

**CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS
CAMPUS DIVINÓPOLIS**

Sávio Rodrigues

CLASSIFICADOR FUZZY EVOLUTIVO MULTICLASSE

Divinópolis

2023

SÁVIO RODRIGUES

CLASSIFICADOR FUZZY EVOLUTIVO MULTICLASSE

Trabalho de Conclusão de Curso apresentado no curso de Graduação em Engenharia de Computação do Centro Federal de Educação Tecnológica de Minas Gerais como requisito parcial para obtenção do título de Bacharel em Engenharia de Computação.

Orientador: Alisson Marques da Silva

DIVINÓPOLIS

2023

SÁVIO RODRIGUES

CLASSIFICADOR FUZZY EVOLUTIVO MULTICLASSE

Trabalho de Conclusão de Curso apresentado no curso de Graduação em Engenharia de Computação do Centro Federal de Educação Tecnológica de Minas Gerais como requisito parcial para obtenção do título de Bacharel em Engenharia de Computação.

Aprovado em 12 de dezembro de 2023



Alisson Marques da Silva
CEFET-MG

Thabatta Moreira Alves de Araújo
CEFET-MG

Tiago Alves de Oliveira
CEFET-MG

RESUMO

Este trabalho propõe um classificador fuzzy evolutivo denominado eMGMC (*evolving Multivariable Gaussian Multiclass Classifier*), que se baseia em um algoritmo de agrupamento de aprendizado participativo. A estrutura de regras do eMGMC é gerada do zero, em que os antecedentes são modelados por funções gaussianas multivariáveis, enquanto os consequentes representam as classes. O algoritmo incorpora mecanismos de atualização, mesclagem e exclusão de grupos e regras. A criação de grupos e regras é guiada pela medida de compatibilidade, calculada por meio da distância de Mahalanobis, e um método de procrastinação. Para a exclusão de grupos inativos, são utilizados conceitos de idade e população, enquanto a mesclagem de grupos redundantes é realizada com base em um limiar de sobreposição. Os experimentos realizados visam avaliar o desempenho do algoritmo proposto em comparação com outros modelos do estado da arte. Os resultados destacam a superioridade ou, no mínimo, comparabilidade do eMGMC em relação a outros modelos, evidenciando a confiabilidade do algoritmo em diversas configurações de dados de entrada.

Palavras-chave: Classificador , Fuzzy evolutivo, multiclasse, funções multivariáveis.

ABSTRACT

This work proposes an evolutionary fuzzy classifier named eMGMC (evolving Multivariable Gaussian Multiclass Classifier), which is based on a participatory learning clustering algorithm. The rule structure of eMGMC is generated from scratch, where antecedents are modeled by multivariate Gaussian functions, and consequents represent classes. The algorithm incorporates mechanisms for updating, merging, and deleting groups and rules. The creation of groups and rules is guided by the compatibility measure, calculated through the Mahalanobis distance, and a procrastination method. For the deletion of inactive groups, concepts of age and population are employed, while the merging of redundant groups is performed based on an overlap threshold. The experiments conducted aim to assess the performance of the proposed algorithm compared to other state-of-the-art models. The results highlight the superiority or, at the very least, comparability of eMGMC to other models, demonstrating the reliability of the algorithm across various input data configurations.

Keywords: Classifier, Evolutionary Fuzzy, Multiclass, Multivariate Functions.

LISTA DE ILUSTRAÇÕES

Figura 1 – Fluxograma do eMGMC.	10
Figura 2 – Evolução da estrutura do eMGMC.	28

LISTA DE TABELAS

Tabela 1 – Conjuntos de dados.	24
Tabela 2 – Variação e melhores valores dos parâmetros.	25
Tabela 3 – Desempenho do classificadores.	26
Tabela 4 – Posição dos algoritmos	27

LISTA DE ABREVIATURAS E SIGLAS

eMGMC	<i>Evolving Multivariable Gaussian Multiclass Classifier</i>
eMG	<i>Evolving Multivariable Gaussian</i>
eFS	<i>Evolving Fuzzy Systems</i>
eTS	<i>Evolving Takagi-sugeno</i>
FLEXFIS	<i>Flexible Evolving Fuzzy Inference Systems Classifier</i>
eT2Class	<i>Evolving Type-2 Classifier</i>
ALMMo	<i>Autonomous Learning Multi-Model Systems</i>
eiX	<i>Evolving Internal-eXternal Fuzzy Clustering</i>
eNFN	<i>Evolving Neuro-Fuzzy</i>
eFMC	<i>evolving Fuzzy Mean Classifier</i>
FBeM	<i>Fuzzy-set-Based evolving Modeling</i>
eOGS	<i>Evolving optimal granular systems</i>
IBeM	<i>Interval Based evolving Modeling</i>
eGNN	<i>Evolving Neo-Fuzzy Neuron</i>

LISTA DE SÍMBOLOS

v	Centros dos grupos
x	Vetor de variáveis de entrada
t	Passo atual
i	Índice dos centros, grupos e regras
j	Índice dos centros, grupos e regras
m	Número de variáveis de entrada
c	Número de grupos ativos criados
k	Número classes criados
a	Quantidade de amostras por grupo ativo
Σ	Matriz de dispersão de grupos ativos
Ω	Vetor de índices de grupos ativos
Θ	Vetor de classes de grupos ativos
y	Saída desejada
\hat{y}	Saída do modelo
l	Índice das classes dos grupos ativos
\hat{y}_i	Saída da i -ésima regra
R	Regra
B	Função de pertinência gaussiana multivariável
$g\Theta$	Vetor de número de grupos por classe
τ	Grau de ativação normalizado
D	Distância de Mahalanobis
i^+	Regra/Grupo com maior grau de ativação
S	Compatibilidade da amostra em relação aos grupos
λ	Taxa de aprendizado
γ	Medida de compatibilidade
$\neg\exists$	Não existe
\exists	Existe
o	Número de classes dos grupos ativos
$\bar{\Theta}$	Vetor de classes inativos
\bar{l}	Índice das classes dos grupos inativos
\bar{o}	Número de classes dos grupos inativos
\bar{v}	Centros dos grupos inativos
\bar{i}	Índice dos grupos inativos
\bar{c}	Número de grupos inativos criados
\bar{a}	Quantidade de amostras por grupo inativo
$\bar{\Sigma}$	Matriz de dispersão de grupos inativos
$\bar{\Omega}$	Vetor de índices de grupos inativos

$g\bar{\Theta}$	Vetor de números de grupos por classe
$\bar{\eta}$	Limiar de ativação de grupo inativo
ρ	Limiar de medida de compatibilidade/sobreposição
i^*	Matriz de dispersão de grupos inativos
A_{cc}	Acurácia
σ	Desvio Padrão
P	Numero total de amostras
f	Valores obtidos nas execuções
\bar{f}	Média dos valores obtidos nas execuções
N	Numero de execuções
<i>idade</i>	Idade dos grupos
ω	Limiar de deleção de grupos e regras

SUMÁRIO

1	INTRODUÇÃO	1
1.1	Contextualização e Relevância	1
1.2	Objetivos	2
1.2.1	Objetivo Geral	2
1.2.2	Objetivos Específicos	2
1.3	Organização do Trabalho	3
2	REVISÃO DA LITERATURA	4
2.1	Sistemas Fuzzy	4
2.2	Sistemas <i>Fuzzy</i> Evolutivos	5
2.3	Trabalhos Relacionados	6
3	ABORDAGEM PROPOSTA	9
3.1	Introdução	9
3.2	Inicialização	11
3.3	Cálculo de Saída	12
3.4	Criação e Atualização de Grupos e Regras	13
3.5	Mesclagem de Grupos e Regras	18
3.6	Exclusão de Grupos e Regras	20
3.7	Parâmetros do eMGMC	21
4	EXPERIMENTOS E RESULTADOS	23
4.1	Metodologia	23
4.2	Resultados Experimentais	26
5	CONCLUSÃO	29
	REFERÊNCIAS	30

1 INTRODUÇÃO

Neste capítulo, é apresentada uma breve introdução sobre o problema abordado neste trabalho, bem como os principais desafios envolvidos. Na Seção 1.1 é realizada uma contextualização sobre tema e apresentada a relevância desta pesquisa. A Seção 1.2 apresenta os objetivos gerais e específicos deste trabalho. Por fim, na Seção 1.3 é descrita a organização dos capítulos subsequentes.

1.1 Contextualização e Relevância

Com o crescimento exponencial na geração e armazenamento de informações, os conjuntos de dados estão se tornando cada vez mais ricos e complexos, desempenhando um papel fundamental no funcionamento de diversas aplicações e sistemas (CHEN; MAO; LIU, 2014). Nesse contexto, o uso de técnicas computacionais, como algoritmos de aprendizado de máquina, para análise e interpretação desses dados é crucial. Esses algoritmos desempenham um papel essencial ao identificar padrões e gerar conhecimento a partir dos dados, seja em tarefas de regressão ou classificação.

Em problemas de regressão, os algoritmos desempenham a função de realizar previsões e gerar saídas contínuas, abrangendo números inteiros ou reais. Essas tarefas de regressão são aplicadas em uma variedade de domínios, como previsão de temperatura (DAWOOD *et al.*, 2023), identificação de sistemas não lineares (RODRIGO; PATINO; SCHWEICKARDT, 2021), previsão de valores em séries financeiras (ROSA *et al.*, 2023) e previsão de séries temporais (SHYAMSUKHA *et al.*, 2021), apenas para citar alguns exemplos.

Por outro lado, em problemas de classificação, os algoritmos têm como objetivo atribuir rótulos ou classes a amostras em um conjunto de dados com base nas suas características (ROUSSELL, 2022). As tarefas de classificação podem ser binárias, quando existem apenas duas classes possíveis (por exemplo, sim ou não, verdadeira ou falso, positivo ou negativo), ou multiclasse, quando existem mais de duas classes (por exemplo, classificar imagens em categorias como cachorro, gato, pássaro). O reconhecimento de imagens (WANG; MA, 2023), análise de sentimentos (SHARMA; GHOSE, 2023), detecção de fraudes (BAABDULLAH; ALZHRANI; RAWAT, 2020) e detecção de *spam* (RAZA; JAYASINGHE; MUSLAM, 2021) são apenas algumas das várias aplicações em que as tarefas de classificação são utilizadas.

Algoritmos classificadores podem ser categorizados em dois tipos principais com base no método de aprendizado adotado: *offline* e *online*. O treinamento de algoritmos de aprendizado *offline* envolve várias passagens pelo conjunto de dados, conhecidas como épocas de treinamento. Essa abordagem é adequada para conjuntos de dados estáticos, que não sofrem variações ao longo do tempo, e requer que o conjunto completo de dados

de treinamento esteja disponível *a priori*. Por outro lado, em algoritmos de aprendizado *online*, também conhecidos como aprendizado incremental ou tempo real, as amostras são recebidas sequencialmente em um fluxo de dados contínuo. O treinamento é realizado à medida que as amostras são apresentadas ao algoritmo, permitindo que o classificador se adapte a mudanças e variações nos dados ao longo do tempo (NGUYEN; WOON; NG, 2015). Os classificadores com aprendizado *online* possuem a capacidade de aprendizado contínuo, realizando apenas uma passagem nos dados. Esse tipo de abordagem elimina a necessidade de retreinamento dos classificadores, uma vez que o aprendizado ocorre de forma constante.

