism Detection [103], da Universidade da Califórnia, Irvine, é amplamente utilizado em estudos que investigam correlações entre padrões de EEG e predisposição genética ao alcoolismo. A base contém gravações de sinais de EEG obtidas de 122 sujeitos, divididos em dois grupos: indivíduos diagnosticados com alcoolismo e um grupo controle. Os sinais foram registrados por 64 eletrodos posicionados no couro cabeludo, com taxa de amostragem de 256 Hz durante janelas de 1 segundo. Cada participante foi submetido a 120 ensaios com estímulos visuais oriundos de um conjunto de imagens padronizadas, incluindo imagens de bebidas alcoólicas. Os ensaios apresentavam um único estímulo (S1) ou dois estímulos (S1 e S2), exibidos em condições pareadas (S1 = S2) ou não pareadas (S1  $\neq$  S2), com o objetivo de investigar respostas diferenciais entre os grupos.

#### Resultados da Literatura

Vários estudos exploraram este conjunto para testar metodologias inovadoras. Zhu et al. [104] utilizaram o *Horizontal Visibility Graph Entropy* (HVGE) para capturar características não lineares, alcançando uma acurácia de 95,8% com os classificadores KNN e SVM. Mukhtar et al. [105] empregaram CNNs para classificar diretamente os sinais brutos de EEG, obtendo uma acurácia média de 98%.

Bavkar et al. [106] propuseram uma abordagem inovadora para otimizar a seleção de canais, utilizando o *Improved Binary Gravitational Search Algorithm* (IBGSA). Esse método reduziu o número de canais para 13 e alcançou uma acurácia de 92,5% por meio

da validação cruzada LOSO. Além disso, destaca-se a aplicabilidade prática em sistemas de detecção de alcoolismo em tempo real.

# Capítulo 3

# Solução Proposta

Neste trabalho, desenvolvemos e avaliamos um algoritmo baseado em PGA, denominado Genetic Programming-Based Feature Selection and Construction (GP-FeSCo), para construção e seleção de atributos em dados complexos, com foco em dados de EEG. O objetivo é automatizar a identificação de atributos relevantes, reduzir a dimensionalidade e aprimorar a generalização dos modelos de classificação. Nesta seção, detalhamos as etapas de desenvolvimento e validação do GP-FeSCo, incluindo sua estrutura, operadores genéticos, avaliação de aptidão e a abordagem de validação cruzada.

# 3.1 Estrutura e Funcionamento do GP-FeSCo

O algoritmo proposto, GP-FeSCo, realiza a construção e seleção automática de atributos a partir de dados de entrada, gerando atributos informativos e reduzindo a dimensionalidade. O GP-FeSCo utiliza uma representação em forma de árvore, onde os nós terminais representam atributos originais e constantes, enquanto os nós intermediários aplicam operações matemáticas. Essa configuração permite que o GP-FeSCo construa transformações complexas de atributos de forma autônoma, simplificando a identificação dos atributos mais relevantes. Cada árvore representa um conjunto de atributos construídos, avaliados por classificadores como Naive Bayes, Random Forest e K Vizinhos mais Próximos - K-Nearest Neighbors (KNN), com o desempenho do modelo medido pela métrica de F1-score. Essa métrica é adequada para conjuntos de dados complexos, como EEG, onde equilibrar precisão e recall é essencial para a simplicidade e interpretabilidade do modelo [41, 54].

O fluxo do framework GP-FeSCo, ilustrado na Figura 3.1, apresenta um processo estruturado e evolutivo para a análise de dados de EEG. O ponto de partida é um conjunto de dados brutos de EEG, submetido a uma etapa inicial de segmentação do sinal, na qual os registros contínuos são divididos em trechos menores, seguidos da extração de atributos básicos que descrevem características fundamentais dos dados. Esses atribu-

tos pré-processados alimentam o GP-FeSCo, que opera em três estágios principais: (1) Seleção de Atributos, onde os atributos mais relevantes são identificados; (2) Construção de Atributos, fase em que novos atributos são gerados por meio da combinação e transformação dos atributos selecionados; e (3) Classificação de Atributos, onde os atributos construídos são avaliados quanto à sua capacidade de melhorar a performance de classificadores de AM. O GP-FeSCo gera como saída um conjunto de atributos otimizados e um classificador treinado, prontos para aplicação em tarefas de classificação de EEG, oferecendo uma abordagem automatizada e eficaz para problemas complexos de análise de sinais.

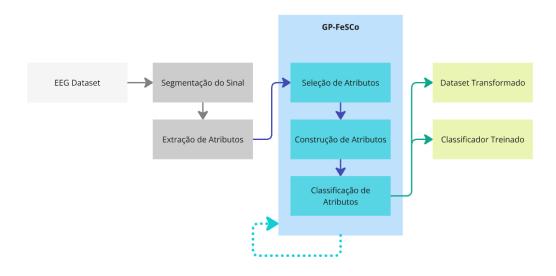


Figura 3.1: Fluxograma do framework GP-FesCo para análise de dados de EEG.

No cerne do framework, o operador F permite que cada árvore expresse múltiplos atributos, marcando subárvores como independentes, conforme proposto por Guo et al. [41]. O operador F é definido pela equação F(x) = x. Embora não modifique os elementos da subárvore, ele desempenha um papel crucial na identificação dos trechos considerados atributos.

Como exemplo prático, considere a aplicação do GP-FeSCo em um conjunto de dados com 10 atributos iniciais, representados como  $A_1, A_2, \ldots, A_{10}$ . Após a execução do algoritmo, a árvore resultante, ilustrada na Figura 3.2, demonstra como o GP-FeSCo construiu novos atributos a partir das características originais. A árvore revela a geração de diversos atributos por meio de uma série de operadores (círculos azuis) e constantes (losangos amarelos) aplicados aos atributos de entrada (retângulos verdes). Nessa abordagem, operações como adição (+), subtração (-), multiplicação (×), divisão (/), além de funções como logaritmo (log) e valor absoluto (abs), combinam atributos de entrada,

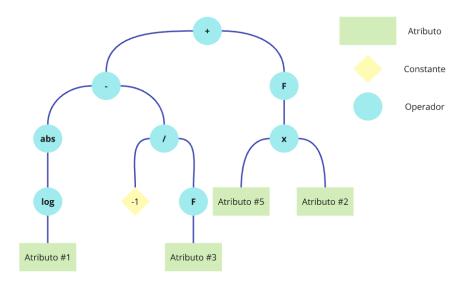


Figura 3.2: Ilustração de uma árvore de construção de atributos no framework GP-FesCo.

possibilitando a construção de novos atributos complexos, promovendo um processo de engenharia de atributos adaptável e eficiente.

Nesse caso, os novos atributos identificados pelo operador F foram:

- $F(A_5, A_2) = A_5 \times A_2$
- $F(A_3) = A_3$

Esses novos atributos foram selecionados e construídos automaticamente pelo GP-FeSCo com o objetivo de maximizar o desempenho do modelo de classificação. A inclusão de operações matemáticas, como logaritmos e valores absolutos, possibilitou a geração de combinações não triviais adaptadas às dinâmicas do conjunto de dados. Esse exemplo ilustra como o GP-FeSCo não apenas identifica os atributos mais relevantes, mas também cria novos atributos que capturam interações complexas, otimizando a representação dos dados para aplicações avançadas, como a classificação de sinais de EEG. Uma característica interessante desse método é que nem todas as operações contidas na árvore contribuem diretamente para a seleção e construção de atributos. No exemplo ilustrado na Figura 3.2, o trecho  $abs(\log(A_1))$  não possui o operador F. Contudo, devido às operações de crossover e mutação, essa expressão pode ser relevante e passada para gerações futuras, contribuindo para a criação de um novo indivíduo ainda mais especializado.

Um dos desafios ao trabalhar com sinais de EEG é a extração de atributos representativos de uma janela de sinal, especialmente devido à alta presença de ruído e às características não estacionárias. Um número reduzido de atributos pode ser insuficiente para discriminar os diferentes rótulos presentes no problema. Por outro lado, a adição de

atributos redundantes ou desnecessários prejudica o desempenho do modelo, conforme demonstrado pelo efeito Hughes [107]. Essa adição aumenta o tempo necessário para ajustar o *pipeline*, podendo até inviabilizar o processo em cenários com recursos computacionais limitados.

Nossa abordagem identifica automaticamente o número necessário de atributos, eliminando a configuração manual e permitindo que o GP-FeSCo se adapte flexivelmente aos dados. O operador F facilita combinações não lineares de atributos e aprimora a modularidade, possibilitando a propagação de subárvores relevantes entre gerações. Essa estratégia contribui para a eficiência computacional ao representar múltiplos atributos em uma única árvore.

É importante ressaltar que, apesar de a PG permitir a construção de atributos complexos e não lineares, a utilização excessiva desses atributos pode comprometer a interpretabilidade do modelo. Assim, existe um *trade-off* natural entre desempenho preditivo e interpretabilidade. Portanto, é fundamental adotar estratégias de controle de complexidade, como limitar a profundidade das árvores, para preservar a compreensão das soluções geradas.

Para promover a diversidade e evitar a convergência prematura, parâmetros genéticos essenciais, como a profundidade máxima da árvore e as taxas de *crossover* e mutação, são configurados estrategicamente com base em experimentações prévias [54, 53]. Essas operações favorecem a exploração do espaço de atributos por meio de dois mecanismos principais:

- *Crossover*: Combina subárvores de dois indivíduos selecionados, permitindo a troca de atributos entre indivíduos de alto desempenho.
- Mutação: Altera nós dentro da árvore para gerar combinações de atributos inovadoras.

Além disso, o GP-FeSCo utiliza operadores protegidos, como divisão e raiz quadrada, adaptados para evitar erros numéricos, o que constitui uma prática padrão em aplicações de PG. Em síntese, o GP-FeSCo combina a seleção e construção de atributos, modularidade e redução de redundância, otimizando a criação de *pipelines* preditivos para alcançar alto desempenho em dados complexos, como sinais de EEG.

Para ilustrar a dinâmica do GP-FeSCo, apresentamos o pseudocódigo da abordagem no Algoritmo 1. Esse esquema resume as principais etapas do processo evolutivo, desde a criação da população até a seleção dos atributos mais promissores.

```
Algorithm 1 GP-FeSCo: Construção e Seleção de Atributos com Programação Genética
Require: Conjunto de dados de treinamento e validação (X_{\text{train}}, y_{\text{train}}, X_{\text{val}}, y_{\text{val}})
Require: Conjunto de operadores e terminais
Require: Número de gerações G
Require: Tamanho da população N
Require: Classificador base.
Ensure: Conjunto de atributos construídos e indivíduo com melhor desempenho.
 1: Inicializar a população com N indivíduos (árvores de construção de atributos).
 2: for geração g = 1 até G do
       for cada indivíduo na população do
 3:
          Identificar subestruturas da árvore iniciadas pelo operador marcador 'F'.
 4:
          if Número de subestruturas marcadas com 'F' > 0 then.
 5:
              Eliminar subárvores duplicadas.
 6:
              Compilar cada subárvore em uma função matemática.
 7:
              Aplicar cada função aos dados de entrada para transformar os atributos.
 8:
              Armazenar os atributos recém-calculados e evitar recalcular os já existentes.
 9:
              Concatenar os novos atributos aos anteriores.
10:
              Remover atributos altamente correlacionados (Coef. Correlação > 85%).
11:
12:
          else
13:
              A árvore inteira é utilizada como um único atributo
          end if
14:
          Treinar o classificador com os dados transformados e avaliar seu desempenho
15:
    no conjunto de validação.
          Atribuir ao indivíduo um valor de fitness com base no valor de F1-score sobre
16:
    o conjunto de validação.
       end for
17:
18:
       Selecionar os melhores indivíduos para reprodução.
       Aplicar crossover e mutação para gerar nova população.
19:
20: end for
21: Retornar os atributos construídos e o indivíduo com melhor fitness.
```

### 3.1.1 Melhorias de Eficiência e Redução de Redundância

Para aprimorar a eficiência computacional e minimizar redundâncias, o GP-FeSCo incorpora duas inovações principais:

- Remoção de Atributos Altamente Correlacionados: Eliminando automaticamente atributos construídos com correlações acima de 85%, o GP-FeSCo garante que os atributos finais sejam mais independentes e interpretáveis, reduzindo redundâncias e o risco de *overfitting*. Esse processo melhora a generalização do modelo ao focar em padrões únicos e informativos.
- Feature Cache: O GP-FeSCo inclui um cache para armazenar atributos previamente calculados entre gerações. Isso evita cálculos redundantes e acelera o tempo

de execução, especialmente em problemas complexos que exigem extensivas avaliações de aptidão.

A avaliação da aptidão de cada árvore é realizada com base na métrica de *F1-score*, com o objetivo de maximizar a generalização ao otimizar o equilíbrio entre precisão e recall.

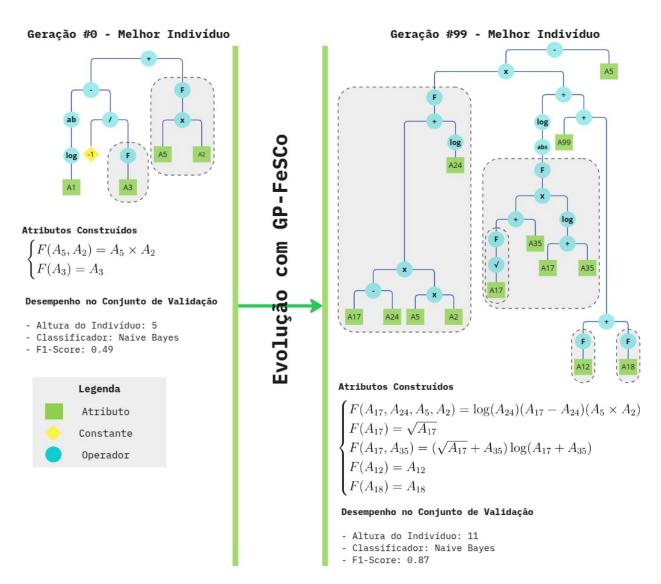


Figura 3.3: Evolução do melhor indivíduo ao longo das gerações no framework GP-FeSCo. À esquerda, a árvore gerada na geração inicial (#0), composta por operadores simples e estruturas rasas, resulta em baixo desempenho preditivo (F1-score = 0.49). À direita, após 99 gerações, observa-se o aumento de complexidade da árvores, com combinações mais elaboradas de atributos, elevando significativamente o desempenho (F1-score = 0.87).

A Figura 3.3 ilustra a evolução dos indivíduos no processo de PG aplicado à construção de atributos no framework GP-FeSCo. À esquerda, observa-se o melhor indivíduo da

geração inicial, caracterizado por estruturas simples e expressões pouco informativas. Nesse estágio, os atributos construídos são formados por operações básicas entre variáveis originais, resultando em baixo desempenho no conjunto de validação (F1-score = 0.49). Após 99 gerações de evolução, o melhor indivíduo (à direita) apresenta uma estrutura mais profunda e complexa, integrando múltiplos operadores matemáticos e combinações não-lineares entre atributos mais informativos. Essa complexificação proporcionou um ganho significativo de desempenho (F1-score = 0.87), demonstrando a capacidade do algoritmo de explorar o espaço de soluções e otimizar representações simbólicas relevantes. Além disso, como os atributos construídos mantêm forma analítica, a solução final preserva a interpretabilidade — característica desejável em aplicações biomédicas e clínicas.

### 3.1.2 Aplicação de LOSO

Para avaliar a eficácia do GP-FeSCo em cenários do mundo real, utilizamos a Validação Cruzada LOSO, uma abordagem recomendada para estudos com dados de EEG, especialmente quando a variabilidade entre sujeitos é significativa. A LOSO assegura que os dados de teste de cada sujeito sejam independentes dos dados de treinamento, proporcionando uma avaliação robusta da generalização do modelo. Internamente, dentro do conjunto de treino, realizamos uma divisão do tipo *holdout* tradicional (80/20) para separar parte dos dados para validação, possibilitando a avaliação intermediária de desempenho ao longo das gerações da PG. Esse duplo esquema de validação garante maior robustez e reduz o risco de sobreajuste.

Como destacado por Kunjan et al. [1] e Kinahan et al. [25], a LOSO é fundamental para prevenir que o modelo se baseie em características específicas dos sujeitos, permitindo a captura de padrões generalizáveis e relevantes para tarefas de classificação. Isso é especialmente importante em aplicações como o diagnóstico baseado em EEG.

### 3.1.3 Processo de Extração de Atributos

Para analisar eficazmente os sinais complexos de EEG, aplicamos o GP-FeSCo em conjuntos distintos de atributos, cada um fornecendo *insights* únicos sobre os dados. Os conjuntos — Estatístico, Espectral, Complexidade e *CAnonical Time-series CHaracteristics* (catch22) [2] — são pré-extraídos dos dados brutos de EEG, oferecendo ao GP-FeSCo uma base abrangente para a seleção e construção de atributos. Antes da extração, cada canal do EEG é decomposto em cinco bandas de frequência tradicionalmente utilizadas na análise do sono: *delta* (0–4 Hz), *theta* (5–7 Hz), *alpha* (8–12 Hz), *beta* (13–30 Hz) e *gamma* (>30 Hz). Essas bandas são amplamente utilizadas nos critérios visuais de pontuação dos sinais EEG do sono, conforme discutido por Motamedi et al. [18]. Essa decomposição pos-

sibilita uma análise detalhada da dinâmica dos sinais em diferentes faixas de frequência, capturando informações relevantes para a classificação e interpretação dos dados.

