



**UNIVERSIDADE ESTADUAL PAULISTA  
"JÚLIO DE MESQUITA FILHO"**

Rafael Junqueira Martarelli

**Estratégias para Seleção de Classificadores  
Baseadas em Programação Genética para  
Reconhecimento de Dados Multimídia**

Bauru  
2021

Rafael Junqueira Martarelli

# **Estratégias para Seleção de Classificadores Baseadas em Programação Genética para Reconhecimento de Dados Multimídia**

Dissertação apresentada como parte dos requisitos para obtenção do título de Mestre em Ciência da Computação, junto ao Programa de Pós-Graduação em Ciência da Computação, da Universidade Estadual Paulista "Júlio de Mesquita Filho".

Financiadora: FAPESP - Proc. 2021/02023-4

Orientador: Prof. Dr. João Paulo Papa

M375e      Martarelli, Rafael Junqueira  
Estratégias para Seleção de Classificadores Baseadas  
em Programação Genética para Reconhecimento de Dados  
Multimídia / Rafael Junqueira Martarelli. -- , 2024  
43 p. : il., tabs., fotos

Dissertação (mestrado) - Universidade Estadual Paulista  
(UNESP), Faculdade de Ciências, Bauru,  
Orientador: João Paulo Papa

1. Comitê de classificadores. 2. Otimização de funções  
de combinação de predição. 3. Aprendizado de máquina. I.  
Título.

Sistema de geração automática de fichas catalográficas da Unesp. Biblioteca da  
Universidade Estadual Paulista (UNESP), Faculdade de Ciências, Bauru. Dados  
fornecidos pelo autor(a).

Essa ficha não pode ser modificada.

**ATA DA DEFESA PÚBLICA DA DISSERTAÇÃO DE MESTRADO DE RAFAEL JUNQUEIRA MARTARELLI, DISCENTE DO PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO, DA FACULDADE DE CIÊNCIAS - CÂMPUS DE BAURU.**

Ao vigésimo (20) dia do mês de fevereiro do ano de 2024, às 13:00 horas, por meio de Videoconferência, realizou-se a defesa de DISSERTAÇÃO DE MESTRADO de RAFAEL JUNQUEIRA MARTARELLI, intitulada **Estratégias para Seleção de Classificadores Baseadas em Programação Genética para Reconhecimento de Dados Multimídia**. A Comissão Examinadora foi constituída pelos seguintes membros: Prof. Dr. JOAO PAULO PAPA (Orientador(a) - Participação Virtual) do(a) FC / UNESP Bauru SP, Prof. Dr. JOÃO BAPTISTA CARDIA NETO (Participação Virtual) do(a) Faculdade de Tecnologia de Catanduva. / Centro Estadual de Educação Tecnológica Paula Souza, Prof. Dr. KELTON AUGUSTO PONTARA DA COSTA (Participação Virtual) do(a) Departamento de Ciência da Computação / UNESP Bauru. Após a exposição pelo mestrando e arguição pelos membros da Comissão Examinadora que participaram do ato, de forma presencial e/ou virtual, o discente recebeu o conceito final APROVADO. Nada mais havendo, foi lavrada a presente ata, que após lida e aprovada, foi assinada pelo Presidente da Comissão Examinadora.

Documento assinado digitalmente  
 **JOAO PAULO PAPA**  
Data: 20/02/2024 14:30:01-0300  
Verifique em <https://validar.it.gov.br>

Prof. Dr. JOAO PAULO PAPA

# Agradecimentos

Gostaria de expressar minha sincera gratidão à Deus, cuja presença foi meu guia e força constante ao longo do mestrado e na minha vida. Um agradecimento especial ao Prof. Dr. João Paulo Papa, cuja orientação foi essencial, e a todos no laboratório, especialmente Douglas, por seu apoio inestimável. À Fapesp pelo financiamento desta pesquisa de mestrado via processo Proc. 2021/02023-4 e à minha família merece minha eterna gratidão pelo amor e apoio incansável. Agradeço também à banca examinadora por aceitar participar da minha defesa.

*“... não é possível decidir racionalmente se estamos na estrada certa a não ser que estejamos dispostos a considerar a possibilidade de estarmos na estrada errada.” -*

*Phillip E. Johnson*

# Resumo

A era atual da tecnologia da informação trouxe um dilúvio de dados sem precedentes, abrangendo diversas áreas, desde pesquisa científica até negócios e vida cotidiana. No entanto, a análise manual de volumes tão massivos torna-se inviável, criando a necessidade de soluções de aprendizado de máquina, como algoritmos de classificação. Embora muitos desses algoritmos tenham sido desenvolvidos, avaliar sua qualidade é multifacetado, envolvendo diversas métricas. Dada a complexidade, a formação de comitês de classificadores, onde vários classificadores trabalham juntos, se apresenta como uma solução eficaz, mas requer a escolha adequada dos classificadores e funções de combinação de predições. A programação genética, uma técnica de otimização baseada em algoritmos genéticos, surge como uma abordagem inovadora para encontrar a configuração ideal desses comitês e funções de combinação de predições, buscando maximizar o desempenho, enquanto reduz a necessidade de experimentação manual dispendiosa. A dissertação busca contribuir para o avanço do campo de aprendizado de máquina, enfrentando os desafios da análise de dados em grande escala e promovendo melhorias significativas nas classificações por meio da programação genética.

**Palavras-chave:** Classificadores. Otimização. Seleção de Classificadores. Funções de Combinação de Predições. Programação Genética.

# Abstract

The current era of information technology has brought an unprecedented deluge of data, spanning various areas from scientific research to business and everyday life. However, manually analyzing such massive volumes becomes impractical, creating the need for machine learning solutions, like classification algorithms. Although many of these algorithms have been developed, evaluating their quality is multifaceted, involving various metrics. Given the complexity, the formation of classifier committees, where multiple classifiers work together, presents as an effective solution, but it requires the appropriate choice of classifiers and prediction combination functions. Genetic programming, an optimization technique based on genetic algorithms, emerges as an innovative approach to finding the optimal configuration of these committees and prediction combination functions, aiming to maximize performance while reducing the need for costly manual experimentation. The dissertation seeks to contribute to the advancement of the machine learning field, tackling the challenges of large-scale data analysis and promoting significant improvements in classifications through genetic programming.

**Keywords:** Classifiers. Optimization. Classifier Selection. Prediction Combination Functions. Genetic Programming.

# Lista de ilustrações

|                                                                                               |    |
|-----------------------------------------------------------------------------------------------|----|
| Figura 1 – Estrutura da Matriz de Confusão. . . . .                                           | 16 |
| Figura 2 – Exemplo de uma função com mínimos e máximos. . . . .                               | 20 |
| Figura 3 – Exemplo de indivíduo da programação genética. . . . .                              | 22 |
| Figura 4 – Exemplo de cruzamento em indivíduos da programação genética. . . . .               | 23 |
| Figura 5 – Exemplo de mutação em indivíduos da programação genética. . . . .                  | 24 |
| Figura 6 – Fluxo de trabalho completo referente à metodologia empregada neste estudo. . . . . | 27 |

# Lista de abreviaturas e siglas

|    |                        |
|----|------------------------|
| GP | Programação genética   |
| ML | Aprendizado de máquina |

# Sumário

|            |                                                |           |
|------------|------------------------------------------------|-----------|
|            | <b>Lista de ilustrações</b>                    | <b>6</b>  |
| <b>1</b>   | <b>INTRODUÇÃO</b>                              | <b>10</b> |
| <b>1.1</b> | <b>Justificativa</b>                           | <b>12</b> |
| <b>1.2</b> | <b>Objetivo Geral</b>                          | <b>12</b> |
| <b>1.3</b> | <b>Hipótese de pesquisa</b>                    | <b>12</b> |
| <b>1.4</b> | <b>Organização do Documento</b>                | <b>13</b> |
| <b>2</b>   | <b>REFERENCIAL TEÓRICO</b>                     | <b>14</b> |
| <b>2.1</b> | <b>Classificadores de padrões</b>              | <b>14</b> |
| 2.1.1      | Métricas de Avaliação de Classificadores       | 16        |
| 2.1.1.1    | Matriz de Confusão                             | 16        |
| 2.1.1.2    | Acurácia                                       | 17        |
| 2.1.1.3    | Precisão                                       | 17        |
| 2.1.1.4    | Revocação                                      | 17        |
| 2.1.1.5    | Métrica F1                                     | 17        |
| 2.1.1.6    | Índice de Jaccard ( <i>Jaccard Score</i> )     | 17        |
| <b>2.2</b> | <b>Comitês de Classificadores</b>              | <b>18</b> |
| 2.2.1      | Seleção de Classificadores                     | 18        |
| 2.2.2      | Funções de Combinação de Predição              | 19        |
| <b>2.3</b> | <b>Otimização</b>                              | <b>19</b> |
| 2.3.1      | Algoritmos Evolutivos                          | 21        |
| <b>2.4</b> | <b>Programação genética</b>                    | <b>22</b> |
| 2.4.1      | Cruzamento                                     | 23        |
| 2.4.2      | Mutação                                        | 23        |
| <b>3</b>   | <b>METODOLOGIA</b>                             | <b>25</b> |
| <b>3.1</b> | <b>Conjuntos de Dados</b>                      | <b>26</b> |
| <b>3.2</b> | <b>Configuração Experimental</b>               | <b>26</b> |
| <b>4</b>   | <b>CONFIGURAÇÃO E RESULTADOS EXPERIMENTAIS</b> | <b>31</b> |
| <b>4.1</b> | <b>Classificadores</b>                         | <b>31</b> |
| <b>4.2</b> | <b>HMDB51</b>                                  | <b>32</b> |
| <b>4.3</b> | <b>UCF101</b>                                  | <b>32</b> |
| <b>4.4</b> | <b>ABLAÇÃO</b>                                 | <b>34</b> |
| <b>5</b>   | <b>DISCUSSÃO</b>                               | <b>36</b> |

|     |                             |    |
|-----|-----------------------------|----|
| 6   | CONCLUSÃO . . . . .         | 38 |
| 6.1 | Trabalhos futuros . . . . . | 38 |
|     | REFERÊNCIAS . . . . .       | 39 |

# 1 Introdução

Em um contexto histórico, a evolução da era digital é marcada por avanços significativos, como a invenção do pixel por Russell Kirsch em 1957 e a criação de "Toy Story", o primeiro longa-metragem de animação da Pixar totalmente produzido com imagens digitais. Esses marcos ilustram a extraordinária expansão de dados em formatos variados, como texto, imagens, áudio e vídeo. A transformação digital abrange desde grandes organizações como a NASA, que capta imagens do sol a cada poucos segundos, até o uso diário da internet por indivíduos, gerando uma quantidade imensa de dados ([SIMPSON, 2020](#); [TOY... , 2016](#); [BORMIDA, 2021](#)).