Os algoritmos de aprendizado *online* tradicionais, conhecidos como sistemas adaptativos, ajustam seus parâmetros continuamente, mantendo uma estrutura fixa. No entanto, recentemente, uma nova classe de algoritmos com aprendizado *online* surgiu, apresentando a capacidade de adaptar/evoluir simultaneamente e continuamente tanto seus parâmetros quanto sua estrutura. Os sistemas *fuzzy* com essa característica são chamados de sistemas *fuzzy* evolutivos (EFS - *evolving fuzzy systems*).

Este trabalho introduz um novo classificador *fuzzy* evolutivo multiclasse denominado eMGMC (*evolving Multivariable Gaussian Multiclass Classifier*). O classificador proposto utiliza uma versão modificada do algoritmo de agrupamento com aprendizado participativo do eMG (*evolving Multivariable Gaussian*) (LE MOS; CAMINHAS; GOMIDE, 2011). Ao contrário do eMG, que é um preditor, no classificador proposto a criação de grupos e regras é baseada no conceito de procrastinação e não no de índice de alerta. No algoritmo do eMGMC foi incorporado um método para exclusão de grupos e regras inativas baseado no conceito de idade (LUGHOFER; ANGELOV, 2011) e população (LUGHOFER; ANGELOV, 2011). O método de mesclagem foi adaptado para utilização de uma medida de sobreposição de grupos (AGUIAR; LEITE, 2020).

1.2 Objetivos

1.2.1 Objetivo Geral

O objetivo geral deste trabalho é propor e implementar um novo classificador multiclasse *fuzzy* evolutivo.

1.2.2 Objetivos Específicos

Para alcançar o objetivo geral, propõem-se os seguintes objetivos específicos:

- a) Revisar a literatura sobre sistemas evolutivos.
- b) Propor um algoritmo classificador multiclasse *fuzzy* evolutivo.
- c) Realizar simulações computacionais para avaliar o desempenho do classificador

proposto.

1.3 Organização do Trabalho

O restante deste trabalho está dividido em 5 capítulos:

- a) O Capítulo 2 apresenta conceitos relacionados ao problema tratado, além de apresentar uma revisão da literatura sobre sistemas *fuzzy* evolutivos.
- b) O Capítulo 3 descreve o funcionamento eMGMC. Assim, são descritas todas as etapas de execução, como a inicialização, criação, mesclagem e deleção de grupos e regras.
- c) O Capítulo 4 destaca a metodologia dos experimentos computacionais realizados e os resultados obtidos.
- d) Por fim, o Capítulo 5 descreve as considerações finais e sugestões para trabalhos futuros.

2 REVISÃO DA LITERATURA

Este capítulo tem como objetivo introduzir os conceitos básicos de sistemas *fuzzy* evolutivos e apresentar uma breve revisão da literatura sobre o tema. Na Seção 2.2 os sistemas *fuzzy* evolutivos são introduzidos. A Seção 2.3 apresenta o estado da arte sobre sistemas evolutivo caracterizando alguns dos principais sistemas evolutivos.

2.1 Sistemas Fuzzy

A linguagem natural possui como característica inerente a imprecisão. Essa imprecisão, que muitas vezes se aproxima de algo verdadeiramente humano, se torna imprescindível na criação de sistemas e deve ser levada em consideração (OKAMOTO *et al.*, 1995). A teoria fuzzy propõe uma abordagem para representar e processar a imprecisão inerente à linguagem natural, buscando assim aprimorar a capacidade de sistemas inteligentes em se assemelharem à compreensão humana.

O primeiro sistema *Fuzzy* foi introduzido por Zadeh (1965). O algoritmo utiliza de modelos matemáticos capazes de lidar com problemas de incerteza, imprecisão e subjetividade. Sistemas *fuzzy* fundamentam-se em proposições lógicas que não se limitam a serem estritamente verdadeiras ou falsas, mas, de fato, apresentam graus de verdade. Ao contrário de sistemas com fronteiras definidas entre elementos que pertencem ou não a um conjunto, os conjuntos *fuzzy* lidam com valores contínuos ou discretos, atribuindo graus de pertinência por meio de uma base de regras *fuzzy*. Esses valores variam em um intervalo de $[0,1]$.

A base de regras *fuzzy* é responsável por armazenar e lidar com todo o conhecimento por meio de uma coleção de regras *fuzzy*. Cada regra é do tipo:

SE < *antecedente* > **Então** < *consequente* >

em que o antecedente é a premissa que descreve a condição para que a regra seja satisfeita. O consequente, por sua vez, é responsável por definir o resultado ou ação a ser tomada quando o antecedente da regra é atendido, no caso desse trabalho realizar o classificação.

Os sistemas *Fuzzy* apresentam uma estrutura composta por etapas claramente definidas durante sua execução, aprimorando a capacidade de lidar com dados inicialmente precisos. A primeira fase, conhecida como fuzzificação, desempenha um papel crucial ao converter dados exatos em variáveis linguísticas, utilizando funções de pertinência provenientes da base de conhecimentos *Fuzzy*. Na sequência, a etapa de inferência assume a responsabilidade de mapear as entradas fuzzificadas em regras lógicas do tipo SE-ENTÃO. Essas regras, fundamentadas na base de conhecimentos

Fuzzy, capturam relações complexas e não-lineares entre as variáveis de entrada e saída. Por fim, o processo de defuzzificação é responsável por converter a saída do sistema *Fuzzy*, em um resultado interpretável e utilizável.

2.2 Sistemas *Fuzzy* Evolutivos

Os Sistemas *Fuzzy* Evolutivos (EFS), também conhecidos como Evolving *Fuzzy* Systems, apresentam uma estrutura que pode se expandir e/ou contrair em resposta aos dados apresentados (SILVA *et al.*, 2014). Esse comportamento adaptativo é fundamental para lidar com ambientes complexos e em constante mudança. Um dos métodos comuns para a construção da estrutura do EFS envolve algoritmos de agrupamento, que organizam a informação em elementos como neurônios, grânulos, funções de pertinência, grupos ou nuvem de dados (ŠKRJANC *et al.*, 2019a).

É interessante observar que, enquanto muitos EFS utilizam algoritmos de agrupamento para a construção inicial da estrutura, alguns utilizam o erro de modelagem calculado recursivamente (SILVA *et al.*, 2014). Essa abordagem revela a versatilidade e diversidade de estratégias adotadas pelos EFS para se adaptarem suas estruturas conforme os dados de entrada.

A evolução dos EFS é um processo dinâmico, envolvendo a criação, realocação, união e/ou deleção de elementos estruturais. Essas operações possibilitam a contínua adaptação do sistema às mudanças e variações ao longo do tempo. O aprendizado dos EFS é caracterizado como *online* e incremental, o que significa que apenas a amostra atual é considerada no processo de aprendizado. Isso não apenas reduz o custo computacional, pois a amostra pode ser descartada após o processamento, mas também elimina a necessidade de retreinamento periódico. A base de regras de um sistema *Fuzzy* evolutivo inicia-se do zero e, simultaneamente à sua função primária de realizar classificação ou previsão, ela evolui dinamicamente sua estrutura ao longo do tempo.

É digno de nota que a capacidade de ajuste em tempo real dos EFS é potencializada pelos algoritmos de cálculo recursivo dos parâmetros do consequente, frequentemente baseados em métodos como o gradiente, mínimos quadrados ou suas variações (LEITE; COSTA; GOMIDE, 2012; LEITE; DECKER *et al.*, 2020). Essa abordagem proporciona uma adaptação contínua às mudanças nos padrões dos dados, garantindo a relevância e a precisão do modelo ao longo do tempo.

Em resumo, os EFS representam uma classe dinâmica e adaptativa de sistemas inteligentes, capazes de se ajustar continuamente às complexidades e variações do ambiente, proporcionando um aprendizado eficiente e em tempo real. Essa característica os torna particularmente valiosos em aplicações onde a natureza dos dados e as condições operacionais são sujeitas a mudanças imprevisíveis.

2.3 Trabalhos Relacionados

O eTS (*evolving Takagi-sugeno*) (ANGELOV, P. P.; FILEV, D. P., 2004) representa um marco no desenvolvimento dos EFS. O eTS utiliza funções de pertinência Gaussianas e emprega um algoritmo de agrupamento recursivo não supervisionado. Nesse algoritmo, o potencial de cada amostra é calculado recursivamente em relação a todos os grupos existentes para atualização e/ou criação grupos. Caso o potencial de uma amostra seja maior que o de todos os grupos, um novo grupo é criado. Caso contrário, a amostra é adicionada ao grupo mais próximo. Quando o potencial de uma amostra é menor que o de todos os grupos, o consequente do grupo com a menor distância euclidiana é atualizado utilizando um método baseado em mínimos quadrados.

O FLEXFIS-Class (*Flexible Evolving Fuzzy Inference Systems Classifier*) (LUGHOFER; ANGELOV; ZHOU, 2007) é uma variação do preditor FLEXFIS (LUGHOFER; KLEMENT, 2005). Esse classificador possui uma arquitetura *fuzzy* com funções Gaussianas e regras Takagi-sugeno de ordem zero e usa uma versão evolutiva de quantização vetorial. Na abordagem com uma arquitetura de modelo único, a evolução dos consequentes e dos pesos das regras é baseada em uma escolha por maioria e na frequência relativa das classes nos diferentes clusters.

O eMG (*evolving Multivariable Gaussian*), proposto por Lemos, Caminhas e Gomide (2011), é um preditor que utiliza um algoritmo de agrupamento não supervisionado baseado no conceito de aprendizado participativo (YAGER, 1990) para construção de sua base de regras. Sua estrutura é composta por regras de Takagi-Sugeno de Ordem 1, cujo antecedente é definido por funções de pertinência Gaussiana multivariável. O algoritmo de aprendizado emprega um índice de alerta para determinar a criação de novos grupos e lidar com dados atípicos (*outliers*). Além disso, o eMG incorpora um mecanismo de mesclagem de grupos com base em uma medida de compatibilidade entre eles. Os parâmetros do consequente das regras são atualizados pelo algoritmo de mínimos quadrados recursivos ponderados.

O eT2Class (*evolving Type-2 Classifier*) (PRATAMA; LU; ZHANG, 2015) é um classificador evolutivo que utiliza dois métodos distintos para a geração de regras. O primeiro método estima a contribuição estatística do fluxo de dados, enquanto o segundo método monitora essa contribuição, quantificando a proximidade espacial de um dado em relação aos padrões de treinamento anteriores, de maneira semelhante a uma medida de densidade de dados. O eT2Class utiliza funções Gaussianas Multivariáveis no antecedente das regras e o consequente das regras é construído ou atualizado a partir do polinômio de Chebyshev.

ALMMo (*Autonomous Learning Multi-Model Systems*)(ANGELOV; GU, 2017) é um classificador que utiliza de regras *fuzzy* do tipo AnYa de ordem 0 e uma estrutura de agrupamento denominada uma nuvem de dados. A criação de novas regras ocorre com

base na densidade de dados onde a nova amostra está inserida. Caso nenhuma nova regra seja criada, a amostra é adicionada à nuvem de dados mais próxima. O processo de exclusão de regras é realizado por meio de um mecanismo que identifica regras menos importantes, utilizando uma soma acumulada normalizada dos valores dos parâmetros. Em seguida, essa soma é comparada com a média das somas de todas as regras. Caso a soma seja menor que essa média, a regra é excluída.

No modelo funcional evolutivo, descrito por Torres e Serra (2018), os parâmetros do antecedente são estimados com base na densidade das amostras, o que permite a generalização e representatividade dessas amostras. Quando uma amostra é selecionada como ponto focal, sua densidade precisa ser atualizada. A sobreposição de regras é evitada sempre que uma nova regra é criada. Assim, todas as regras cujo grau de pertinência esteja acima de um determinado limiar são substituídas pela nova regra criada. Além disso, o consequente é estimado utilizando um modelo linear do espaço de estados da *i*-ésima regra, com a utilização de parâmetros de Markov.