A Figura 3.4 ilustra o processo de pré-processamento e extração de atributos a partir de um único canal de EEG (FP1). O sinal original é decomposto em cinco faixas de frequência, permitindo capturar diferentes aspectos da atividade cerebral. Para cada faixa, são extraídos atributos em quatro grupos: catch22, atributos estatísticos, espectrais e de complexidade. Esses grupos serão discutidos nesta seção. Os mapas de calor à direita apresentam os valores normalizados desses atributos, permitindo visualizar sua variação entre faixas. Essa representação evidencia que, mesmo com um único canal, é possível gerar centenas de atributos, resultando em uma explosão dimensional ao analisar múltiplos canais simultaneamente. Essa característica impõe um desafio significativo para algoritmos de AM, exigindo estratégias eficientes para seleção, construção e redução de atributos, a fim de evitar sobreajuste e garantir uma boa generalização dos modelos.

Motamedi et al. [18] destacaram a necessidade de técnicas diversificadas de extração de atributos para capturar a natureza não linear e não estacionária dos sinais de EEG, especialmente em aplicações como a análise do sono. Ao utilizar esses conjuntos variados de atributos e a decomposição espectral prévia, o GP-FeSCo é capaz de captar dinâmicas essenciais do sinal, fundamentais para uma classificação precisa e uma interpretação clara.

A separação de grupos de atributos possibilita compreender a importância de cada grupo no *pipeline* treinado em seu espaço de busca, por meio da interpretação no domínio dos sinais. As estatísticas instantâneas estão diretamente relacionadas à morfologia do segmento, enquanto os atributos espectrais descrevem aspectos da Transformada de Fourier. O desempenho das soluções geradas pelo GP-FeSCo em diferentes tarefas de classificação de EEG contribui para a compreensão do domínio em questão.

### Conjunto de Atributos Estatísticos

O Conjunto de Atributos Estatísticos abrange medidas fundamentais que caracterizam as propriedades de distribuição do sinal, como percentis e medidas de tendência central. Este conjunto é essencial na análise de EEG, pois fornece um resumo de alto nível da variabilidade e das tendências centrais dos dados. Os principais atributos desse conjunto incluem:

- Percentis (5°, 25°, 75° e 95°): Oferecem uma visão geral da distribuição do sinal em diferentes quantis, ajudando a descrever variações em amplitude.
- Média, Mediana e Variância: Capturam a tendência central e a dispersão, características frequentemente usadas na análise de EEG para distinguir diferentes estados cerebrais.

- Entropia: Quantifica a incerteza ou desordem no sinal. É especialmente útil para identificar mudanças no estado do cérebro, como transições entre estágios de sono [18].
- Raiz do Valor Quadrático Médio Root Mean Square (RMS) e Desvio padrão Standard deviation (STD): Medem a variação da amplitude, útil para identificar anomalias como eventos transientes de alta energia.
- *Kurtosis* e Assimetria: Descrevem a forma da distribuição, indicando a concentração de valores em torno da média e a assimetria dos dados.
- Crossings (Zero e Média): Contam transições do sinal através de pontos de referência, o que é indicativo de comportamento oscilatório.

Essas características são essenciais para distinguir padrões de EEG associados a diferentes estados cerebrais, proporcionando informações sobre a forma geral e a variabilidade do sinal. Estudos indicam que a combinação dessas métricas com análises mais complexas, como transformadas de Fourier, aprimora a precisão na classificação de estágios de sono e eventos transientes [18].

### Conjunto de Atributos Espectrais

O Conjunto de Atributos Espectrais foca em características no domínio da frequência, essenciais para identificar padrões oscilatórios dentro dos dados de EEG. As características espectrais possibilitam a análise de bandas de frequência específicas, como alfa, beta e theta, que são relevantes no EEG, uma vez que diferentes ritmos cerebrais estão associados a diversos estados fisiológicos. Este conjunto inclui:

- Percentis (5°, 25°, 75° e 95°): Representam a distribuição da potência espectral em diferentes quantis.
- Média, Mediana e Variância da Potência Espectral: Resumem o conteúdo de frequência central e disperso.
- Entropia no Domínio da Frequência: Mede a complexidade da distribuição espectral, ajudando a caracterizar eventos como espículas e fusos de sono.
- RMS e STD no Domínio da Frequência: Descrevem variações de potência em frequências específicas.
- *Kurtosis* e Assimetria no Domínio da Frequência: Caracterizam a forma da distribuição espectral, fornecendo *insights* sobre padrões de atividade cerebral.

• Crossings (Zero e Média) no Domínio da Frequência: Contam oscilações de frequência ao redor de pontos de referência, úteis na detecção de fusos de sono [18].

Essas características permitem que o GP-FeSCo detecte padrões em bandas de frequência específicas do EEG, como ondas alfa e beta, que indicam diferentes estados cognitivos e fisiológicos. Métodos como a Transformada de Fourier e a Transformada Wavelet são frequentemente utilizados para extrair essas características, melhorando a resolução temporal e espectral [18].

### Conjunto de Atributos de Complexidade

O Conjunto de Atributos de Complexidade captura a complexidade estrutural e dinâmica dos sinais de EEG. Inclui métricas avançadas que avaliam dinâmicas não lineares, úteis para identificar padrões irregulares ou caóticos frequentemente associados a anomalias neurológicas. Os atributos deste conjunto são:

- Medidas de Entropia (Shannon, Baseada em FFT, SVD, Fisher) [108, 109, 110, 111]: Diferentes métodos de quantificar a aleatoriedade e o conteúdo informacional.
- Dimensão Fractal (Higuchi e Petrosian) [112, 113]: Mede a auto-semelhança e a complexidade dentro da estrutura do sinal, detalhando o sinal em várias escalas [18].

Estas métricas fornecem *insights* sobre a imprevisibilidade e a estrutura complexa do sinal, auxiliando na distinção entre atividade cerebral normal e anormal, como demonstrado em aplicações de detecção de epilepsia e análise de fusos de sono [18].

#### Conjunto de Atributos catch22

O conjunto de atributos catch22, desenvolvido por Lubba et al. [2], oferece uma abordagem eficiente para a análise de séries temporais, condensando características dinâmicas dos dados em um vetor de atributos altamente interpretável e de baixa dimensionalidade. Os autores selecionaram, através de um processo sistemático e baseado em desempenho, 22 atributos canônicos (*CAnonical Time-series CHaracteristics*) que encapsulam as propriedades fundamentais das séries temporais. Esse conjunto é derivado de uma redução dimensional a partir de 4791 atributos iniciais da biblioteca *Highly comparative time-series analysis* (HCTSA) [114], mantendo um desempenho competitivo com significativa redução no tempo de computação.

Os atributos do catch22 abrangem uma diversidade de características, incluindo propriedades lineares e não lineares de autocorrelação, diferenças sucessivas, distribuições de

valores e análise de flutuações. Esses atributos oferecem uma base interpretável e eficiente para tarefas de classificação, como a análise de sinais EEG, onde a identificação de padrões dinâmicos é essencial. A seguir, estão descritos os principais atributos e suas contribuições:

- Distribuição e Estatísticas Temporais Simples: Incluem o cálculo de modas em distribuições normalizadas, períodos consecutivos de valores acima ou abaixo da média, e intervalos entre eventos extremos. Essas métricas permitem identificar padrões de recorrência e valores fora do comum na série temporal, úteis para detectar eventos significativos nos sinais EEG.
- Autocorrelação Linear e Não Linear: Atributos como o primeiro mínimo da função de autocorrelação ajudam a capturar dependências temporais e padrões oscilatórios complexos, fornecendo informações valiosas sobre a dinâmica neuronal.
- Diferenças Sucessivas e Entropia: Medidas como a proporção de diferenças sucessivas acima de um limiar e entropia avaliam a regularidade e complexidade estrutural do sinal, auxiliando na identificação de comportamentos caóticos ou de sistemas estáveis.
- Flutuações Escalares: Incluem métricas de ajuste a escalas temporais, como a análise de flutuação de escalas, fornecendo informações sobre a auto-similaridade e as variações de longo prazo no sinal.

O catch22 destaca-se por sua eficiência computacional, apresentando um tempo de execução quase linear em relação ao comprimento da série temporal. Essa característica torna o *framework* aplicável a sinais de EEG, que frequentemente contêm grandes volumes de dados temporais.

Em estudos comparativos, o conjunto apresentou desempenho similar ao de conjuntos maiores, com uma redução de apenas 7% na acurácia média para tarefas de classificação. Simultaneamente, o tempo de execução foi reduzido em aproximadamente mil vezes para séries de 10.000 amostras. Essa eficiência permite que catch22 seja amplamente utilizado em diversas áreas, como finanças, medicina e indústria, sendo uma ferramenta de apoio na análise de dados de EEG. A capacidade de representar propriedades complexas de forma interpretável e eficiente torna esse conjunto uma solução promissora para análises preditivas e diagnósticas [2].

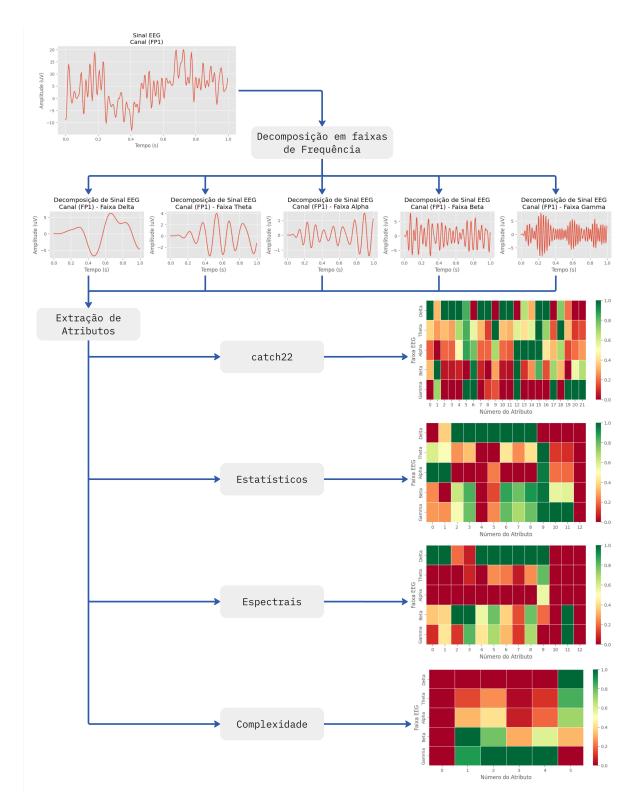


Figura 3.4: Fluxo de decomposição e extração de atributos a partir de sinais EEG. O sinal bruto do canal FP1 é segmentado em cinco faixas de frequência, e cada componente é submetido à extração de atributos em quatro grupos distintos. Os mapas de calor representam os valores normalizados dos atributos extraídos (linhas: faixas de frequência; colunas: atributos), permitindo visualizar a distribuição e intensidade de cada característica em diferentes bandas.

# Capítulo 4

# Metodologia

Nesta seção, apresentamos os procedimentos metodológicos adotados para a realização deste estudo, desde a seleção dos conjuntos de atributos até a análise quantitativa e qualitativa dos resultados. O objetivo dessa abordagem estruturada é garantir a reprodutibilidade e a qualidade dos experimentos realizados.

# 4.1 Pré-processamento e Extração de Atributos

Nesta etapa, apresentamos os procedimentos adotados para o pré-processamento dos sinais de EEG e a extração de atributos. Diferentemente das abordagens tradicionais que aplicam filtragem e remoção de artefatos, mantivemos os dados brutos em sua forma original. Essa decisão visa avaliar a capacidade de criação de atributos do *GP-FeSCo* ao lidar com sinais que refletem condições mais próximas da realidade de aquisição, frequentemente impactados por ruídos fisiológicos e ambientais. Realizamos apenas a decomposição de cada canal EEG nas cinco bandas de frequência: *delta* (0–4 Hz), *theta* (5–7 Hz), *alpha* (8–12 Hz), *beta* (13–30 Hz) e *gamma* (> 30 Hz). Na Tabela 4.2, essa decomposição resulta em um aumento de cinco vezes no número de atributos gerados por canal.

A partir dessa decomposição, os atributos são extraídos de diversos domínios, incluindo estatísticas temporais, espectrais, medidas de complexidade e o conjunto catch22. Essa abordagem estruturada permite que o *GP-FeSCo* maximize as informações presentes nos sinais EEG, selecionando e combinando atributos de forma otimizada, melhorando o desempenho dos classificadores.

### 4.1.1 Segmentação dos Sinais

Os sinais de EEG foram segmentados em janelas temporais não sobrepostas, permitindo a análise em intervalos discretos. A escolha do tamanho da janela foi baseada em estudos

Tabela 4.1: Características das Bases de Dados

Base de Dados	Número de Pacientes	Grupos de Pacientes (LOSO)	Pacientes por Grupo	Frequência de Coleta	Número de Canais
Sleep Spindles	8	8	1	50 Hz (Paciente 3), 100 Hz (Paciente 0) e 200 Hz	3
Detecção de Alcoolismo	120	10	12	256 Hz	64
Imagética Motora	109	10	10-11	160 Hz	64

Tabela 4.2: Características dos Grupos de Atributos

Grupo de Atributos	Número de Atributos	Atributos Gerados por Canal EEG	Total de Atributos Sleep Spindles	Total de Atributos Detecção de Alcoolismo e Imagética Motora
catch22	22	$22 \times 5 = 110$	330	7040
statistic	13	$13 \times 5 = 65$	195	4160
spectral	13	$13 \times 5 = 65$	195	4160
complexity	6	$6 \times 5 = 30$	90	1920
all	54	$54 \times 5 = 270$	810	17280

prévios, buscando um equilíbrio entre a preservação das características temporais do sinal e a viabilidade computacional. Cada segmento representou um evento ou tarefa específica, alinhando-se aos contextos descritos na seção anterior sobre os conjuntos de dados.

### 4.1.2 Descrição dos Atributos

A extração de atributos concentrou-se nos quatro grupos apresentados no Capítulo anterior: estatísticos, espectrais, de complexidade e a coleção catch22, conforme detalhado nas Tabelas 4.3, 4.4, 4.5 e 4.6. Esses atributos foram selecionados por sua capacidade de capturar diferentes aspectos dos sinais de EEG, abrangendo desde tendências gerais até padrões complexos que caracterizam fenômenos específicos.

- Estatísticos: Incluem métricas como média, variância e curtose, fornecendo uma visão geral da distribuição do sinal (Tabela 4.3).
- Espectrais: Derivados da transformada de Fourier, esses atributos capturam informações sobre a composição de frequências e a potência do sinal (Tabela 4.4).

- Complexidade: Indicadores como entropia e dimensões fractais foram utilizados para descrever a natureza dinâmica e não linear dos sinais de EEG (Tabela 4.5).
- catch22: Um conjunto compacto de 22 atributos que engloba medidas estatísticas, dinâmicas e espectrais, otimizadas para eficiência em tarefas de classificação (Tabela 4.6).

Cada segmento resultou em um vetor de atributos. Esses vetores foram então utilizados como entrada para o GP-FeSCo, permitindo a avaliação de sua capacidade de generalização em dados com ruídos e artefatos.

Tabela 4.3: Atributos Estatísticos Extraídos

Atributo	Descrição
n5	Percentil 5% do conjunto de valores.
n25	Percentil 25% do conjunto de valores.
n75	Percentil 75% do conjunto de valores.
n95	Percentil 95% do conjunto de valores.
median	Mediana do conjunto de valores.
mean	Média aritmética do conjunto de valores.
std	Desvio padrão do conjunto de valores.
var	Variância do conjunto de valores.
rms	Média quadrática (root mean square) dos valores.
skewness	Assimetria (skewness) do conjunto de valores.
kurtosis	Curtose do conjunto de valores.
no_zero_crossings	Número de cruzamentos do eixo zero no conjunto de valores.
no_mean_crossings	Número de cruzamentos em relação à média dos valores.
en	Entropia do conjunto de valores

Tabela 4.4: Atributos Espectrais Extraídos

Atributo	Descrição
n5	Percentil 5% da PSD.
n25	Percentil 25% da PSD.
n75	Percentil 75% da PSD.
n95	Percentil 95% da PSD.
median	Mediana da PSD.
mean	Média aritmética da PSD.
std	Desvio padrão da PSD.
var	Variância da PSD.
rms	Média quadrática (root mean square) da PSD.
skewness	Assimetria (skewness) da PSD.
kurtosis	Curtose da PSD.
$no\_zero\_crossings$	Número de cruzamentos do eixo zero na PSD.
$no\_mean\_crossings$	Número de cruzamentos em relação à média da PSD.
en	Entropia da PSD.

Tabela 4.5: Atributos de Complexidade Extraídos

Categoria	Atributo	Descrição
	Shannon Entropy	Limite absoluto sobre o comprimento médio
Entropia		ideal para a codificação sem perda de uma
Епиоріа		fonte de informação.
	Sample Entropy	Mede a probabilidade condicional de que dois
		padrões similares continuem similares, indi-
		cando auto-similaridade.
	Multiscale Entropy	Mede a complexidade de uma série temporal
		em diferentes escalas.
	SVD Entropy	Mede a "riqueza"das informações, avaliando
		quantos vetores ortogonais são necessários
		para explicar os dados.
Dimensão Fractal	Higushi	Algoritmo para medir a Dimensão Frac-
Dimensao Tractar		tal, avaliando auto-similaridade de sequên-
		cias temporais.
	Petrosian Fractal	Método rápido para calcular a Dimensão
	Dimension	Fractal convertendo a série em uma sequên-
		cia binária.
	Fisher Information	Mede a quantidade de informação que uma
Outros		variável observada carrega sobre um parâme-
		tro desconhecido.
	Hurst Exponent	Mede a "memória de longo prazo" de uma
		série temporal, indicando persistência ou an-
	777	tipersistência.
	DFA	Adaptação do Hurst Parameter para proces-
		sos não estacionários.
	Lyapunov Expo-	Mede caos e imprevisibilidade; exponentes
	nent	positivos indicam sensibilidade a condições
		iniciais.