Diante do cenário atual, com a proliferação massiva de dados em diversas estruturas, a análise manual que antes era viável tornou-se um desafio quase impossível. A capacidade humana de processamento e análise é insuficiente para lidar com a imensa quantidade e complexidade desses dados. Isso é evidente em exemplos como o projeto da NASA, que coleta dados solares em intervalos de segundos, gerando volumes imensos de informações. Estes dados, embora ricos em potencial, requerem métodos avançados para sua análise e interpretação eficazes ([YOO; LEUNG; NASRIDINOV, 2022](#)).

No contexto da era atual de dados massivos, o campo do aprendizado de máquina (ML) emerge como uma solução crucial. Esta área da tecnologia tem se destacado na revolução digital, oferecendo ferramentas capazes de processar e analisar grandes volumes de dados com eficiência. O ML permite automatizar tarefas que tradicionalmente requerem intervenção humana, adaptando-se a uma variedade de aplicações e tipos de dados. Esse avanço representa um salto significativo na forma como interagimos e extraímos informações dos dados ([BALOGLU; LATIFI; NAZHA, 2022](#)).

No ML, classificadores são algoritmos projetados para categorizar dados em diferentes classes ou grupos. Eles são fundamentais para a análise e interpretação de dados, sendo aplicados em uma variedade de contextos, desde reconhecimento de imagens até análise de sentimentos em textos. A eficácia de um classificador depende da sua capacidade de corretamente identificar e separar dados em categorias relevantes, uma habilidade crítica em um mundo onde os dados são cada vez mais complexos e abundantes.

A qualidade de um classificador é medida por meio de métricas como acurácia, medida-F, revocação, coeficiente Kappa, dentre outras ([VUJOVIĆ et al., 2021](#)). Vários classificadores podem apresentar resultados considerados bons, influenciados por

múltiplas variáveis. Estas incluem a qualidade do algoritmo, o tipo de dados sendo classificado, a base de dados usada para treinamento, os hiperparâmetros escolhidos, e a ocorrência de fenômenos como sobreajuste e subajuste. Estes aspectos são cruciais na determinação da eficácia de um classificador no processamento de dados (ALZUBAIDI et al., 2021).

A eficácia de um único classificador pode ser limitada devido às suas características intrínsecas ou à natureza dos dados em questão. Por isso, uma abordagem que ganha cada vez mais destaque é o uso de comitês de classificadores, também conhecidos como *ensembles*. Essa estratégia capitaliza na ideia de que a combinação de múltiplos classificadores, cada um com seus pontos fortes e fracos, pode resultar em um sistema mais robusto e eficiente. Ao trabalhar em conjunto, os classificadores em um comitê podem complementar-se mutuamente, melhorando a precisão geral e reduzindo as vulnerabilidades de modelos individuais (MIENYE; SUN, 2022). Esse método é particularmente valioso em cenários complexos onde a diversidade de abordagens pode enriquecer a análise e a tomada de decisões baseada em dados.

Contudo, nem todos os classificadores em um comitê contribuem igualmente para o desempenho geral. Alguns podem ser redundantes ou até mesmo prejudiciais à precisão do grupo. Surge, portanto, a necessidade de uma seleção de classificadores, onde o objetivo é identificar e manter apenas aqueles classificadores que efetivamente melhoram o desempenho coletivo. Este processo de seleção e eliminação não é apenas uma questão de maximizar a acurácia, mas também de otimizar recursos computacionais. A poda de classificadores é realizada por meio de algoritmos específicos que avaliam a contribuição de cada classificador individualmente e em conjunto, buscando a combinação ótima que ofereça o melhor equilíbrio entre precisão e eficiência.

Uma parte essencial do funcionamento de comitês de classificadores são as funções de combinação de predição. Essas funções determinam como os votos de cada classificador individual são ponderados e combinados para produzir uma decisão coletiva. Existem várias estratégias, como a votação majoritária, onde a classe mais votada é escolhida, ou a votação ponderada, que atribui diferentes pesos aos votos dos classificadores com base em sua confiabilidade ou desempenho histórico. A escolha da função de votação desempenha um papel crítico na eficácia do comitê e é um ponto de considerável pesquisa no campo do ML.

É neste ponto que a programação genética (GP) apresenta uma proposta inovadora. A GP pode ser aplicada tanto na seleção dos classificadores quanto na geração de funções de combinação de predições específicas para cada conjunto de dados, representando uma abordagem de otimização central nesta dissertação. Nosso foco é explorar como a GP pode ser utilizada para aprimorar o desempenho dos comitês de classificadores, oferecendo soluções mais eficazes e adaptadas às necessidades

específicas de cada problema de classificação.

## 1.1 Justificativa

A era da tecnologia da informação tem sido marcada por uma explosão sem precedentes de dados, impactando desde a pesquisa científica até o cotidiano empresarial e pessoal. Um exemplo vívido disso é a coleta massiva de dados solares pela NASA, ou a quantidade massiva de vídeos sendo postados diariamente em sites de hospedagem de vídeo, como o YouTube, ilustrando a necessidade urgente de métodos mais avançados de processamento de dados. Diante da impraticabilidade da análise manual, o ML e, em particular, os comitês de classificadores, surgem como uma solução promissora. Contudo, a configuração ideal desses comitês e a seleção eficaz de funções de combinação de predição permanecem como desafios notáveis. Aqui, é destacado como a GP pode inovar na seleção de classificadores e na otimização da função de combinação de predição, melhorando o desempenho e economizando recursos. Esta seção busca enfatizar a relevância da GP no aprimoramento dos comitês de classificadores, frente às limitações das técnicas convencionais.

## 1.2 Objetivo Geral

Esta dissertação tem como objetivo explorar a aplicação da GP para otimizar comitês de classificadores em contextos de análise de dados em larga escala. Especificamente, investigaremos como a GP pode ser utilizada para identificar a combinação ideal de classificadores e desenvolver funções de combinação de predição que maximizem a eficácia da classificação. Pretendemos quantificar essa melhoria utilizando métricas específicas como precisão, revocação e medida-F. Além disso, buscaremos demonstrar como a GP pode substituir experimentações manuais demoradas e onerosas, consolidando-se como uma ferramenta valiosa no domínio do ML.

## 1.3 Hipótese de pesquisa

Partindo do pressuposto de que a GP pode ser eficientemente aplicada na otimização de comitês de classificadores, nossa hipótese é que esta abordagem resultará em melhorias significativas no desempenho da classificação em comparação com métodos convencionais. Esta hipótese baseia-se na capacidade da GP de explorar efetivamente o espaço de soluções para encontrar classificadores ótimos e funções de combinação de predição adaptativas, potencialmente superando as limitações das abordagens tradicionais. Planejamos validar esta hipótese através de experimentos rigorosos, comparando o desempenho dos comitês otimizados pela GP com comitês de

classificadores que utilizam funções de combinação padrão, como a média e a média ponderada.

## 1.4 Organização do Documento

Esta dissertação é estruturada da seguinte forma:

1. No Capítulo 2, é apresentado o referencial teórico, onde se exploram os conceitos fundamentais para o entendimento desta pesquisa.
2. No Capítulo 3, descreve-se a metodologia adotada, detalhando as técnicas e abordagens utilizadas no estudo.
3. No Capítulo 4, são expostos os resultados experimentais alcançados, fornecendo uma visão quantitativa da pesquisa.
4. No Capítulo 5, procede-se à discussão desses resultados, interpretando-os à luz dos objetivos e hipóteses do estudo.
5. Por fim, o Capítulo 6 conclui a dissertação, destacando as principais descobertas, contribuições e possíveis direções para pesquisas futuras.

## 2 Referencial Teórico

Neste referencial teórico, são exploradas as áreas-chave e os conceitos fundamentais associados ao ML, fundamentais para o entendimento do escopo e das inovações propostas nesta pesquisa. Inicialmente, são abordados os classificadores de padrões, elementos cruciais do ML e deste estudo, com uma ênfase especial na sua relevância e na diversidade de suas aplicações. Além disso, é dada atenção especial às métricas de avaliação de classificadores, visando oferecer uma compreensão clara de como avaliar e interpretar o desempenho desses modelos.

Em sequência, o método Fast-CoViAR é introduzido, detalhando suas contribuições significativas para a classificação de vídeos. Este método estabelece um pano de fundo para as propostas aplicadas nesta dissertação. O conceito de comitês de classificadores é então aprofundado, com subseções focadas na seleção de classificadores e nas funções de combinação de predições, visando destacar as potenciais oportunidades de otimização nestes processos.

Finalmente, a seção sobre GP oferece um exame detalhado deste algoritmo evolutivo. A discussão se concentra em como a GP aplica processos de evolução para solucionar problemas complexos, estabelecendo uma conexão direta com as inovações propostas na pesquisa.

Este referencial teórico foi elaborado com o objetivo de fornecer uma visão abrangente e detalhada das áreas mais relevantes ao tema desta dissertação, ressaltando tanto os conceitos fundamentais quanto as inovações mais recentes no campo. Assim, estabelece-se uma base sólida para a compreensão das técnicas avançadas que serão discutidas nas seções subsequentes do trabalho.

### 2.1 Classificadores de padrões

Com o avanço contínuo da tecnologia de computadores e da internet, a quantidade de dados gerados diariamente tem crescido exponencialmente. Estima-se que a população mundial produza cerca de 2.5 quintilhões de bytes por dia ([TECH, 2021](#)). Esse fenômeno torna a análise manual impraticável em diversos contextos, especialmente em tarefas de classificação de dados, como a verificação de conformidade de imagens em redes sociais ([DEFENDANTS'OPPOSITION, 2018](#)).