O *eiX* (*evolving Internal-eXternal Fuzzy Clustering*) é um classificador evolutivo, introduzido em Aguiar e Leite (2020), baseado no conceito de computação granular. O modelo é caracterizado como não supervisionado e apresenta um princípio de grânulos *fuzzy* de contorno duplo com funções de pertinência trapezoidal. Assim, o conhecimento é representado pelos grânulos denominado hiper-caixas de contorno duplo, em que sistemas de inferência tipo 1 e tipo 2 é obtido por uma projeção dos grânulos interno e externo. O princípio da granularidade é utilizado para balancear os grânulos ao longo de todas as dimensões à medida que novos dados são apresentados. A criação de novos grânulos é realizada quando a amostra não está localizada dentro dos limites dos intervalos internos e externos de qualquer grânulo. A mesclagem dos grânulos ocorre com base na proximidade entre os centros, a qual é determinada pela norma entre eles, desde que seja inferior a um limite predefinido.

Adaptive Evolving Fuzzy (AHWIADI; WANG, 2022) é um preditor evolutivo que utiliza regras Takagi-Sugeno de ordem 1. Essa abordagem apresenta um método de avaliação de erro que monitora a tendência dos erros de treinamento. Além disso, é proposto um filtro de partículas adaptativas, que explora o espaço de regras *fuzzy* para melhorar a flexibilidade da modelagem e a capacidade adaptativa do modelo.

O *eNFN* (*evolving Neuro-Fuzzy Network*) (SHAH; WANG, 2022) utiliza funções de pertinência Gaussiana em sua abordagem. Nesse modelo, os parâmetros do grupo são atualizados com base no potencial calculado para cada amostra. Se a amostra não pertencer a nenhum conjunto *fuzzy* existente, um novo grupo é criado, resultando na formação de uma nova regra. Além disso, os pesos presentes na matriz de parâmetros são reajustados para melhorar a precisão do modelo.

O *eFMC* (*evolving Fuzzy Mean Classifier*) (TAVARES; SILVA; MOITA, 2022) é um classificador evolutivo multiclasse autônomo, i.e., não requer a definição de

hiper-parâmetros. Nesse modelo, os grupos são criados automaticamente sempre que uma nova classe é descoberta. A atualização dos centros dos grupos ocorre a cada nova amostra, utilizando a média dos valores das amostras associadas a cada grupo. Cada grupo representa exclusivamente uma classe, e a saída do modelo é determinada com base no grau de pertinência da amostra em relação aos grupos existentes, o qual é calculado usando a distância euclidiana.

3 ABORDAGEM PROPOSTA

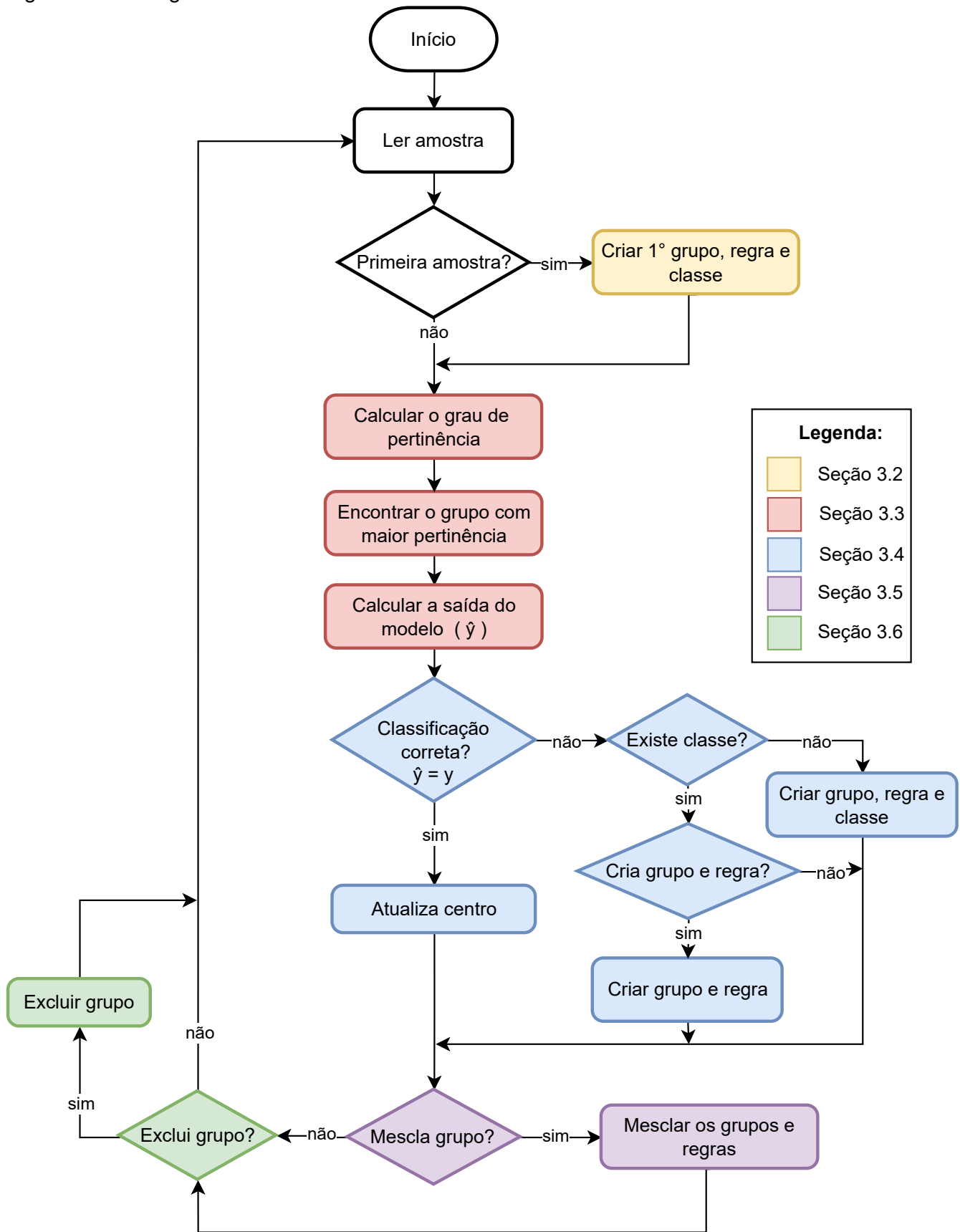
Este capítulo tem como objetivo descrever as etapas de execução do eMGMC. Para isso, o capítulo está dividido da seguinte forma: na Seção 3.1, é realizada uma introdução e a sumarização das etapas do algoritmo, fornecendo uma visão geral do processo, juntamente com a apresentação de um fluxograma da abordagem; na Seção 3.2, aborda-se a inicialização do classificador; na Seção 3.3, descreve-se o cálculo da saída; na Seção 3.4, apresenta-se o método para atualização e criação de grupos, regras e classes; na Seção 3.5, descreve-se a abordagem utilizada para a mesclagem de grupos e regras; na Seção 3.6, é descrito o processo de deleção de grupos e regras. Por fim, na Seção 3.7, são apresentados os parâmetros e o algoritmo do eMGMC.

3.1 Introdução

A abordagem proposta possui como característica uma estrutura dinâmica, que se adapta de maneira contínua à medida que as amostras são processadas, permitindo uma única passagem pelos dados. O classificador usa uma versão modificada do algoritmo de agrupamento evolutivo participativo gaussiano multivariável. A estrutura do classificador se desenvolve adicionando um novo grupo, mesclando ou excluindo uma regra existente. A figura 1 mostra o fluxograma eFCMG e as etapas a seguir resumem o procedimento de aprendizagem.

- a) Defina os valores iniciais dos hiperparâmetros. Em seguida, inicialize o primeiro centro com a amostra de dados e crie o grupo, a classe e a regra. Esta etapa é realizada apenas uma vez.
- b) Depois, para cada nova amostra de dados, calcule o grau de ativação da amostra atual em todas as funções de pertinência existentes. Encontre a regra mais ativa i^+ . Posteriormente, obtenha a saída \hat{y} .
- c) Verifique a classificação. Se a classificação estiver correta, ou seja, $\hat{y} = y$, atualize o centro indexado por i^+ . Por outro lado, se a classificação estiver incorreta, ou seja, $\hat{y} \neq y$, decida se um novo grupo ativo, classe e regra devem ser criados, se um novo grupo inativo deve ser criado, ou se um grupo inativo deve ser ativado.
- d) Verifica grupos redundantes com a mesma classe. Decida se dois grupos com sobreposição notável devem ser mesclados.
- e) Encontre o grupo inativo mais antigo i^- . Verifique se a *idade* do i^- -ésimo grupo é maior que um limite e o número de amostras do grupo é menor que 1% do total de amostras. Decida se o grupo i^- -th deve ser excluído.

Figura 1 – Fluxograma do eMGMC.



Fonte: próprio autor.

3.2 Inicialização

O classificador evolutivo multiclasse inicia o aprendizado do zero e a primeira amostra de dados é usada para criar o primeiro grupo/regra. O centro do grupo é obtido utilizando os valores da amostra, ou seja:

$$v_{ij}^t \leftarrow x_j^t, \quad (3.1)$$

onde v é o vetor com o centro do grupo é definido como $[v_1^t \dots v_i^t \dots v_{c^t}^t]^T$; t é o passo atual; i indexa o centro do grupo; c^t é o número de grupos criados até a etapa t ; j indexa as variáveis de entrada, com $j = 1 \dots m$; m é o número de variáveis de entrada; x é o vetor com as variáveis de entrada e é descrito por $[x_1^t \dots x_j^t \dots x_m^t]^T$.

Para criar o primeiro grupo, o número de grupos criados c e o número de classes k , inicialmente definidos como zero, são incrementados em um, da seguinte forma:

$$c^t = c^{t-1} + 1, \quad (3.2)$$

e,

$$k^t = k^{t-1} + 1. \quad (3.3)$$

Em seguida, o contador de amostras do grupo a é incrementado em um:

$$a_i^t = a_i^{t-1} + 1. \quad (3.4)$$

Posteriormente, a matriz de dispersão do grupo é inicializada por:

$$\sum_i^t \leftarrow \sum_{init}, \quad (3.5)$$

onde \sum_{init} é a matriz de dispersão inicial de dimensão $m \times m$. Em seguida, o número de centros ativos p é incrementado em um:

$$p^t = p^{t-1} + 1. \quad (3.6)$$

Então, o vetor de grupos ativos Ω recebe o número do grupo criado:

$$\Omega_{p^t}^t \leftarrow c^t. \quad (3.7)$$

A classe do grupo é definida pela classe da saída desejada y . Assim, o vetor de classe recebe a classe da amostra da seguinte forma:

$$\Theta_{k^t}^t \leftarrow y^t. \quad (3.8)$$

Depois de criar o primeiro grupo, a primeira regra é criada. A antecedente da regra é representada por uma função de pertinência gaussiana multivariada, cujo valor modal é definido pelo centro do grupo, e a dispersão é obtida pela matriz de dispersão. Uma função de pertinência gaussiana multivariada pode ser descrita por:

$$f(x) = e^{-\frac{1}{2}(x^t - v_i^t)(\Sigma_i^t)^{-1}(x^t - v_i^t)^T}, \quad (3.9)$$

na qual x^t é a amostra atual, v são os centros dos grupos (valor modal das funções de pertinência) e Σ é a matriz de dispersão. O consequente da regra é definido pela classe, ou seja, $\hat{y}_i = \Theta_l$, onde l indexa as classes. As regras fuzzy podem ser descritas por:

$$R_i \quad : \quad \text{Se } x^t \text{ é } B_i \text{ então } \hat{y}_i^t = \Theta_l^t,$$

em que R_i é a i -ésima regra, x^t é a amostra atual, B_i é a i -ésima função de pertinência gaussiana multivariada, \hat{y}_i^t é a saída da i -ésima regra, e Θ_l^t é o consequente. Finalmente, g_Θ , o número de grupos da classe l , é atualizado por:

$$g_{\Theta_l^t} = g_{\Theta_l^{t-1}} + 1. \quad (3.10)$$

O Algoritmo 1 ilustra o procedimento para criar grupos, classes e regras.