Tabela 4.6: Atributos do Grupo catch<br/>22  $\left[2\right]$ 

Categoria	Atributo	Descrição
Propriedades de	DN_HistogramMode	Modo da distribuição normalizada (z-score,
Distribuição	_5	histograma com 5 bins).
	DN_HistogramMode	Modo da distribuição normalizada (z-score,
	_10	histograma com 10 bins).
Estatísticas	SB_BinaryStats	Maior sequência de valores consecutivos
Temporais	_mean _longstretch1	acima da média.
Simples	DN_OutlierInclude	Intervalos entre eventos extremos sucessivos
	_p_001_mdrmd	acima da média.
	DN_OutlierInclude	Intervalos entre eventos extremos sucessivos
	_n_001_mdrmd	abaixo da média.
	CO_flecac	Primeiro cruzamento $1/e$ da função de auto-
		correlação.
Correlação	CO_FirstMin_ac	Primeiro mínimo da função de autocorrela-
Linear		ção.
	SP_Summaries_welch	Potência total no quinto mais baixo da
	$\_{rect}\_{area}\_5\_1$	frequência na análise espectral de Welch.
	SP_Summaries	Centróide do espectro de potência de Fourier.
	_welch_rect_centroid	
	FC_LocalSimple	Erro médio de uma previsão local com janela
	_mean3_stderr	de 3 amostras.
C 1 ~ N~	CO_trev_1_num	Estatística de reversibilidade temporal.
Correlação Não	CO_HistogramAM	Informação mútua automatizada, $(m = $
Linear	I_even_2_5	$2, \tau = 5$ ).
	IN_AutoMutualInfo	Primeiro mínimo da função de informação
	Stats_40_gaussian	mútua automatizada.
	MD_hrv_classic	Proporção de diferenças sucessivas que exce-
D:C	_pnn40	dem $0.04\alpha$ [115].
Diferenças	SB_BinaryStats	Maior sequência de decréscimos incrementais
Sucessivas	_diff_longstretch0	sucessivos.
	SB_MotifThree	Entropia de Shannon de dois padrões sucessi-
	_quantile_hh	vos na simbolização de 3 letras equiprováveis.
	FC_LocalSimple	Mudança no comprimento da correlação após
	_mean1_tauresrat	diferenciação iterativa.
	CO_Embed2_Dist	Ajuste exponencial às distâncias sucessivas
	_tau_d_expfit	em espaço de incorporação 2D.
Análise de Flu-	SC_FluctAnal	Proporção de flutuações de escala mais lenta
tuações	_2_dfa_50_1_2_logi	que seguem DFA (50% de amostragem).
	SC_FluctAnal_2	Proporção de flutuações de escala mais lenta
	_rsrangefit_50_1	ajustadas por alcance rescalado.
Outres	SB_TransitionMatrix	Traço da covariância da matriz de transição
Outros	_3ac_sumdiagcov	entre símbolos em um alfabeto de 3 letras.
	PD_PeriodicityWang	Medida de periodicidade de Wang et al.
	_th0_01	(2007) [116].

## 4.2 Aplicação do GP-FeSCo

Nesta seção, detalhamos a aplicação do *GP-FeSCo* nos experimentos realizados, destacando a utilização de cinco classificadores: Árvore de decisão - *Decision tree* (DT), K Vizinhos mais Próximos - *K-Nearest Neighbors* (KNN), *Random Forest* (RF), Máquina de Vetores de Suporte - *Support Vector Machine* (SVM) e *Naive Bayes* (NB). A escolha desses classificadores foi guiada por suas características complementares e no desempenho em diversas tarefas de classificação, conforme descrito na literatura.

### 4.2.1 Classificadores Utilizados

Os cinco classificadores selecionados possuem abordagens e vantagens distintas para tarefas de classificação [117, 118, 119, 120]:

- Árvore de decisão Decision tree (DT): DT são métodos de classificação que utilizam uma estrutura hierárquica para particionar o espaço de características. Cada nó interno representa uma característica do conjunto de dados, enquanto os ramos indicam os valores dessa característica. A classificação ocorre ao percorrer a árvore desde o nó raiz até os nós folha, que correspondem às classes preditas. Este método é amplamente utilizado por sua interpretabilidade e eficácia em dados não-lineares. No entanto, pode sofrer com overfitting, que pode ser mitigado com técnicas de poda.
- K Vizinhos mais Próximos K-Nearest Neighbors (KNN): O KNN é um algoritmo baseado em instâncias que classifica um exemplo de teste com base nas classes dos exemplos de treinamento mais próximos, utilizando uma métrica de distância como a euclidiana. Ele é simples e intuitivo, mas sua eficiência diminui em grandes conjuntos de dados devido ao alto custo computacional de calcular distâncias para cada amostra de teste.
- Random Forest (RF): Esse método combina múltiplas árvores de decisão para formar um modelo de conjunto, onde cada árvore é treinada em diferentes subconjuntos do conjunto de dados (amostragem com reposição). As previsões são obtidas por votação majoritária ou média das saídas das árvores individuais. O RF melhora a precisão e reduz o overfitting das árvores de decisão individuais, sendo adequado para grandes volumes de dados e tarefas complexas.
- Máquina de Vetores de Suporte Support Vector Machine (SVM): A SVM é uma técnica que busca encontrar um hiperplano ótimo que maximize a margem entre diferentes classes no espaço de características. É eficaz em problemas

lineares e pode ser estendido para não-linearidades usando funções *kernel*. A SVM é frequentemente utilizada em conjuntos de dados de alta dimensionalidade, mas seu treinamento pode ser computacionalmente intensivo.

• Naive Bayes (NB): Este é um classificador probabilístico baseado no Teorema de Bayes, que assume independência condicional entre as características. Apesar de essa suposição raramente ser verdadeira, o NB tem bom desempenho em várias aplicações práticas, incluindo classificação de textos. É simples, eficiente e requer poucos dados para treinamento.

A diversidade desses classificadores possibilita explorar diferentes perspectivas e lidar com a variabilidade inerente aos sinais de EEG, permitindo uma avaliação abrangente do desempenho do *GP-FeSCo*.

### 4.2.2 Parâmetros de Programação Genética

Para a aplicação do GP-FeSCo, configuramos a Programação Genética (PG) com parâmetros cuidadosamente selecionados para equilibrar a exploração e a convergência do algoritmo. A escolha desses parâmetros influencia diretamente a capacidade do modelo em construir atributos discriminativos sem gerar expressões excessivamente complexas que possam levar ao *overfitting*. A Tabela 4.7 apresenta os principais hiperparâmetros utilizados.

Tabela 4.7: Parâmetros configurados para a Programação Genética no GP-FeSCo.

Parâmetro	Valor
Tamanho da População	50
Número de Gerações	100
Taxa de Mutação	15%
Taxa de Crossover	85%
Seleção	Torneio
Controle de Bloat	Altura máxima de árvore: 16
Conjunto de Operadores	$+, -, \times, \div *, \sqrt{*, \log^*, F}$
Critério de Parada	Geração Máxima
Outros parâmetros	Elitismo

A elevada taxa de mutação (15%) foi adotada para incentivar a diversidade da população, promovendo uma maior exploração do espaço de busca e evitando a convergência prematura para soluções sub-ótimas. Para mitigar o efeito do bloat — o crescimento excessivo das expressões sem ganho significativo de desempenho — foi imposta uma altura máxima de 16 para as árvores evoluídas. Essa limitação garante que as soluções geradas

permaneçam interpretáveis e computacionalmente eficientes, sem comprometer a expressividade do modelo. No entanto, essa restrição pode limitar a capacidade de aprendizado dos indivíduos.

Selecionamos operações matemáticas básicas como operadores da PG. Porém, algumas delas não são definidas para todo número real (marcadas com \* na Tabela 4.7). Dessa forma, para evitar exceções durante os cálculos, transformamos esses operadores em funções protegidas:

• Divisão protegida:

$$\operatorname{div}_{protected}(a, b) = \begin{cases} a \div b, & \text{se } b \neq 0 \\ 1, & \text{se } b = 0 \end{cases}$$

• Logaritmo Natural protegido:

$$\log_{protected}(a) = \begin{cases} \log|a|, & \text{se } a \neq 0\\ 1, & \text{se } a = 0 \end{cases}$$

• Raíz Quadrada protegida:

$$\operatorname{sqrt}_{protected}(a) = \sqrt{|a|}$$

Além de permitir o cálculo por meio dos operadores, essa abordagem possibilita a inclusão de não-linearidades nos novos atributos gerados pelo GP-FeSCo.

## 4.3 Validação Cruzada por Paciente

Para garantir a generalização dos resultados, foi utilizada a validação cruzada por paciente (Leave-One-Subject-Out Cross-Validation, LOSO). Essa abordagem é amplamente empregada em estudos com dados de EEG, pois avalia o desempenho do modelo em novos indivíduos e simula cenários do mundo real. Em cada iteração, os dados de um paciente são separados como conjunto de teste, enquanto os dados dos demais formam o conjunto de treino. Essa estratégia assegura a independência dos dados entre os conjuntos e permite avaliar a capacidade de generalização do modelo para indivíduos não vistos durante o treinamento.

A validação LOSO é especialmente relevante para este estudo devido à alta variabilidade interpessoal dos sinais de EEG. A metodologia adotada minimiza viés e reflete um cenário prático de aplicação, como em sistemas personalizados de BCI.

### 4.4 Análise dos Resultados

Após a execução dos experimentos, os resultados foram analisados de forma quantitativa e qualitativa, considerando métricas de desempenho como acurácia, precisão e *F1-score*. Essas métricas foram selecionadas por sua relevância em tarefas de classificação, permitindo uma avaliação detalhada do comportamento de cada classificador em relação aos diferentes conjuntos de dados.

Os experimentos realizados são comparativos, contrastando um cenário baseline em que a tarefa de classificação é executada por meio dos classificadores, sem a intervenção do nosso framework e um cenário que aplica o GP-FeSCo. Adicionalmente, comparamos os resultados obtidos com abordagens distintas da literatura que utilizam os mesmos dados deste estudo.

A definição de um cenário baseline é fundamental para a avaliação do desempenho do nosso framework, pois estabelece um ponto de referência claro e replicável, a partir do qual as melhorias introduzidas pela abordagem baseada em PG podem ser quantificadas. Ao comparar os resultados do GP-FeSCo com esse cenário de base, onde os classificadores são aplicados diretamente, é possível isolar o impacto específico da seleção e construção automática de atributos sobre seu desempenho. Esse processo não apenas valida a eficácia da proposta em relação a métodos tradicionais, mas também fornece uma medida robusta de progresso em direção a modelos mais precisos, interpretáveis e generalizáveis.

Além disso, foi realizada uma análise comparativa entre os classificadores, evidenciando suas forças e limitações no contexto dos sinais de EEG. Essa abordagem possibilitou identificar os cenários em que cada técnica apresentou melhor desempenho, oferecendo *insights* valiosos para aplicações futuras.

### 4.5 Visão Geral

A Figura 4.1 ilustra a metodologia adotada neste estudo, que combina um fluxo estruturado de pré-processamento, extração diversificada de atributos e otimização evolutiva via PG, garantindo a construção de representações eficientes para a classificação de EEG. A validação LOSO assegura que os modelos sejam testados em condições próximas da realidade, evitando vieses associados à segmentação tradicional dos dados. Além disso, a engenharia de atributos automatizada pelo GP-FeSCo não apenas proporciona ganhos de desempenho, mas também identifica características mais interpretáveis, favorecendo aplicações biomédicas. O detalhamento dos parâmetros utilizados na PG e a diversidade de classificadores avaliados reforçam a flexibilidade e aplicabilidade do GP-FeSCo para diferentes desafios em neurociência computacional.

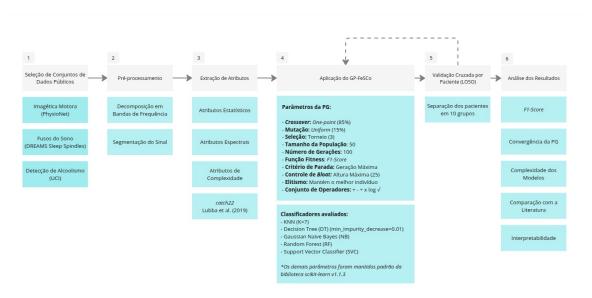


Figura 4.1: Fluxo detalhado da metodologia aplicada no GP-FeSCo.

# Capítulo 5

# Resultados

Neste capítulo, apresentamos os resultados da metodologia descrita no Capítulo 4. Os experimentos foram realizados para avaliar o desempenho do *GP-FeSCo* em distintos conjuntos de dados de EEG, incluindo *Physionet EEG Motor Movement/Imagery*, DREAMS *Sleep Spindles* e *UCI Alcoholism Detection*. Utilizamos os classificadores DT, KNN, RF, SVM e NB, implementando a LOSO para garantir uma análise adequada aos cenários de produção. Os resultados estão organizados em uma análise comparativa entre um cenário *baseline* e GP-FeSCo por conjunto de dados, acompanhada de uma discussão geral dos achados.

O cenário baseline é uma configuração de referência utilizada para comparar os resultados do GP-FeSCo. Nesse contexto, aplica-se diretamente os grupos de atributos, sem passar pela etapa de otimização e construção evolutiva realizada pela aplicação da PG. Assim, os modelos de AM são treinados e avaliados apenas com os atributos brutos.

O objetivo do cenário baseline é servir como um ponto de partida para avaliar os benefícios da abordagem proposta. Nos experimentos, a comparação entre os resultados obtidos com e sem o GP-FeSCo permite quantificar os ganhos em termos de desempenho, interpretabilidade e eficiência. Isso se reflete nas análises apresentadas, onde os mapas de calor e gráficos ilustram como a introdução da PG impacta as métricas de classificação, como F1-score e número de atributos selecionados.

# 5.1 Análise Comparativa por Conjunto de Dados

Abaixo, apresentamos uma análise detalhada dos resultados para cada conjunto de dados, organizada em três subseções: desempenho geral, convergência e interpretabilidade.

### 5.1.1 Sleep Spindles

### Desempenho Geral

A Figura 5.1 ilustra o desempenho médio do F1-Score no conjunto de teste para a tarefa de detecção de fusos do sono, considerando os cenários Baseline (à esquerda) e GP-FeSCo (à direita). Cada mapa de calor compara diferentes classificadores em combinação com os grupos de atributos disponíveis. A utilização do GP-FeSCo resulta em um aumento consistente no F1-Score, indicando que o método melhora a capacidade dos modelos em extrair e combinar atributos. A análise detalhada do desempenho dos classificadores, apresentada na Figura 5.2, reforça essa observação. Neste caso, a figura ilustra a aplicação específica do classificador NB, que foi o mais impactado pela aplicação do GP-FeSCo. A avaliação por paciente revela que, em todos os casos, a combinação dos atributos extraídos pelo GP-FeSCo resulta em desempenho semelhante ou superior ao do Baseline.

A Figura 5.3 ilustra a diferença média do F1-Score entre os conjuntos de validação e teste, comparando os cenários Baseline e GP-FeSCo. As barras representam o F1-Score médio, calculado para cada grupo de atributos em diferentes pacientes (0 a 7). No cenário com GP-FeSCo, observa-se uma melhoria geral nos valores do F1-Score, indicando que o método otimizou a combinação de atributos para a detecção de fusos do sono. Esse avanço resultou em um desempenho mais consistente entre os diversos pacientes e grupos de atributos. Embora tenha sido notada uma melhora no desempenho do modelo durante a fase de teste, desafios relacionados à generalização persistem, especialmente para os pacientes 3, 6 e 7, onde a diferença entre os conjuntos de validação e teste é mais acentuada. Identificamos um desafio específico para esses pacientes: o paciente 3 foi submetido a uma taxa de amostragem de 50 Hz, distinta das demais. Os pacientes 6 e 7 tiveram suas amostras rotuladas por apenas um especialista médico, ao invés de dois. Essas condições contribuíram para as dificuldades na generalização dos modelos. Apesar disso, houve uma melhora no desempenho em relação ao Baseline.

#### Convergência

A convergência do GP-FeSCo foi avaliada principalmente pelo aumento da função *fitness* ao longo das gerações, conforme ilustrado na Figura 5.4. Essa métrica reflete a melhoria progressiva das soluções geradas pelo algoritmo, evidenciando o aprimoramento dos atributos selecionados e construídos durante o processo evolutivo.

Para todos os classificadores avaliados, observa-se um crescimento acentuado da função fitness nas primeiras 20 gerações, seguido por uma estabilização. Esse comportamento sugere que o GP-FeSCo detecta rapidamente padrões relevantes nos dados, otimizando a combinação de atributos nas primeiras iterações. Após cerca de 50 gerações, o ganho

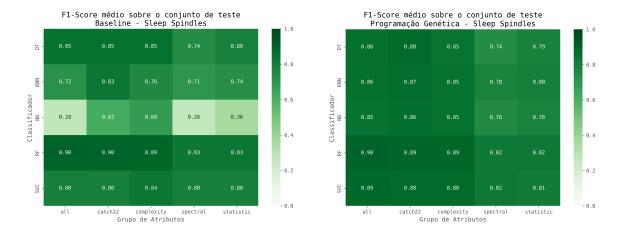


Figura 5.1: Desempenho médio de *F1-Score* sobre o conjunto de teste para o problema de detecção de fusos do sono (Sleep Spindles) nos cenários Baseline (à esquerda) e com a utilização do GP-FeSCo (à direita).

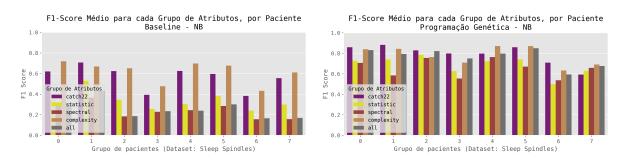


Figura 5.2: Desempenho médio do F1-Score para o classificador NB aplicado ao conjunto de dados  $Sleep\ Spindles$ , comparando os cenários Baseline (à esquerda) e GP-FeSCo (à direita).