A resposta a esta necessidade de processamento e classificação eficiente de dados está nos classificadores de padrões, algoritmos projetados para identificar e categorizar padrões em variados tipos de dados, incluindo imagens, textos, audios e

vídeos ([BALOGLU; LATIFI; NAZHA, 2022](#)). Um exemplo de uso dos classificadores é no caso da autenticação por biometria, onde características específicas são analisadas para confirmar a identidade de um indivíduo ([DARGAN; KUMAR, 2020](#)). Esta aplicação prática evidencia a necessidade de classificadores robustos e precisos, adaptados a diferentes tipos de dados e contextos.

A busca por soluções eficazes para lidar com a complexidade dos dados leva à diversificação dos tipos de algoritmos de classificação de padrões. Esta diversidade é uma resposta direta à necessidade de adaptar os classificadores a diferentes contextos e tipos de dados. Os algoritmos variam significativamente em seus fundamentos teóricos e práticos, abrangendo desde métodos baseados em análises cartesianas e espaciais, como as Máquinas de Vetores de Suporte, até aqueles inspirados em processos naturais, como as Redes Neurais Artificiais ([BOATENG; OTOO; ABAYE, 2020](#)).

Esta multiplicidade de abordagens reflete a natureza dinâmica do campo do Aprendizado de Máquina, onde a inovação contínua é essencial. Pesquisadores em todo o mundo dedicam-se não apenas a aprimorar algoritmos existentes, mas também a explorar novas fronteiras, desenvolvendo técnicas inovadoras. Exemplos recentes dessa tendência incluem a evolução da floresta de caminhos ótimos ([FALCAO; PAPA, 2022](#)) e o desenvolvimento de novas arquiteturas para redes neurais. Esses avanços representam um esforço contínuo para melhorar a eficácia, eficiência e aplicabilidade dos classificadores de padrões, adaptando-os a desafios cada vez mais complexos.

Há diversas pesquisas que buscam aumentar o conhecimento sobre ML, um exemplo é o trabalho de [Rosa, Roder e Papa \(2022\)](#) exploraram o uso de redes neurais convolucionais combinadas com a técnica de sacola de amostras baseada na vizinhança para classificar escritas manuais. No decorrer de sua pesquisa, os autores realizaram experimentos com nove classificadores diferentes, cada um treinado por quinhentas épocas em duas bases de dados distintas. Os resultados obtidos foram notáveis: eles conseguiram desenvolver um classificador que superou o modelo anterior em eficiência, alcançando uma acurácia de 66,6%, um aumento de 14,7% em relação ao desempenho anterior de 51,9%.

Em 2019, [Santos, Sebe e Almeida \(2019\)](#), implementaram e testaram uma arquitetura neural chamada de CV-C3D, uma rede neural convolucional que classifica ações em vídeos comprimidos 3D. Para verificar a viabilidade desta nova rede, foram realizados testes em duas bases, UCF-101 e HMDB-51, e compararam os resultados gerados com outras redes já conhecidas na literatura. Ao final observaram que seu novo método obteve um resultado semelhante na base UCF-101 (83,9%) e obtiveram o segundo melhor resultado na base HMDB-51 (59,1%).

Alcançar a perfeição em classificadores de padrões é uma meta raramente

atingida, devido à complexidade e variedade dos desafios aos quais esses sistemas são submetidos. A tolerância a erros varia significativamente dependendo do contexto de aplicação: em algumas situações, como na segmentação de potenciais clientes para campanhas publicitárias, certa margem de erro pode ser aceitável. Por outro lado, em contextos críticos, como na identificação e classificação de objetos por veículos autônomos, um único erro pode ter consequências desastrosas. Essa variabilidade nos requisitos e impactos dos erros traz à tona duas questões fundamentais na área de aprendizado de máquina: como quantificar efetivamente a qualidade de um classificador e quais estratégias podem ser empregadas para aproximar esses sistemas da perfeição ou, pelo menos, melhorar significativamente suas soluções em face de problemas complexos de classificação.

## 2.1.1 Métricas de Avaliação de Classificadores

A eficácia de um classificador é um aspecto crucial no desenvolvimento de sistemas de aprendizado de máquina. Diversas métricas são empregadas para quantificar a qualidade de um classificador, cada uma oferecendo *insights* distintos sobre seu desempenho. Métricas fundamentais como a Matriz de Confusão, Acurácia, Precisão, Revocação, Métrica F1 e o Índice de Jaccard são essenciais para uma avaliação abrangente da eficiência do classificador (VUJOVIĆ et al., 2021).

### 2.1.1.1 Matriz de Confusão

A Matriz de Confusão é uma ferramenta analítica primária na avaliação de classificadores. Ela fornece uma visão clara da relação entre as previsões feitas pelo classificador e os valores reais. Organizada em duas linhas e duas colunas, em problemas de classificação binária, a matriz categoriza as previsões em Verdadeiro Positivo (TP), Falso Positivo (FP), Falso Negativo (FN) e Verdadeiro Negativo (TN), como ilustrado na Figura 1.

Figura 1 – Estrutura da Matriz de Confusão.

|               | Predito positivo         | Predito negativo         |
|---------------|--------------------------|--------------------------|
| Real positivo | Verdadeiro Positivo (TP) | Falso Negativo (FN)      |
| Real negativo | Falso Positivo (FP)      | Verdadeiro Negativo (TN) |

Fonte: Elaborada pelo autor.

Em um cenário de classificação multiclasse, a Matriz de Confusão é expandida para acomodar o número de classes envolvidas. Ela se torna uma matriz  $N \times N$ , onde  $N$  é o número de classes. Cada linha representa as instâncias da classe real, enquanto cada coluna representa as instâncias da classe prevista. As células da matriz

mostram o número de previsões feitas para cada combinação de classe real e classe prevista. Isso permite uma análise mais detalhada do desempenho do classificador em cada classe, identificando não apenas erros gerais, mas também erros específicos de classificação entre as classes.

#### 2.1.1.2 Acurácia

A Acurácia é uma das métricas mais utilizadas para avaliar classificadores, fornecendo uma visão geral sobre a proporção de previsões corretas em relação ao total. Calcula-se pela fórmula:

$$Acurcia = \frac{TP + TN}{TP + TN + FP + FN}. \quad (1)$$

#### 2.1.1.3 Precisão

A Precisão é crucial em contextos onde distinguir entre verdadeiros e falsos positivos é vital, como em diagnósticos médicos. Ela é calculada pela proporção de verdadeiros positivos em relação ao total de positivos identificados pelo classificador:

$$Preciso = \frac{TP}{TP + FP}. \quad (2)$$

#### 2.1.1.4 Revocação

A Revocação, similar à Precisão, foca na proporção de verdadeiros positivos em relação à soma dos verdadeiros positivos e falsos negativos, sendo fundamental para entender a habilidade do classificador em identificar todos os casos positivos:

$$Revocao = \frac{TP}{TP + FN}. \quad (3)$$

#### 2.1.1.5 Métrica F1

A Métrica F1 é um balanço entre Precisão e Revocação, fornecendo uma medida harmônica das duas. É particularmente útil quando a distribuição de classes é desigual e é calculada pela fórmula:

$$F1 = 2 \times \frac{\text{Precisão} \times \text{Revocação}}{\text{Precisão} + \text{Revocação}}. \quad (4)$$

#### 2.1.1.6 Índice de Jaccard (*Jaccard Score*)

O Índice de Jaccard, conhecido também como *Jaccard Score*, é uma métrica que se concentra na proporção de interseção entre os verdadeiros positivos e a união

dos verdadeiros positivos, falsos positivos e falsos negativos. É particularmente útil para medir a precisão dos classificadores em situações onde as classes são desbalanceadas ou em problemas onde a distinção clara entre as classes é crucial. O *Jaccard Score* é calculado utilizando a seguinte fórmula:

$$Jaccard\ Score = \frac{TP}{TP + FP + FN}. \quad (5)$$

## 2.2 Comitês de Classificadores

Assim como a colaboração humana supera limitações individuais por meio da discussão coletiva e da combinação de conhecimentos, no campo do Aprendizado de Máquina, um conceito similar é aplicado através dos comitês de classificadores, também conhecidos como *ensembles*. Estes comitês são conjuntos de classificadores que, ao trabalharem juntos, potencializam suas habilidades na resolução de tarefas complexas de classificação, mitigando fraquezas individuais e melhorando a precisão geral (DONG et al., 2019).

### 2.2.1 Seleção de Classificadores

A seleção de classificadores para um comitê é um processo que visa identificar e combinar diferentes modelos de tal forma que suas forças sejam maximizadas e suas fraquezas, minimizadas. Este processo é análogo à formação de equipes em contextos humanos, onde a diversidade de habilidades e perspectivas contribui para um melhor desempenho geral. Na literatura, vários métodos têm sido propostos para a seleção efetiva de classificadores, incluindo os seguintes:

- **Ensemble Pruning Baseado em Ordenação:** Classifica os classificadores com base em métricas de desempenho (como acurácia ou erro) e seleciona os melhores. Exemplo: selecionar os top 5 classificadores com base na acurácia.
- **Ensemble Pruning Baseado em Agrupamento:** Agrupa classificadores com base na semelhança de suas previsões e seleciona representantes de cada grupo. Isso promove diversidade no comitê. Exemplo: agrupar classificadores usando k-means e escolher um representante de cada cluster.
- **Ensemble Pruning Baseado em Otimização:** Utiliza algoritmos de otimização (como algoritmos genéticos) para encontrar a melhor combinação de classificadores. Considera critérios como desempenho e diversidade. Exemplo: usar programação genética para encontrar a combinação ótima de classificadores.

## 2.2.2 Funções de Combinação de Predição

Após a seleção dos classificadores, o próximo passo crucial é determinar como suas previsões individuais serão combinadas. As funções de combinação de predição são essenciais para o sucesso do comitê, pois definem como as informações de diferentes fontes serão sintetizadas para formar uma decisão unificada. Técnicas comuns de combinação incluem votação os seguintes:

- **Votação Majoritária:** Cada classificador vota em uma classe, e a classe com a maioria dos votos é escolhida como a previsão final. Exemplo: Em um comitê de cinco classificadores, se três classificadores votam na classe A e dois na classe B, a classe A é selecionada.
- **Médias Ponderadas:** As previsões dos classificadores são combinadas com base em pesos atribuídos a cada um, refletindo sua confiabilidade ou desempenho. Exemplo: Classificadores com maior acurácia recebem pesos maiores na combinação das previsões.
- **Métodos Estatísticos Complexos:** Abordagens mais sofisticadas, como métodos bayesianos ou de fusão de evidências, que consideram a incerteza e a correlação entre as previsões dos classificadores. Exemplo: Combinação de previsões usando modelos probabilísticos para estimar a probabilidade de cada classe.