Algoritmo 1: Criação grupos, classes e regras.

- Atualizar o número de grupos criados c - Eq. (3.2);
 - Criar o centro/grupo v - Eq. (3.1);
 - Atualizar o número de classes k - Eq. (3.3);
 - Atualizar o contador de amostras do grupo a - Eq. (3.4);
 - Inicializar a matriz de dispersão Σ - Eq. (3.5);
 - Atualizar o número de centros ativos p - Eq. (3.6);
 - Atualizar o vetor de índices de grupo Ω - Eq. (3.7);
 - Atualizar o vetor de classes Θ - Eq. (3.8);
 - Criar a regra;
 - Atualizar o número de grupos da classe g_Θ - Eq. (3.10);
-

3.3 Cálculo de Saída

Para cada nova amostra de dados x^t , o grau de ativação normalizado do antecedente das regras τ é calculado por:

$$\tau_i^t = \frac{e^{D_{(x^t, v_i^t)}}}{\sum_{i=1}^{p^t} e^{D_{(x^t, v_i^t)}}}, \quad (3.11)$$

no qual i indexa as regras, Σ é a matriz de dispersão e p^t é o número de regras ativas na etapa t . $D_{(x^t, v_i^t)}$ é a distância de Mahalanobis entre a amostra atual x^t e o centro v_i^t e é

calculada por:

$$D_{(x^t, v_i^t)} = (x^t - v_i^t) \left(\sum_i^{p^t} \right)^{-1} (x^t - v_i^t)^T. \quad (3.12)$$

Para obter a saída do classificador, encontra-se o índice da regra mais ativa i^+ , ou seja:

$$i^+ = \arg \max_{i=1,2,\dots,p^t} \tau_i^t. \quad (3.13)$$

A classe representada pelo conseqüente da regra i^+ é a classe prevista pelo classificador. Em outras palavras, a saída do classificador é tal que:

$$\hat{y}^t \leftarrow \hat{y}_{i^+}^t = \Theta_{i^+}^t, \quad (3.14)$$

no qual $\hat{y}_{i^+}^t$ é a saída da regra mais ativa.

Algoritmo 2: Cálculo da Saída.

// Para cada grupo ativo

for $i = 1$ to p **do**

 | Calcular a distância de Mahalanobis D - Eq. (3.12);

 | Calcular o grau de ativação τ - Eq. (3.11);

end

Encontrar o índice da regra mais ativa i^+ - Eq. (3.13);

Obter a saída \hat{y} - Eq. (3.14);

3.4 Criação e Atualização de Grupos e Regras

A estrutura de grupo representa o conhecimento atual e é atualizada conforme cada amostra de dados é apresentada. A cada nova amostra, a estrutura evolui com base em um algoritmo de agrupamento com aprendizado participativo (YAGER, 1990). Este procedimento utiliza uma medida de compatibilidade (LEMOS; CAMINHAS; GOMIDE, 2011) e um conceito de procrastinação (LUGHOFER; PRATAMA; SKRJANC, 2018). A medida de compatibilidade determina o quão compatível a nova amostra é com o conhecimento atual e pode ajustar a taxa de aprendizado, reduzindo a aprendizagem quando entradas anômalas ocorrem e, de outra forma, acelerando o processo de aprendizagem (RODRIGUES, F.; SILVA, A.; LEMOS, A., 2022). O conceito de procrastinação é um mecanismo robusto para lidar com valores atípicos. Na procrastinação, os grupos são criados inativos, sem associação com nenhuma regra, e, conseqüentemente, não contribuem para a saída do classificador. Esses grupos permanecem inativos até terem amostras suficientes associadas a eles. O algoritmo de

agrupamento pode (i) atualizar um grupo ativo, (ii) criar um grupo ativo, (iii) criar um grupo inativo, (iii) ativar um grupo inativo e (iv) atualizar um grupo inativo. Esses procedimentos são realizados com base na saída do classificador da seguinte forma:

- **Caso 1: Se** $\hat{y}^t = y^t$, ou seja, a amostra x^t foi classificada corretamente, **então** o grupo mais ativo indexado por i^+ tem seu centro v_{i^+} atualizado por:

$$v_{i^+}^{t+1} = v_{i^+}^t + S_{i^+}^t (x^t - v_{i^+}^t), \quad (3.15)$$

onde $S_{i^+}^t$ é dado por:

$$S_{i^+}^t = \lambda \gamma_{i^+}^t, \quad (3.16)$$

em que λ é a taxa de aprendizado e γ é a medida de compatibilidade com valores entre $[0, 1]$. A medida de compatibilidade é obtida por:

$$\gamma_i^t = e^{-\frac{1}{2}D(x^t, v_i^t)}. \quad (3.17)$$

Em seguida, a matriz de dispersão \sum_{i^+} é atualizada usando:

$$\sum_{i^+}^t = (1 - S_{i^+}^t) \left(\sum_{i^+}^{t-1} - S_{i^+}^t (x^t - v_{i^+}^t)(x^t - v_{i^+}^t)^T \right). \quad (3.18)$$

- **Caso 2: Se** $\hat{y}^t \neq y^t$ e $\neg \exists y^t = \Theta_l^t$ ($l = 1 \dots k^t$), ou seja, se a amostra x^t foi classificada incorretamente e a classe da amostra é nova e, portanto, não existe no vetor de classes, **então** um novo grupo ativo, uma nova classe e uma nova regra são criados. Este procedimento é, *mutatis mutandis*, o mesmo descrito pelo Algoritmo 1 (consulte a Seção 3.2) e, portanto, não será descrito novamente.
- **Caso 3: Se** $\hat{y}^t \neq y^t$ e $\exists y^t = \Theta_l^t$ ($l = 1 \dots k^t$) e $\neg \exists y^t = \bar{\Theta}_{\bar{l}}^t$, ($\bar{l} = 1 \dots \bar{k}^t$), ou seja, se a amostra x^t foi classificada incorretamente e a classe da amostra existe nos grupos ativos e a classe da amostra não existe nos grupos inativos, **então** um grupo inativo é criado. O procedimento para criar grupos inativos é semelhante ao de criar grupos ativos. No entanto, o procedimento é detalhado para facilitar a compreensão. Inicialmente, o número de grupos inativos criados \bar{c} é aumentado em um:

$$\bar{c}^t = \bar{c}^{t-1} + 1, \quad (3.19)$$

e o centro do grupo inativo $\bar{v}_{\bar{i}j}$ é encontrado por:

$$\bar{v}_{\bar{i}j}^t \leftarrow x_j^t, \quad (3.20)$$

onde \bar{v} é o centro do grupo inativo, t é a etapa atual, \bar{i} é o índice dos grupos inativos ($\bar{i} = 1 \dots \bar{p}^t$, com \bar{p}^t sendo o número de grupos inativos na etapa t), x é o vetor com as variáveis de entrada e j é o índice das variáveis de entrada. Em seguida, o

número atual de centros inativos \bar{p} é atualizado usando:

$$\bar{p}^t = \bar{p}^{t-1} + 1. \quad (3.21)$$

Em seguida, o contador de amostras do grupo inativo \bar{a}_i é incrementado:

$$\bar{a}_i^t = \bar{a}_i^{t-1} + 1. \quad (3.22)$$

Posteriormente, a matriz de dispersão do grupo inativo $\bar{\Sigma}_i$ é inicializada por:

$$\bar{\Sigma}_i^t \leftarrow \bar{\Sigma}_{init}. \quad (3.23)$$

A seguir, o vetor de índices de grupos inativos $\bar{\Omega}$ é atualizado por:

$$\bar{\Omega}_{\bar{p}^t}^t = \bar{c}^t. \quad (3.24)$$

A classe do grupo inativo é definida da seguinte forma:

$$\bar{\Theta}_i^t \leftarrow y^t. \quad (3.25)$$

onde \bar{i} é o índice do grupo inativo e y^t é a classe da amostra x^t . Por fim, $\bar{g}\bar{\Theta}$ é incrementado em um:

$$g\bar{\Theta}_{\bar{l}} = g\bar{\Theta}_{\bar{l}} + 1, \quad (3.26)$$

onde \bar{l} indexa as classes dos grupos inativos.

- **Caso 4: Se** $\hat{y}^t \neq y^t$ e $\exists y^t = \Theta_l^t$ ($l = 1 \dots k^t$) e $\exists y^t = \bar{\Theta}_{\bar{l}}^t$ ($\bar{l} = 1 \dots \bar{k}^t$), ou seja, se a amostra x^t foi classificada incorretamente e a classe da amostra existe nos grupos ativos e a classe da amostra existe nos grupos inativos, **então** verifique se o grupo inativo deve ser ativado ou atualizado. A distância de Mahalanobis D é calculada entre a amostra atual x^t e o centro de todos os grupos inativos \bar{v}_i que têm a mesma classe que a amostra atual por:

$$D_{(x^t, \bar{v}_i^t)} = (x^t - \bar{v}_i^t) \left(\sum_{\bar{i}}^{\bar{c}^t} \right)^{-1} (x^t - \bar{v}_i^t)^T. \quad (3.27)$$

Em seguida, calcule para os grupos inativos \bar{v}_i que têm a mesma classe que a amostra atual o grau de ativação normalizado:

$$\tau_i^t = \frac{e^{D_{(x^t, \bar{v}_i^t)}}}{\sum_{\bar{i}=1}^{\bar{c}^t} e^{D_{(x^t, \bar{v}_i^t)}}}, \quad (3.28)$$

onde x^t é a amostra atual, \bar{v}_i^t é o centro do \bar{i} -ésimo grupo inativo, $\bar{\Sigma}$ é a matriz de

dispersão do grupo inativo e D é a distância de Mahalanobis. Em seguida, entre os grupos da mesma classe, encontre o grupo com a maior ativação indexado por \bar{i}^+ , ou seja:

$$\bar{i}^+ = \arg \max_{i=1,2,\dots,\bar{p}^t} \tau_i^t. \quad (3.29)$$

Se o número de amostras do grupo inativo indexado por \bar{i}^+ for menor que o limiar de procrastinação η , ou seja, se $\bar{a}_{\bar{i}^+}^t < \eta$, o número de amostras do grupo é aumentado por:

$$\bar{a}_{\bar{i}^+}^t = \bar{a}_{\bar{i}^+}^{t-1} + 1. \quad (3.30)$$

Então, o centro do grupo inativo indexado por \bar{i}^+ é atualizado por:

$$\bar{v}_{\bar{i}^+}^t = \bar{v}_{\bar{i}^+}^{t-1} + S_{\bar{i}^+}^t (x^t - \bar{v}_{\bar{i}^+}^t), \quad (3.31)$$

onde $S_{\bar{i}^+}^t$ é obtido por:

$$S_{\bar{i}^+}^t = \lambda \gamma_i^t, \quad (3.32)$$

onde λ é a taxa de aprendizado e γ é a medida de compatibilidade obtida por:

$$\gamma_i^t = e^{-\frac{1}{2}D(x^t, \bar{v}_i^t)}. \quad (3.33)$$

Em seguida, a matriz de dispersão do grupo indexado por \bar{i}^+ é atualizada usando:

$$\bar{\Sigma}_{\bar{i}^+}^t = (1 - S_{\bar{i}^+}^t) (\bar{\Sigma}_{\bar{i}^+}^{t-1} - S_{\bar{i}^+}^t (x^t - \bar{v}_{\bar{i}^+}^t)(x^t - \bar{v}_{\bar{i}^+}^t)^T). \quad (3.34)$$