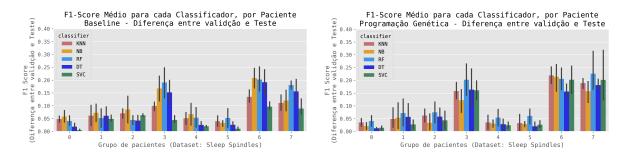


Figura 5.3: Diferença média do F1-Score entre os conjuntos de validação e teste para cada classificador no conjunto de dados  $Sleep\ Spindles$ , comparando os cenários Baseline (à esquerda) e GP-FeSCo (à direita). Cada barra representa a diferença para um paciente  $(0\ a\ 7)$ .

adicional de desempenho torna-se marginal, indicando que o algoritmo atingiu um ponto de saturação na otimização dos atributos. A partir desse ponto, há risco de *overfitting*.

Além disso, na Figura 5.5, analisa-se a evolução do número de atributos gerados e a

altura média dos indivíduos. Observa-se um crescimento rápido no número de atributos nas primeiras gerações, que se estabiliza quando o algoritmo identifica um conjunto eficiente de características para classificação. De modo semelhante, a altura das árvores apresenta um padrão análogo, indicando que a complexidade das soluções aumenta até certo ponto, quando o modelo começa a representar os padrões dos dados de forma eficaz. Esse fenômeno pode refletir um ganho na capacidade de modelagem, mas também implica um maior risco de perda de interpretabilidade e sobreajuste. Portanto, é fundamental monitorar esse crescimento e aplicar restrições, como a profundidade máxima, para garantir um equilíbrio entre complexidade e clareza.

Esses resultados indicam que o GP-FeSCo gera atributos de forma eficiente e converge para soluções otimizadas em um número reduzido de gerações. Esse comportamento é crucial para equilibrar o tempo de execução e a qualidade dos atributos gerados, especialmente em aplicações que exigem processamento eficiente.

Embora o número de gerações tenha sido fixado em 100 para todos os experimentos, notamos que, em vários casos, a função fitness já apresentava estabilização antes do término. Apesar dessa tendência à estabilização após certo número de gerações, interromper o processo prematuramente pode impedir a convergência de soluções mais refinadas. O número de gerações foi definido com base em testes preliminares [54, 53], que demonstraram que, mesmo com pequenas melhorias, as gerações finais ainda contribuem para a exploração do espaço de busca. Em aplicações futuras, técnicas como early stopping poderão ser consideradas para melhorar a eficiência.

### Interpretabilidade

A interpretabilidade dos atributos gerados pelo GP-FeSCo é um diferencial crucial para aplicações clínicas e neurocientíficas. Além de melhorar o desempenho dos classificadores, o método possibilita uma análise aprofundada das características dos sinais de EEG relevantes para a detecção de fusos do sono. Esse entendimento pode ser utilizado tanto na construção de modelos mais eficientes quanto na otimização do processo de aquisição de dados, reduzindo a coleta de informações desnecessárias.

A Figura 5.6 apresenta a distribuição dos atributos mais selecionados pelo melhor indivíduo no classificador NB, segmentados por grupos de atributos. Os resultados indicam que alguns atributos específicos dentro desses grupos foram frequentemente selecionados, sugerindo que determinadas propriedades do sinal são mais relevantes para a tarefa de classificação.

A análise se torna ainda mais interessante ao observar a Figura 5.7, que apresenta os 20 atributos mais selecionados pelo melhor indivíduo para os classificadores, aplicando o GP-FeSCo sobre todos os conjuntos de atributos combinados. Além de permitir a compa-

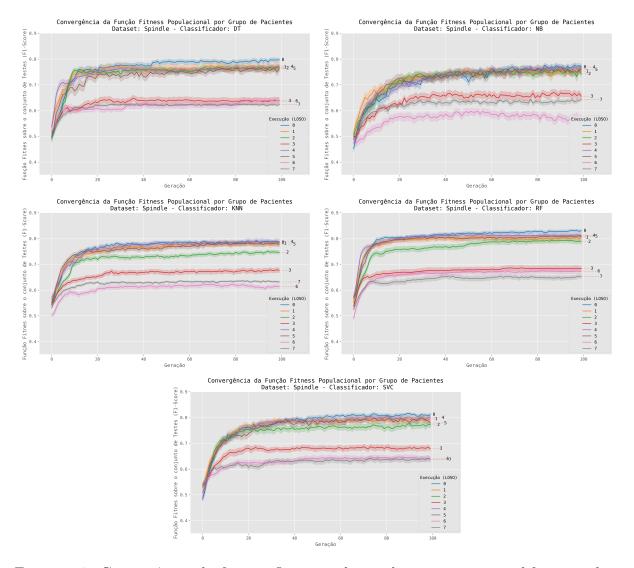
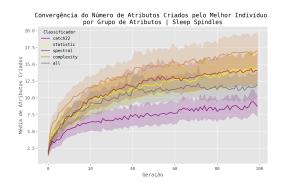


Figura 5.4: Convergência da função fitness ao longo das gerações para diferentes classificadores no conjunto de dados  $Sleep\ Spindles$ . Cada gráfico representa a evolução do F1-Score populacional médio para um classificador específico avaliado sobre os diferentes pacientes.

ração entre diferentes modelos, esses resultados revelam um aspecto poderoso do método proposto: a identificação não apenas dos atributos mais relevantes, mas também dos canais de EEG nos quais esses atributos são extraídos com maior frequência. Essa perspectiva é promissora para otimizar a coleta de sinais cerebrais, permitindo que estudos futuros foquem apenas nos canais mais informativos, reduzindo a quantidade de eletrodos necessários, como demonstrado em trabalhos anteriores [54, 121]. Essa abordagem oferece benefícios significativos do ponto de vista econômico, ao minimizar a necessidade de recursos, e do ponto de vista clínico, proporcionando maior conforto aos pacientes ao reduzir o número de eletrodos aplicados ao escalpo.



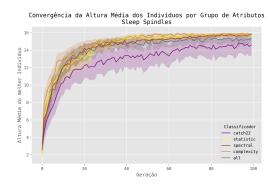


Figura 5.5: Evolução do número de atributos criados (esquerda) e da altura média dos indivíduos (direita) ao longo das gerações para diferentes grupos de atributos no conjunto Sleep Spindles.

Além disso, a comparação entre os atributos gerados pelo GP-FeSCo e aqueles utilizados no baseline demonstra que o método evolutivo seleciona subconjuntos mais informativos, reduzindo redundâncias. Isso resulta em modelos mais enxutos, eficientes e interpretáveis, sem comprometer o desempenho preditivo. A eliminação de atributos irrelevantes, sem perda significativa de desempenho, torna a proposta uma alternativa promissora para aplicações biomédicas, nas quais a precisão dos modelos e a facilidade de interpretação dos resultados são essenciais.

Dessa forma, o GP-FeSCo se destaca não apenas como um método eficaz para otimização de atributos, mas também como uma ferramenta que transforma a coleta e a análise dos dados de EEG. Ao identificar automaticamente os atributos e canais mais relevantes, possibilita a criação de sistemas mais eficientes, acessíveis e confortáveis, alinhando avanços tecnológicos às necessidades práticas da neurociência e da medicina do sono.

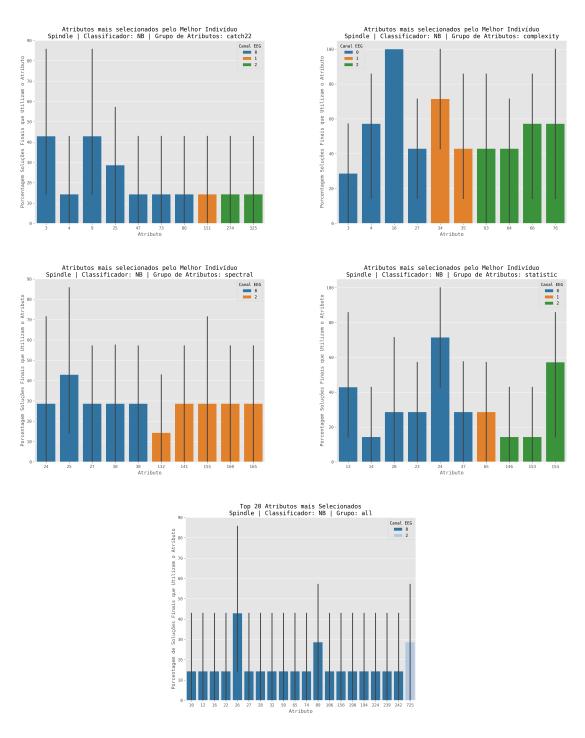


Figura 5.6: Distribuição dos atributos mais selecionados pelo melhor indivíduo para o classificador NB no conjunto de dados *Sleep Spindles*. Cada gráfico representa um grupo de atributos distinto, além de uma visão geral com os 20 atributos mais selecionados considerando todos os grupos de atributos como entrada.

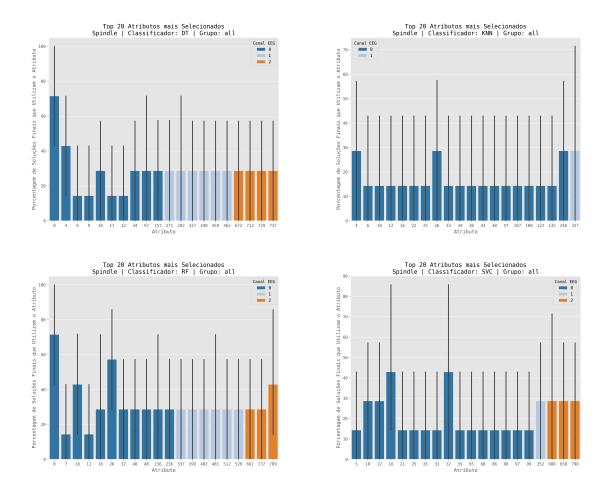


Figura 5.7: Frequência dos 20 atributos mais selecionados pelo melhor indivíduo para os classificadores no conjunto de dados  $Sleep\ Spindles$  ao usar todos os conjuntos de atributos como entrada

O uso de uma representação baseada em árvores, conforme ilustrado nas Figuras 5.8 e 5.9, oferece diversas vantagens no contexto da PG aplicada à engenharia de atributos para EEG. Diferentemente das abordagens baseadas em DL, que atuam como caixaspretas, a representação em árvore permite visualizar diretamente a combinação de atributos para a geração de novas características. Essa visibilidade facilita a análise do comportamento do modelo e sua interpretação por especialistas. A estrutura hierárquica das árvores possibilita a identificação de subconjuntos funcionais (destacados em azul nas figuras), que representam partes reutilizáveis da solução. Esses módulos podem ser analisados individualmente, simplificando a compreensão da lógica desenvolvida pelo GP-FeSCo. Nas Equações 5.1 e 5.2 estão representados os novos atributos artificiais gerados pelo GP-FeSCo. Assim, o primeiro exemplo demonstra a redução de um conjunto de dados de 330 atributos para apenas 3  $(F_1, F_2, F_3)$ , resultantes da combinação de apenas 9 dos atributos originais  $(A_6, A_9, A_{33}, A_{60}, A_{107}, A_{137}, A_{138}, A_{249}, A_{315})$ . Da mesma forma, para o segundo exemplo (Figura 5.9 e Equação 5.2), podemos aplicar a mesma análise. Neste caso, fica evidente uma repetição de atributo, que é removido pelo GP-FeSCo durante a sua execução.

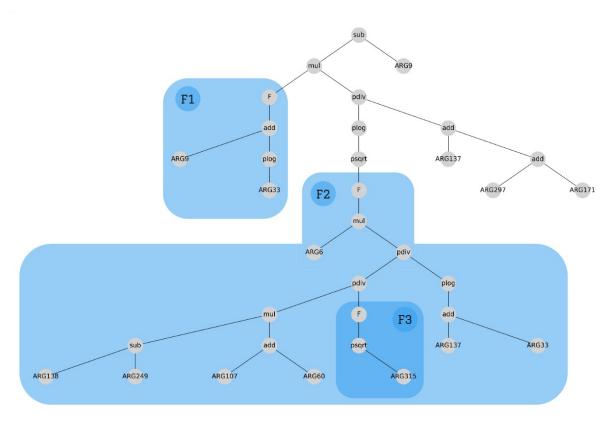


Figura 5.8: Melhor indivíduo gerado pelo GP-FeSCo para o problema de detecção de *Sleep Spindles*, utilizando o classificador DT sobre o conjunto de atributos *catch22*. A árvore representa a expressão evoluída pelo algoritmo, destacando os novos atributos construídos (F1, F2 e F3) que contribuem para a tomada de decisão do modelo.

$$\begin{cases}
F_1 = A_9 + \log(A_{33}), \\
F_2 = A_6 \times \frac{(A_{138} - A_{249})(A_{107} + A_{60})}{\sqrt{A_{315}} \log(A_{137} + A_{33})}, \\
F_3 = \sqrt{A_{315}}
\end{cases} (5.1)$$

A PG é capaz de construir atributos não lineares por meio da combinação de operações matemáticas e estatísticas. Essa capacidade é especialmente útil para sinais de EEG, onde padrões relevantes frequentemente envolvem interações complexas entre diferentes canais e características temporais. O GP-FeSCo não apenas seleciona atributos relevantes, mas também cria novas representações a partir da combinação dos atributos originais. Isso reduz a redundância e aprimora a discriminação entre classes, resultando em classificadores mais eficientes. Além disso, a estrutura em árvore permite que o algoritmo descubra interações entre atributos que não são evidentes em abordagens tradicionais de seleção de características. Essa característica é particularmente relevante em sinais biomédicos, onde relações sutis são críticas para diferenciar estados fisiológicos.

Por fim, a utilização de atributos compostos informativos, por meio da representação em árvore, permite que os modelos de classificação operem com um número reduzido de variáveis. Isso reduz o custo computacional e mitiga o risco de *overfitting*. Essas vantagens tornam essa abordagem altamente eficaz para problemas de classificação de EEG, onde a extração automática de atributos relevantes é um dos principais desafios.

$$\begin{cases} F_{1} = A_{399}, \\ F_{2} = \log \left( \log \left( \log \left( \sqrt{A_{0}} + A_{227} \right) \right) - 2 \log(2) \right), \\ F_{3} = \frac{\log(A_{512})}{2}, \\ F_{4} = \log(A_{789}), \\ F_{5} = A_{48}, \\ F_{6} = \frac{\log(\sqrt{A_{0}} + A_{227})}{4}, \\ F_{7} = \log(A_{352}), \\ F_{8} = A_{48}, \\ F_{9} = A_{26} \end{cases}$$

$$(5.2)$$

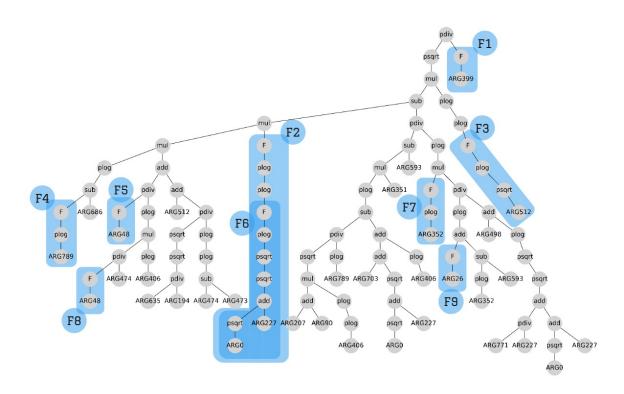


Figura 5.9: Melhor indivíduo gerado pelo GP-FeSCo para o problema de detecção de *Sleep Spindles*, utilizando o classificador RF sobre o conjunto de atributos *all*. A árvore representa a expressão evoluída pelo algoritmo, destacando os novos atributos construídos (F1, ..., F9) que contribuem para a tomada de decisão do modelo.

# 5.1.2 Detecção de Alcoolismo

## Desempenho Geral

A Figura 5.10 apresenta a média obtida na detecção de alcoolismo, calculada com base no conjunto de teste. Diferentemente do que foi observado para  $Sleep\ Spindles$ , não há um incremento consistente no F1-Score ao utilizar o GP-FeSCo. Em cenários específicos, verifica-se uma melhora pontual no desempenho. Entretanto, de modo geral, os valores do F1-Score permanecem semelhantes aos do Baseline.

No entanto, a principal vantagem do uso do GP-FeSCo neste contexto torna-se evidente ao analisarmos, em paralelo, a redução no número de atributos utilizados e controle de overfitting. O método manteve o desempenho dos classificadores e reduziu o número de atributos, tornando os modelos mais enxutos e eficientes. Essa seleção otimizada de atributos é crucial para aplicações que requerem menor custo computacional e maior interpretabilidade.

A Figura 5.11 reforça essa observação ao destacar o desempenho do classificador SVC em distintos grupos de atributos. Embora existam flutuações de desempenho entre os pacientes, o método proporcionou aumento de F1-Score em todas combinações de grupos de pacientes e grupos de atributos.

A Figura 5.12 apresenta a diferença média do F1-Score entre os conjuntos de validação e teste, comparando os cenários baseline e GP-FeSCo. Os resultados indicam que não houve aumento no overfitting, e a diferença entre os conjuntos de validação e teste permaneceu estável. Isso sugere que a otimização realizada pelo GP-FeSCo não resultou em superespecialização dos modelos em relação aos dados de treino, demonstrando uma seleção de atributos mais informativos.

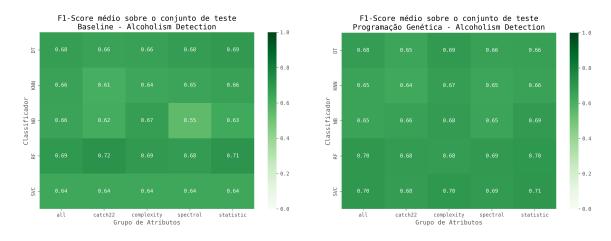
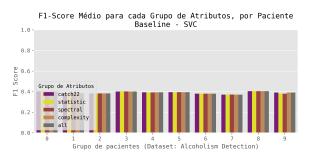


Figura 5.10: Desempenho médio de F1-Score sobre o conjunto de teste para o problema de detecção de alcoolismo nos cenários Baseline (à esquerda) e com a utilização do GP-FeSCo (à direita).



