Universidade Federal de São Carlos– UFSCar
Centro de Ciências Exatas e de Tecnologia– CCET
Departamento de Computação– DC
Programa de Pós-Graduação em Ciência da Computação– PPGCC

Bruno Mass

Classificação Semissupervisionada Baseada em Densidade com Reconhecimento de Anomalias

São Carlos
2025

Bruno Mass

Classificação Semissupervisionada Baseada em Densidade com Reconhecimento de Anomalias

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação do Centro de Ciências Exatas e de Tecnologia da Universidade Federal de São Carlos, como parte dos requisitos para a obtenção do título de Mestre em Ciência da Computação.

Área de concentração: AMPLN

Orientador: Prof. Dr. Murilo C. Naldi

São Carlos
2025

Mass, Bruno

Classificação semi-supervisionada baseada em densidade com reconhecimento de anomalias / Bruno Mass -- 2025.
93f.

Dissertação (Mestrado) - Universidade Federal de São Carlos, campus São Carlos, São Carlos
Orientador (a): Murilo Coelho Naldi
Banca Examinadora: Murilo Coelho Naldi, Alan Demétrius Baria Valejo, Pablo Andretta Jaskowiak
Bibliografia

1. Aprendizado Semissupervisionado. 2. Classificação Semissupervisionada. 3. Detecção de Anomalias. I. Mass, Bruno. II. Título.

Ficha catalográfica desenvolvida pela Secretaria Geral de Informática (SIn)

DADOS FORNECIDOS PELO AUTOR

Bibliotecário responsável: Arildo Martins - CRB/8 7180

UNIVERSIDADE FEDERAL DE SÃO CARLOS

Centro de Ciências Exatas e de Tecnologia
Programa de Pós-Graduação em Ciência da Computação

Folha de Aprovação

Defesa de Dissertação de Mestrado do candidato Bruno Mass, realizada em 05/12/2025.

Comissão Julgadora:

Prof. Dr. Murilo Coelho Naldi (UFSCar)

Prof. Dr. Alan Demétrius Baria Valejo (UFSCar)

Prof. Dr. Pablo Andretta Jaskowiak (UFSC)

O Relatório de Defesa assinado pelos membros da Comissão Julgadora encontra-se arquivado junto ao Programa de Pós-Graduação em Ciência da Computação.

Dedico este trabalho à minha esposa Isabelle. Sem ela, não teria sido possível.

Agradecimentos

Agradeço à minha esposa Isabelle e minhas duas filhas Sofia e Olívia por toda a paciência e compreensão durante minhas muitas ausências necessárias para a realização do curso.

Agradeço também ao professor Murilo Naldi por todo o conhecimento e orientações transmitidas durante todo o decorrer do curso.

Agradeço ao professor Jadson Gertrudes por toda a paciência e orientações recebidas.

Agradeço a minha mãe Vera por toda a sua ajuda com meus almoços e café no período das aulas presenciais.

Agradeço a meu pai Baker por todos os ensinamentos e por ser o exemplo de pessoa que me fez quem sou.

Agradeço à UFSCar e ao Departamento de Computação por me proporcionar a oportunidade de realizar um sonho - o de me formar como mestre em um dos melhores cursos do mundo.

"It is the responsibility of scientists never to suppress knowledge, no matter how awkward that knowledge is, no matter how it may bother those in power; we are not smart enough to decide which pieces of knowledge are permissible and which are not."
(Carl Sagan)

Resumo

No contexto de mineiração de dados, a tarefa de detecção de anomalias é importante pois observações divergentes do todo podem afetar negativamente de modelos de aprendizado de máquina ou constituir o principal objeto de interesse em diversos cenários reais. Ao mesmo tempo, tarefas de classificação semissupervisionada mostram-se essenciais em contextos nos quais os dados rotulados são escassos. Neste trabalho, sugerimos a unificação das duas tarefas em um processo único integrado: propomos a combinação de um algoritmo considerado estado da arte em agrupamento baseado em densidade, capaz de detectar anomalias, com dois conhecidos classificadores semissupervisionados baseados em densidade, com o objetivo de produzir métodos híbridos capazes de desempenhar ambas as tarefas. Experimentos conduzidos em 42 conjuntos de dados semissintéticos com diferentes proporções de objetos rotulados e dois tipos distintos de anomalias mostraram que o método de detecção de anomalias investigado apresenta desempenho superior ao de métodos similares, principalmente em conjuntos de dados contendo anomalias globais. Os resultados também comprovam que, quando o método de detecção de anomalias é combinado com os classificadores semissupervisionados, há um baixo impacto na qualidade da tarefa de classificação. Desta forma demonstramos que as abordagens híbridas propostas constituem alternativas viáveis aos seus respectivos métodos originais, permitindo a identificação explícita de anomalias sem comprometer de forma significativa a qualidade da tarefa de classificação.

Palavras-chave: Aprendizado semissupervisionado. Classificação semissupervisionada. Detecção de anomalias.

Abstract

In the context of data mining, the task of anomaly detection is important because observations that deviate from the majority can negatively affect machine learning models or represent the main object of interest in various real-world scenarios. At the same time, semi-supervised classification tasks are essential in situations where labeled data are scarce. In this work, we suggest unifying these two tasks into a single integrated process: we propose combining a state-of-the-art density-based clustering algorithm capable of detecting outliers with two well-known density-based semi-supervised classifiers, with the goal of producing hybrid methods capable of performing both tasks. Experiments conducted on 42 semi-synthetic datasets with different proportions of labeled objects and two distinct types of anomalies showed that the investigated anomaly detection method outperforms similar approaches, especially on datasets containing global anomalies. The results also demonstrate that when the outlier detection method is combined with the semi-supervised classifiers, there is only a minor impact on classification quality. Thus, we show that the proposed hybrid approaches constitute viable alternatives to their respective original methods, enabling explicit identification of anomalies without significantly compromising classification performance.

Keywords: Semi-supervised learning. Semi-supervised classification. Anomaly detection.