Por outro lado, se o número de amostras do grupo inativo indexado por \bar{i}^+ for igual ou maior que o limiar de procrastinação, ou seja, se $\bar{a}_{\bar{i}^+}^t \geq \eta$, o grupo indexado por \bar{i}^+ é ativado. Para fazer isso, inicialmente, o número de grupos ativos criados e o número de grupos ativos são aumentados em um:

$$c^t = c^{t-1} + 1, \quad (3.35)$$

e,

$$p^t = p^{t-1} + 1. \quad (3.36)$$

Em seguida, o contador de amostras do novo grupo ativo é atualizado da seguinte forma:

$$a_i^t = \bar{a}_{\bar{i}^+}^t + 1. \quad (3.37)$$

Posteriormente, a nova matriz de dispersão do grupo ativo recebe o valor da matriz de dispersão do grupo sendo ativado, ou seja:

$$\sum_i^t \leftarrow \sum_{\bar{i}^+}^t. \quad (3.38)$$

Em seguida, o vetor de índices de grupos ativos Ω é atualizado por:

$$\Omega_{\bar{p}^t}^t \leftarrow c^t. \quad (3.39)$$

Após criar o grupo, a regra é criada e consequente definida pela classe e grupo indexado por \bar{i}^+ , ou seja, $\hat{y}_{\bar{i}^+} = \bar{\Theta}_{\bar{l}}$. Finalmente, o número de grupos da classe l é atualizado da seguinte forma:

$$g\Theta_l^t = g\Theta_l^t + 1. \quad (3.40)$$

em que l indexa a classe da nova regra. O Algoritmo 3 descreve os passos de atualização e criação de grupo ativos e inativos.

Algoritmo 3: Atualização e criação grupo e regras.

```

if  $\hat{y}^t = y^t$  then
  | Atualizar o centro  $v_{i^+}$  - Eq. (3.17), (3.16), e (3.15);
  | Atualizar a matriz de dispersão  $\sum_{i^+}$  - Eq. (3.18);
else if  $\hat{y} \neq y$  e  $\neg \exists y = \Theta_l$  then
  | Criar grupo classe e regra - Algoritmo 1;
end
else if  $\hat{y} \neq y$  e  $\exists y = \Theta_l$  e  $\neg \exists y = \bar{\Theta}_l$ , then
  | Atualizar o número de grupos inativos criados  $\bar{c}$  - Eq. (3.19);
  | Criar o centro inativo  $\bar{v}$  - Eq. (3.20);
  | Atualizar o número de centros inativos  $\bar{p}$  - Eq. (3.21);
  | Atualizar o contador de amostras do grupo inativo  $\bar{a}$  - Eq. (3.22);
  | Atualizar a matriz de dispersão  $\bar{\sum}$  - Eq. (3.23);
  | Atualizar o vetor de índices de grupo inativo  $\bar{\Omega}$  - Eq. (3.24);
  | Atualizar o vetor de classe  $\bar{\Theta}$  - Eq. (3.25);
  | Atualizar o número de grupos da classe  $g\bar{\Theta}$  - Eq. (3.26);
end
else if  $\hat{y} \neq y$  e  $\exists y = \Theta_l$  e  $\exists y = \bar{\Theta}_l$  then
  | Calcular a distância de Mahalanobis  $D$  - Eq. (3.27) e (3.28);
  | Encontrar o grupo mais compatível indexado por  $\bar{i}^+$  - Eq (3.29);
  if  $\bar{a}_{\bar{i}^+} < \eta$  then
    | Atualizar o contador de amostras  $\bar{a}_{\bar{i}^+}$  - Eq. (3.30);
    | Atualizar o centro inativo  $\bar{v}_{\bar{i}^+}$  - Eq. (3.33), (3.32), (3.31);
    | Atualizar a matriz de dispersão  $\bar{\sum}_{\bar{i}^+}$  - Eq. (3.34);
  else
    | Atualizar o número de clusters ativos criados  $c$  - Eq. (3.35);
    | Atualizar o número de clusters ativos atuais  $p$  - Eq. (3.36);
    | Inicializar o número de amostras  $a$  - Eq. (3.37);
    | Inicializar a matriz de dispersão  $\sum$  - Eq. (3.38);
    | Atualizar o vetor de índices de clusters ativos  $\Omega$  - Eq. (3.39);
    | Atualizar o número de clusters da classe  $g\Theta$  - Eq. (3.40);
  end
end

```

3.5 Mesclagem de Grupos e Regras

A mesclagem ocorre sempre que dois grupos i e i^* com a mesma consequência (i.e, classe) têm uma sobreposição notável (AGUIAR; LEITE, 2020). Mais precisamente, os grupos i e i^* com a mesma consequência são mesclados se a distância euclidiana entre seus centros v_{i^*} e v_i for menor ou igual a um limiar predefinido, ou seja, se:

$$\|v_{i^*}^t - v_i^t\| \leq \rho \text{ e } \hat{y}_i^t = \hat{y}_{i^*}^t,$$

em que ρ é o limiar de sobreposição, $i = 1 \dots p^t$ e $i \neq i^*$. O centro do grupos $v_{i \cup i^*}^t$ é obtido pela média ponderada entre os centros dos dois grupos mesclados:

$$v_{i \cup i^*}^t = v_{i^*}^t - \frac{a_i^t}{a_{i^*}^t + a_i^t} (v_{i^*}^t - v_i^t), \quad (3.41)$$

onde a_i^t e $a_{i^*}^t$ são o número de amostras dos grupos i e i^* . Posteriormente, a matriz de dispersão $\sum_{i^* \cup i}^t$ é definida como:

$$\sum_{i^* \cup i}^t = \frac{\sum_{i^*}^t + \sum_i^t}{2}, \quad (3.42)$$

e o número de amostras $a_{i^* \cup i}^t$ é obtido adicionando o número de amostras dos dois grupos mesclados, ou seja:

$$a_{i^* \cup i}^t = a_{i^*}^t + a_i^t. \quad (3.43)$$

O número de grupos criados c é incrementado em um:

$$c^t = c^{t-1} + 1, \quad (3.44)$$

O número de centros ativos p e o número de grupos da classe do novo grupo criado $g\Theta$ são decrementados em um:

$$p^t = p^{t-1} - 1, \quad (3.45)$$

e,

$$g\Theta_l^t = g\Theta_l^{t-1} - 1. \quad (3.46)$$

Em seguida, o vetor de grupos ativos Ω recebe o número do grupo mesclado:

$$\Omega_{p^t}^t \leftarrow c^t. \quad (3.47)$$

Por fim, a regra é mesclada, os grupos indexados i e i^* são excluídos do Ω , e os índices são atualizados. O Algoritmo 4 descreve o procedimento para a fusão de grupos e regras.

Algoritmo 4: Mesclagem de grupos e regras.

```

//  $i^*$  - índice do grupo atualizado ou criado
for  $i = 1$  até  $p$  do
  if  $i \neq i^*$  e  $\|v_{i^*} - v_i\| \leq \rho$  e  $\hat{y}_i = \hat{y}_{i^*}$  then
    Criar o centro mesclado  $v_{i \cup i^*}$  - Eq. (3.41);
    Calcular a matriz de dispersão  $\sum_{i^* \cup i}$  - Eq. (3.42);
    Calcular o número de amostras  $a_{i^* \cup i}$  - Eq. (3.43);
    Atualizar o número de grupo criados  $c$  - Eq. (3.44);
    Atualizar o número de centros ativos  $p$  - Eq. (3.45);
    Atualizar o número de grupos da classe  $g_{\Theta}$  - Eq. (3.46);
    Atualizar o vetor de índice de grupo  $\Omega$  - Eq. (3.47);
    Excluir os clusters indexados  $i$  e  $i^*$  do  $\Omega$ ;
    Criar a regra mesclada;
  end
end

```

3.6 Exclusão de Grupos e Regras

O método de exclusão de grupos e regras é baseado no conceito de idade (LUGHOFER; ANGELOV, 2011) e no de população (ANGELOV, P.; FILEV, D., 2005). No eMGMC o conceito de idade é utilizado para definir o período de inatividade de um grupo. Em outras palavras, a idade de um grupo é definida pelo número de amostras que um grupo que ficou um tempo sem ser ativado e pode ser obtida por:

$$idade_i^t = t - A_i, \quad (3.48)$$

em que i é o índice do grupo, t o passo atual e A_i é a instância de tempo de ativação do i -ésimo grupo. Assim, para cada amostra o grupo com maior tempo de inatividade será excluído se:

$$idade_i^t > \omega,$$

em que ω é a limiar de exclusão do grupo.

A população de um grupo é representada pelo número de amostras associadas ao grupo. Nessa estratégia um grupo i será elegível para exclusão quando sua população a_i^t for inferior a 1% do número total de amostras. Portanto, o grupo i será elegível se:

$$\frac{a_i^t}{t} < 0.01.$$

Para excluir um grupo, encontra-se a cada amostra x^t grupo com maior tempo de inatividade indexado por i^- , i.e.:

$$i^- = \arg \max_{i=1,2,\dots,c^t} idade_i^t. \quad (3.49)$$

Em seguida, verifica se o grupo i^- é elegível para exclusão simultaneamente pelos critérios de idade e população. Depois, analisa se existem dois ou mais grupos que representam a mesma classe do grupo i^- . Por fim, o grupo indexado por i^- será excluído se:

$$idade_{i^-}^t > \omega \text{ e } \frac{a_{i^-}^t}{t} < 0.01 \text{ e } g_{\Theta_{i^-}} > 1,$$

em que $g_{\Theta_{i^-}}$ é o número de grupos que representam a mesma classe do grupo i^- . Após exclusão de um grupo/regra os índices devem ser atualizados. O Algoritmo 5 descreve o procedimento para exclusão de grupos e regras.

Algoritmo 5: Exclusão de grupo e regras.

Atualizar *idade* do grupo atualizado ou criado - Eq. (3.48);

Encontrar o índice do grupo menos ativo i^- - Eq. (3.49);

se $idade_{i^-} > \omega$ e $\frac{a_{i^-}}{t} < 0.01$ e $g_{\Theta_{i^-}} > 1$ **então**

 Excluir o grupo indexado por i^- ;

 Excluir a regra indexada por i^- ;

 Atualizar o número de grupos ativos p - Eq. (3.45);

 Atualizar o número de grupos da classe $g_{\Theta_{i^-}}$ - Eq. (3.46);

 Atualizar o vetor de grupo ativos Ω ;

fim

3.7 Parâmetros do eMGMC

O eMGMC possui 5 hiper-parâmetros:

- \sum_{init} : é matriz de dispersão inicial utilizada para o cálculo da distância de Mahalanobis. \sum_{init} é inicializada como uma matriz identidade ajustada, com valores compreendidos entre 10^{-1} e 10^{-4} .
- λ : é a taxa de aprendizado definida para controlar a velocidade de atualização dos grupos e regras. Quanto menor a taxa de aprendizado menor as oscilações causada por *outliers* e mais lento são os ajustes dos centros no processo de atualização. Os valores de λ variam entre 10^{-1} e 10^{-5} (LEMONS; CAMINHAS; GOMIDE, 2011).
- ρ : é o parâmetro de mesclagem de novos grupos, ou seja, o limiar da medida de compatibilidade. ρ é um valor responsável por definir o quanto sobrepostos os grupos estão ou não para realização da mesclagem (AGUIAR; LEITE, 2020). Quando o valor de ρ está mais próximo de 1, significa que os grupos mesclados estão mais distantes entre si. Já quando ρ está mais próximo de 0, indica que os grupos estão

mais sobrepostos, ou seja, mais próximos uns dos outros. A prática sugere-se $p \in [0.25, 0.45]$.