## 2.3 Otimização

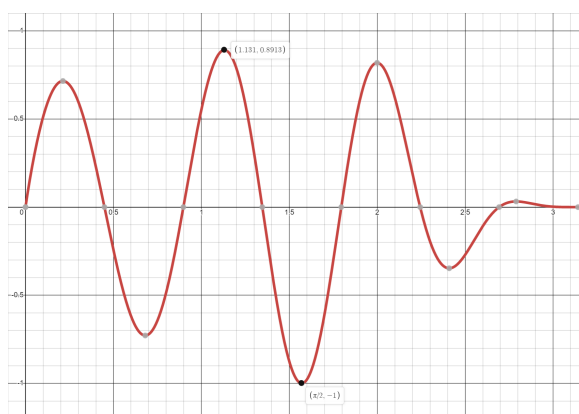
Outro tema importante para ser abordado nesta dissertação é sobre a otimização. Esta é um aspecto fundamental em diversas áreas, incluindo o Aprendizado de Máquina e a solução de problemas complexos do dia a dia. Este processo envolve a busca pela melhor solução possível dentro de um conjunto de opções, cada uma com seus próprios méritos e limitações. Por exemplo, um viajante que precisa determinar a melhor rota para maximizar as vendas ou um escoteiro que deve escolher os itens mais essenciais para carregar em sua mochila são situações clássicas de otimização (CHEIKHROUHOU; KHOUI, 2021; CACCHIANI et al., 2022).

Em um universo com recursos finitos nem sempre é possível investir tempo e/ou recurso com soluções ruins para problemas, cabe compreender o problema e buscar uma boa forma de resolvê-lo de forma, relativamente, boa ou ótima. Visto a infinidade de soluções que tendem a ter os problemas, é possível compreender o problema como uma função, onde a entrada são os parâmetros escolhidos e o resultado é o valor de qualidade que essas escolhas resultaram. Por exemplo, no caso do viajante, uma

sequência de cidades pode resultar em mais lucro ou maior velocidade de conclusão que outra sequência. (CHEIKHROUHOU; KHOUFI, 2021)

O conjunto de todas as soluções possíveis é denominado espaço de busca. Ao observar o espaço de busca localmente é possível encontrar ótimos locais, parâmetros iniciais que resultam em um valor melhor que a vizinhança apresenta. Esses ótimos locais podem ser ótimos globais caso, ao observar todo o espaço de busca, não seja possível encontrar nenhum outro valor melhor para a resolução do problema (DU; PARDALOS; WU, 2009). O espaço de busca geralmente é representado em um plano cartesiano, como na Figura 2, nesta consta uma função a qual apresenta mínimos e máximos locais e globais.

Figura 2 – Exemplo de uma função com mínimos e máximos.



Fonte: Elaborada pelo autor.

Visto as limitações de tempo e recurso, é compreensível a importância de se encontrar o ótimo global para resolver os problemas de minimização, diminuir um certo valor, ou maximização, maximizar um certo valor. Porém, como encontrar um ótimo local ou até mesmo um ótimo global? Talvez percorrer todo o espaço de busca seja uma solução viável.

Em alguns casos a força bruta, percorrer todo o espaço de busca, é viável. Afinal, ao obtermos todos os valores do espaço de busca será possível saber qual é realmente o ótimo global e assim a melhor forma de resolver o problema (CORMEN; LEISERSON, 2022). Porém, não é sempre que a força bruta é uma possível solução. Há problemas que apresentam uma complexidade que mesmo com um supercomputador levaria milhares de anos para ser concluído.

Visto a impossibilidade de utilizar a força bruta para a otimização de diversos problemas, algoritmos de otimização tem surgido para enfrentar estes desafios. Um dos primeiros algoritmos de otimização que foi cogitado foi o algoritmo guloso. Este parte do pressuposto que, em alguns casos, fazer escolhas que parecem localmente

ótimas podem levar a uma solução ótima ([CORMEN; LEISERSON, 2022](#)).

Em alguns casos nem a força bruta nem os algoritmos gulosos podem otimizar o problema. Visto a necessidade de encontrar novos algoritmos de otimização, diversos pesquisadores propuseram e implementaram novos algoritmos de otimização para diversos problemas. Há diversos tipos de algoritmos, alguns deles são:

- **Algoritmos Genéticos:** Baseiam-se em princípios de evolução biológica, utilizando operadores como seleção, cruzamento e mutação para evoluir uma população de soluções e encontrar a melhor.
- **Otimização por Colônia de Formigas:** Inspirada no comportamento das formigas ao encontrar caminhos para alimentos. Utiliza a ideia de feromônio para encontrar soluções ótimas em problemas de otimização, especialmente útil em problemas de roteamento.
- **Otimização por Enxame de Partículas:** Simula o comportamento social de enxames, como pássaros ou peixes. Cada partícula na população ajusta sua posição com base em sua experiência e na dos vizinhos para encontrar soluções ótimas.
- **Programação Genética:** É uma técnica de algoritmo genético que evolui programas de computador para resolver problemas. Ela manipula representações de programas e utiliza operações evolutivas para otimizar soluções.

### 2.3.1 Algoritmos Evolutivos

Os algoritmos evolutivos são uma categoria de algoritmos de otimização que simulam o processo de evolução biológica. Eles são utilizados para resolver problemas complexos em diversas áreas, aplicando princípios como seleção natural, mutação, reprodução e recombinação ([EIBEN; SMITH, 2015](#)).

Um algoritmo evolutivo é um algoritmo que tenta simular a evolução das espécies, de Charles Darwin. Na biologia, cada indivíduo tem suas características que podem ou não ajudá-lo a gerar descendentes. A geração da prole pode ser dada por meio da reprodução sexuada ou assexuada, sendo a primeira a mais interessante para a diversidade populacional. Gerar indivíduos é uma tarefa complicada na natureza, os mais adaptados têm mais chance de gerar filhos do que os menos adaptados. Conforme as gerações passam é notável que a população de indivíduos tendem a se tornarem cada vez mais adaptados para aquele ambiente ([ZHAN et al., 2022](#)).

O universo em que os seres humanos e todos os outros animais estão inseridos é um universo escasso, seja em recursos tanto em tempo. Devido a realidade deste

universo o fenômeno da evolução levou a ocorrência de animais que realizam ações de forma boa ou ótima, como a aerodinâmica de um pássaro, a busca por alimentos feitas por formigas, entre diversas outras características animais. Portanto é compreensível a utilização tanto da simulação da ação dos animais quanto da evolução, as quais geraram tanta diversidade, para a resolução de problemas de otimização (NOSIL et al., 2020).

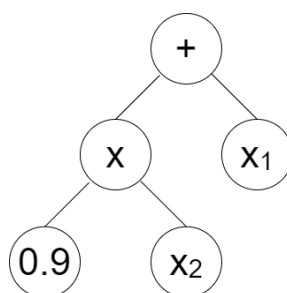
Geralmente um algoritmo evolutivo tem três partes importantes: função de recompensa, indivíduos e algoritmo principal. A função de recompensa é um cálculo que, baseado nas características de um indivíduo, irá gerar um valor de qualidade do indivíduo, em biologia, seria como o quão bem adaptado aquele indivíduo é. Os indivíduos têm suas características ímpares os quais deixam estes mais ou menos adaptados (BACK, 2018).

## 2.4 Programação genética

Geralmente a evolução tecnológica parte de um trabalho totalmente manual humano, passando por terceirização do trabalho a outras entidades biológicas ou sintéticas, finalmente, o fim da necessidade do ser humano na realização do trabalho. O processo de evolução das tecnologias existe em diversas áreas e não está sendo diferente na computação, atualmente é utilizado diversas ferramentas que automatizam diversos trabalhos que os desenvolvedores antes faziam no desenvolvimento de sistemas. Uma das que se destacam pela terceirização do problema é o algoritmo evolutivo bio-inspirado chamado de GP.

O diferencial do algoritmo da GP aos demais algoritmos evolutivos bio-inspirados são seus indivíduos. Os quais são árvores de expressão, composta de dois tipos de nós: nós do tipo função e do tipo terminais. Na Figura 3 tem-se um exemplo de um indivíduo e a solução gerada por ele é dada percorrendo a árvore, neste caso a solução é dada por:  $0.9 * x_2 + x_1$ .

Figura 3 – Exemplo de indivíduo da programação genética.



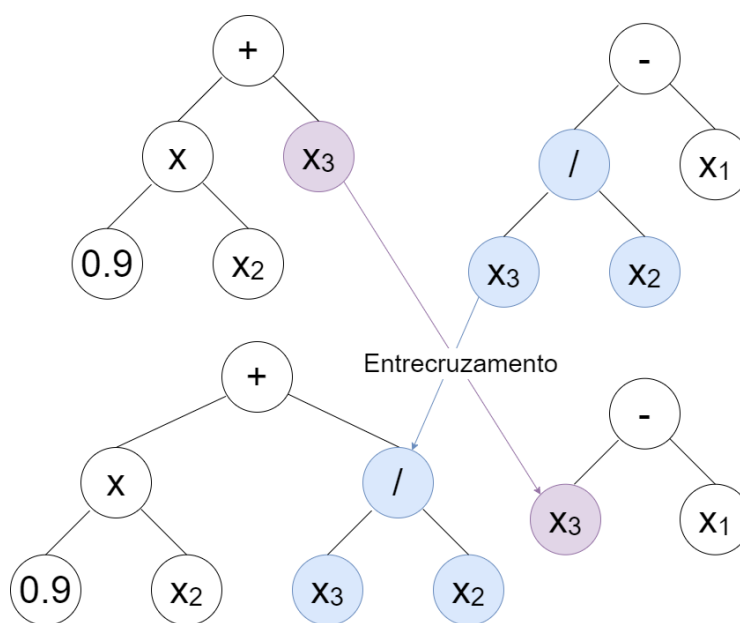
Fonte: Elaborada pelo autor.