Lista de ilustrações

Lista de tabelas

Lista de siglas

AGM Árvore Geradora Mínima

ARI *Adjusted Rand Index*

CoreSG *Core-distance based Spanning Graph*

DAM Distância de Alcançabilidade Mútua

FOSC *Framework for Optimal Selection of Clusters*

GAM Grafo de Alcançabilidade Mútua

GLOSH *Global-Local Outlier Scores from Hierarchies*

IEEE Institute of Electrical and Electronics Engineers

*k***NNG** *k Nearest Neighbors Graph*

MMG Modelo de Mistura Gaussiana

R*k***NN** *Reverse k Nearest Neighbors*

RkNNG *Reverse k Nearest Neighbors Graph*

Sumário

Capítulo 1

Introdução

Nos últimos anos, a quantidade de dados produzidos pelas atividades humanas tem aumentado significativamente. Como consequência, são cada vez mais frequentes situações em que dados devidamente rotulados são escassos e difíceis de obter, pois dependem de especialistas humanos ou de experimentos dispendiosos, enquanto dados não rotulados são relativamente baratos e mais acessíveis (ZHU; GOLDBERG, 2009). Em tais cenários, a quantidade de dados rotulados geralmente não é representativa o suficiente para o uso de métodos tradicionais de aprendizado supervisionado. De acordo com a publicação de Chapelle, Schölkopf e Zien (2006), para lidar com problemas dessa natureza, é mais apropriado adotar o paradigma de aprendizado semissupervisionado, que combina características dos paradigmas tradicionais de aprendizado supervisionado e não supervisionado. O paradigma de aprendizado semissupervisionado representa um meio-termo entre os dois e envolve técnicas que aproveitam a porção não rotulada dos dados em conjunto com os rótulos previamente conhecidos, produzindo resultados de maior qualidade do que aqueles obtidos por abordagens puramente não supervisionadas (CHAPELLE; SCHöLKOPF; ZIEN, 2006). Isso desde que a distribuição das observações seja relevante para o problema em questão (CHAPELLE; SCHöLKOPF; ZIEN, 2006) e desde que a estrutura do problema seja consistente com as premissas do modelo (GERTRUDES et al., 2019).

O aprendizado semissupervisionado pode se manifestar de duas formas: aprendizado indutivo e aprendizado transdutivo. Ambos buscam construir um modelo a partir de todas as instâncias, rotuladas e não rotuladas. No entanto, o primeiro visa prever o objetivo de instâncias futuras, desconhecidas para o modelo, enquanto o segundo busca prever o valor objetivo apenas das instâncias não rotuladas conhecidas (ZHU; GOLDBERG, 2009). De forma ampla, problemas de aprendizado semissupervisionado podem ser divididos em

problemas de regressão e de classificação, dependendo da natureza do objetivo (ZHU; GOLDBERG, 2009). Enquanto a regressão é útil para prever valores numéricos contínuos, como preços de imóveis ou consumo de energia, a classificação busca solucionar problemas em que o objetivo é uma lista finita de opções discretas. Nosso trabalho tem como foco a transdução semissupervisionada para classificação, que tem sido amplamente aplicada com sucesso em desafios reais, como a classificação de textos (JOACHIMS et al., 1999), o diagnóstico de imagens (FILIPOVYCH; DAVATZIKOS, 2011) e a detecção de fraudes (XIANG et al., 2023).

Uma suposição comum entre os métodos transdutivos semissupervisionados existentes é que os dados de treinamento fornecidos foram curados previamente e, portanto, são confiáveis. Na prática, isso raramente ocorre. Conjuntos de dados reais frequentemente contêm anomalias (*outliers*): observações cujas características incomuns desviam significativamente da distribuição geral dos dados, a ponto de levantar a suspeita de que foram geradas por um processo diferente (HAWKINS, 1980). Anomalias são observações raras, geralmente representando apenas uma fração muito pequena do conjunto total de dados (AGYEMANG; BARKER; ALHAJJ, 2006). Apesar de sua raridade, em muitos cenários reais, a identificação de anomalias é de grande importância: anomalias podem refletir erros de medição ou de coleta que degradam a qualidade de modelos de aprendizado de máquina (GHOSH et al., 2024). Nesses casos, a detecção de anomalias é fundamental para permitir sua remoção dos dados de entrada. Em outros contextos, como a detecção de fraudes em sistemas financeiros (NGAI et al., 2011) ou a identificação de doenças raras (SCHLEGL et al., 2017), o comportamento diferente representado pelas anomalias pode ser de maior interesse do que o de observações normais (KNORR; NG, 1998).

Muito embora a tarefa de detecção de anomalias tenha considerável representatividade na literatura (HODGE; AUSTIN, 2004; CHANDOLA; BANERJEE; KUMAR, 2009; VINCES et al., 2025), relativamente pouca atenção tem sido dada à integração direta da detecção de anomalias a processos de classificação semissupervisionada. Na literatura, a classificação semissupervisionada e a detecção de anomalias são comumente tratadas como tarefas independentes: a maioria dos métodos de classificação semissupervisionada não é capaz de detectar anomalias nativamente, e a maioria dos métodos de detecção de anomalias se concentra apenas na decisão sobre se uma observação é ou não uma anomalia, desconsiderando semelhanças e diferenças estruturais entre observações normais. Uma abordagem popular para lidar com anomalias em contextos de classificação é removê-los em uma etapa de pré-processamento antes de aplicar o classificador. Esta abordagem em duas etapas é, por definição, uma solução não ótima: são necessários esforços adicionais de integração entre processos, e os métodos de detecção de anomalias utilizados podem ser baseados em suposições centrais diferentes das do algoritmo principal de classificação. Também podem requerer parâmetros diferentes, implicando em trabalho adicional de exploração para a obtenção de resultados adequados. Em contraste, propomos uma

abordagem diferente: a integração direta das duas tarefas, ambas baseando-se nas mesmas suposições centrais, com o objetivo de produzir um único mecanismo não paramétrico capaz de realizar classificação semissupervisionada e simultaneamente detectar a presença de anomalias nos dados.