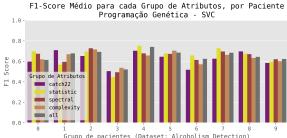
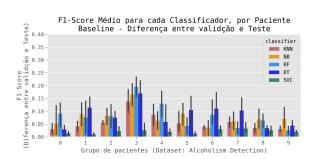


Figura 5.11: Desempenho médio do F1-Score para o classificador SVC aplicado ao conjunto de dados Detecção de Alcoolismo, comparando os cenários Baseline (à esquerda) e GP-FeSCo (à direita).



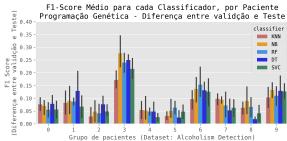


Figura 5.12: Diferença média do F1-Score entre os conjuntos de validação e teste para cada classificador no conjunto de dados Detecção e Alcoolismo, comparando os cenários Baseline (à esquerda) e GP-FeSCo (à direita). Cada barra representa a diferença para um grupo de pacientes (0 a 9).

### Convergência

Assim como no caso anterior e ilustrado na Figura 5.13, em todos os classificadores analisados, a função *fitness* apresenta um crescimento expressivo nas primeiras 20 gerações, seguido por uma fase de estabilização. Esse comportamento sugere que o GP-FeSCo consegue identificar rapidamente padrões relevantes nos dados, ajustando a combinação de atributos de maneira eficiente logo nas primeiras iterações. Novamente, após cerca de 50 gerações, os ganhos de desempenho tornam-se insignificantes, indicando que o algoritmo atingiu um ponto de saturação na otimização. Consequentemente, há um aumento no risco de *overfitting*, uma vez que o modelo pode começar a se ajustar excessivamente às particularidades dos dados de treinamento, podendo comprometer sua capacidade de generalização.

A análise dos atributos otimizados mostra que o GP-FeSCo reduziu significativamente a quantidade de atributos necessários para a classificação, selecionando apenas os mais relevantes. Essa redução é evidenciada na Figura 5.14, onde o número de atributos se estabiliza em um nível inferior aos atributos originais, utilizados pelo *baseline* (Tabela 4.2).

Além disso, a comparação entre os atributos gerados pelo GP-FeSCo e os utilizados no

baseline demonstra que a abordagem evolutiva seleciona subconjuntos mais informativos, reduz redundâncias e melhora a eficiência computacional dos modelos.

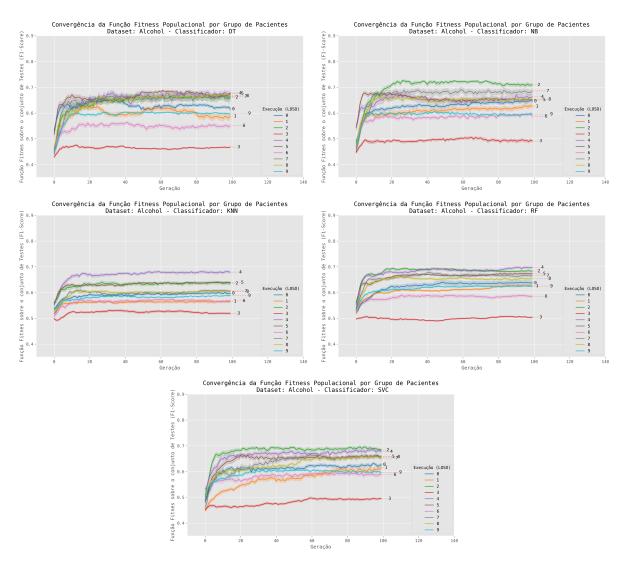
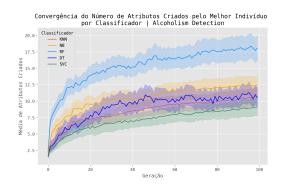


Figura 5.13: Convergência da função fitness ao longo das gerações para diferentes classificadores no conjunto de dados Detecção de Alcoolismo. Cada gráfico representa a evolução do F1-Score populacional médio para um classificador específico avaliado sobre os diferentes pacientes.



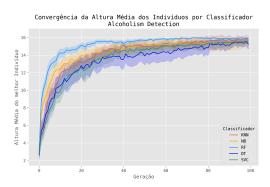


Figura 5.14: Evolução do número de atributos criados (esquerda) e da altura média dos indivíduos (direita) ao longo das gerações para diferentes grupos de atributos no conjunto Detecção de Alcoolismo.

## Interpretabilidade

A análise dos atributos selecionados pelo GP-FeSCo no conjunto de dados de Detecção de Alcoolismo permite identificar quais características dos sinais de EEG são mais relevantes para distinguir sujeitos alcoólatras de não alcoólatras. A Figura 5.15 mostra a distribuição dos atributos mais frequentemente escolhidos pelo melhor indivíduo para os classificadores, considerando o grupo de atributos complexity. É possível observar que certos atributos são selecionados de maneira recorrente, evidenciando sua relevância na discriminação entre os grupos.

Uma das principais vantagens do GP-FeSCo, evidenciada neste conjunto de dados, é a capacidade de identificar os canais mais relevantes para a tarefa de classificação. Canais específicos de EEG aparecem com maior frequência nas melhores soluções encontradas pelo método proposto. Esse comportamento sugere que certas regiões cerebrais podem fornecer informações mais relevantes para a identificação do alcoolismo, alinhando-se a estudos prévios que destacam padrões característicos no córtex frontal e nas regiões associadas ao controle inibitório e à tomada de decisão. Como destacado no caso dos Sleep Spindles, essa informação pode ser utilizada para otimizar a coleta de dados, reduzindo o número de eletrodos necessários e, assim, minimizando o custo e o desconforto do paciente durante a aquisição do sinal. No contexto da detecção de alcoolismo, essa economia de recursos pode ser especialmente vantajosa, uma vez que avaliações desse tipo podem ser realizadas em diversos ambientes clínicos, não necessitando de laboratórios especializados em neurociência.

Outro aspecto importante a ser destacado é a relação entre os atributos selecionados e a estabilidade do modelo. A identificação de um subconjunto otimizado de atributos permite melhorar a interpretabilidade dos modelos e reduzir a complexidade computacional, sem perda significativa de desempenho. A comparação com o baseline reforça que o GP-FeSCo

não apenas seleciona atributos informativos, mas também reduz redundâncias, resultando em modelos mais enxutos e eficazes.

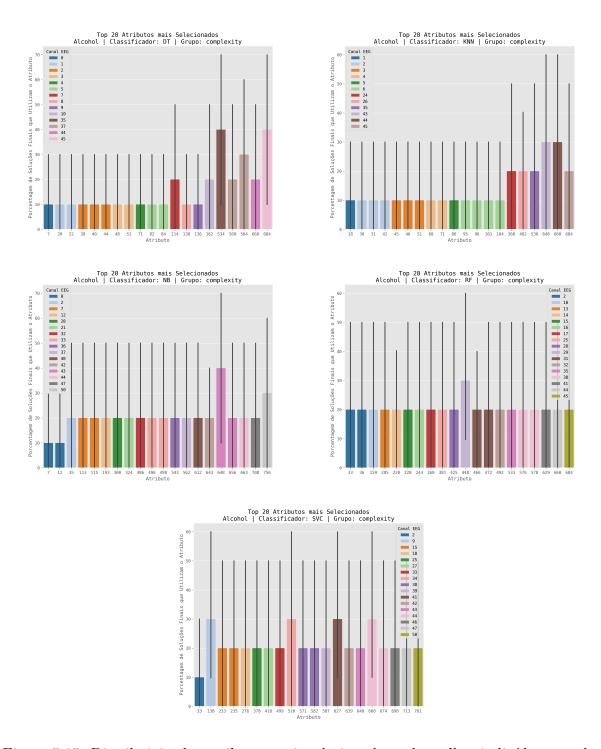


Figura 5.15: Distribuição dos atributos mais selecionados pelo melhor indivíduo por classificador no conjunto de dados Detecção de Alcoolismo sobre o conjunto de atributos complexity

.

# 5.1.3 Imagética Motora

### Desempenho Geral

A Figura 5.16 mostra o desempenho médio do F1-Score no conjunto de teste na tarefa de Imagética Motora, comparando os cenários baseline (à esquerda) e GP-FeSCo (à direita). Observa-se que a utilização do GP-FeSCo resultou em um aumento significativo no desempenho da classificação, com melhorias expressivas para todos os classificadores e grupos de atributos. O aumento do F1-Score é particularmente evidente para os classificadores RF e SVC, que apresentaram os maiores ganhos quando combinados com o grupo de atributos complexity e catch22.

A Figura 5.17 apresenta a análise detalhada do desempenho dos classificadores para diferentes grupos de pacientes. A aplicação do GP-FeSCo resultou em melhorias na estabilidade do desempenho entre os diferentes grupos, reduzindo a variabilidade observada no cenário *baseline*. Isso sugere que a construção automática de atributos ajudou a identificar representações mais discriminativas dos sinais de EEG, tornando os classificadores mais resistentes à variabilidade inter-paciente.

A Figura 5.18 ilustra a diferença entre os conjuntos de validação e teste, comparando os cenários baseline e GP-FeSCo. Observa-se que, apesar do aumento na precisão dos modelos, a diferença entre os conjuntos de validação e teste permaneceu estável, indicando que o GP-FeSCo não aumentou o overfitting. Esse resultado é crucial para garantir que as melhorias no desempenho dos modelos sejam genuínas e não decorram de uma especialização excessiva nos dados de treinamento.

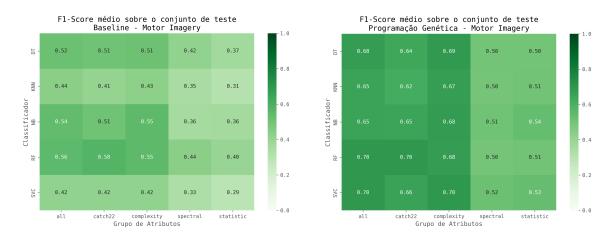
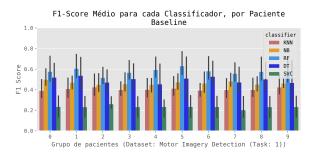


Figura 5.16: Desempenho médio de F1-Score sobre o conjunto de teste para o problema de Imagética Motora nos cenários Baseline (à esquerda) e com a utilização do GP-FeSCo (à direita).



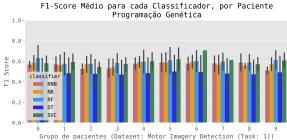
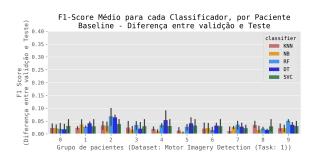


Figura 5.17: Desempenho médio do *F1-Score* por classificador aplicado ao conjunto de dados Imagética Motora, comparando os cenários *Baseline* (à esquerda) e GP-FeSCo (à direita).



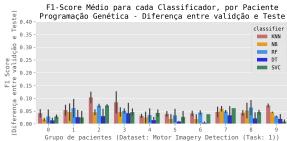


Figura 5.18: Diferença média do F1-Score entre os conjuntos de validação e teste para cada classificador no conjunto de dados Imagética Motora, comparando os cenários Baseline (à esquerda) e GP-FeSCo (à direita). Cada barra representa a diferença para um grupo de pacientes (0 a 9).

### Convergência

A convergência do GP-FeSCo foi avaliada pela evolução da função fitness ao longo das gerações, conforme ilustrado na Figura 5.19. Para todos os classificadores avaliados, observa-se um rápido aumento da função fitness nas primeiras 20 gerações, seguido de um período de estabilização, comportamento já observado nos outros dois conjuntos de dados. No entanto, surge uma diferença importante ao analisarmos a taxa de crescimento da fitness: no caso da Imagética Motora, a melhora no desempenho ao longo das gerações foi mais gradual e menos acentuada em comparação com os conjuntos de dados de Fusos do Sono e Detecção de Alcoolismo.

Essa diferença pode ser atribuída à maior complexidade da tarefa de Imagética Motora, na qual os padrões de ativação cerebral associados ao movimento imaginado são mais sutis e difíceis de capturar por atributos tradicionais. Para Sleep Spindles e Detecção de Alcoolismo, a evolução do GP-FeSCo identificou rapidamente características relevantes. Em contraste, na Imagética Motora, o processo de otimização exigiu um maior número de gerações para construir representações mais discriminativas.

Além disso, conforme ilustrado na Figura 5.20, a evolução do número de atributos

gerados e da altura média dos indivíduos apresenta comportamentos distintos neste conjunto de dados. Diferentemente dos conjuntos de *Sleep Spindles* e Detecção de Alcoolismo, onde o número médio de atributos se estabilizou em gerações mais iniciais, na Imagética Motora observa-se um crescimento contínuo ao longo de um maior número de gerações, especialmente para os classificadores RF e SVC. Esse comportamento sugere que, neste caso, o GP-FeSCo explorou um espaço de atributos mais amplo antes de convergir para uma solução otimizada.

Outra distinção relevante é que, enquanto nos outros conjuntos de dados a altura média das árvores estabilizou rapidamente, na Imagética Motora, observa-se um crescimento mais expressivo da profundidade das árvores evoluídas. Isso sugere que, para capturar padrões mais complexos da Imagética Motora, o algoritmo evolutivo precisou desenvolver expressões mais elaboradas, incorporando interações não lineares entre os atributos.

Em resumo, a convergência do GP-FeSCo no conjunto de Imagética Motora apresentou um comportamento distinto em relação aos conjuntos de Fusos do Sono e Detecção de Alcoolismo. A evolução foi mais lenta, demandando estruturas de atributos mais complexas, o que reflete a natureza desafiadora desta tarefa. Esses resultados indicam que, embora o GP-FeSCo tenha demonstrado eficácia na construção de atributos relevantes para este conjunto de dados, o processo de otimização requer maior flexibilidade e um número superior de gerações para alcançar soluções satisfatórias.

#### Interpretabilidade

A análise dos atributos otimizados indica que o GP-FeSCo selecionou predominantemente atributos do grupo *complexity*. Isso sugere que medidas de não-linearidade e variabilidade dos sinais são fundamentais para a diferenciação das classes. Esse resultado corrobora estudos anteriores [122], que demonstraram que a atividade motora imaginada apresenta padrões complexos, dificultando a captura por estatísticas tradicionais de séries temporais.

Além disso, a comparação entre os atributos gerados pelo GP-FeSCo e os utilizados no baseline indica que a abordagem evolutiva eliminou atributos redundantes, mantendo apenas os mais informativos para a classificação. Isso reforça o potencial do GP-FeSCo não apenas como um método de seleção de atributos, mas também como uma ferramenta para compreender melhor as características subjacentes dos sinais de EEG.

A Figura 5.21 ilustra a distribuição dos atributos mais frequentemente escolhidos pelo melhor indivíduo para diferentes classificadores, considerando os grupos de atributos catch22 e spectral. Observa-se que, além da seleção recorrente de atributos específicos nesses grupos, há uma concentração significativa de atributos associados a determinados canais de EEG. Isso reforça a hipótese de que certas regiões do cérebro desempenham um

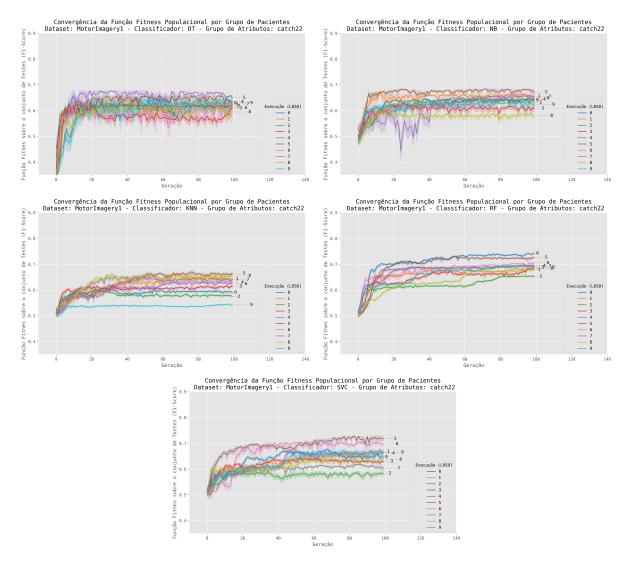
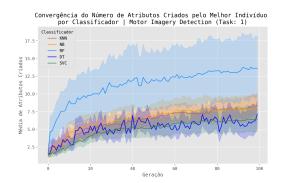


Figura 5.19: Convergência da função fitness ao longo das gerações para diferentes classificadores no conjunto de dados Imagética Motora. Cada gráfico representa a evolução do F1-Score populacional médio para um classificador específico avaliado sobre os diferentes pacientes.

papel mais relevante na diferenciação dos padrões de Imagética Motora, implicando na otimização da aquisição de dados e na configuração dos sistemas de classificação.

A identificação dos canais mais informativos representa uma vantagem significativa do GP-FeSCo, pois permite reduzir o número de eletrodos necessários para a coleta de sinais EEG. Essa abordagem se diferencia de conjuntos de dados anteriores, uma vez que, na Imagética Motora, a distribuição dos canais mais relevantes é heterogênea, indicando a participação de diversas regiões cerebrais no processamento do movimento imaginado. A Figura 5.22 ilustra um indivíduo criado com base no grupo de atributos *catch22* para o classificador SVC. Neste contexto, a flexibilidade do GP-FeSCo permite a seleção direta de atributos, bem como a criação de novos com complexidade variada, conforme evidenciado



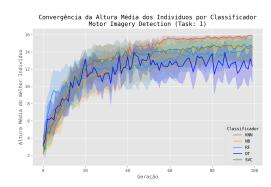


Figura 5.20: Evolução do número de atributos criados (esquerda) e da altura média dos indivíduos (direita) ao longo das gerações para diferentes grupos de atributos no conjunto Imagética Motora.

na Equação 5.3. Na equação, os atributos riscados representam repetições de atributos no mesmo indivíduo. Como apresentado no Capítulo 3, repetições são eliminadas antes do treinamento do classificador. Entretanto, uma análise para trabalhos futuros é entender se essas repetições podem ser interpretadas como uma medida de importância dos atributos criados.