- ω : é o limiar de deleção de novos grupos e define a idade máxima que os grupos podem ter antes de serem excluídos. O valores de ω variam de 5 a 200 (RODRIGUES, F. P.; SILVA, A. M.; LEMOS, A. P., 2021).
- η : é o parâmetro de procrastinação responsável por determinar a quantidade de amostras necessárias para a criação de novos grupos. η é representada por valores inteiro positivos e seu valor padrão é 1. A prática sugere-se $\eta \in [0, 3]$.

O Algoritmo 6 sumariza todas as etapas do eMGMC.

Algoritmo 6: Procedimento de Aprendizado EFCMG.

Entrada: $x^t, y^t, \lambda, \sum_{init}, \omega, \rho, \eta$;

Saída: \hat{y}^t ;

for $t = 1, 2, \dots$ **do**

 Ler x^t ;

if $t == 1$ **then**

 Criar primeiro grupo, classe e regra - Alg. 1;

else

 Calcular saída \hat{y}^t - Alg. 2

 Ler y^t ;

 Criar grupos, classe e regras - Alg. 3;

 Mesclar grupos e regras - Alg. 4;

 Excluir grupos e regras - Alg. 5;

end

end

4 EXPERIMENTOS E RESULTADOS

Este capítulo apresenta os resultados obtidos durante a execução de experimentos no algoritmo implementado com diferentes bases de dados. A Seção 4.1 descreve detalhadamente a metodologia utilizada para a realização do experimento, enquanto a Seção 4.2 apresenta e analisa o desenvolvimento da estrutura do classificador proposto durante a execução e os resultados obtidos com o eMGMC e os demais modelos do estado da arte.

4.1 Metodologia

Nesta seção, é descrita a metodologia adotada nos experimentos realizados para avaliar a classificação do eMGMC em relação a outros algoritmos do estado da arte, são eles : ALMMo (**almmo**), FBEM (LEITE; BALLINI *et al.*, 2012), eOGS (LEITE; ŠKRJANC, 2019), IBEM (LEITE; COSTA JR; GOMIDE, 2010) e eGNN (LEITE; COSTA; GOMIDE, 2010). Os códigos desses algoritmos foram disponibilizados pelos respectivos autores e todos foram executados na plataforma de desenvolvimento Matlab. O desempenho de todos os modelos foram avaliados a partir da acurácia dado pela Equação 4.1 e o desvio padrão pela Equação 4.2:

$$Acc(y, \hat{y}) = \frac{1}{T} \sum_{t=1}^T 1(\hat{y}^t = y^t) \quad (4.1)$$

em que T é o número total de amostras.

$$\sigma = \sqrt{\frac{\sum_{i=1}^N (f_i - \bar{f})^2}{N}} \quad (4.2)$$

em que N é o número de execuções f_i são os valores obtidos nas execuções e \bar{f} a média dos resultados.

Nos experimentos, adotou-se um conjunto diversificado composto por 10 bases de dados, que abrangem configurações tanto binárias quanto multi-classes além de distintas quantidades de amostras e atributos. O propósito foi avaliar o desempenho do eMGMC em ambientes heterogêneos. Todos os conjuntos de dados foram normalizados para o intervalo $[0,1]$ e seus dados foram embaralhados de forma aleatória. A Tabela 1 apresenta as bases utilizadas, acompanhadas de suas respectivas características.

Devido à ausência de padronização no desenvolvimento de testes de avaliação de algoritmos em ambientes não estacionários (ŠKRJANC *et al.*, 2019b), a opção foi realizar esses testes de maneira similar ao método empregado por Fernanda Rodrigues, Alisson Silva e Andre Lemos (2022). Dessa forma, conduziram-se simulações computacionais de processamento online, em que os parâmetros iniciais mais eficazes - aqueles que

Tabela 1 – Conjuntos de dados.

Nome	Classes	Amostras	Atributos
Ceasarian (AMIN; ALI, 2018)	2	80	5
Imonotherapy (IMMUNOTHERAPY DATASET..., 2018)	2	90	7
Fertility (GIL; GIRELA, 2013)	2	100	9
Transfusion (YEH, 2008)	2	748	3
Iris (FISHER, 1988)	3	150	4
Wine (AEBERHARD; FORINA, 1991)	3	178	13
Phishing (ABDELHAMID, 2016)	3	1353	9
Zoo (FORSYTH, 1990)	7	101	16
Dry Bean (DRY BEAN DATASET..., 2020)	7	13611	16
OptDigits (ALPAYDIN; KAYNAK, 1998)	10	5620	63

resultarem na menor taxa de erro e/ou maior acurácia - foram selecionados para avaliação. O método de seleção utilizado foi a busca exaustiva que, de forma sistemática, executa os possíveis valores de entrada.

A metodologia foi implementada em duas etapas distintas. Inicialmente, todas as bases selecionadas foram divididas em duas partes. Na primeira metade dos dados, foi empregada uma busca exaustiva para identificar os melhores valores dos parâmetros para os seis modelos, conforme organizados na Tabela 2. Os melhores parâmetros de cada base foram selecionados a partir da maior acurácia obtida após a execução dos algoritmos.

Na segunda etapa, os parâmetros selecionados foram empregados na segunda metade das bases de dados. Esse experimento foi repetido em 30 iterações, mediante a aleatorização das bases. Todos os resultados de acurácia são reunidos e o desvio padrão calculado para análise. Tal metodologia não apenas destaca o desempenho em cenários específicos, mas também revela a consistência e eficácia dos algoritmos em contextos diversos, contribuindo para uma compreensão mais profunda de sua aplicabilidade.

Tabela 2 – Variação e melhores valores dos parâmetros.

Models	Parameters	Passo	Caesarian	Dry Bean	Fertility	Imonotherapy	Iris	OptiDigits	Phishing	Transfusion	Wine	Zoo
FBEM	$\omega = 60, \dots, 200$	20	20	200	40	20	160	200	200	80	60	60
	$\alpha = 0$ (fixed)		0	0	0	0	0	0	0	0	0	0
	$\rho = 0.1, \dots, 1$	0.10	0.60	0.20	0.80	0.80	1	1	0.30	0.10	0.80	0.10
eOGS	$v = 100, \dots, 1000$	100	100	400	100	100	100	100	200	300	100	100
	$\beta = 2$ (fixed)		2	2	2	2	2	2	2	2	2	2
	$\gamma = 0.01$ (fixed)		0.010	0.010	0.010	0.010	0.010	0.010	0.010	0.010	0.010	0.010
	$\alpha = 0.1, \dots, 1$	0.1	1	0.70	0.10	0.40	0.40	0.10	0.80	0.10	0.80	0.10
	$\omega = 1$ (fixed)		1	1	1	1	1	1	1	1	1	1
ALMMO	$\omega = 10, \dots, 100$	10	90	90	70	60	10	40	40	80	70	50
	$\eta_0 = 0.10$ (fixed)		0.10	0.10	0.10	0.10	0.10	0.10	0.10	0.10	0.10	0.10
	$n^* = 0.1, \dots, 1$	0.60	0.70	0.90	0.20	0.60	0.20	0.10	0.40	0.60	0.40	0.9
	$\omega = 20, \dots, 200$	20	20	200	40	20	60	120	200	180	40	20
IBEM	$\eta = 3$ (fixed)		3	3	3	3	3	3	3	3	3	3
	$\rho = 0.1, \dots, 1$	0.10	0.70	0.20	0.40	0.10	0.10	0.10	0.1	0.90	0.1	0.1
	$\omega = 20, \dots, 200$	20	20	20	20	20	20	20	20	180	200	40
eGNN	$\eta = 2$ (fixed)		2	2	2	2	2	2	2	2	2	2
	$\rho = 3, \dots, 21$	3	18	6	12	12	6	9	3	3	3	3
	$\beta = 0.9$ (fixed)		0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.90
	$\chi = 0.1$ (fixed)		0.10	0.10	0.10	0.10	0.10	0.10	0.10	0.1	0.1	0.1
	$\zeta = 0.9$ (fixed)		0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.90
	$\omega = 20, \dots, 200$	20	20	120	100	20	40	80	140	20	20	20
eMGMC	$\eta = 2$ (fixed)		2	2	2	2	2	2	2	2	2	2
	$\lambda = 0.050$ (fixed)		0.050	0.050	0.050	0.050	0.050	0.050	0.050	0.050	0.050	0.050
	$\alpha = 0.10$ (fixed)		0.10	0.10	0.10	0.10	0.10	0.10	0.10	0.10	0.10	0.10
	$K_{init} = 0.1, \dots, 0.0001$	10^{-1}	0.00010	0.0010	0.00010	0.00010	0.00010	0.010	0.01	0.0010	0.010	0.10

4.2 Resultados Experimentais

A Tabela 3 apresenta os resultados experimentais derivados da avaliação de desempenho do eMGMC, incluindo uma análise comparativa com os resultados obtidos por classificadores do estado da arte. Além disso, a tabela detalha a Acurácia Média e o Desvio Padrão considerando todos os experimentos realizados por cada classificador, i.e., os experimentos de cada modelo nas 30 execuções para cada uma das 10 bases.

Tabela 3 – Desempenho do classificadores.

Datasets	eMGMC	FBEM	eOGS	ALMMO	IBEM	eGNN
Caesarian	58,1 ± 6,28	57,5 ± 0,00	51,8 ± 3,85	62,5 ± 4,91	50,0 ± 0,00	50,9 ± 7,32
Dry_Bean	82,4 ± 0,51	74,4 ± 0,14	24,0 ± 0,21	18,2 ± 0,56	41,1 ± 2,16	64,4 ± 2,09
Fertility	87,9 ± 1,04	85,0 ± 0,00	84,1 ± 1,84	81,8 ± 18,84	25,9 ± 0,36	86,0 ± 8,00
Immunotherapy	76,3 ± 5,68	48,9 ± 0,20	69,6 ± 2,32	77,7 ± 4,87	68,8 ± 0,40	70,7 ± 19,56
Iris	93,3 ± 2,45	72,7 ± 0,00	37,4 ± 2,33	66,3 ± 4,67	86,8 ± 0,40	83,3 ± 9,93
OptDigits	91,1 ± 0,40	93,3 ± 0,04	*	12,3 ± 2,53	86,3 ± 0,02	65,6 ± 5,28
Phishing	77,6 ± 1,14	82,0 ± 0,00	54,2 ± 1,13	39,2 ± 12,63	71,8 ± 0,16	48,2 ± 6,03
Transfusion	68,0 ± 2,95	63,44 ± 0,41	65,9 ± 1,30	77,6 ± 1,85	75,31 ± 3,4	36,7 ± 19,59
Wine	93,1 ± 1,45	92,6 ± 0,40	35,5 ± 2,67	32,0 ± 3,74	92,8 ± 3,74	48,7 ± 7,17
Zoo	79,0 ± 2,80	72,5 ± 0,18	35,3 ± 2,26	39,6 ± 5,12	56,7 ± 0,70	47,5 ± 7,25
MÉDIA	80,7 ± 2,52	74,2 ± 0,14	50,9 ± 2,00	50,7 ± 5,97	65,5 ± 1,20	60,2 ± 8,10