Para alcançar o objetivo proposto, recorremos a métodos semissupervisionados de agrupamento e classificação baseados em densidade, fundamentados nos seguintes argumentos: em primeiro lugar, a transdução semissupervisionada baseada em densidade para agrupamento é bastante similar à transdução baseada em densidade para classificação, a principal diferença sendo que, em tarefas de agrupamento, nem todas as classes precisam ser previamente conhecidas durante o treinamento, podendo haver classes não observadas nos dados de teste (GERTRUDES et al., 2019). Em segundo lugar, o trabalho de Gertrudes et al. (2019) demonstrou que o agrupamento baseado em densidade pode ser utilizado para classificação semissupervisionada, desde que as classes sigam a estrutura natural de agrupamentos dos dados. Em terceiro lugar, o paradigma de agrupamento baseado em densidade permite, de forma nativa, que alguns objetos permaneçam sem rótulo, sendo considerados "ruído" (ESTER et al., 1996; LELIS; SANDER, 2009; CAMPELLO et al., 2015), conceito equivalente ao de anomalias do ponto de vista de densidade (BREUNIG et al., 2000): as mesmas suposições centrais que definem um agrupamento também são utilizadas para definir anomalias. Por fim, as capacidades nativas de detecção de anomalias de métodos de agrupamento baseados em densidade ainda não foram totalmente exploradas na literatura, o que sugere tratar-se de um campo promissor de investigação.

1.1 Objetivos

- ❏ Introduzir dois novos métodos híbridos por meio da combinação de um arcabouço não supervisionado baseado em densidade capaz de detectar anomalias com dois conhecidos classificadores semissupervisionados baseados em densidade.

- ❏ Comparar os dois métodos propostos com algoritmos similares baseados em densidade e demonstrar que são superiores em detectar anomalias e competitivos em termos de classificação semissupervisionada.

- ❏ Demonstrar que o arcabouço não supervisionado baseado em densidade utilizado neste trabalho para detectar anomalias apresenta melhor desempenho em identificar anomalias globais do que em identificar anomalias locais.

O restante deste trabalho está organizado da seguinte forma: o Capítulo 2 revisa os conceitos e trabalhos anteriores relevantes para nosso trabalho, o Capítulo 3 discute o método baseado em densidade que utilizamos para detecção de anomalias e apresenta os métodos híbridos propostos, o Capítulo 4 descreve a metodologia experimental utilizada

e discute os resultados obtidos, e por fim o Capítulo 5 traz as conclusões e sugestões sobre pesquisas futuras.

Capítulo 2

Trabalhos Relacionados

Neste capítulo, apresentamos os principais conceitos e trabalhos científicos que serviram como base para nosso trabalho. Em primeiro lugar, detalhamos um importante arcabouço para classificação semissupervisionada baseada em densidade, discutimos o funcionamento do classificador produzido por uma de suas variações e suas limitações específicas no contexto de detecção de anomalias. Em seguida, discutimos o funcionamento de um classificador semissupervisionado baseado nos k vizinhos, seu relacionamento com o paradigma de densidade e seu comportamento em relação à tarefa de identificação de anomalias.

2.1 Arcabouço Unificado e HDBSCAN*(cd,-)

O trabalho de Gertrudes et al. (2019) apresenta contribuições importantes na área de agrupamento e classificação baseados em densidade. Os autores demonstram que há uma relação próxima entre os paradigmas de agrupamento semissupervisionado por densidade e classificação semissupervisionada por densidade. Além disso, Gertrudes et al. (2019) observam que conhecidos algoritmos de agrupamento por densidade, como DBSCAN (ESTER et al., 1996), OPTICS (ANKERST et al., 1999), HISSCLU (BÖHM; PLANT, 2008), SSDBSCAN (LELIS; SANDER, 2009) e HDBSCAN* (CAMPELLO; MOULAVI; SANDER, 2013), possuem diversas semelhanças entre si, como, por exemplo, o uso de um parâmetro de entrada fornecido pelo usuário, denominado m_{Pts}, para estimar a densidade dos objetos e a representação da estrutura de densidade dos dados através de um grafo. Com base nessas semelhanças, Gertrudes et al. (2019) propõem um arcabouço unificado para classificação semissupervisionada baseada em densidade por meio da generalização do algoritmo HDBSCAN*. A seguir, apresentamos os conceitos fundamentais do para-

digma de agrupamento por densidade adotados pelo HDBSCAN* e, consequentemente, também pelo arcabouço unificado.

A densidade de uma região no espaço pode ser definida como a quantidade de objetos existentes nesta região dividida pelo volume da região no espaço. De forma análoga, a densidade ao redor de um objeto no espaço pode ser expressada através da quantidade de objetos (ou vizinhos) ao redor deste objeto, considerando um raio ε como limite superior. A figura 1 exemplifica este conceito, onde para o cálculo da densidade da região ao redor do ponto A dado um raio ε, somente os pontos B, C, D, E, e F vizinhos de A (e A incluso) contribuem para o cálculo da densidade de A.

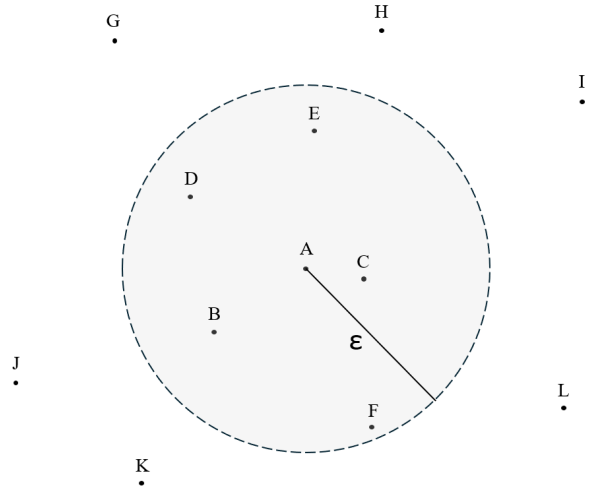


Figura 1 – Definição de densidade do ponto A em função de seus vizinhos e de um raio ε.