Na Figura 3 as variáveis assumem quaisquer dados, no caso podem ser as conclusões realizadas pelos classificadores. Para cada variável será gerada uma solução da árvore, dada esse conjunto de soluções pode-se utilizar qualquer função para gerar a qualidade do indivíduo. Podendo ser, por exemplo, a função de acurácia, Equação 1.

### 2.4.1 Cruzamento

Os indivíduos da programação genética, assim como os biológicos, reproduzem e geram descendentes com características de ambos os pais. Na biologia há o cruzamento, onde metade do gene do indivíduo é do pai e metade da mãe, na programação genética é semelhante. Dado duas árvores, o indivíduo terá parte da árvore do pai e parte da mãe. Um exemplo de cruzamento pode ser visto na Figura 4.

Figura 4 – Exemplo de cruzamento em indivíduos da programação genética.

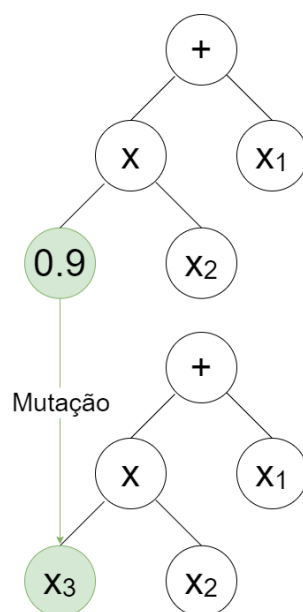


Fonte: Elaborada pelo autor.

### 2.4.2 Mutação

Se os indivíduos fossem cópias exatas dos genes dos pais, a quantidade de variantes de indivíduos seria extremamente limitada, para resolver este problema há a mutação, uma pequena probabilidade de que trechos do gene se modifiquem. Na programação genética a mutação é dada em uma região da árvore, modificando a região para outra estrutura. Um exemplo de mutação pode ser visto na Figura 5.

Figura 5 – Exemplo de mutação em indivíduos da programação genética.



Fonte: Elaborada pelo autor.

### 3 Metodologia

O sistema inicialmente, constrói uma matriz tridimensional concatenando as previsões dos classificadores, abrangendo as fases de treinamento e teste. Nesta matriz, a primeira dimensão é representativa dos classificadores, a segunda dimensiona os objetos de amostra, e a terceira corresponde as possíveis classes. Portanto, cada célula da matriz simboliza o valor prognosticado para uma categoria específica por um classificador, dado um objeto de amostra.

O segundo passo concerne à normalização da matriz de previsão tridimensional. Emprega-se uma função inicial que ajusta as previsões iniciais ao intervalo  $[0, 1]$ . Para este propósito, optou-se pela função softmax, embora outras funções, tais como normalização e sigmóide, possam ser igualmente aplicadas.

No terceiro estágio, inicia-se o processo de otimização. O algoritmo desenvolve uma população aleatória cujo tamanho é definido por *population\_size* adicionado a *new\_individuals*. A geração aleatória de cada árvore obedece a determinadas restrições: (i) o nó raiz é invariavelmente um nó de função (cálculo), (ii) o nó com uma profundidade equivalente a *max\_depth* é sempre um nó de extração, e (iii) os nós restantes são selecionados aleatoriamente entre as funções de cálculo e a função de extração. Essas restrições garantem a diversidade e a eficácia da população gerada. Os dois tipos principais de nós utilizados no processo são detalhados a seguir:

- Nós de extração da matriz de previsões: esses nós são sempre folhas da árvore e copiam uma das matrizes bidimensionais de algum classificador. Durante a geração desses nós, um classificador é selecionado aleatoriamente.
- Nós de função: quando este tipo de nó é gerado, uma das funções é escolhida aleatoriamente. As Tabelas 1 e 2 listam todas as funções disponíveis utilizadas neste.

A convergência do algoritmo genético neste estudo é eficiente, permitindo um tamanho populacional menor. A relação entre o tamanho da população e o número de gerações é inversamente proporcional, o que significa que uma população maior pode reduzir a necessidade de muitas gerações encontrarem soluções ótimas, e vice-versa. Além disso, a capacidade de memória disponível da GPU é um fator crucial na determinação do tamanho da população. Uma maior capacidade de GPU permite populações maiores, facilitando uma exploração mais abrangente do espaço de soluções em menos gerações.

Uma vez gerada a população, procede-se ao cálculo da qualidade de cada árvore, baseando-se na precisão no conjunto de treinamento da matriz de previsões resultante da árvore. Consequentemente, o algoritmo executa o operador de cruzamento com base na aptidão de cada árvore e aplica o operador de mutação a cada árvore resultante. Adicionalmente, novos indivíduos aleatórios são gerados (*new\_individuals*) e adicionados à população. Estes processos repetem-se até a obtenção do número estipulado de gerações.

É importante salientar que esta abordagem é aplicável a qualquer problema de classificação, seja ele de texto, imagem, vídeo, áudio ou uma combinação de classificadores projetados para diferentes tipos de dados. Por exemplo, pode-se empregar classificadores de vídeo para a classificação do vídeo em si, enquanto classificadores de texto são utilizados para analisar e categorizar a descrição do vídeo. O único requisito é que a ordem dos objetos de amostra e das classes seja a mesma em todos os classificadores.

A Figura 6 ilustra o fluxo de trabalho relacionado à metodologia deste estudo.

## 3.1 Conjuntos de Dados

Os experimentos foram realizados em dois conjuntos de dados:

- O conjunto de dados HMDB51 é uma grande coleção de vídeos realistas provenientes de filmes e vídeos da web, compreendendo 6.766 clipes em 51 categorias de ação, com uma taxa de quadros fixa de 30 FPS, altura fixa de 240 e largura escalada para manter a proporção original. Essas categorias abrangem uma ampla gama de ações humanas, como dirigir, lutar, correr e beber, entre outras classes ([KUEHNE et al., 2011](#)).
- O conjunto de dados UCF101, uma extensão do UCF50, contém 13.320 clipes de vídeo classificados em 101 categorias. Todos os vídeos são originários do YouTube, com uma taxa de quadros fixa de 25 FPS e resolução de 320×240. Alguns vídeos podem apresentar desafios como iluminação inadequada, fundos movimentados e movimento significativo da câmera ([SOOMRO; ZAMIR; SHAH, 2012](#)).

## 3.2 Configuração Experimental

Nesta seção, é apresentado a configuração experimental relativa ao processo de otimização utilizando o algoritmo de PG:

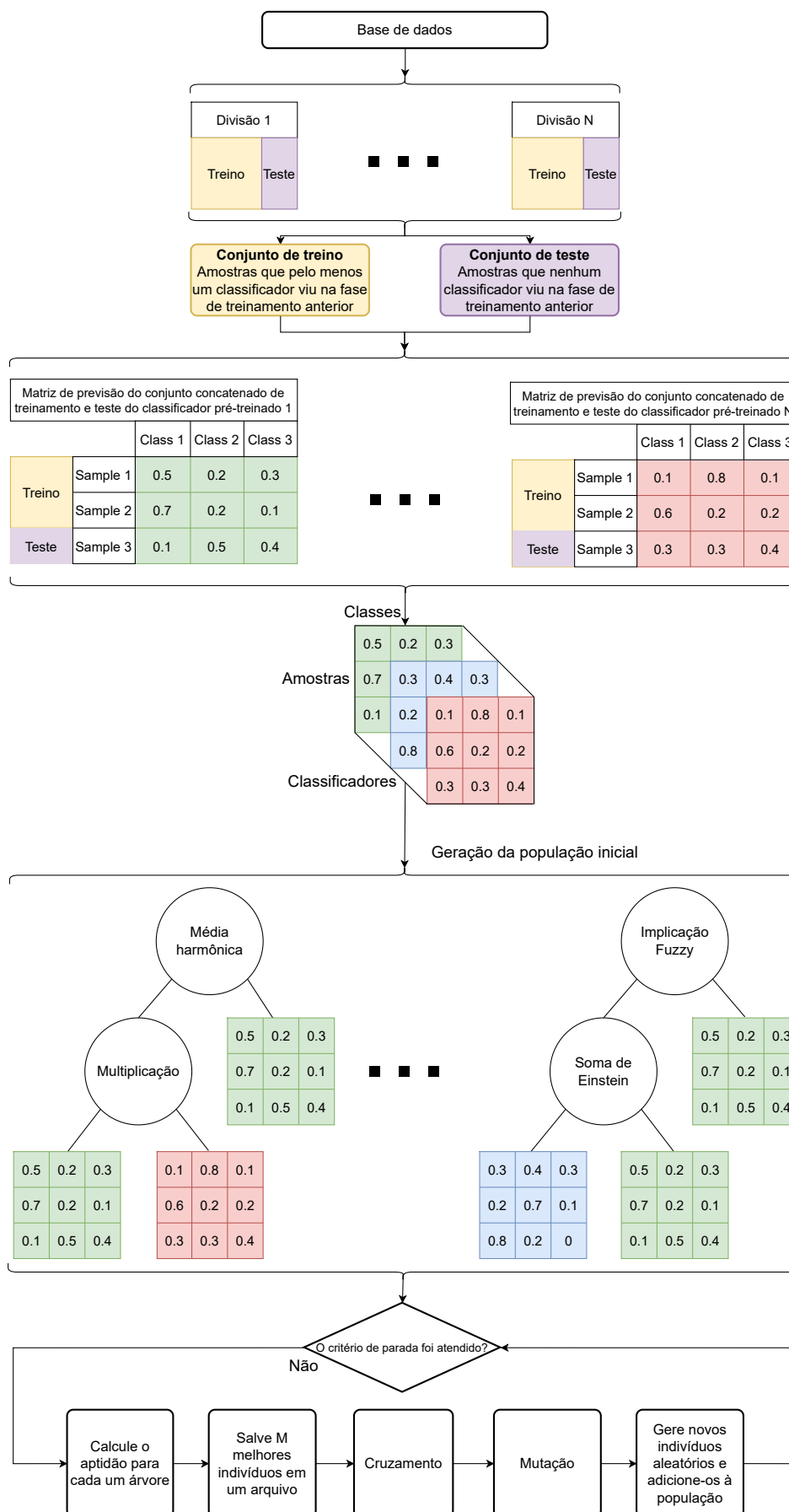


Figura 6 – Fluxo de trabalho completo referente à metodologia empregada neste estudo.