Os resultados apresentados na Tabela 3 mostram que o eMGMC obteve o melhor desempenho em 5 das 10 bases analisadas. Na outras 5 bases, ele alcançou a segunda melhor Acurácia em 4 bases e o terceiro melhor desempenho em 1 base. Por outro lado, o ALMMo obteve a maior Acurácia em 3 bases. Na outras 7 bases, o ALMMo ficou entre o 5º e 6º melhor desempenho. Já o FBEM obteve o melhor desempenho em 2 bases, o segundo melhor em 3 bases. Nas 5 restantes, o FBEM alcançou entre o 3º e 6º melhor desempenho. Os classificadores eOGS, IBEM e eGNN não obtiveram a maior Acurácia em nenhuma das bases e seus desempenhos alternaram entre o 2º e 6º melhores. Na média de todos os experimentos o melhor desempenho foi obtido pelo eMGMC, seguido pelo FBEM, IBEM, eGNN, eOGS e ALMMo. Destaca-se que o eMGMC alcançou uma Acurácia média 6,5% superior o segundo melhor colocado e 30,0% superior ao 6º colocado. Note que, apesar do ALMMo ter obtido o melhor desempenho em 2 bases de dados, no geral seu desempenho foi inferior aos demais algoritmos. Por fim, avaliando o desvio padrão, percebe-se que o classificador FBEM possui a menor variação, seguido pelo IBEM, eOGS e eMGMC. O desvio padrão desses três classificadores são comparáveis e inferiores aos apresentados pelo eGNN e ALMMo. A Tabela 4 ilustra a ordem de classificação para cada uma das bases de dados. Os resultados apresentados na tabela corroboram os descritos no parágrafo anterior.

Tabela 4 – Posição dos algoritmos

Datasets	eMGMC	FBEM	eOGS	ALMMO	IBEM	eGNN
Caesarian	2	3	4	1	6	5
Dry_Bean	1	2	5	6	4	3
Fertility	1	3	4	5	6	2
Immunothrapy	2	6	4	1	5	3
Iris	1	4	6	5	2	3
OptDigits	2	1	6	5	3	4
Phishing	2	1	4	6	3	5
Transfusion	3	5	4	1	2	6
Wine	1	2	5	6	3	4
Zoo	1	2	6	5	3	4

A Figura 2 ilustra representações gráficas da evolução da estrutura do eMGC em cada base de dados. Nela, é visível a capacidade do algoritmo em ajustar sua estrutura de acordo com a apresentação dos dados o que evidencia a capacidade do modelo em excluir e/ou mesclar grupos conforme necessário. Vale destacar que, ao término da execução, o número de grupos nunca é inferior ao número de classes na base, respeitando, assim, uma das restrições do modelo, que consiste em não excluir o único grupo associado a uma determinada classe.

Em bases como Caesarian, Fertility, Iris, Wine, Immunotherapy e Zoo, observou-se que a estrutura do modelo experimentou poucas adaptações, o que resultou em um número de grupos próximo ao número de classes e manteve-se estável ao longo da execução. Nas bases binárias, como Caesarian, Immunotherapy e Fertility, a execução foi concluída com 4, 3 e 2 grupos, respectivamente. Já em bases com um maior número de classes, como Iris, Wine, que possuem 3 classes, e a Zoo, com 7 classes, a execução foi finalizada com 3, 4, 4 e 9 grupos, respectivamente. Por outro lado, as bases restantes, transfusion, Phishing, DryBean, e OptDigits, que apresentam um maior número de amostras, conseqüentemente, apresentaram um número maior de grupos em sua estrutura, totalizando 18, 43, 87 e 102 grupos.

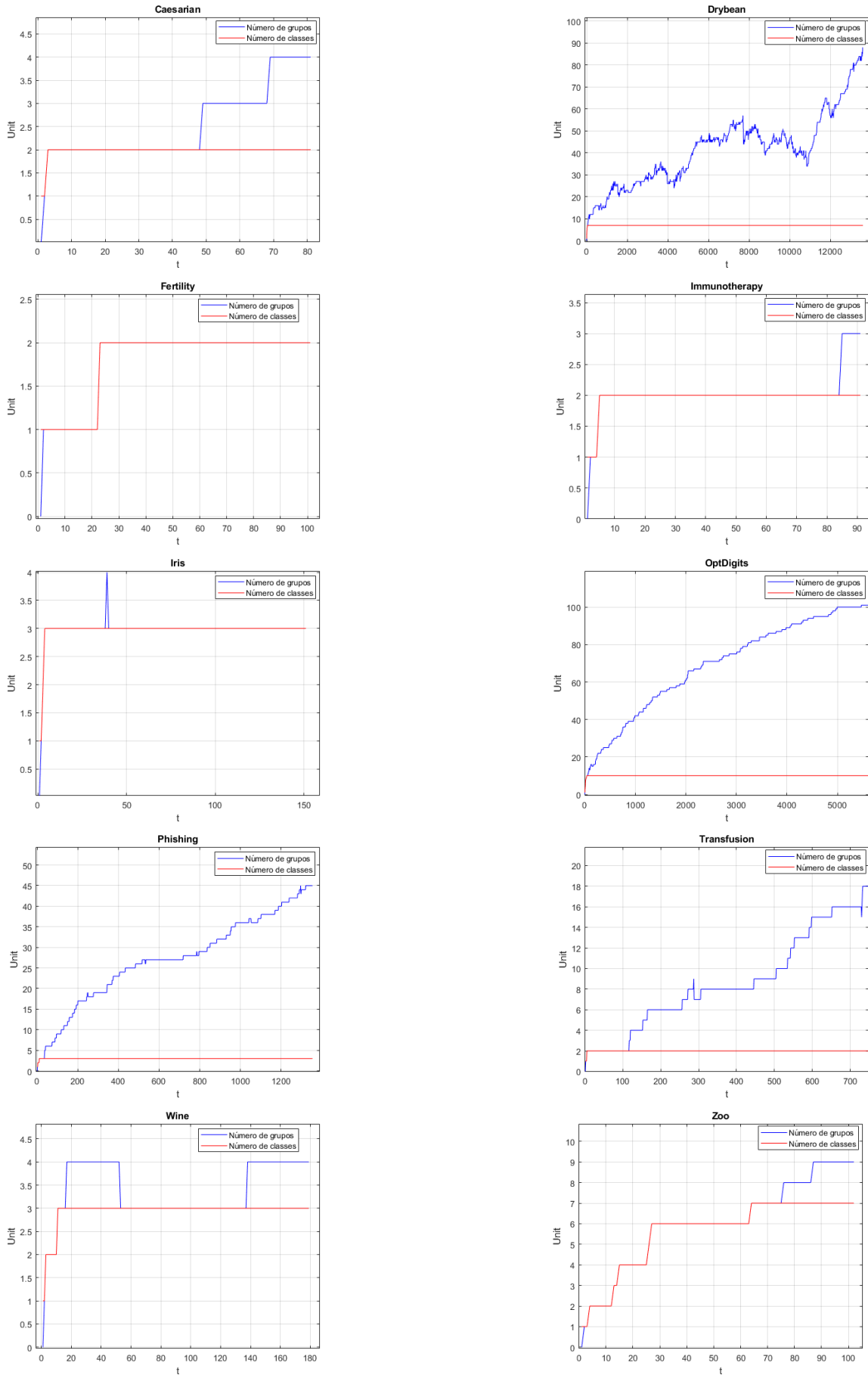


Figura 2 – Evolução da estrutura do eMGMC.

5 CONCLUSÃO

Este trabalho propõe um classificador evolutivo que incorpora um algoritmo de agrupamento baseado em aprendizado participativo. A base de regras gerada por este algoritmo apresenta o antecedente como funções de pertinência gaussiana multivariável, enquanto os consequentes são representados pelas classes. A formação de grupos e regras emprega um mecanismo de procrastinação, responsável por garantir a integridade do modelo de dados em relação a *outliers*. Dois mecanismos que controlam o crescimento da estrutura foram implementados: o método de exclusão de grupos e regras, que incorpora conceitos de idade e população, e o método de mesclagem de grupos e regras, baseado no conceito de sobreposição. Essa estratégia, aliada ao método de atualização dos grupos, confere dinamismo à estrutura do eMGMC, permitindo sua adaptação ao fluxo de amostras de entrada.

O desempenho do eMGMC foi avaliado e comparado com outros algoritmos do estado da arte para a tarefa de classificação. Experimentos foram conduzidos com 10 bases para avaliar e comparar a execução do algoritmo proposto. Os resultados computacionais mostram que o algoritmo proposto possui as características desejáveis de um EFS, que é a evolução de sua estrutura simultaneamente com os ajustes de seus parâmetros. O classificador consegue obter uma boa acurácia em diferentes bases de dados com um resultado que é superior ou comparável aos demais algoritmos do estado da arte utilizados nesse trabalho.

Como propostas de continuidade da proposta no contexto de aperfeiçoamento do eMGMC tem-se as seguintes perspectivas:

- avaliar outros métodos para exclusão e união de grupos e regras. Por exemplo, a técnica do casco convexo por ser utilizada como mecanismo para união de grupos e regras.
- utilizar outras forma para o cálculo da distância para a medida de pertinência.
- implementar um método de seleção de variáveis de entrada ao eMGMC. A utilização da razão das médias amostrais pode ser uma abordagem interessante para esse fim.

REFERÊNCIAS

ABDELHAMID, Neda. **Website Phishing**. [S. l.: s. n.], 2016. UCI Machine Learning Repository. DOI: <https://doi.org/10.24432/C5B301>.

AEBERHARD, Stefan; FORINA, M. **Wine**. [S. l.: s. n.], 1991. UCI Machine Learning Repository. DOI: <https://doi.org/10.24432/C5PC7J>.

AGUIAR, Charles; LEITE, Daniel. Unsupervised Fuzzy eIX: Evolving Internal-eXternal Fuzzy Clustering. **IEEE Conference on Evolving and Adaptive Intelligent Systems**, Institute of Electrical e Electronics Engineers Inc., 2020-May, maio 2020. ISSN 24734691. DOI: 10.1109/EAIS48028.2020.9122774.

AHWIADI, Mohamed; WANG, Wilson. An Adaptive Evolving Fuzzy Technique for Prognosis of Dynamic Systems. **IEEE Transactions on Fuzzy Systems**, Institute of Electrical e Electronics Engineers Inc., v. 30, p. 841–849, 3 mar. 2022. ISSN 19410034. DOI: 10.1109/TFUZZ.2021.3049916.

ALPAYDIN, E.; KAYNAK, C. **Optical Recognition of Handwritten Digits**. [S. l.: s. n.], 1998. UCI Machine Learning Repository. DOI: <https://doi.org/10.24432/C50P49>.

AMIN, Muhammad; ALI, Amir. **Caesarian Section Classification Dataset**. [S. l.: s. n.], 2018. UCI Machine Learning Repository. DOI: 10.24432/C5N59X.

ANGELOV, Plamen; FILEV, Dimitar. Simpl-eTS A simplified method for learning evolving Takagi-Sugeno fuzzy models. **IEEE International Conference on Fuzzy Systems**, p. 1068–1073, 2005. ISSN 10987584. DOI: 10.1109/FUZZY.2005.1452543.

ANGELOV, Plamen; GU, Xiaowei. Autonomous learning multi-model classifier of 0-Order (ALMMo-0). **IEEE Conference on Evolving and Adaptive Intelligent Systems**, Institute of Electrical e Electronics Engineers Inc., 2017-May, jun. 2017. ISSN 24734691. DOI: 10.1109/EAIS.2017.7954832.