Fundamental para o paradigma de agrupamento por densidade é a diferenciação entre o que são regiões densas (agrupamentos) e o que são regiões esparsas (ruído). A abordagem do HDBSCAN* (assim como o DBSCAN, OPTICS, SSDBSCAN, e HISSCLU) é a utilização de um parâmetro de entrada fornecida pelo usuário, denominado m_{pts}, para determinar o que é considerado uma região densa e o que é considerado ruído. O parâmetro m_{pts} define o limite mínimo de objetos vizinhos que devem existir ao redor de um objeto, distantes no máximo ε deste objeto, para que a região ao redor deste objeto seja considerada densa. As figuras 2(a), 2(b) e 2(c) ilustram este comportamento para $m_{pts} = 4$ e valores diferentes de ε. Nas figuras 2(a) e 2(b), a vizinhança do objeto A é considerada esparsa, enquanto na figura 2(c) a vizinhança do objeto A é considerada densa. Quando a vizinhança de um objeto é considerada densa (de acordo com m_{pts} e ε), o objeto é chamado de objeto núcleo (*core object*), e quando a vizinhança é considerada esparsa, o objeto é chamado de ruído. Estes conceitos serão definidos formalmente posteriormente nesta mesma seção. O algoritmo HDBSCAN* funciona fixando o parâmetro m_{pts} e variando ε, capturando iterativamente as estruturas dos objetos de forma hierárquica em cada nível diferente de ε. Desta forma, o HDBSCAN* elimina a necessidade do parâmetro de entrada ε (necessário para outros algoritmos de agrupamento por densidade, como o

DBSCAN, por exemplo), já que são capturadas hierarquias para todos os possíveis níveis de $\varepsilon \in [0, \infty)$ para um dado conjunto de dados. O algoritmo HDBSCAN* será discutido em maior detalhe na Seção 3.1.

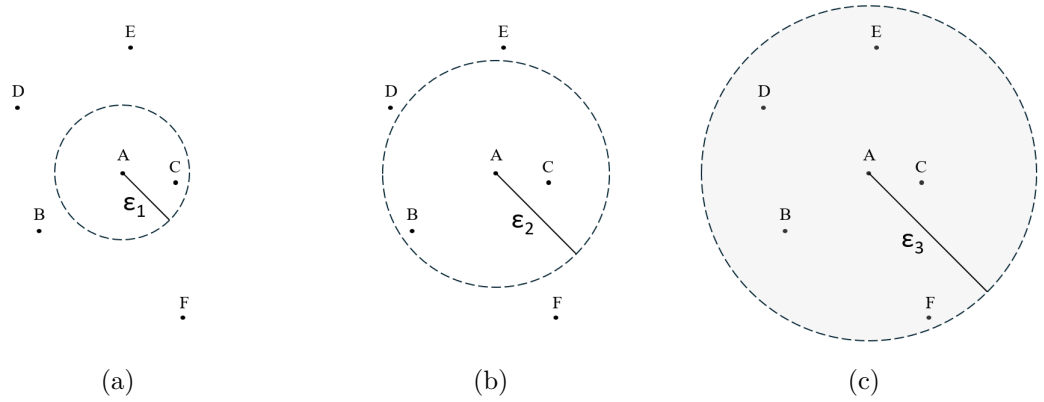


Figura 2 – Classificação de um objeto A conforme m_{pts} e ε. (a) $m_{pts} = 4$, $\varepsilon = \varepsilon_1$. O objeto A é ruído. (b) $m_{pts} = 4$, $\varepsilon = \varepsilon_2$. O objeto A é ruído. (c) $m_{pts} = 4$, $\varepsilon = \varepsilon_3$. O objeto A é um objeto núcleo (*core object*).

Com os conceitos fundamentais do paradigma de densidade apresentados em mente, dado um conjunto de dados $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n\}$ com n objetos, podemos formalizar as seguintes definições-chave, extraídas dos trabalhos de Campello, Moulavi e Sander (2013), Campello et al. (2015):

Definição 1: Objeto núcleo: Um objeto $\mathbf{x}_p$ é denominado objeto *núcleo* em relação a ε e m_{pts} se sua vizinhança ε contém pelo menos m_{pts} objetos. Ou seja, se $|N_\varepsilon(\mathbf{x}_p)| \geq m_{pts}$, onde $N_\varepsilon(\mathbf{x}_p) = \{\mathbf{x} \in \mathbf{X} \mid d(\mathbf{x}, \mathbf{x}_p) \leq \varepsilon\}$ e $|\cdot|$ denotam cardinalidade. Objetos que não satisfazem as condições citadas são considerados ruído.

Definição 2: ε-alcançável: Dois objetos núcleo $\mathbf{x}_p$ e $\mathbf{x}_q$ são ε-alcançáveis com relação a ε e m_{pts} se $\mathbf{x}_p \in N_\varepsilon(\mathbf{x}_q)$ e $\mathbf{x}_q \in N_\varepsilon(\mathbf{x}_p)$.

Definição 3: Conectados por densidade: Dois objetos núcleo $\mathbf{x}_p$ e $\mathbf{x}_q$ são conectados por densidade com relação a ε e m_{pts} se são direta ou transitivamente ε-alcançáveis entre si.

Definição 4: Grupo: Um grupo $\mathbf{C}$ com relação a ε e m_{pts} é um subconjunto não vazio e maximal de $\mathbf{X}$ tal que todos os pares de objetos pertencentes a $\mathbf{C}$ estão conectados por densidade.

Definição 5: Distância núcleo: A distância núcleo $d_{core}(\mathbf{x}_p)$ de um objeto $\mathbf{x}_p \in \mathbf{X}$ com relação a m_{pts} é a distância de $\mathbf{x}_p$ ao seu m_{pts}-ésimo vizinho mais próximo, considerando o próprio $\mathbf{x}_p$ como um dos vizinhos.

Definição 6: Distância de Alcançabilidade Mútua (DAM): A distância de alcançabilidade mútua entre dois objetos $\{\mathbf{x}_p, \mathbf{x}_q\} \in \mathbf{X}$ com respeito a m_{pts} é definida por $d_{mreach}(\mathbf{x}_p, \mathbf{x}_q) = \max\{d_{core}(\mathbf{x}_p), d_{core}(\mathbf{x}_q), d(\mathbf{x}_p, \mathbf{x}_q)\}$.