Os resultados obtidos para a Imagética Motora indicam que o GP-FeSCo melhorou significativamente o desempenho da classificação, sem elevar o risco de *overfitting*, além de manter um conjunto de atributos enxuto e interpretável. Esses achados ressaltam a aplicabilidade do método em cenários biomédicos, onde a interpretabilidade das decisões do modelo é tão importante quanto a precisão.

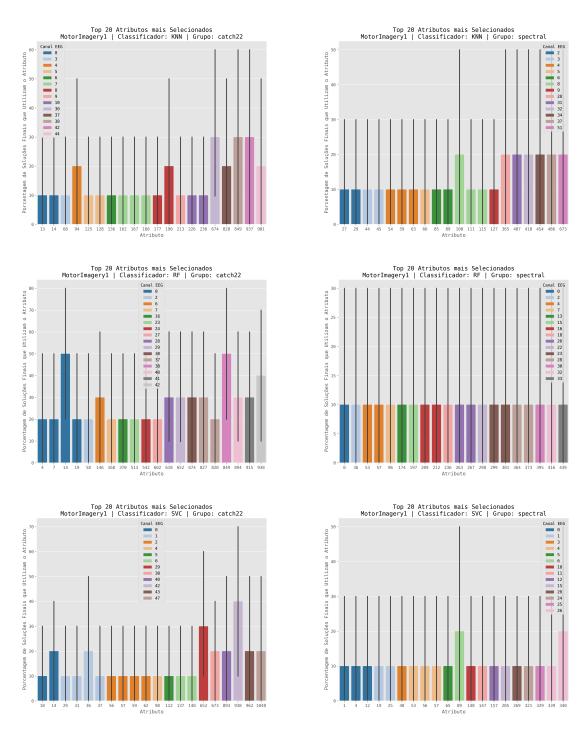


Figura 5.21: Distribuição dos atributos mais selecionados pelo melhor indivíduo por classificador no conjunto de dados Imagética Motora sobre os conjunto de atributos catch22 e spectral.

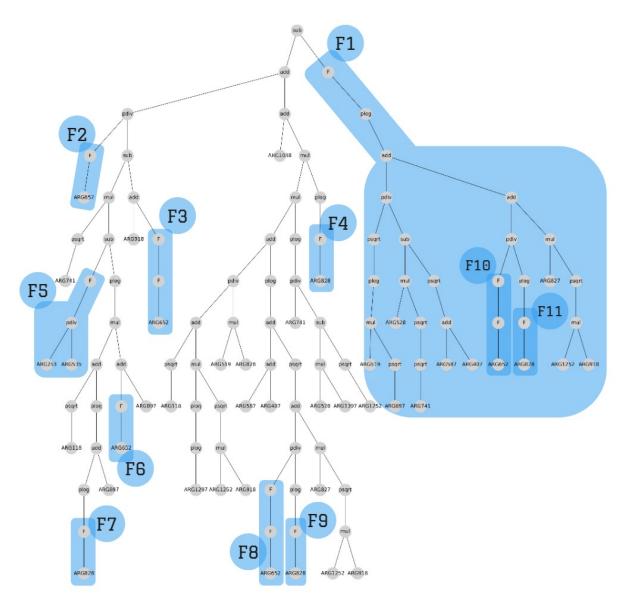


Figura 5.22: Melhor indivíduo gerado pelo GP-FeSCo para o problema de detecção de Imagética Motora, utilizando o classificador SVC sobre o conjunto de atributos *catch22*. A árvore representa a expressão evoluída pelo algoritmo, destacando os novos atributos construídos (F1, ..., F11) que contribuem para a tomada de decisão do modelo.

$$\begin{cases} F_{1} = \log\left(\frac{A_{652}}{\log(A_{828})} + A_{827} \cdot \sqrt{A_{1252} \cdot A_{918}} + \frac{\sqrt{\log(A_{519} \cdot \sqrt{A_{897}})}}{A_{528} \cdot A_{741}^{1/4} - \sqrt{A_{407} + A_{587}}} \right) \\ F_{2} = A_{652}, \\ F_{3} = A_{652}, \\ F_{4} = A_{828}, \\ F_{5} = \frac{A_{253}}{A_{535}}, \\ F_{6} = A_{652}, \\ F_{7} = A_{828}, \\ F_{8} = A_{652}, \\ F_{9} = A_{828}, \\ F_{10} = A_{652}, \\ F_{11} = A_{828} \end{cases}$$

$$(5.3)$$

## 5.2 Discussão Geral dos Resultados

Os resultados apresentados neste capítulo evidenciam a eficácia do GP-FeSCo na seleção e construção de atributos para diversas tarefas de classificação de sinais de EEG. A abordagem proposta foi avaliada em três conjuntos de dados distintos: Sleep Spindles, Detecção de Alcoolismo e Imagética Motora. Cada conjunto apresenta desafios específicos, como a variabilidade dos sinais, a complexidade da modelagem e o impacto dos atributos na performance dos classificadores. A validação LOSO garantiu uma avaliação rigorosa dos modelos, simulando um cenário mais realista, no qual os classificadores precisam generalizar para sujeitos não vistos anteriormente.

A análise dos resultados revelou diferenças significativas entre as aplicações. No caso dos Sleep Spindles, a utilização do GP-FeSCo resultou em um aumento consistente no F1-score, indicando que a construção evolutiva de atributos extraiu informações relevantes para a classificação. Os atributos do grupo catch22 e as métricas de complexidade foram frequentemente selecionados, reforçando sua importância na discriminação entre segmentos de EEG com e sem fusos do sono. Além disso, a convergência do algoritmo ocorreu rapidamente, estabilizando a função fitness e o número médio de atributos em poucas gerações. Esse comportamento sugere que, para esta tarefa, a PG identificou de forma ágil um conjunto eficiente de atributos, possibilitando a construção de modelos mais interpretáveis e com melhor desempenho.

Em contraste, na detecção de alcoolismo, a principal vantagem do GP-FeSCo não foi o aumento expressivo no desempenho dos classificadores, mas sim a redução da dimensiona-

Tabela 5.1: Comparação de Resultados da Literatura

Base de Dados	Referência	Folds (Validação Cruzada)	LOSO	Acurácia	F1-Score
Detecção de Alcoolismo	Zhu, Guohun, et al.(2014) [104]	10 Fold	Não	98%	-
	Bavkar, Sandeep, et al.(2019) [106]	10 Fold	Não	93%	-
	Mukhtar, Hamid, et al.(2021) [105]	10 Fold	Não	98%	98.4%
	GP-FeSCo	10 Fold	Sim	78%	75%
Imagética Motora	Wang, Xiaying, et al.(2020) [97]	5 Fold	Sim	87.49%	-
	Taşar, Beyda, et al.(2022) [98]	Não informado	Não	99.51%	99.51%
	GP-FeSCo	10 Fold	Sim	76%	76%
Sleep Spindle	Lachner-Piza, Daniel, et al.(2018) [101]	Train/Test	Não	95%	43%
	Chen, Peilu, et al.(2021) [102]	8 Fold	Sim	-	66%
	Shimizu, Riki, et al.(2024) [123]	8 Fold	Sim	-	74.9%
	Wang, Fei, et al.(2023) [124]	8 Fold	Sim	94.7%	55.4%
	GP-FeSCo	8 Fold	Sim	88%	84%

lidade dos atributos utilizados, sem comprometer o F1-score. A manutenção da acurácia com um menor número de atributos indica que a abordagem evolutiva conseguiu eliminar características redundantes, tornando os modelos mais eficientes e interpretáveis. Além disso, não houve indícios de aumento do overfitting, sugerindo que a seleção de atributos realizada pela PG contribuiu para o desenvolvimento de um classificador mais simples, sem favorecer uma especialização excessiva nos dados de treino. Esse achado é particularmente relevante para aplicações biomédicas, onde a complexidade computacional pode ser um fator limitante e a interpretabilidade dos modelos é essencial para a aceitação clínica.

A Imagética Motora revelou um cenário desafiador. Diferentemente dos outros dois conjuntos de dados, a evolução do F1-score foi mais gradual e menos acentuada, refletindo a complexidade da tarefa de distinguir padrões cerebrais associados ao movimento imaginado. A convergência do GP-FeSCo foi mais lenta, exigindo um maior número de gerações para estabilização. Isso sugere que a estrutura dos atributos relevantes é mais complexa e difícil de modelar. Além disso, ao contrário do observado nos demais con-

juntos de dados, o número médio de atributos continuou a crescer por um período mais extenso antes de se estabilizar. As árvores geradas pelo algoritmo apresentaram maior profundidade. Esse comportamento indica que, para a Imagética Motora, a extração de atributos eficazes exige expressões matemáticas mais elaboradas, incorporando interações não lineares que capturam variações sutis nos sinais de EEG.

Esses achados ressaltam algumas características principais do GP-FeSCo. A capacidade de construir atributos não lineares foi crucial para a tarefa de *Sleep Spindles*, na qual padrões complexos precisavam ser diferenciados com precisão. A redução da dimensionalidade mostrou-se especialmente benéfica na detecção de alcoolismo, evidenciando que o método pode ser aplicado eficazmente para selecionar subconjuntos informativos de atributos, sem comprometer o desempenho dos classificadores. A imagética motora, por sua vez, evidenciou a necessidade de adaptações no processo evolutivo para lidar com tarefas mais complexas, sugerindo que ajustes nos hiperparâmetros da programação genética poderiam acelerar a convergência e otimizar a construção de atributos para essa aplicação específica.

Outro aspecto relevante nas análises é o desempenho do GP-FeSCo em relação à generalização dos modelos. A utilização da validação LOSO assegurou que os classificadores fossem avaliados de forma mais rigorosa do que em abordagens tradicionais baseadas em K-fold. Esse ponto é especialmente importante no contexto dos sinais de EEG, onde a variabilidade entre sujeitos pode inflacionar métricas quando os modelos são testados em dados já conhecidos. Os resultados demonstram que, embora a validação LOSO tenha reduzido ligeiramente os valores absolutos do F1-score em comparação a estudos anteriores, essa abordagem proporcionou uma avaliação mais realista e confiável da capacidade de generalização dos modelos para novos indivíduos.

A Tabela 5.1 apresenta uma comparação do desempenho do GP-FeSCo em relação a outros estudos da literatura, ressaltando o impacto da validação LOSO e a eficácia do método na otimização de atributos. Observa-se que, nos estudos que empregaram K-fold, os valores de acurácia são consistentemente mais altos. No entanto, conforme discutido anteriormente, esse método pode superestimar a performance dos modelos devido à sobreposição de dados entre os conjuntos de treinamento e teste. Em contraste, a utilização do LOSO permitiu uma avaliação mais rigorosa da capacidade de generalização dos classificadores, mitigando possíveis vieses e assegurando um cenário mais próximo do uso real em aplicações biomédicas.

No conjunto de dados de *Sleep Spindles*, o GP-FeSCo alcançou um *F1-score* de 84%, superando todos os métodos analisados que também utilizaram LOSO. Em comparação, os trabalhos de Chen et al. [102] e Shimizu et al. [123] obtiveram *F1-scores* de 66% e 74.9%, respectivamente. Essa diferença destaca o impacto da construção evolutiva de

atributos na melhoria da capacidade discriminativa dos modelos, demonstrando que o GP-FeSCo conseguiu extrair informações mais relevantes dos sinais de EEG. Além disso, ao contrário do estudo de Lachner-Piza et al. [101], que utilizou um esquema de divisão train/test e reportou uma acurácia de 95%, o GP-FeSCo apresentou uma avaliação mais conservadora, mas mais realista, ao adotar o LOSO.

Na detecção de alcoolismo, estudos prévios apresentam acurácias elevadas, superiores a 93%. Entretanto, todos esses trabalhos utilizaram K-fold como método de validação, o que pode ter gerado estimativas inflacionadas. O GP-FeSCo, ao empregar LOSO, alcançou uma acurácia de 78% e um F1-score de 75%. Ademais, a abordagem proposta reduziu significativamente o número de atributos utilizados, tornando os modelos mais eficientes e interpretáveis, sem comprometer a performance.

Na Imagética Motora, os resultados mostram um cenário semelhante. O estudo de Taşar et al.[98], que não especifica o método de validação, reportou uma acurácia extremamente alta de 99,51%, sugerindo possível viés decorrente da divisão inadequada dos dados. Por sua vez, Wang et al.[97], que empregou LOSO, apresentou uma acurácia de 87,49%, mais próxima do valor obtido pelo GP-FeSCo, que alcançou 76%. Esse resultado reforça a importância da validação LOSO para assegurar que os modelos sejam efetivamente avaliados em novos sujeitos, evitando métricas artificialmente elevadas.

Os achados deste capítulo indicam que o GP-FeSCo é uma abordagem promissora para a análise de sinais de EEG, combinando a flexibilidade da PG com estratégias eficazes de seleção de atributos. No entanto, algumas limitações foram identificadas. O tempo de execução do algoritmo, embora não tenha sido explorado neste estudo, pode ser um fator limitante em aplicações em tempo real devido à natureza evolutiva da PG. Adicionalmente, a taxa de convergência variou significativamente entre os diferentes conjuntos de dados, sugerindo a necessidade de ajustes específicos na parametrização, dependendo da tarefa. Trabalhos futuros podem explorar técnicas de paralelização para acelerar a evolução dos atributos e investigar ajustes automáticos dos hiperparâmetros, visando otimizar a eficiência do processo de busca.

De forma geral, os resultados obtidos reforçam o potencial do GP-FeSCo como uma alternativa viável para otimizar classificadores tradicionais, aumentando sua eficiência e interpretabilidade. A aplicação bem-sucedida do método em diversos domínios sugere sua extensão a outras tarefas biomédicas, como diagnósticos assistidos por inteligência artificial e análise de sinais fisiológicos. Assim, este estudo contribui para a exploração de abordagens baseadas em Programação Genética na modelagem e interpretação de sinais de EEG, abrindo caminho para investigações futuras voltadas ao aprimoramento da eficiência e aplicabilidade do método.

# Capítulo 6

# Conclusão

Neste estudo, desenvolvemos e avaliamos o framework Genetic Programming-Based Feature Selection and Construction (GP-FeSCo) para a seleção e construção de atributos em tarefas de classificação de sinais de Eletroencefalograma (EEG), explorando sua aplicabilidade em três conjuntos de dados distintos: Detecção de Alcoolismo, Imagética Motora e Sleep Spindles. O principal objetivo foi investigar a eficácia do GP-FeSCo na criação de atributos otimizados e interpretáveis, assegurando avaliações robustas por meio da validação cruzada Leave-One-Subject-Out (LOSO). Essa abordagem garantiu que os modelos fossem testados em pacientes ainda não analisados, refletindo de forma mais precisa as condições reais de aplicação.

A utilização da estratégia LOSO é especialmente relevante em tarefas baseadas em EEG, pois características individuais dos sujeitos podem levar à superestimação do desempenho quando a separação entre treino e teste não é realizada corretamente. Trabalhos anteriores que utilizaram validação K-fold frequentemente relataram acurácias inflacionadas devido à sobreposição de dados dos mesmos pacientes nas etapas de treino e teste. Esse problema é contornado pelo GP-FeSCo, que assegura que cada iteração avalie exclusivamente novos indivíduos. Os resultados apresentados na Tabela 5.1 evidenciam essa diferença, mostrando que, embora LOSO possa gerar métricas ligeiramente inferiores em comparação com métodos tradicionais de validação, oferece estimativas mais realistas e generalizáveis do desempenho dos modelos.

A aplicação do GP-FeSCo apresentou desempenho competitivo em diversas tarefas de classificação de EEG, alcançando 76% no conjunto de dados de Imagética Motora e 84% no conjunto de Fusos do Sono. Esses resultados evidenciam sua capacidade de adaptação a diferentes domínios de EEG. Além do desempenho preditivo, um dos principais diferenciais do GP-FeSCo é a habilidade de construir modelos interpretáveis que combinam atributos de forma não-linear e identificam características relevantes para a classificação. Essa interpretabilidade é crucial em aplicações biomédicas, pois a compreensão dos padrões

extraídos auxilia os especialistas na tomada de decisões.

Outro aspecto fundamental do GP-FeSCo foi a capacidade de reduzir a dimensionalidade, essencial para aprimorar o desempenho de modelos de AM tradicionais. A técnica baseada em PG eliminou atributos redundantes, preservando apenas as características mais relevantes. Isso permitiu que modelos como KNN, NB e SVM alcançassem resultados competitivos com um número reduzido de atributos. Essa redução resultou em ganhos de eficiência computacional e melhorias na generalização dos modelos, evitando o risco de overfitting. No conjunto de Sleep Spindles, por exemplo, o GP-FeSCo superou abordagens anteriores que utilizavam conjuntos de atributos maiores, reforçando a eficácia da otimização realizada.

Além disso, a Figura 5.10 demonstra que, no caso da Detecção de Alcoolismo, embora o GP-FeSCo não tenha promovido um aumento expressivo no F1-score em comparação ao baseline, houve uma redução significativa no número de atributos utilizados. Essa redução não comprometeu o desempenho e evitou um aumento de overfitting. Isso evidencia que, mesmo em situações onde o ganho de precisão não é notável, o método proporciona benefícios importantes em eficiência e complexidade, tornando os modelos mais enxutos e interpretáveis.

Uma das vantagens mais relevantes do GP-FeSCo é a identificação dos canais de EEG significativos para a classificação, por meio da análise da frequência de uso de cada canal na construção de atributos discriminativos. Essa capacidade impacta a otimização da aquisição de sinais EEG, reduzindo a complexidade do exame ao permitir a coleta de dados de um subconjunto menor de eletrodos. Além de diminuir custos e a carga computacional associada ao processamento dos sinais, essa abordagem aumenta o conforto do paciente, tornando o exame mais acessível. Em aplicações clínicas, onde a duração e o desconforto na aquisição de sinais são críticos, a identificação dos canais mais informativos possibilita soluções mais eficientes para o monitoramento e diagnóstico.