ANGELOV, Plamen P.; FILEV, Dimitar P. An Approach to Online Identification of Takagi-Sugeno Fuzzy Models. **IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics**, v. 34, p. 484–498, 1 fev. 2004. ISSN 10834419. DOI: 10.1109/TSMCB.2003.817053.

BAABDULLAH, Tahani; ALZHRANI, Amani; RAWAT, Danda B. On the Comparative Study of Prediction Accuracy for Credit Card Fraud Detection wWith Imbalanced Classifications, p. 1–12, 2020. DOI: 10.22360/SpringSim.2020.CSE.004.

CHEN, Min; MAO, Shiwen; LIU, Yunhao. Big data: A survey. **Mobile Networks and Applications**, Kluwer Academic Publishers, v. 19, p. 171–209, 2 abr. 2014. ISSN 1383469X. DOI: 10.1007/S11036-013-0489-0/FIGURES/6.

DAWOOD, Aisha *et al.* Prediction of Environmental Earth Surface Temperature using hybrid machine learning model. **2023 Sixth International Conference of Women in Data Science at Prince Sultan University (WiDS PSU)**, IEEE, p. 127–132, mar. 2023. DOI: 10.1109/WIDS-PSU57071.2023.00036.

DRY BEAN DATASET. [S. l.: s. n.], 2020. UCI Machine Learning Repository. DOI: <https://doi.org/10.24432/C50S4B>.

FISHER, R. A. **Iris**. [S. l.: s. n.], 1988. UCI Machine Learning Repository. DOI: <https://doi.org/10.24432/C56C76>.

FORSYTH, Richard. **Zoo**. [S. l.: s. n.], 1990. UCI Machine Learning Repository. DOI: <https://doi.org/10.24432/C5R59V>.

GIL, David; GIRELA, Jose. **Fertility**. [S. l.: s. n.], 2013. UCI Machine Learning Repository. DOI: <https://doi.org/10.24432/C5Z01Z>.

IMMUNOTHERAPY DATASET. [S. l.: s. n.], 2018. UCI Machine Learning Repository. DOI: <https://doi.org/10.24432/C5DC72>.

LEITE, Daniel; BALLINI, Rosangela *et al.* Evolving fuzzy granular modeling from nonstationary fuzzy data streams. **Evolving Systems**, v. 3, jun. 2012. DOI: 10.1007/s12530-012-9050-9.

LEITE, Daniel; COSTA, Pyramo; GOMIDE, Fernando. Evolving granular neural network for semi-supervised data stream classification, p. 1–8, 2010. DOI: 10.1109/IJCNN.2010.5596303.

LEITE, Daniel; COSTA, Pyramo; GOMIDE, Fernando. Interval approach for evolving granular system modeling. **Learning in non-stationary environments: methods and applications**, Springer, p. 271–300, 2012.

LEITE, Daniel; COSTA JR, Pyramo Pires da Costa Júnior; GOMIDE, Fernando. Granular Approach for Evolving System Modeling, p. 340–349, jun. 2010. DOI: 10.1007/978-3-642-14049-5_35.

LEITE, Daniel; DECKER, Leticia *et al.* EGFC: Evolving Gaussian Fuzzy Classifier from Never-Ending Semi-Supervised Data Streams – With Application to Power Quality Disturbance Detection and Classification. IEEE, Glasgow, UK, p. 1–9, 2020. DOI: 10.1109/FUZZ48607.2020.9177847.

LEITE, Daniel; ŠKRJANC, Igor. Ensemble of evolving optimal granular experts, OWA aggregation, and time series prediction. **Information Sciences**, v. 504, p. 95–112, 2019. ISSN 0020-0255. DOI: <https://doi.org/10.1016/j.ins.2019.07.053>.

LEMOS, Andre; CAMINHAS, Walmir; GOMIDE, Fernando. Multivariable gaussian evolving fuzzy modeling system. **IEEE Transactions on Fuzzy Systems**, v. 19, p. 91–104, 1 fev. 2011. ISSN 10636706. DOI: 10.1109/TFUZZ.2010.2087381.

LUGHOFER, E.; ANGELOV, P. Handling drifts and shifts in on-line data streams with evolving fuzzy systems. **Applied Soft Computing**, Elsevier, v. 11, p. 2057–2068, 2 mar. 2011. ISSN 1568-4946. DOI: 10.1016/J.ASOC.2010.07.003.

LUGHOFER, Edwin; ANGELOV, Plamen; ZHOU, Xiaowei. Evolving single- and multi-model fuzzy classifiers with FLEXFIS-class. **IEEE International Conference on Fuzzy Systems**, 2007. ISSN 10987584. DOI: 10.1109/FUZZY.2007.4295393.

LUGHOFER, Edwin; KLEMENT, Erich Peter. FLEXFIS: A variant for incremental learning of Takagi-Sugeno fuzzy systems. **IEEE International Conference on Fuzzy Systems**, p. 915–920, 2005. ISSN 10987584. DOI: 10.1109/FUZZY.2005.1452516.

LUGHOFER, Edwin; PRATAMA, Mahardhika; SKRJANC, Igor. Incremental rule splitting in generalized evolving fuzzy systems for autonomous drift compensation. **IEEE Transactions on Fuzzy Systems**, Institute of Electrical e Electronics Engineers Inc., v. 26, p. 1854–1865, 4 ago. 2018. ISSN 10636706. DOI: 10.1109/TFUZZ.2017.2753727.

NGUYEN, Hai Long; WOON, Yew Kwong; NG, Wee Keong. A survey on data stream clustering and classification. **Knowledge and Information Systems**, Springer London, v. 45, p. 535–569, 3 dez. 2015. ISSN 02193116. DOI: 10.1007/S10115-014-0808-1/FIGURES/8.

OKAMOTO, W. *et al.* FLINS-fuzzy natural language communication system. v. 5, 1–2 vol.5, 1995. DOI: 10.1109/FUZZY.1995.410014.

PRATAMA, Mahardhika; LU, Jie; ZHANG, Guangquan. An incremental interval Type-2 neural fuzzy Classifier. **IEEE International Conference on Fuzzy Systems**, Institute of Electrical e Electronics Engineers Inc., 2015-November, nov. 2015. ISSN 10987584. DOI: 10.1109/FUZZ-IEEE.2015.7337801.

RAZA, Mansoor; JAYASINGHE, Nathali Dilshani; MUSLAM, Muhana Magboul Ali. A Comprehensive Review on Email Spam Classification using Machine Learning Algorithms, p. 327–332, 2021. DOI: 10.1109/ICOIN50884.2021.9334020.

RODRIGO, Rodolfo Horacio; PATINO, Hector; SCHWEICKARDT, Gustavo. Contractive Method for Identification of Nonlinear Regressors in Dynamic Systems. Institute of Electrical e Electronics Engineers (IEEE), p. 1–1, ago. 2021. DOI: 10.1109/ARGENCON49523.2020.9505394.

RODRIGUES, Fernanda; SILVA, Alisson; LEMOS, Andre. Evolving fuzzy predictor with multivariable Gaussian participatory learning and multi-innovations recursive weighted

least squares: eFMI. **Evolving Systems**, v. 13, p. 1–20, fev. 2022. DOI: 10.1007/s12530-022-09421-9.

RODRIGUES, Fernanda P.S.; SILVA, Alisson Marques; LEMOS, Andre Paim. Evolving Fuzzy System with Multivariable Gaussian Participatory Learning and Recursive Maximum Correntropy - EFCE. **IEEE International Conference on Fuzzy Systems**, Institute of Electrical e Electronics Engineers Inc., 2021-July, jul. 2021. ISSN 10987584. DOI: 10.1109/FUZZ45933.2021.9494451.

ROSA, Gabriel *et al.* Buying and Selling Decision in the Brazilian Stock Exchange Financial Market by a Neuro Fuzzy Network (NFN) Applied to the Hurwicz Criterion. João Pessoa, Paraíba, ago. 2023.

ROUSSELL, Peter Novig Stuart J. Inteligência artificial: Uma abordagem moderna, p. 591, 2022.

SHAH, Jital; WANG, Wilson. An evolving neuro-fuzzy classifier for fault diagnosis of gear systems. **ISA Transactions**, Elsevier, v. 123, p. 372–380, abr. 2022. ISSN 0019-0578. DOI: 10.1016/J.ISATRA.2021.05.019.

SHARMA, Ankita; GHOSE, Udayan. Toward Machine Learning Based Binary Sentiment Classification of Movie Reviews for Resource Restraint Language (RRL)—Hindi. **IEEE Access**, v. 11, p. 58546–58564, 2023. DOI: 10.1109/ACCESS.2023.3283461.

SHYAMSUKHA, Udith *et al.* Modeling of Predictable Variations in Multi-Time Instant Ambient Temperature Time Series. **3rd International Conference on Energy, Power and Environment: Towards Clean Energy Technologies, ICEPE 2020**, Institute of Electrical e Electronics Engineers Inc., mar. 2021. DOI: 10.1109/ICEPE50861.2021.9404497.

SILVA, Alisson Marques *et al.* A fast learning algorithm for evolving neo-fuzzy neuron. **Applied Soft Computing**, v. 14, p. 194–209, 2014. Evolving Soft Computing Techniques and Applications. ISSN 1568-4946. DOI: <https://doi.org/10.1016/j.asoc.2013.03.022>.

ŠKRJANC, Igor *et al.* Evolving fuzzy and neuro-fuzzy approaches in clustering, regression, identification, and classification: A Survey. **Information Sciences**, Elsevier, v. 490, p. 344–368, jul. 2019. ISSN 0020-0255. DOI: 10.1016/J.INS.2019.03.060.

ŠKRJANC, Igor *et al.* Evolving fuzzy and neuro-fuzzy approaches in clustering, regression, identification, and classification: A Survey. **Information Sciences**, v. 490, p. 344–368, 2019. ISSN 0020-0255. DOI: <https://doi.org/10.1016/j.ins.2019.03.060>.

TAVARES, Emmanuel; SILVA, Alisson Marques; MOITA, Gray Farias. A fast clustering algorithm for evolving fuzzy classifier based on samples mean. **Journal of Intelligent Fuzzy Systems**, IOS Press, v. 43, p. 6897–6908, 6 jan. 2022. ISSN 1064-1246. DOI: 10.3233/JIFS-212831.

TORRES, Luís M.M.; SERRA, Ginalber L.O. A novel approach for online multivariable evolving fuzzy modeling from experimental data. **IEEE International Conference on Fuzzy Systems**, Institute of Electrical e Electronics Engineers Inc., 2018-July, out. 2018. ISSN 10987584. DOI: 10.1109/FUZZ-IEEE.2018.8491470.

WANG, Huazhe; MA, Li. Image Generation and Recognition Technology Based on Attention Residual GAN. **IEEE Access**, p. 1–1, 2023. ISSN 2169-3536. DOI: 10.1109/ACCESS.2023.3287854.

YAGER, Ronald R. A Model of Participatory Learning. **IEEE Transactions on Systems, Man and Cybernetics**, v. 20, p. 1229–1234, 5 1990. ISSN 21682909. DOI: 10.1109/21.59986.

YEH, I-Cheng. **Blood Transfusion Service Center**. [S. l.: s. n.], 2008. UCI Machine Learning Repository. DOI: <https://doi.org/10.24432/C5GS39>.

ZADEH, L. A. Fuzzy sets. **Information and Control**, Academic Press, v. 8, p. 338–353, 3 jun. 1965. ISSN 0019-9958. DOI: 10.1016/S0019-9958(65)90241-X.