Definição 7: Grafo de Alcançabilidade Mútua (GAM): É um grafo completo $G_{m_{pts}}$ cujos vértices são os objetos de $\mathbf{X}$ e cujos pesos das arestas correspondem às distâncias de alcançabilidade mútua (com relacão a m_{pts}) entre cada par de objetos.

Definição 8: Árvore Geradora Mínima (AGM): A árvore geradora mínima de um grafo não direcionado é um subgrafo conectado que abrange todos os vértices, não possui ciclos e que possui a menor soma possível de pesos, considerando todas as arestas.

A versão estendida do HDBSCAN* publicada por Campello et al. (2015) é capaz de realizar agrupamento semissupervisionado por meio de restrições fracas *should-link* e *should-not-link* para orientar o agrupamento. Em contraste, o arcabouço proposto por Gertrudes et al. (2019) inclui modificações no algoritmo original que permitem que o HDBSCAN* comece a realizar tarefas de classificação semissupervisionada diretamente a partir dos rótulos conhecidos. Os quatro componentes principais do arcabouço são:

1. A definição de distância núcleo e de distância de alcançabilidade adotadas;
2. Uma etapa que calcula a árvore geradora mínima no espaço transformado das distâncias de alcançabilidade mútua;
3. Uma etapa de expansão de rótulos;
4. Uma etapa opcional de pré-processamento para aplicar pesos às distâncias com base nos rótulos conhecidos.

Os componentes citados podem ser vistos como blocos de construção independentes que compõem um processo de classificação semissupervisionada, tendo o HDBSCAN* como ponto central. As configurações individuais de cada componente, quando utilizadas em conjunto, produzem variações nas características e no comportamento do HDBSCAN*, resultando em saídas distintas. Relevante para nosso trabalho é a variação mais simples do arcabouço, denominada HDBSCAN*(cd,-) pelos autores. Esta variação adota as definições de distância núcleo (Definição 5) e distância de alcançabilidade simétrica, ou mútua (Definição 6) conforme Campello, Moulavi e Sander (2013), Campello et al. (2015), e não faz uso do componente opcional 4 (aplicação de pesos às distâncias).

O HDBSCAN*(cd,-) tem como primeiro passo o cálculo de uma árvore geradora mínima (Definição 8) no espaço transformado das distâncias de alcançabilidade mútua, a partir do grafo completo $G_{m_{pts}}$ (Definição 7). Em seguida, um processo de transdução é realizado através da AGM por meio de uma etapa denominada propagação de rótulos, conforme a seguinte definição:

Definição 9: Propagação de rótulos do HDBSCAN*(cd,-): Seja $\mathbf{X}_U$ o conjunto de todos os objetos não rotulados e $\mathbf{X}_L$ o conjunto de todos os objetos rotulados. Com base na árvore geradora mínima do espaço transformado de distâncias de alcançabilidade mútua,

para cada caminho entre um objeto $\mathbf{x}_i \in \mathbf{X}_U$ e um objeto $\mathbf{x}_j \in \mathbf{X}_L$, encontra-se a maior aresta desse caminho e define-se classe($\mathbf{x}_i$) = classe($\mathbf{x}_j$), onde $\mathbf{x}_j$ é o objeto rotulado cuja maior aresta no caminho até $\mathbf{x}_i$ é a menor entre todas as maiores arestas dos caminhos entre os outros objetos rotulados e $\mathbf{x}_i$.

Intuitivamente, os rótulos são propagados de objetos rotulados para objetos não rotulados com base na densidade: o objeto rotulado que se encontra no caminho mais denso até um objeto não rotulado acaba por definir seu rótulo. A Figura 3 ilustra o funcionamento do processo: As cores cinza claro, cinza escuro, e preto representam objetos rotulados de diferentes classes e a cor branca representa objetos com ausência de rótulos. A Figura 3(a) exibe um grafo representando a AGM de um dado conjunto de dados em sua situação inicial, com três objetos previamente rotulados $\mathbf{x}_j, \mathbf{x}_p, \mathbf{x}_q$. Considerando que deseja-se rotular o objeto $\mathbf{x}_i$, o algoritmo percorre a AGM e identifica a maior aresta entre $\mathbf{x}_i$ e $\mathbf{x}_j$, entre $\mathbf{x}_i$ e $\mathbf{x}_p$ e entre $\mathbf{x}_i$ e $\mathbf{x}_q$. Em seguida, o processo de propagação do HDBSCAN*(cd,-) atribui a $\mathbf{x}_i$ o mesmo rótulo que o objeto rotulado $\mathbf{x}_p$, pois a menor aresta entre todas as maiores arestas é a que liga $\mathbf{x}_i$ a $\mathbf{x}_p$ (Figura 3(a)). Em caso de empates, estes são resolvidos escolhendo a menor entre as segundas maiores arestas entre os pares de objetos rotulados e não rotulados, e assim por diante. A partir do momento em que todos os objetos não rotulados receberem um rótulo por meio do processo descrito pela Definição 9, é produzida uma partição plana como a saída final do HDBSCAN(cd,-). É importante realçar que o método de expansão de rótulos proposto por Gertrudes et al. (2019) atribui um rótulo a todos os objetos $\mathbf{x}_i \in \mathbf{X}_U$, o que implica que o método não é nativamente capaz de identificar objetos como ruído (ou anomalias). Dada esta característica, neste trabalho investigamos a combinação do HDBSCAN*(cd,-) com um método não supervisionado de detecção de anomalias baseado em densidade, com o objetivo de expandir suas capacidades de classificação para também incluir a detecção automática de anomalias.