Além das aplicações específicas em EEG, a abordagem utilizada pelo GP-FeSCo pode ser estendida a outros domínios de biossinais, como ECG e EMG, assim como a tarefas gerais de AM em dados tabulares. A PG é capaz de construir atributos personalizados e interpretar padrões relevantes do conjunto de dados, beneficiando qualquer aplicação onde a seleção de características é crucial para o sucesso do modelo. No contexto de biossinais, essa habilidade auxilia na identificação de padrões clínicos relevantes, reduzindo a necessidade de engenharia manual de atributos e tornando os sistemas mais automatizados e eficientes.

Em resumo, o GP-FeSCo demonstrou ser uma ferramenta poderosa para a construção de atributos, redução de dimensionalidade e melhoria da interpretabilidade dos modelos de aprendizado de máquina aplicados à análise de EEG. A validação LOSO assegurou que

os modelos fossem avaliados em condições realistas, aumentando sua confiabilidade para aplicações práticas. Esses achados reforçam o potencial dos métodos baseados em PG para o desenvolvimento de modelos generalistas, eficientes e interpretáveis, contribuindo para avanços na análise automatizada de sinais biomédicos. Adicionalmente, a identificação dos canais mais relevantes para a classificação de padrões de EEG representa um avanço significativo na otimização dos processos de aquisição de sinais, proporcionando benefícios tanto para pesquisadores quanto para aplicações clínicas, onde a eficiência e o conforto do paciente são essenciais. Por fim, a flexibilidade do GP-FeSCo sugere que sua aplicabilidade pode se estender a uma ampla gama de problemas em aprendizado de máquina, desde a análise de biossinais até a modelagem de dados tabulares complexos, tornando-o uma ferramenta versátil e promissora para diversas áreas do conhecimento.

# 6.1 Limitações e Trabalhos Futuros

Embora o *framework* proposto, GP-FeSCo, tenha demonstrado resultados promissores na seleção e construção de atributos para classificação de sinais de EEG, algumas limitações precisam ser consideradas.

Primeiro, o custo computacional da abordagem ainda é elevado, especialmente em cenários com grandes volumes de dados ou múltiplas janelas temporais. Apesar das otimizações empregadas — como o uso de feature cache e a remoção de atributos altamente correlacionados — o tempo de execução pode ser um obstáculo para aplicações em tempo real ou com recursos computacionais limitados.

Adicionalmente, o processo evolutivo depende de uma configuração fixa de hiperparâmetros da PG, o que pode limitar sua adaptabilidade a diferentes bases ou domínios. Pesquisas futuras podem explorar a adaptação automática desses hiperparâmetros durante a execução, otimizando tanto o tempo de convergência quanto a qualidade das soluções geradas.

A estrutura baseada em árvores favorece a interpretabilidade, mas impõe limitações na diversidade de atributos que podem ser construídos. A inclusão de operadores mais expressivos, como convoluções, filtros parametrizados ou transformações no domínio tempofrequência, pode ampliar o espaço de busca sem comprometer a interpretabilidade.

Além disso, estudos mais aprofundados sobre os indivíduos gerados pelo GP-FeSCo podem contribuir para sua compreensão e aprimoramento. Por exemplo, análises estatísticas sobre a recorrência de subestruturas na população podem ser utilizadas para inferir a importância dos atributos construídos. Paralelamente, estratégias de simplificação simbólica podem ser incorporadas para controlar o inchaço (bloat), reduzindo redundâncias e melhorando a eficiência computacional.

Por fim, embora este trabalho tenha se concentrado em sinais de EEG, a arquitetura do GP-FeSCo é passível de adaptação para outros tipos de séries temporais e dados tabulares. Isso amplia as possibilidades de aplicação em diferentes domínios do AM, como finanças, indústria, medicina e ciências ambientais, onde a construção automática e interpretável de atributos é igualmente desejável.

A incorporação de estratégias multiobjetivo no processo evolutivo do GP-FeSCo representa uma direção promissora para aprimorar a qualidade das soluções geradas. Atualmente, a função de aptidão do algoritmo considera predominantemente o desempenho do classificador (por exemplo, F1-score). No entanto, em cenários práticos, outros critérios também são relevantes, como a interpretabilidade dos atributos construídos e o custo computacional associado à avaliação dos indivíduos. Estratégias multiobjetivo permitem que múltiplos critérios sejam otimizados simultaneamente, promovendo o equilíbrio entre acurácia, simplicidade estrutural das árvores (como profundidade, número de nós ou complexidade simbólica) e tempo de execução. Essa abordagem pode ser implementada por meio de algoritmos evolutivos específicos, como NSGA-II ou SPEA2 [125, 126], que mantêm um conjunto de soluções não-dominadas (fronteira de Pareto) ao longo da execução. Com isso, o usuário pode selecionar soluções de acordo com restrições ou preferências específicas da aplicação, como priorizar modelos mais rápidos e interpretáveis em detrimento de pequenas perdas de desempenho. Além disso, o uso de métricas explícitas de complexidade pode evitar o inchaço estrutural indesejado sem recorrer a limites artificiais de profundidade, incentivando a geração de soluções mais eficientes e práticas para aplicações em domínios sensíveis, como sistemas embarcados ou ambientes clínicos.

Essas direções visam ampliar a aplicabilidade e desempenho do GP-FeSCo, contribuindo para o avanço de soluções automatizadas e interpretáveis em AM.

# Referências

- [1] Kunjan, Sajeev, Tyler S Grummett, Kenneth J Pope, David MW Powers, Sean P Fitzgibbon, T Bastiampillai, M Battersby e Trent W Lewis: The necessity of leave one subject out (loso) cross validation for eeg disease diagnosis. Em Brain Informatics: 14th International Conference, BI 2021, Virtual Event, September 17–19, 2021, Proceedings 14, páginas 558–567. Springer, 2021. xiv, 2, 20, 21, 22, 35
- [2] Lubba, Carl H, Sarab S Sethi, Philip Knaute, Simon R Schultz, Ben D Fulcher e Nick S Jones: catch22: Canonical time-series characteristics. Data Mining and Knowledge Discovery, 33(6):1821–1852, 2019. xiv, 35, 38, 39, 46
- [3] Carbonell, Jaime G, Ryszard S Michalski e Tom M Mitchell: An overview of machine learning. Machine learning, páginas 3–23, 1983. 1
- [4] Simon, Phil: Too big to ignore: the business case for big data, volume 72. John Wiley & Sons, 2013. 1
- [5] Wang, Hua, Cuiqin Ma e Lijuan Zhou: A brief review of machine learning and its application. Em 2009 international conference on information engineering and computer science, páginas 1–4. IEEE, 2009. 1
- [6] Ribeiro, Mauro, Katarina Grolinger e Miriam AM Capretz: Mlaas: Machine learning as a service. Em 2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA), páginas 896–902. IEEE, 2015. 1
- [7] Mahesh, Batta: *Machine learning algorithms-a review*. International Journal of Science and Research (IJSR).[Internet], 9:381–386, 2020. 1
- [8] Rashid, Mamunur, Norizam Sulaiman, Anwar PP Abdul Majeed, Rabiu Muazu Musa, Ahmad Fakhri Ab Nasir, Bifta Sama Bari e Sabira Khatun: Current status, challenges, and possible solutions of eeg-based brain-computer interface: A comprehensive review. Frontiers in neurorobotics, 14, 2020. 1, 8, 9
- [9] Ng, Andrew Y et al.: Preventing" overfitting" of cross-validation data. Em ICML, volume 97, páginas 245–253. Citeseer, 1997. 1
- [10] Ying, Xue: An overview of overfitting and its solutions. Em Journal of physics: Conference series, volume 1168, página 022022. IOP Publishing, 2019. 1
- [11] Tang, Jiliang, Salem Alelyani e Huan Liu: Feature selection for classification: A review. Data classification: Algorithms and applications, página 37, 2014. 1

- [12] Li, Jundong, Kewei Cheng, Suhang Wang, Fred Morstatter, Robert P Trevino, Jiliang Tang e Huan Liu: Feature selection: A data perspective. ACM computing surveys (CSUR), 50(6):1–45, 2017. 1
- [13] Zheng, Alice e Amanda Casari: Feature engineering for machine learning: principles and techniques for data scientists. "O'Reilly Media, Inc.", 2018. 1
- [14] Khalid, Samina, Tehmina Khalil e Shamila Nasreen: A survey of feature selection and feature extraction techniques in machine learning. Em 2014 science and information conference, páginas 372–378. IEEE, 2014. 1
- [15] Guyon, Isabelle, Steve Gunn, Masoud Nikravesh e Lofti A Zadeh: Feature extraction: foundations and applications, volume 207. Springer, 2008. 1
- [16] Duboue, Pablo: The art of feature engineering: essentials for machine learning. Cambridge University Press, 2020. 1
- [17] Kevric, Jasmin e Abdulhamit Subasi: Comparison of signal decomposition methods in classification of eeg signals for motor-imagery bci system. Biomedical Signal Processing and Control, 31:398–406, 2017. 1
- [18] Motamedi-Fakhr, Shayan, Mohamed Moshrefi-Torbati, Martyn Hill, Catherine M Hill e Paul R White: Signal processing techniques applied to human sleep eeg signals—a review. Biomedical Signal Processing and Control, 2014. 2, 18, 24, 35, 36, 37, 38
- [19] Bontempi, Gianluca, Souhaib Ben Taieb e Yann Aël Le Borgne: Machine learning strategies for time series forecasting. Em European business intelligence summer school, páginas 62–77. Springer, 2012. 2
- [20] Zöller, Marc André e Marco F Huber: Benchmark and survey of automated machine learning frameworks. Journal of Artificial Intelligence Research, 70:409–472, 2021.
- [21] Hutter, Frank, Lars Kotthoff e Joaquin Vanschoren: Automated Machine Learning. Springer, 2019. 2
- [22] Poli, Riccardo, William B. Langdon e Nicholas Freitag McPhee: A field guide to genetic programming. 2008. http://www.gp-field-guide.org.uk, (With contributions by J. R. Koza). 2, 11, 14, 15
- [23] Koza, John R: Genetic programming as a means for programming computers by natural selection. Statistics and computing, 4(2):87–112, 1994. 2, 11, 14
- [24] Poli, Riccardo, William B Langdon, Nicholas F McPhee e John R Koza: Genetic programming: An introductory tutorial and a survey of techniques and applications. Univ. Essex School of Computer Science and Eletronic Engineering Technical Report No. CES-475, páginas 1–112, 2007.

- [25] Kinahan, Sean, Pouria Saidi, Ayoub Daliri, Julie Liss e Visar Berisha: Achieving reproducibility in eeg-based machine learning. Em The 2024 ACM Conference on Fairness, Accountability, and Transparency, páginas 1464–1474, 2024. 2, 9, 20, 35
- [26] Brookshire, Geoffrey, Jake Kasper, Nicholas M Blauch, Yunan Charles Wu, Ryan Glatt, David A Merrill, Spencer Gerrol, Keith J Yoder, Colin Quirk e Ché Lucero: Data leakage in deep learning studies of translational eeg. Frontiers in Neuroscience, 18:1373515, 2024. 2, 20
- [27] Lee, Hyung Tak, Hye Ran Cheon, Seung Hwan Lee, Miseon Shim e Han Jeong Hwang: Risk of data leakage in estimating the diagnostic performance of a deep-learning-based computer-aided system for psychiatric disorders. Scientific Reports, 13(1):16633, 2023. 2, 20
- [28] Siuly, Siuly, Yan Li e Yanchun Zhang: *Eeg signal analysis and classification*. IEEE Trans Neural Syst Rehabilit Eng, 11:141–144, 2016. 6
- [29] Khosla, Ashima, Padmavati Khandnor e Trilok Chand: A comparative analysis of signal processing and classification methods for different applications based on eeg signals. Biocybernetics and Biomedical Engineering, 40(2):649–690, 2020. 6
- [30] Alturki, Fahd A, Khalil AlSharabi, Akram M Abdurraqeeb e Majid Aljalal: Eeg signal analysis for diagnosing neurological disorders using discrete wavelet transform and intelligent techniques. Sensors, 20(9):2505, 2020. 7
- [31] Teplan, Michal et al.: Fundamentals of eeg measurement. Measurement science review, 2(2):1–11, 2002. 7
- [32] Rundo, Jessica Vensel e Ralph Downey III: *Polysomnography*. Handbook of clinical neurology, 160:381–392, 2019. 7
- [33] Mane, Akshaya R, SD Biradar e RK Shastri: Review paper on feature extraction methods for eeg signal analysis. Int. J. Emerg. Trend Eng. Basic Sci, 2(1):545–552, 2015. 8
- [34] Pahuja, SK, Karan Veer et al.: Recent approaches on classification and feature extraction of eeg signal: A review. Robotica, 40(1):77–101, 2022. 8
- [35] Al-Fahoum, Amjed S e Ausilah A Al-Fraihat: Methods of eeg signal features extraction using linear analysis in frequency and time-frequency domains. International Scholarly Research Notices, 2014(1):730218, 2014. 8
- [36] Urigüen, Jose Antonio e Begoña Garcia-Zapirain: Eeg artifact removal—state-of-the-art and guidelines. Journal of neural engineering, 12(3):031001, 2015. 8
- [37] Hervé, Estelle, Giovanni Mento, Béatrice Desnous e Clément François: Challenges and new perspectives of developmental cognitive eeg studies. NeuroImage, 260:119508, 2022. 8

- [38] Hwang, Sunhee, Sungho Park, Dohyung Kim, Jewook Lee e Hyeran Byun: Mitigating inter-subject brain signal variability for eeg-based driver fatigue state classification. Em ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), páginas 990–994. IEEE, 2021. 9
- [39] ŠŤASTNÝ, Jakub, Pavel Sovka e Milan Kostilek: Overcoming inter-subject variability in bci using eeg-based identification. Radioengineering, 23(1), 2014. 9
- [40] Misiūnas, AV, T Meškauskas e R Samaitienė: Machine learning based eeg classification by diagnosis: Approach to eeg morphological feature extraction. Em AIP Conference Proceedings, volume 2164. AIP Publishing, 2019. 9
- [41] Guo, Ling, Daniel Rivero, Julián Dorado, Cristian R Munteanu e Alejandro Pazos: Automatic feature extraction using genetic programming: An application to epileptic eeg classification. Expert Systems with Applications, 38(8):10425–10436, 2011. 9, 10, 12, 18, 19, 29, 30
- [42] Sanei, Saeid e Jonathon A Chambers: *EEG signal processing and machine learning*. John Wiley & Sons, 2021. 9
- [43] Gendreau, Michel e Jean Yves Potvin: *Handbook of Metaheuristics*. Springer Publishing Company, Incorporated, 2nd edição, 2010, ISBN 1441916636. 10
- [44] Harris, Sean, Travis Bueter e Daniel R. Tauritz: A comparison of genetic programming variants for hyper-heuristics. Em Proceedings of the Companion Publication of the 2015 Annual Conference on Genetic and Evolutionary Computation, GECCO Companion '15, página 1043–1050, New York, NY, USA, 2015. Association for Computing Machinery, ISBN 9781450334884. https://doi.org/10.1145/2739482.2768456. 10, 12, 13
- [45] Burke, Edmund K., Mathew R. Hyde, Graham Kendall, Gabriela Ochoa, Ender Ozcan e John R. Woodward: Exploring Hyper-heuristic Methodologies with Genetic Programming, páginas 177–201. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009, ISBN 978-3-642-01799-5. https://doi.org/10.1007/978-3-642-01799-5\_6. 10
- [46] Espejo, Pedro G, Sebastián Ventura e Francisco Herrera: A survey on the application of genetic programming to classification. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 40(2):121–144, 2009. 10
- [47] Lee, Kwang Y e Frank F Yang: Optimal reactive power planning using evolutionary algorithms: A comparative study for evolutionary programming, evolutionary strategy, genetic algorithm, and linear programming. IEEE Transactions on power systems, 13(1):101–108, 1998. 10
- [48] Nag, Kaustuv e Nikhil R Pal: A multiobjective genetic programming-based ensemble for simultaneous feature selection and classification. IEEE transactions on cybernetics, 46(2):499–510, 2015. 11, 19