- Profundidade máxima da árvore (`max_depth`):  $[2, 7]$ .
- Tamanho da população (`population_size`): 20 para HMDB51 e 10 para UCF101, respectivamente.
- Novos indivíduos (`new_individuals`): 10 para HMDB51 e 5 para UCF101, respectivamente.
- Número de gerações: 400.
- Taxa de mutação (`mutation_rate`): 0.5.

Seis grupos de funções são utilizados neste trabalho: Matemáticas, Fuzzy, Geométricas, Médias, Médias Ponderadas e Funções Autoaplicáveis. Todas essas funções seguem uma premissa crucial: elas aceitam valores de entrada no intervalo  $[0, 1]$  e retornam valores dentro do mesmo intervalo. As funções específicas podem ser encontradas nas Tabelas 1 e 2. Funções de média ponderada e funções matemáticas autoaplicáveis são a base para as funções centrais do sistema, que utilizam métricas para suas operações. As métricas disponíveis no sistema incluem acurácia, pontuação F1, pontuação de precisão, pontuação de revocação, pontuação de Jaccard e suas inversas ( $1 - \text{métrica}$ ).

O projeto do algoritmo GP garante que todos os indivíduos ou soluções gerados sejam válidos. Isto é conseguido através do emprego de funções que operam na faixa de 0 a 1, tanto para entradas quanto para saídas, conforme detalhado nas Tabelas 1 e 2. A seleção aleatória dessas funções durante a criação de um nó é parte integrante do algoritmo, potencializando a diversidade das soluções. Esta abordagem elimina a possibilidade de geração de indivíduos inválidos, mantendo a consistência e validade das combinações de classificadores ao longo do processo.

Tabela 1 – Funções Matemáticas e Fuzzy

| Matemática             |                                                                                |
|------------------------|--------------------------------------------------------------------------------|
| Nome                   | Função                                                                         |
| Mult                   | $x \times y$                                                                   |
| Fuzzy                  |                                                                                |
| Nome                   | Função                                                                         |
| OR                     | $\max(x, y)$                                                                   |
| NOR                    | $1 - \max(x, y)$                                                               |
| AND                    | $\min(x, y)$                                                                   |
| NAND                   | $1 - \min(x, y)$                                                               |
| NOT                    | $1 - x$                                                                        |
| XOR                    | $\text{abs}(x - y)$                                                            |
| XNOR                   | $1 - \text{abs}(x - y)$                                                        |
| Implicação             | $\min(1 - x, y)$                                                               |
| Concentração           | $x^2$                                                                          |
| Dilatação              | $\sqrt{x}$                                                                     |
| Soma Algébrica         | $x + y - x \times y$                                                           |
| Soma Limitada          | $\min(x + y, 1)$                                                               |
| Diferença Limitada     | $\max(x - y, 0)$                                                               |
| Mais ou Menos          | $0.5 \times (\sqrt{x} + x)$                                                    |
| Implicação Godel       | $1 \text{ se } x \leq y \text{ senão } y$                                      |
| Implicação Lukasiewicz | $\min(1, 1 - x + y)$                                                           |
| Soma de Einstein       | $\frac{x+y}{1+x \times y}$                                                     |
| Produto de Einstein    | $\frac{x \times y}{2 - (x+y-x \times y)}$                                      |
| Negação Yager          | $\sqrt{1 - x^2}$                                                               |
| Implicação Mamdani     | $\min(x, y)$                                                                   |
| Implicação Zadeh       | $\max(1 - x, \min(x, y))$                                                      |
| Operador Gamma         | $(x^z \times y^z)^{\frac{1}{z}}$                                               |
| Implicação Goguen      | $\min\left(\frac{y}{x}, 1\right) \text{ if } x > 0 \text{ else } 1$            |
| Soma Hamacher Limitada | $\frac{x+y-2 \times x \times y}{\max(1-x \times y, 1e-6)}$                     |
| Produto Hamacher       | $\text{clamp}\left(\frac{x \times y}{\max(x+y-x \times y, 1e-6)}, 0, 1\right)$ |
| Exponencial            | $\exp(-(1-x)^2)$                                                               |
| Logística              | $\frac{1}{1+\exp(-x)}$                                                         |
| Contraste Sigmoidal    | $\frac{1}{1+\exp(-z \times (x-y))}$                                            |

Tabela 2 – Funções Geométricas, Média, Média Ponderada e Matemática Autoaplicável

| Geométrica                 |                                                                                           |
|----------------------------|-------------------------------------------------------------------------------------------|
| Nome                       | Função                                                                                    |
| tanh                       | $\frac{\tanh(x)+1}{2}$                                                                    |
| circular                   | $\sqrt{\text{clamp}(1-x^2, 0, 1)}$                                                        |
| elíptica                   | $\sqrt{\text{clamp}(1-x^2-y^2, 0, 1)}$                                                    |
| Parabólica                 | $\text{clamp}(1-x^2, 0, 1)$                                                               |
| Seno                       | $\frac{\sin(x)+1}{2}$                                                                     |
| Cosseno                    | $\frac{\cos(x)+1}{2}$                                                                     |
| Sigmoidal                  | $\frac{1}{1+\exp(-10 \times (x-0.5))}$                                                    |
| Média                      |                                                                                           |
| Nome                       | Função                                                                                    |
| Média                      | $\frac{x+y}{2}$                                                                           |
| Média Geométrica           | $\sqrt{x \times y}$                                                                       |
| Média Harmônica            | $\frac{2}{\frac{1}{\max(x,\epsilon)} + \frac{1}{\max(y,\epsilon)}}$                       |
| Média Quadrática           | $\sqrt{\frac{x^2+y^2}{2}}$                                                                |
| Média Cúbica               | $\left(\frac{x^3+y^3}{2}\right)^{\frac{1}{3}}$                                            |
| Média Ponderada            |                                                                                           |
| Nome                       | Função                                                                                    |
| Média Ponderada            | $\frac{x \times wx + y \times wy}{\max(wx+wy, \epsilon)}$                                 |
| Média Geométrica Ponderada | $\left(\sqrt{x^{wx} \times y^{wy}}\right)^{\frac{1}{\max(wx+wy, \epsilon)}}$              |
| Média Harmônica Ponderada  | $\frac{\frac{wx}{\max(x,\epsilon)} + \frac{wy}{\max(y,\epsilon)}}{\max(wx+wy, \epsilon)}$ |
| Média Quadrática Ponderada | $\sqrt{\frac{x^2 \times wx + y^2 \times wy}{\max(wx+wy, \epsilon)}}$                      |
| Média Cúbica Ponderada     | $\left(\frac{x^3 \times wx + y^3 \times wy}{\max(wx+wy, \epsilon)}\right)^{\frac{1}{3}}$  |
| Matemática Autoaplicável   |                                                                                           |
| Nome                       | Função                                                                                    |
| Auto Multiplicação         | $x \times f(x)$                                                                           |

## 4 Configuração e Resultados experimentais

Esta seção aprofunda-se na análise experimental abrangente realizada utilizando os conjuntos de dados HMDB51 e UCF101. Empregando classificadores avançados e Programação Genética (PG), os experimentos visam explorar e aprimorar as capacidades do nosso algoritmo. As descobertas destes experimentos fornecem insights valiosos sobre a eficácia da PG na otimização de comitês de classificadores, evidenciando o potencial destes métodos no campo do aprendizado de máquina.

### 4.1 Classificadores

Os classificadores utilizados neste estudo foram os mesmos empregados por (SANTOS; ALMEIDA, 2020), conforme descrito a seguir:

- Classificador MV: utiliza Vetores de Movimento (MV) para classificação de vídeo. A arquitetura ResNet-18 se concentra na captura e análise de variações de movimento entre quadros consecutivos. Neste contexto, MV serve como um meio para destacar diferenças dinâmicas e mudanças entre os quadros, essencial para compressão eficiente de vídeo e para o reconhecimento de padrões de movimento (SANTOS; ALMEIDA, 2020).
- Classificador DCT: emprega a Transformada Discreta de Cosseno para classificar vídeos na arquitetura ResNet-50. Esta abordagem usa DCT para converter informações visuais de  $n$ -frames em uma representação de frequência. Esta conversão permite a identificação mais eficiente de características visuais importantes para compressão e análise de vídeo, tornando o DCT um método eficaz para processar grandes volumes de dados visuais e reduzi-los a componentes essenciais para a classificação (SANTOS; ALMEIDA, 2020).
- Classificadores DCT com FBS (DCT com Seleção de Banda de Frequência) 16 e 32: representam variações avançadas da arquitetura ResNet-50, adaptadas para incluir a técnica de Seleção de Banda de Frequência (FBS). O FBS é usado para selecionar os coeficientes DCT mais relevantes em cada  $n$ -frames. As variantes "DCT com FBS 16" e "DCT com FBS 32" diferem no número de coeficientes DCT processados por canal de cor. Essas abordagens são projetadas para análises mais precisas e eficientes de dados de vídeo comprimidos, focando na otimização do reconhecimento de ações em vídeos (SANTOS; ALMEIDA, 2020).

## 4.2 HMDB51

Esta seção apresenta os resultados experimentais obtidos com os 12 classificadores treinados fornecidos por (SANTOS; ALMEIDA, 2020). Os classificadores foram treinados em três divisões criadas pela seleção aleatória de vídeos do conjunto de dados completo do HMDB51, denotadas como  $D$ . Cada divisão consiste em seus próprios subconjuntos de treinamento e teste, denotados como  $D_i^{Treino}$  e  $D_i^{Teste}$ , respectivamente, para  $(i = 1, 2, 3)$ .

Para avaliar a GP na poda de conjunto, definimos um novo conjunto de treinamento como a união dos subconjuntos de treinamento de todas as três divisões,  $D^{GPTreino} = D_1^{Treino} \cup D_2^{Treino} \cup D_3^{Treino}$ , o que corresponde a aproximadamente 90% do conjunto de dados HMDB51 completo. O novo conjunto de teste,  $D^{GPTeste}$ , compreende os elementos em  $D$  que não estão presentes em  $D^{GPTreino}$ , formalmente representado como  $D^{GPTeste} = D - D^{GPTreino}$ , constituindo os 10% restantes do conjunto de dados.