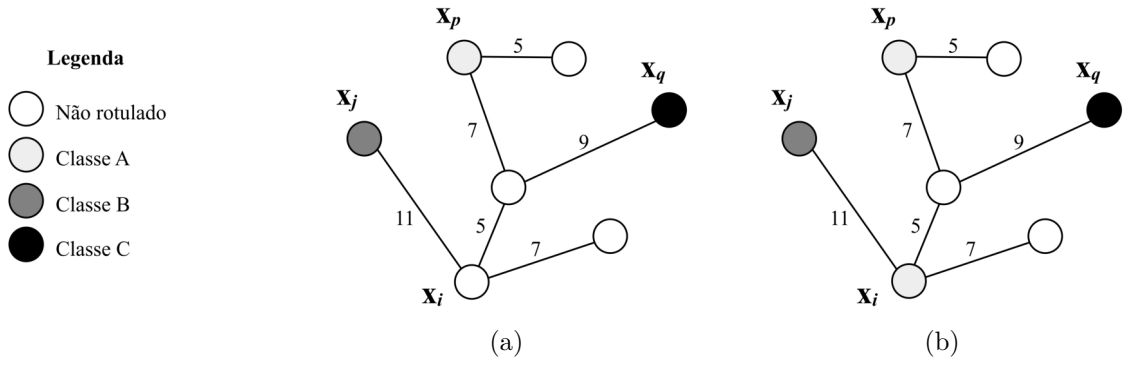


Figura 3 – Exemplo do processo de propagação do HDBSCAN*(cd,-).

2.2 *k*NN_LDP

*k*NN_LDP (GøTTCKE; ZIMEK; CAMPELLO, 2025) é um algoritmo da família dos *k* vizinhos mais próximos, capaz de realizar transdução e indução com base em um subconjunto de objetos previamente rotulados. O principal aspecto que diferencia o *k*NN_LDP de outros classificadores *k*NN reside em sua etapa de propagação de rótulos: o algoritmo utiliza uma abordagem probabilística baseada no teorema de Bayes para propagar não os rótulos em si, mas as distribuições de probabilidade dos rótulos (incluindo uma classe especial "desconhecido"), estimadas considerando os *k* vizinhos mais próximos de cada objeto. Outro aspecto do *k*NN_LDP relevante para nosso trabalho é o conceito de abstenção de classificação (PIETRASZEK, 2007) adotado pelo algoritmo: o *k*NN_LDP se abstém de rotular observações cujas distribuições de probabilidade são desconhecidas.

O *k*NN_LDP modela a estrutura de densidade dos dados de entrada por meio de um grafo especial denominado grafo *k*NN (ou *k*NNG), definido por Dong, Moses e Li (2011) da seguinte forma:

Definição 10: *k Nearest Neighbors Graph* (*k*NNG): O grafo dos *k* vizinhos mais próximos para um conjunto de objetos $\mathbf{X}$ é um grafo direcionado com conjunto de vértices $\mathbf{X}$ e arestas partindo de cada $\mathbf{x} \in \mathbf{X}$ e chegando em seus *k* objetos em $\mathbf{X}$ mais similares, dada uma medida de similaridade qualquer.

De acordo com a publicação de Gøttcke, Zimek e Campello (2025), a medida de similaridade usada para o *k*NNG no *k*NN_LDP é, na realidade, uma medida de dissimilaridade, pois o algoritmo utiliza, por padrão, a distância euclidiana. O primeiro passo do *k*NN_LDP consiste então do cálculo do *k*NNG a partir dos dados de entrada. Como o grafo será consultado extensivamente nas etapas subsequentes, Gøttcke, Zimek e Campello (2025) sugerem indexar os *k* vizinhos mais próximos e os *k* vizinhos reversos - *Reverse k Nearest Neighbors* (R*k*NN) de cada instância para melhorar o desempenho. Uma vez computado o *k*NNG e o grafo dos *k* vizinhos reversos - *Reverse k Nearest Neighbors Graph* (R*k*NNG), o *k*NN_LDP dá início ao seu processo de transdução, que funciona da seguinte forma: inicialmente todos os objetos não rotulados são percorridos através do *k*NNG uma vez pelo algoritmo, e a cada objeto são atribuídas probabilidades de distribuição das classes (ou rótulos) possíveis, conforme a Equação 1:

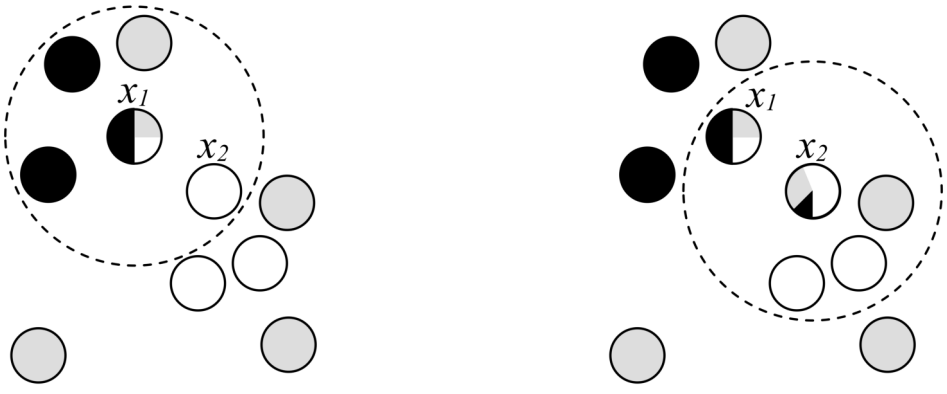
$$\Pr(c \mid \mathbf{x}) = \frac{\sum_{\mathbf{x}_\ell \in k\mathrm{NN}(\mathbf{x})} \Pr(c \mid \mathbf{x}_\ell)}{|k\mathrm{NN}(\mathbf{x})|} \tag{1}$$

onde $\mathbf{c}$ representa uma dada classe e $\mathbf{x}_\ell$ representa a probabilidade de distribuição de classes do objeto $\mathbf{x}$. As probabilidades de distribuição de classes são calculadas com base nas classes (ou probabilidades de distribuição de classes para objetos que já receberam probabilidades) de objetos pertencentes à vizinhança definida pelos *k* vizinhos. Neste contexto, cada objeto vizinho contribui igualmente e de forma proporcional conforme *k* para