- [49] Sadiq, Muhammad Tariq, Xiaojun Yu e Zhaohui Yuan: Exploiting dimensionality reduction and neural network techniques for the development of expert brain-computer interfaces. Expert Systems with Applications, 164:114031, 2021. 11
- [50] Hu, Ting: Can genetic programming perform explainable machine learning for bioinformatics? Genetic Programming Theory and Practice XVII, páginas 63–77, 2020.
- [51] Fernández-Blanco, Enrique, Daniel Rivero, Marcos Gestal e Julián Dorado: Classification of signals by means of genetic programming. Soft Computing, 17:1929–1937, 2013. 11
- [52] Bhardwaj, Arpit, Aruna Tiwari, M Vishaal Varma e M Ramesh Krishna: Classification of eeg signals using a novel genetic programming approach. Em Proceedings of the companion publication of the 2014 annual conference on genetic and evolutionary computation, páginas 1297–1304, 2014. 11
- [53] Miranda, I. M., C. Aranha, A. P. L. de Carvalho e L. P. F. Garcia: Genetic programming-based automl for eeg signal classification a comparative study. Em Anais do X Symposium on Knowledge Discovery, Mining and Learning. SBC, 2022. 11, 18, 32, 55
- [54] Miranda, Ícaro Marcelino, Claus Aranha e Marcelo Ladeira: Classification of eeg signals using genetic programming for feature construction. Em Proceedings of the Genetic and Evolutionary Computation Conference, páginas 1275–1283, 2019. 11, 19, 29, 32, 55, 56
- [55] Bhardwaj, Harshit, Aditi Sakalle, Arpit Bhardwaj e Aruna Tiwari: Classification of electroencephalogram signal for the detection of epilepsy using innovative genetic programming. Expert Systems, 36(1):e12338, 2019. 12, 24
- [56] Wilson, Garnett e Wolfgang Banzhaf: A comparison of cartesian genetic programming and linear genetic programming. Em O'Neill, Michael, Leonardo Vanneschi, Steven Gustafson, Anna Isabel Esparcia Alcázar, Ivanoe De Falco, Antonio Della Cioppa e Ernesto Tarantino (editores): Genetic Programming, páginas 182–193, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg, ISBN 978-3-540-78671-9. 12, 13
- [57] Ahvanooey, Milad Taleby, Qianmu Li, Ming Wu e Shuo Wang: A survey of genetic programming and its applications. KSII Transactions on Internet and Information Systems (TIIS), 13(4):1765–1794, 2019. 12, 13
- [58] Crane, Ellery Fussell e Nicholas Freitag McPhee: The Effects of Size and Depth Limits on Tree Based Genetic Programming, páginas 223-240. Springer US, Boston, MA, 2006, ISBN 978-0-387-28111-7. https://doi.org/10.1007/0-387-28111-8\_ 15. 12, 17
- [59] Oltean, Mihai e Crina Grosan: A comparison of several linear genetic programming techniques. Advances in Complex Systems ACS, 14(1), janeiro 2003. 13

- [60] Miller, Julian Francis: Cartesian genetic programming: its status and future. Genetic Programming and Evolvable Machines, 21(1):129–168, 2020, ISSN 1573-7632. https://doi.org/10.1007/s10710-019-09360-6. 13
- [61] McKay, Robert I., Nguyen Xuan Hoai, Peter Alexander Whigham, Yin Shan e Michael O'Neill: Grammar-based genetic programming: a survey. Genetic Programming and Evolvable Machines, 11(3):365-396, 2010, ISSN 1573-7632. https://doi.org/10.1007/s10710-010-9109-y. 13
- [62] Perkis, T.: Stack-based genetic programming. Em Proceedings of the First IEEE Conference on Evolutionary Computation. IEEE World Congress on Computational Intelligence, páginas 148–153 vol.1, 1994. 13
- [63] Langdon, William B e Riccardo Poli: Foundations of genetic programming. Springer Science & Business Media, 2013. 14
- [64] Koza, John R e Riccardo Poli: Genetic programming. Em Search Methodologies, páginas 127–164. Springer, 2005. 15, 16
- [65] Vanneschi, Leonardo, Mauro Castelli e Sara Silva: Measuring bloat, overfitting and functional complexity in genetic programming. Em Proceedings of the 12th annual conference on Genetic and evolutionary computation, páginas 877–884, 2010. 17
- [66] Gonçalves, Ivo, Sara Silva, Joana B Melo e João Carreiras: Random sampling technique for overfitting control in genetic programming. Em European Conference on Genetic Programming, páginas 218–229. Springer, 2012. 17
- [67] Lavington, Simon, Neil Dewhurst, Elwood Wilkins e Alex Freitas: *Interfacing knowledge discovery algorithms to large database management systems*. Information and Software Technology, 41(9):605–617, 1999. 17
- [68] Alam, Mohammad Nur, Muhammad Ibn Ibrahimy e SMA Motakabber: Feature extraction of eeg signal by power spectral density for motor imagery based bci. Em 2021 8th International Conference on Computer and Communication Engineering (ICCCE), páginas 234–237. IEEE, 2021. 18
- [69] Tran, Binh, Bing Xue e Mengjie Zhang: Genetic programming for feature construction and selection in classification on high-dimensional data. Memetic Computing, 8(1):3–15, 2016. 18
- [70] Smith, Matthew G e Larry Bull: Genetic programming with a genetic algorithm for feature construction and selection. Genetic Programming and Evolvable Machines, 6:265–281, 2005. 18
- [71] Neshatian, Kourosh: Feature manipulation with genetic programming. Tese de Doutoramento, Open Access Te Herenga Waka-Victoria University of Wellington, 2010.
  18

- [72] Garrett, Deon, David A Peterson, Charles W Anderson e Michael H Thaut: Comparison of linear, nonlinear, and feature selection methods for eeg signal classification. IEEE Transactions on neural systems and rehabilitation engineering, 11(2):141–144, 2003. 19
- [73] Hazarika, Neep, Jean Zhu Chen, Ah Chung Tsoi e Alex Sergejew: Classification of eeg signals using the wavelet transform. Signal processing, 59(1):61–72, 1997. 19
- [74] Sabeti, Malihe, Reza Boostani e Toktam Zoughi: Using genetic programming to select the informative eeg-based features to distinguish schizophrenic patients. Neural Network World, 22(1):3, 2012. 19
- [75] Muni, Durga Prasad, Nikhil R Pal e Jyotirmay Das: Genetic programming for simultaneous feature selection and classifier design. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 36(1):106–117, 2006. 19
- [76] Hu, Ting: Genetic programming for interpretable and explainable machine learning. Em Genetic Programming Theory and Practice XIX, páginas 81–90. Springer, 2023.
- [77] Ferreira, Leonardo Augusto, Frederico Gadelha Guimarães e Rodrigo Silva: Applying genetic programming to improve interpretability in machine learning models. Em 2020 IEEE congress on evolutionary computation (CEC), páginas 1–8. IEEE, 2020. 19
- [78] Vellido, Alfredo: The importance of interpretability and visualization in machine learning for applications in medicine and health care. Neural computing and applications, 32(24):18069–18083, 2020. 19
- [79] Murdoch, W James, Chandan Singh, Karl Kumbier, Reza Abbasi-Asl e Bin Yu: Definitions, methods, and applications in interpretable machine learning. Proceedings of the National Academy of Sciences, 116(44):22071–22080, 2019. 19
- [80] Girard, Maxime, Rémi Nahon, Enzo Tartaglione e Van Tam Nguyen: Enhanced eegbased mental state classification: A novel approach to eliminate data leakage and improve training optimization for machine learning. arXiv preprint arXiv:2312.09379, 2023. 20
- [81] Acı, Çiğdem İnan, Murat Kaya e Yuriy Mishchenko: Distinguishing mental attention states of humans via an eeg-based passive bci using machine learning methods. Expert Systems with Applications, 134:153–166, 2019. 20
- [82] Nguyen, Van Tam, Enzo Tartaglione e Tuan Dinh: Aiot-based neural decoding and neurofeedback for accelerated cognitive training: Vision, directions and preliminary results. Em 2023 IEEE Statistical Signal Processing Workshop (SSP), páginas 705—709. IEEE, 2023. 20
- [83] Saeb, Sohrab, Luca Lonini, Arun Jayaraman, David C Mohr e Konrad P Kording: The need to approximate the use-case in clinical machine learning. Gigascience, 6(5):gix019, 2017. 21

- [84] Lotte, Fabien, Laurent Bougrain, Andrzej Cichocki, Maureen Clerc, Marco Congedo, Alain Rakotomamonjy e Florian Yger: A review of classification algorithms for eeg-based brain-computer interfaces: a 10 year update. Journal of neural engineering, 15(3):031005, 2018. 23
- [85] Hosseini, Mohammad Parsa, Amin Hosseini e Kiarash Ahi: A review on machine learning for eeg signal processing in bioengineering. IEEE reviews in biomedical engineering, 14:204–218, 2020. 23
- [86] Saeidi, Maham, Waldemar Karwowski, Farzad V Farahani, Krzysztof Fiok, Redha Taiar, PA Hancock e Awad Al-Juaid: Neural decoding of eeg signals with machine learning: A systematic review. Brain Sciences, 11(11):1525, 2021. 23, 24
- [87] Devuyst, Stéphanie, Thierry Dutoit, Patricia Stenuit e Myriam Kerkhofs: Automatic sleep spindles detection—overview and development of a standard proposal assessment method. Em 2011 Annual international conference of the IEEE engineering in medicine and biology society, páginas 1713–1716. IEEE, 2011. 23, 26
- [88] Amin, Hafeez Ullah, Aamir Saeed Malik, Rana Fayyaz Ahmad, Nasreen Badruddin, Nidal Kamel, Muhammad Hussain e Weng Tink Chooi: Feature extraction and classification for eeg signals using wavelet transform and machine learning techniques. Australasian physical & engineering sciences in medicine, 38(1):139–149, 2015. 23
- [89] Chowdhury, Anirban e Javier Andreu-Perez: Clinical brain-computer interface challenge 2020 (cbcic at wcci2020): Overview, methods and results. IEEE Transactions on Medical Robotics and Bionics, 2021. 23, 24
- [90] Curran, Eleanor A e Maria J Stokes: Learning to control brain activity: A review of the production and control of eeg components for driving brain-computer interface (bci) systems. Brain and cognition, 51(3):326-336, 2003. 23
- [91] Vaid, Swati, Preeti Singh e Chamandeep Kaur: Eeg signal analysis for bci interface: A review. Em 2015 fifth international conference on advanced computing & communication technologies, páginas 143–147. IEEE, 2015. 24
- [92] Mei, Ning, Michael D Grossberg, Kenneth Ng, Karen T Navarro e Timothy M Ellmore: *Identifying sleep spindles with multichannel eeg and classification optimization*. Computers in biology and medicine, 89:441–453, 2017. 24
- [93] Yang, Fan, Jonathan Elmer e Vladimir I Zadorozhny: Smartprognosis: Automatic ensemble classification for quantitative eeg analysis in patients resuscitated from cardiac arrest. Knowledge-Based Systems, 212:106579, 2021. 24
- [94] Yamazaki, Madoka, Don M Tucker, Ayataka Fujimoto, Tomohiro Yamazoe, Tohru Okanishi, Takuya Yokota, Hideo Enoki e Takamichi Yamamoto: Comparison of dense array eeg with simultaneous intracranial eeg for interictal spike detection and localization. Epilepsy research, 98(2-3):166–173, 2012. 24
- [95] Schalk, Gerwin, Dennis J McFarland, Thilo Hinterberger, Niels Birbaumer e Jonathan R Wolpaw: Bci2000: a general-purpose brain-computer interface (bci) system. IEEE Transactions on biomedical engineering, 51(6):1034–1043, 2004. 25

- [96] Goldberger, Ary L, Luis AN Amaral, Leon Glass, Jeffrey M Hausdorff, Plamen Ch Ivanov, Roger G Mark, Joseph E Mietus, George B Moody, Chung Kang Peng e H Eugene Stanley: *Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals.* circulation, 101(23):e215–e220, 2000. 25
- [97] Wang, Xiaying, Michael Hersche, Batuhan Tömekce, Burak Kaya, Michele Magno e Luca Benini: An accurate eegnet-based motor-imagery brain-computer interface for low-power edge computing. Em 2020 IEEE International Symposium on Medical Measurements and Applications (MeMeA), páginas 1–6. IEEE, 2020. 25, 76, 78
- [98] Taşar, Beyda e Orhan Yaman: Eeg signals based motor imagery and movement classification for bci applications. Em 2022 International Conference on Decision Aid Sciences and Applications (DASA), páginas 1425–1429. IEEE, 2022. 26, 76, 78
- [99] Devuyst, Stéphanie, Thierry Dutoit, Jean François Didier, François Meers, Etienne Stanus, Patricia Stenuit e Myriam Kerkhofs: Automatic sleep spindle detection in patients with sleep disorders. Em 2006 International Conference of the IEEE Engineering in Medicine and Biology Society, páginas 3883–3886. IEEE, 2006. 26
- [100] Himanen, Sari Leena e Joel Hasan: *Limitations of rechtschaffen and kales*. Sleep medicine reviews, 4(2):149–167, 2000. 26
- [101] Lachner-Piza, Daniel, Nino Epitashvili, Andreas Schulze-Bonhage, Thomas Stieglitz, Julia Jacobs e Matthias Dümpelmann: A single channel sleep-spindle detector based on multivariate classification of eeg epochs: Mussdet. Journal of neuroscience methods, 297:31–43, 2018. 26, 76, 78
- [102] Chen, Peilu, Dan Chen, Lei Zhang, Yunbo Tang e Xiaoli Li: Automated sleep spindle detection with mixed eeg features. Biomedical Signal Processing and Control, 70:103026, 2021. 27, 76, 77
- [103] Begleiter, Henri: *EEG Database*. UCI Machine Learning Repository, 1995. DOI: https://doi.org/10.24432/C5TS3D. 27
- [104] Zhu, Guohun, Yan Li, Peng Wen e Shuaifang Wang: Analysis of alcoholic eeg signals based on horizontal visibility graph entropy. Brain informatics, 1(1):19–25, 2014. 27, 76
- [105] Mukhtar, Hamid, Saeed Mian Qaisar e Atef Zaguia: Deep convolutional neural network regularization for alcoholism detection using eeg signals. Sensors, 21(16):5456, 2021. 27, 76
- [106] Bavkar, Sandeep, Brijesh Iyer e Shankar Deosarkar: Rapid screening of alcoholism: An eeg based optimal channel selection approach. IEEE Access, 7:99670–99682, 2019. 27, 76
- [107] Hughes, Gordon: On the mean accuracy of statistical pattern recognizers. IEEE transactions on information theory, 14(1):55–63, 1968. 32

- [108] Li, Yuling, Shasha Xiao, Yingjie Li, Yunxia Li e Banghua Yang: Classification of mild cognitive impairment from multi-domain features of resting-state eeg. Em 2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), páginas 256–259. IEEE, 2020. 38
- [109] Ignaccolo, Massimiliano, Mirek Latka, Wojciech Jernajczyk, Paolo Grigolini e Bruce J West: The dynamics of eeg entropy. Journal of biological physics, 36:185–196, 2010. 38
- [110] Bao, Forrest Sheng, Xin Liu e Christina Zhang: Pyeeg: an open source python module for eeg/meg feature extraction. Computational intelligence and neuroscience, 2011(1):406391, 2011. 38
- [111] Greene, Barry R, Stephen Faul, WP Marnane, Gordon Lightbody, Irina Korotchikova e Geraldine B Boylan: A comparison of quantitative eeg features for neonatal seizure detection. Clinical Neurophysiology, 119(6):1248–1261, 2008. 38
- [112] Uthayakumar, R: Fractal dimension in epileptic eeg signal analysis. Em Applications of Chaos and Nonlinear Dynamics in Science and Engineering-Vol. 3, páginas 103–157. Springer, 2013. 38
- [113] Shamsi, Elham, Mohammad Ali Ahmadi-Pajouh e Tirdad Seifi Ala: Higuchi fractal dimension: An efficient approach to detection of brain entrainment to theta binaural beats. Biomedical signal processing and control, 68:102580, 2021. 38
- [114] Fulcher, Ben D, Max A Little e Nick S Jones: *Highly comparative time-series analysis: the empirical structure of time series and their methods*. Journal of the Royal Society Interface, 10(83):20130048, 2013. 38
- [115] Mietus, JE, Chung Kang Peng, I Henry, RL Goldsmith e AL Goldberger: *The pnnx files: re-examining a widely used heart rate variability measure*. Heart, 88(4):378–380, 2002. 46
- [116] Wang, Xiaozhe, Anthony Wirth e Liang Wang: Structure-based statistical features and multivariate time series clustering. Em Seventh IEEE international conference on data mining (ICDM 2007), páginas 351–360. IEEE, 2007. 46
- [117] Kotsiantis, Sotiris B., Ioannis D. Zaharakis e Panayiotis E. Pintelas: *Machine learning: a review of classification and combining techniques*. Artificial Intelligence Review, 26:159–190, 2006. 47
- [118] Kotsiantis, Sotiris B., Ioannis Zaharakis e P. Pintelas: Supervised machine learning: A review of classification techniques. Emerging Artificial Intelligence Applications in Computer Engineering, 160(1):3–24, 2007. 47
- [119] Mi, Jian Xun, An Di Li e Li Fang Zhou: Review study of interpretation methods for future interpretable machine learning. IEEE Access, 8:191969–191985, 2020. 47
- [120] Pak, Irina e Phoey Lee Teh: Machine learning classifiers: Evaluation of the performance in online reviews. Indian Journal of Science and Technology, 9(45):1–9, 2016. 47

- [121] Ramírez-Arias, Francisco Javier, Enrique Efren García-Guerrero, Esteban Tlelo-Cuautle, Juan Miguel Colores-Vargas, Eloisa García-Canseco, Oscar Roberto López-Bonilla, Gilberto Manuel Galindo-Aldana e Everardo Inzunza-González: Evaluation of machine learning algorithms for classification of eeg signals. Technologies, 10(4):79, 2022. 56
- [122] Kirar, Jyoti Singh e RK Agrawal: Relevant feature selection from a combination of spectral-temporal and spatial features for classification of motor imagery eeg. Journal of medical systems, 42:1–15, 2018. 70
- [123] Shimizu, Riki e Hau Tieng Wu: Unveil sleep spindles with concentration of frequency and time (conceft). Physiological Measurement, 45(8):085003, 2024. 76, 77
- [124] Wang, Fei, Li Li, Yinxing Wan, Zhuorong Li, Lixian Luo, Bangshun Hu, Jiahui Pan, Zhenfu Wen e Haiyun Huang: An efficient sleep spindle detection algorithm based on mp and lsboost. Computers, Materials & Continua, 76(2), 2023. 76
- [125] Deb, Kalyanmoy, Amrit Pratap, Sameer Agarwal e TAMT Meyarivan: A fast and elitist multiobjective genetic algorithm: Nsga-ii. IEEE transactions on evolutionary computation, 6(2):182–197, 2002. 82
- [126] Bleuler, Stefan, Martin Brack, Lothar Thiele e Eckart Zitzler: Multiobjective genetic programming: Reducing bloat using spea2. Em Proceedings of the 2001 Congress on Evolutionary Computation (IEEE Cat. No. 01TH8546), volume 1, páginas 536–543. IEEE, 2001. 82