A Tabela 3 mostra a acurácia de cada classificador nesses novos conjuntos, bem como o melhor resultado obtido pelo algoritmo GP, destacado em negrito. O algoritmo GP foi executado em diferentes profundidades (2 a 7), considerando a função de aptidão como a acurácia no treinamento. Os três melhores resultados obtidos em cada profundidade são apresentados na Tabela 5 (ver Seção 4.4).

No contexto do conjunto de dados HMDB51, conforme ilustrado na Tabela 3, os resultados demonstram a eficácia do algoritmo. A tabela descreve a acurácia de cada classificador em diferentes divisões de treinamento e teste. Isso fornece insights sobre a adaptabilidade e eficiência de nossa abordagem GP ao lidar com características variadas de dados, mostrando sua capacidade de melhorar significativamente a acurácia da classificação.

## 4.3 UCF101

De maneira semelhante à Seção 4.2, esta seção detalha os resultados experimentais obtidos com o uso de classificadores no conjunto de dados UCF101. Para este experimento, foram empregados todos os quatro classificadores da primeira divisão do conjunto de dados UCF101, denominada Divisão 1. O conjunto de treinamento,  $D^{UCFTreino}$ , foi definido como o subconjunto de treinamento da Divisão 1,  $D_1^{Treino}$ . O conjunto de teste,  $D^{UCFTest}$ , foi construído subtraindo o conjunto de treinamento do conjunto de dados UCF101 completo, representado como  $D^{UCFTest} = D^{UCF} - D^{UCFTreino}$ . Essa abordagem foi necessária porque a inclusão de uma divisão adicional resultaria em todos os vídeos da base de dados sendo visualizados por pelo menos um classificador, o que poderia distorcer os resultados da pesquisa.

Tabela 3 – Acurácia dos classificadores nos novos subconjuntos baseado no trabalho de (SANTOS; ALMEIDA, 2020).

| Divisão 1             |               |               |
|-----------------------|---------------|---------------|
| Nome do Classificador | Treinamento   | Teste         |
| MV                    | 70,17%        | 32,84%        |
| DCT c/ FBS (16)       | 70,77%        | 34,27%        |
| DCT c/ FBS (32)       | 75,92%        | 30,87%        |
| DCT                   | 73,95%        | 29,62%        |
| Divisão 2             |               |               |
| Nome do Classificador | Treinamento   | Teste         |
| MV                    | 70,53%        | 37,33%        |
| DCT c/ FBS (16)       | 72,52%        | 35,00%        |
| DCT c/ FBS (32)       | 70,09%        | 29,60%        |
| DCT                   | 71,05%        | 33,87%        |
| Divisão 3             |               |               |
| Nome do Classificador | Treinamento   | Teste         |
| MV                    | 79,07%        | 34,70%        |
| DCT c/ FBS (16)       | 66,43%        | 33,18%        |
| DCT c/ FBS (32)       | 70,79%        | 26,95%        |
| DCT                   | 62,45%        | 24,41%        |
| Conjunto              |               |               |
| Nome do Classificador | Treinamento   | Teste         |
| <b>Nosso</b>          | <b>86,06%</b> | <b>58,59%</b> |

A Tabela 4 apresenta a acurácia de cada um dos quatro classificadores nesses conjuntos de dados. O resultado do método de poda de conjunto utilizando GP é destacado em negrito. É importante ressaltar que o algoritmo GP foi aplicado em profundidades variadas (2 a 7), com as acurácias em treinamento, teste e o valor combinado sendo detalhados na Tabela 6 na Seção 4.4.

Tabela 4 – Acurácia dos 4 classificadores do split 1 no novo subconjunto do UCF101, baseado no trabalho de (SANTOS; ALMEIDA, 2020).

| Divisão 1             |                |               |
|-----------------------|----------------|---------------|
| Nome do classificador | Treinamento    | Teste         |
| MV                    | 97,53%         | 73,86%        |
| DCT c/ FBS (16)       | 99,90%         | 76,84%        |
| DCT c/ FBS (32)       | 99,98%         | 78,77%        |
| DCT                   | 99,97%         | 77,37%        |
| Conjunto              |                |               |
| Nome do classificador | Treinamento    | Teste         |
| <b>Nosso</b>          | <b>100,00%</b> | <b>80,60%</b> |

Em relação ao conjunto de dados UCF101, conforme detalhado na Tabela 4, os resultados destacam a robustez do algoritmo. A tabela exhibe as métricas de desempenho dos classificadores, enfatizando a maior acurácia alcançada através de nossa

abordagem. Isto ressalta o potencial do GP na otimização dos comitês classificadores, mesmo em cenários onde o desempenho inicial do classificador já é alto.

## 4.4 ABLAÇÃO

As Tabelas 5 e 6 apresentam os resultados completos da ablação para os conjuntos de dados HMDB51 e UCF101, respectivamente. Estes resultados foram obtidos executando o algoritmo GP em diferentes profundidades (variando de 2 a 7).

Tabela 5 – Top três árvores por profundidade máxima no conjunto de dados HMDB51 considerando 16 classificadores.

| Profundidade 2 |             |        |        |
|----------------|-------------|--------|--------|
| Classificação  | Treinamento | Teste  | Média  |
| 1              | 88,32%      | 41,08% | 64,70% |
| 2              | 39,15%      | 53,48% | 46,21% |
| 3              | 88,32%      | 41,08% | 64,70% |
| Profundidade 3 |             |        |        |
| Classificação  | Treinamento | Teste  | Média  |
| 1              | 90,87%      | 45,57% | 68,22% |
| 2              | 41,17%      | 63,00% | 55,09% |
| 3              | 82,92%      | 54,57% | 68,74% |
| Profundidade 4 |             |        |        |
| Classificação  | Treinamento | Teste  | Média  |
| 1              | 92,20%      | 45,23% | 68,71% |
| 2              | 50,32%      | 66,19% | 58,25% |
| 3              | 89,80%      | 48,92% | 69,36% |
| Profundidade 5 |             |        |        |
| Classificação  | Treinamento | Teste  | Média  |
| 1              | 91,46%      | 41,50% | 66,48% |
| 2              | 46,46%      | 65,00% | 55,73% |
| 3              | 87,89%      | 50,67% | 69,28% |
| Profundidade 6 |             |        |        |
| Classificação  | Treinamento | Teste  | Média  |
| 1              | 91,98%      | 43,35% | 67,66% |
| 2              | 47,81%      | 66,69% | 57,25% |
| 3              | 91,05%      | 46,98% | 69,01% |
| Profundidade 7 |             |        |        |
| Classificação  | Treinamento | Teste  | Média  |
| 1              | 90,70%      | 38,01% | 64,35% |
| 2              | 49,57%      | 63,46% | 56,52% |
| 3              | 86,06%      | 58,59% | 72,32% |

Tabela 6 – Top três árvores por profundidade máxima no conjunto de dados UCF101 considerando quatro classificadores.

| Profundidade 2 |             |        |        |
|----------------|-------------|--------|--------|
| Classificação  | Treinamento | Teste  | Média  |
| 1              | 100,00%     | 79,79% | 89,89% |
| 2              | 99,56%      | 81,45% | 90,50% |
| 3              | 99,91%      | 81,36% | 90,63% |
| Profundidade 3 |             |        |        |
| Classificação  | Treinamento | Teste  | Média  |
| 1              | 100,00%     | 79,79% | 89,89% |
| 2              | 99,99%      | 82,73% | 91,36% |
| 3              | 99,99%      | 82,73% | 91,36% |
| Profundidade 4 |             |        |        |
| Classificação  | Treinamento | Teste  | Média  |
| 1              | 100,00%     | 80,30% | 90,15% |
| 2              | 99,99%      | 82,89% | 91,44% |
| 3              | 99,99%      | 82,89% | 91,44% |
| Profundidade 5 |             |        |        |
| Classificação  | Treinamento | Teste  | Média  |
| 1              | 100,00%     | 80,77% | 90,38% |
| 2              | 99,96%      | 82,93% | 91,44% |
| 3              | 99,96%      | 82,93% | 91,44% |
| Profundidade 6 |             |        |        |
| Classificação  | Treinamento | Teste  | Média  |
| 1              | 100,00%     | 79,14% | 89,57% |
| 2              | 99,98%      | 83,23% | 91,60% |
| 3              | 99,98%      | 83,23% | 91,60% |
| Profundidade 7 |             |        |        |
| Classificação  | Treinamento | Teste  | Média  |
| 1              | 100,00%     | 80,66% | 90,33% |
| 2              | 99,96%      | 83,20% | 91,58% |
| 3              | 99,96%      | 83,20% | 91,58% |

## 5 Discussão

Pode-se observar que a GP é hábil em identificar novas funções para combinar as previsões dos membros do comitê, embora isso às vezes possa levar a uma redução de desempenho em outros, semelhante ao sobreajuste. Notavelmente, existem casos em que as funções derivadas da GP aumentam o desempenho tanto nos conjuntos de treinamento quanto de teste simultaneamente, indicando um entendimento matizado das características dos dados.

Um exemplo claro é o resultado na profundidade 7 no conjunto de dados HMDB51, observou-se um aumento de 9,01% na acurácia do conjunto de treinamento e um aumento significativo de 21,26% na acurácia do conjunto de teste, considerando o melhor classificador individual. Isso exemplifica a versatilidade da GP como ferramenta, permitindo que os usuários adaptem a função às suas necessidades específicas, seja priorizando a acurácia do conjunto de treinamento, a acurácia do conjunto de teste ou buscando um equilíbrio entre os dois.

No conjunto de dados UCF101, foi encontrado um desafio diferente. Os classificadores já demonstraram altos níveis de acurácia, alcançando até 99,96% de acurácia para treinamento e 83,20% de acurácia para teste, deixando um espaço limitado para melhoria. No entanto, ao considerar a profundidade da árvore de tamanho 7, observou-se um exemplo notável da eficácia da GP em otimizar ainda mais esses resultados. Apesar de uma perda mínima de 0,4% na acurácia do treinamento, houve um ganho significativo de 4,43% na acurácia do conjunto de teste, destacando a capacidade da GP em ajustar o desempenho dos classificadores, mesmo em cenários onde a margem de melhoria parece ser mínima.