o cálculo de probabilidades de distribuição de classes de uma dada observação. Quando um vizinho não contribui com nenhuma informação sobre a distribuição de classes reais, a classe especial "desconhecido" é utilizada para complementar o cálculo das probabilidades de distribuição, de modo que para todos os objetos a somatória total de probabilidades de classes reais e da classe "desconhecido" é sempre 1. Uma vez que todos os objetos não rotulados tenham sido processados, e conseqüentemente, recebido probabilidades de distribuição de classes, é criada uma fila de prioridade em que as observações $\mathbf{x} \in \mathbf{X_U}$ são ordenadas de forma decrescente por um peso calculado através da somatória das probabilidades de distribuição de classes reais (desconsiderando a classe "desconhecido"), definido pela Equação 2:

$$w(\mathbf{x}) = \sum_{c \in \mathbf{C} \setminus \{\text{“}desconhecido''\}} Pr(c|\mathbf{x}) \tag{2}$$

onde $\mathbf{C}$ é o conjunto de todas as classes. A fila de prioridade é então processada, extraindo uma a uma as observações de maior peso. Para cada objeto extraído, as probabilidades de distribuição de rótulos dos vizinhos reversos do objeto são atualizadas juntamente com os pesos na fila de prioridade, possivelmente provocando uma reordenação da fila. O procedimento de propagação de probabilidades descrito é ilustrado na Figura 4, considerando $k = 4$. Círculos de cor branca representam objetos não rotulados, enquanto círculos de cores preta e cinza representam rótulos distintos. Conforme a Figura 4(a), inicialmente, o objeto não rotulado $\mathbf{x}_1$ recebe as probabilidades de distribuição de classes de seus quatro vizinhos mais próximos. A Figura 4(b) exibe as probabilidades de distribuição das classes de $\mathbf{x}_2$ após sua extração da fila de prioridades. Como $\mathbf{x}_1$ faz parte da vizinhança de $\mathbf{x}_2$, as suas probabilidades de distribuição de classes, calculadas em passos anteriores, influenciam a de $\mathbf{x}_2$.



(a) (b)

Figura 4 – Processo de propagação das probabilidades de distribuição de rótulos do kNN_LDP para $k = 4$.

Após o processamento completo da fila, todas as observações cujos k vizinhos mais

próximos possuírem alguma informação de probabilidade de distribuição de rótulos não nula terão recebido, por sua vez, uma probabilidade de distribuição de rótulos não nula proporcional. A conversão das probabilidades em rótulos bem definidos pode, então, ser realizada de forma trivial com base na maior probabilidade entre as classes, desconsiderando a classe "desconhecido". Em caso de empates, Gøttcke, Zimek e Campello (2025) propõem uma possível estratégia de desempate que prioriza classes específicas, embora a implementação pública disponibilizada pelos autores escolha aleatoriamente uma das classes empatadas. Dependendo do conjunto de dados e do valor de k escolhido, alguns objetos podem não ter recebido nenhuma probabilidade de distribuição de rótulos durante o processo de propagação, o que corresponde a $Pr(\text{"desconhecido"} \mid \mathbf{x}) = 1$. O kNN_LDP se abstém de rotular estes objetos, produzindo como sua saída final uma partição plana com rótulos bem definidos atribuídos aos objetos.

Gøttcke, Zimek e Campello (2025) sugerem a generalização do kNN_LDP na forma de um classificador baseado em k vizinhos, cujos componentes podem ser modificados ou substituídos para produzir comportamentos distintos. Os autores demonstram que o HDBSCAN*(cd,-) pode ser obtido como um caso especial desta generalização, mediante as seguintes customizações:

1. Substituição da distância euclidiana pela distância de alcançabilidade mútua;

2. Substituição do esquema de ponderação da Equação 2 pelo cálculo da árvore geradora mínima com base nas distâncias de alcançabilidade mútua;

3. Definição de $k = 1$ para que as probabilidades do kNN_LDP sejam ponderadas por apenas um vizinho, assim transformando-as em rótulos bem definidos.

Desta forma, Gøttcke, Zimek e Campello (2025) estabelecem que, em um contexto mais amplo, métodos de propagação de rótulos baseados em densidade podem ser expressos como variantes de classificadores de k vizinhos mais próximos. Esta perspectiva nos permite considerar as observações não rotuladas pelo kNN_LDP como anomalias em potencial sob a ótica da densidade, conceito central para este trabalho. Gøttcke, Zimek e Campello (2025) ressaltam algumas limitações do kNN_LDP relacionadas à tarefa de classificação, que também podem afetar a identificação de anomalias quando instâncias não rotuladas são interpretadas como tais. Especificamente, em regiões do espaço isoladas contendo alguns objetos não rotulados, mas sem objetos rotulados próximos de acordo com o valor de k escolhido, pode ocorrer que todos os objetos que compõem essa região acabem ficando sem um rótulo atribuído pelo kNN_LDP. Esta característica do kNN_LDP pode ser vantajosa para identificar anomalias bem separadas dos componentes principais de densidade no espaço. Seguindo a mesma lógica, quando as anomalias são locais ou não estão bem separadas de componentes mais densos, é razoável supor que

sejam mais difíceis de identificar pelo kNN_LDP, dada a natureza do algoritmo. Portanto, investigamos neste trabalho: a) se objetos sem um rótulo atribuído pelo kNN_LDP são candidatos adequados a anomalias globais e locais do ponto de vista da densidade; e b) se a combinação do kNN_LDP com um arcabouço baseado em densidade, capaz de detectar anomalias, pode produzir melhores resultados do que o kNN_LDP original em relação à tarefa de detecção de anomalias. O objetivo mais amplo é o mesmo apresentado na Seção 2.1: o desenvolvimento de um método híbrido baseado em densidade capaz de realizar classificação semissupervisionada, ao mesmo tempo em que reconhece anomalias nos dados.