A flexibilidade oferecida pela GP, demonstrada pelas variações de desempenho em diferentes profundidades (notavelmente profundidades 6 e 7), demonstra sua utilidade em diversos cenários de classificação. Essa adaptabilidade pode ser usada para ajustar os modelos, alcançando um equilíbrio ótimo entre os desempenhos de treinamento e teste que estejam alinhados com os objetivos e restrições específicos.

A inovação do método proposto reside na otimização da função de combinação de classificadores. Diferentemente das abordagens convencionais que se concentram apenas na seleção e poda de classificadores, nosso método explora a otimização da função de combinação. Essa abordagem traz uma nova perspectiva para o aprimoramento do desempenho dos sistemas de classificação, destacando-se por não se limitar a uma função de combinação fixa, como a média, mas sim por explorar uma gama mais ampla de possibilidades.

Além disso, o algoritmo desenvolvido se destaca pela sua eficiência, tanto em velocidade quanto em custo computacional. Isso torna a aplicação viável em diversos cenários, especialmente aqueles com classificadores imperfeitos, garantindo que, no mínimo, o método converge para uma função de classificação básica, mas eficiente.

Finalmente, uma das grandes vantagens do método é sua aplicabilidade em classificadores de diferentes tipos de mídia. Por exemplo, ao integrar classificadores de vídeo e áudio, a Programação Genética pode analisar e ponderar a relevância de cada classificador e tipo de mídia, resultando em uma classificação mais precisa e abrangente. Essa capacidade de integrar informações multimídia de forma eficaz é um passo significativo para avançar nas técnicas de reconhecimento de dados.

## 6 Conclusão

Neste estudo, alcançamos com sucesso o objetivo de explorar a aplicação da GP na otimização de comitês de classificadores em análises de dados em larga escala. Demonstramos efetivamente como a GP pode ser utilizada para identificar a combinação ideal de classificadores e desenvolver funções de combinação de predição que maximizam a eficácia da classificação. Os resultados obtidos confirmam que a GP pode substituir experimentações manuais demoradas e onerosas, consolidando-se como uma ferramenta valiosa no domínio do ML.

Nossa pesquisa valida a hipótese de que a aplicação da GP na otimização de comitês de classificadores resulta em melhorias significativas no desempenho da classificação, comparado com métodos convencionais. A capacidade da GP de explorar eficientemente o espaço de soluções, identificando classificadores ótimos e funções de combinação de predição adaptativas, foi claramente demonstrada. Os experimentos conduzidos revelaram que os comitês otimizados pela GP superam aqueles que utilizam funções de combinação padrão, como a média e a média ponderada, corroborando a hipótese inicial e destacando o potencial transformador da GP no campo do ML.

### 6.1 Trabalhos futuros

Em trabalhos futuros, é intrigante expandir o repertório de funções usadas pela GP, potencialmente descobrindo métodos de combinação ainda mais eficazes para classificadores. Além disso, experimentar com diferentes tipos de classificadores além dos usados neste estudo poderia fornecer insights mais profundos sobre a versatilidade e eficácia da GP em vários contextos de classificação.

Outra direção significativa é aplicar a GP a diferentes tipos de dados, como áudio, imagens, texto e vídeo. Investigar como a GP se comporta nessas diferentes modalidades poderia revelar desafios e oportunidades únicos, aprimorando ainda mais a compreensão de sua adaptabilidade e eficácia. Testar a GP nesses variados cenários ampliará sua aplicabilidade e contribuirá para o desenvolvimento de modelos de aprendizado de máquina mais robustos e versáteis, capazes de lidar com conjuntos de dados complexos e multi-modais.

# Referências

ALZUBAIDI, L.; ZHANG, J.; HUMAIDI, A. J.; AL-DUJAILI, A.; DUAN, Y.; AL-SHAMMA, O.; SANTAMARÍA, J.; FADHEL, M. A.; AL-AMIDIE, M.; FARHAN, L. Review of deep learning: concepts, cnn architectures, challenges, applications, future directions. *Journal of Big Data*, v. 8, n. 1, p. 53, Mar 2021. ISSN 2196-1115. Disponível em: <<https://doi.org/10.1186/s40537-021-00444-8>>.

BACK, T. Evolutionary algorithms: Theory and practice. *Evolutionary Computation*, MIT Press, v. 6, n. 3, p. 235–245, 2018.

BALOGLU, O.; LATIFI, S. Q.; NAZHA, A. What is machine learning? *Archives of Disease in Childhood-Education and Practice*, Royal College of Paediatrics and Child Health, v. 107, n. 5, p. 386–388, 2022.

BOATENG, E. Y.; OTOO, J.; ABAYE, D. A. Basic tenets of classification algorithms k-nearest-neighbor, support vector machine, random forest and neural network: a review. *Journal of Data Analysis and Information Processing*, Scientific Research Publishing, v. 8, n. 4, p. 341–357, 2020.

BORMIDA, M. D. The big data world: Benefits, threats and ethical challenges. In: \_\_\_\_\_. Emerald Publishing Limited, 2021. (Advances in Research Ethics and Integrity, v. 8), p. 71–91. Disponível em: <<https://doi.org/10.1108/S2398-601820210000008007>>.

CACCHIANI, V.; IORI, M.; LOCATELLI, A.; MARTELLO, S. Knapsack problems—an overview of recent advances. part ii: Multiple, multidimensional, and quadratic knapsack problems. *Computers & Operations Research*, Elsevier, v. 143, p. 105693, 2022.

CHEIKHROUHOU, O.; KHOUFI, I. A comprehensive survey on the multiple traveling salesman problem: Applications, approaches and taxonomy. *Computer Science Review*, Elsevier, v. 40, p. 100369, 2021.

CORMEN, T. H.; LEISERSON, C. E. *Introduction to Algorithms, fourth edition*. London, England: MIT Press, 2022. 1332 p. ISBN 9780262046305.

DARGAN, S.; KUMAR, M. A comprehensive survey on the biometric recognition systems based on physiological and behavioral modalities. *Expert Systems with Applications*, Elsevier, v. 143, p. 113114, 2020.

DEFENDANTS'OPPOSITION, W. D. I. The youtube service and terms of use. 2018.

DONG, X.; YU, Z.; CAO, W.; SHI, Y.; MA, Q. A survey on ensemble learning. *Frontiers of Computer Science*, Springer Science and Business Media LLC, v. 14, n. 2, p. 241–258, ago. 2019. ISSN 2095-2236. Disponível em: <<http://dx.doi.org/10.1007/s11704-019-8208-z>>.

DU, D.-Z.; PARDALOS, P. M.; WU, W. History of optimizationhistory of optimization. In: \_\_\_\_\_. *Encyclopedia of Optimization*. Boston, MA: Springer US, 2009. p. 1538–1542. ISBN 978-0-387-74759-0. Disponível em: <[https://doi.org/10.1007/978-0-387-74759-0\\_268](https://doi.org/10.1007/978-0-387-74759-0_268)>.

EIBEN, A. E.; SMITH, J. E. *Introduction to Evolutionary Computing*. [S.l.]: Springer, 2015.

FALCAO, A. X.; PAPA, J. P. *Optimum-Path Forest: Theory, Algorithms, and Applications*. [S.l.]: Academic Press, 2022.

KUEHNE, H.; JHUANG, H.; GARROTE, E.; POGGIO, T.; SERRE, T. HMDB: a large video database for human motion recognition. In: *Proceedings of the International Conference on Computer Vision (ICCV)*. [S.l.: s.n.], 2011.

MIENYE, I. D.; SUN, Y. A survey of ensemble learning: Concepts, algorithms, applications, and prospects. *IEEE Access*, IEEE, v. 10, p. 99129–99149, 2022.

NOSIL, P.; FLAXMAN, S. M.; FEDER, J. L.; GOMPERT, Z. Increasing our ability to predict contemporary evolution. *Nat. Commun.*, Springer Science and Business Media LLC, v. 11, n. 1, p. 5592, nov. 2020.

ROSA, G. H. de; RODER, M.; PAPA, J. P. Neighbour-based bag-of-samplings for person identification through handwritten dynamics and convolutional neural networks. *Expert Systems*, v. 39, n. 4, p. e12891, 2022. Disponível em: <<https://onlinelibrary.wiley.com/doi/abs/10.1111/exsy.12891>>.

SANTOS, S. F.; ALMEIDA, J. Faster and accurate compressed video action recognition straight from the frequency domain. In: *2020 33rd SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*. IEEE, 2020. Disponível em: <<https://doi.org/10.1109%2Fsibgrapi51738.2020.00017>>.

SANTOS, S. F. dos; SEBE, N.; ALMEIDA, J. Cv-c3d: Action recognition on compressed videos with convolutional 3d networks. p. 24–30, 2019.

SIMPSON, K. The digital archive as space and place in the constitution, production and circulation of knowledge. *Humanities of the Future*, p. 65–66, 2020.

SOOMRO, K.; ZAMIR, A. R.; SHAH, M. *UCF101: A Dataset of 101 Human Actions Classes From Videos in The Wild*. 2012.

TECH the-next. *How much data is produced every day 2021?* 2021. Disponível em: <<https://www.the-next-tech.com/blockchain-technology/how-much-data-is-produced-every-day-2019/>>.

TOY story. 2016. Disponível em: <<https://www.oscars.org/collection-highlights/toy-story>>.

VUJOVIĆ, Ž. et al. Classification model evaluation metrics. *International Journal of Advanced Computer Science and Applications*, v. 12, n. 6, p. 599–606, 2021.

YOO, K.-H.; LEUNG, C. K.; NASRIDINOV, A. Big data analysis and visualization: Challenges and solutions. *Applied Sciences*, v. 12, n. 16, 2022. ISSN 2076-3417. Disponível em: <<https://www.mdpi.com/2076-3417/12/16/8248>>.

ZHAN, Z.-H.; SHI, L.; TAN, K. C.; ZHANG, J. A survey on evolutionary computation for complex continuous optimization. *Artificial Intelligence Review*, Springer, p. 1–52, 2022.