Capítulo 3

Métodos Propostos

Neste capítulo, começamos com uma breve análise do método baseado em densidade capaz de reconhecer observações anômalas, que escolhemos para integrar com os dois classificadores HDBSCAN*(cd,-) e kNN_LDP discutidos no Capítulo 2. Em seguida, introduzimos os dois métodos híbridos que propomos como fruto desta integração e apresentamos como os métodos individuais de classificação e detecção de anomalias são combinados de acordo com nossa proposta. Por fim, apresentamos uma análise da complexidade dos dois métodos propostos.

3.1 HDBSCAN* e FOSC

O HDBSCAN* proposto por Campello, Moulavi e Sander (2013) é um algoritmo hierárquico de agrupamento baseado em densidade considerado estado da arte. O algoritmo pode ser dividido em duas etapas distintas. A primeira etapa consiste no cálculo de uma árvore geradora mínima a partir do grafo completo de alcançabilidade mútua $G_{m_{pts}}$, conforme as Definições 8 e 7 da Seção 2.1, respectivamente. O cálculo da AGM depende das distâncias núcleo (Definição 5), que precisam ser computadas antes por meio de um kNNG. Para obtenção do kNNG, o HDBSCAN* pode fazer uso de uma matriz de distâncias entre os objetos fornecida como entrada, ou pode computar as distâncias via uma medida de dissimilaridade qualquer caso seja fornecida uma lista de objetos e atributos como entrada. No segundo caso a medida de dissimilaridade padrão utilizada pelo HDBSCAN* para computação do kNNG é a distância euclidiana. A segunda etapa do algoritmo consiste do processamento da árvore geradora mínima calculada na primeira etapa. Nesta etapa as arestas da árvore geradora mínima são removidas uma a uma, em ordem decrescente em relação aos pesos das arestas. À medida que cada aresta é removida, o nível de

densidade associado ao objeto desconectado é registrado. Ao final do processo, é produzida uma árvore hierárquica de grupos contendo todos os níveis de densidade de todos os objetos de forma aninhada, chamada de "hierarquia HDBSCAN*". Esta é a principal saída do algoritmo e pode ser visualizada (por meio de um dendrograma, por exemplo) ou processada externamente para extrair informações adicionais.

Em seu trabalho, Campello, Moulavi e Sander (2013) também propõem uma etapa opcional de pós-processamento denominada *Framework for Optimal Selection of Clusters* (FOSC). Em resumo, o processo FOSC consiste na extração de uma partição plana sem sobreposições entre grupos a partir da hierarquia HDBSCAN*, o que pode ser desejável em muitas situações reais. O FOSC parte da seguinte definição, chamada de estabilidade de um grupo:

$$S(\mathbf{C}_i) = \sum_{\mathbf{x}_j \in \mathbf{C}_i} (\lambda_{\max}(\mathbf{x}_j, \mathbf{C}_i) - \lambda_{\min}(\mathbf{C}_i)) \tag{3}$$

onde $\mathbf{C}_i$, $i \in \{1, 2, ..., \kappa\}$ representam os grupos de objetos da hierarquia HDBSCAN*, $\lambda = \frac{1}{\varepsilon}$ representa um dado nível de densidade da hierarquia, $\lambda_{\min}(\mathbf{C}_i) = \frac{1}{\varepsilon_{\max}(\mathbf{C}_i)}$ é o nível mínimo de densidade no qual $\mathbf{C}_i$ existe, e $\lambda_{\max}(\mathbf{x}_j, \mathbf{C}_i) = \frac{1}{\varepsilon_{\min}(\mathbf{x}_j, \mathbf{C}_i)}$ é o nível máximo de densidade no qual $\mathbf{x}_j$ ainda pertence a $\mathbf{C}_i$. De acordo com esta definição, a estabilidade de um grupo pode ser interpretada como a soma dos tempos de vida de seus objetos ao longo do espectro de densidade da hierarquia: quanto maior a faixa de densidade em que um determinado grupo de objetos existe antes de se dissolver completamente em ruído, maior é sua estabilidade. Neste contexto, a obtenção de uma partição ótima do ponto de vista da estabilidade significa maximizar a soma total das estabilidades de todos os grupos extraídos da hierarquia. Este processo é tratado pelo FOSC como o seguinte problema de otimização:

$$\begin{aligned} &\max_{\delta_2,\ldots,\delta_\kappa} \quad J = \sum_{i=2}^{\kappa} \delta_i \, S(\mathbf{C}_i) \\ &\text{Sujeito a} \begin{cases} \delta_i \in \{0, 1\}, i = 2, ..., \kappa \\ \text{exatamente um } \delta_{(\cdot)} = 1 \text{ em cada caminho de grupo folha até a raíz} \end{cases} \end{aligned} \tag{4}$$

onde δ_i indica se o grupo $\mathbf{C}_i$ pertence ($\delta_i = 1$) ou não pertence ($\delta_i = 0$) à solução final, e as restrições sobre $\delta_2, \delta_3, \ldots, \delta_\kappa$ garantem que grupos na mesma ramificação da árvore não sejam selecionados simultaneamente. Para resolver a Equação 4, o algoritmo FOSC parte dos grupos folhas e percorre a hierarquia de baixo para cima, processando cada grupo, exceto o nó raiz. Para cada grupo $\mathbf{C}_i$, é tomada uma decisão local: incluí-lo (se sua estabilidade for maior que a soma das estabilidades de seus grupos filhos) ou incluir seus grupos filhos em seu lugar (se a soma das estabilidades dos grupos filhos for maior). Desta forma, os cálculos de estabilidade são realizados recursivamente, seguindo as regras:

$$\hat{S}(\mathbf{C}_i) = \begin{cases} S(\mathbf{C}_i), & \text{se } \mathbf{C}_i \text{ é um nó folha,} \\ \max\{S(\mathbf{C}_i), \hat{S}(\mathbf{C}_{i_l}) + \hat{S}(\mathbf{C}_{i_r})\}, & \text{se } \mathbf{C}_i \text{ é um nó interno.} \end{cases} \quad (5)$$

onde $\mathbf{C}_{i_l}$ e $\mathbf{C}_{i_r}$ são os grupos filho esquerdo e direito de $\mathbf{C}_i$, respectivamente. A Equação 5 considera uma estrutura de árvore binária para fins de clareza. Na prática, é possível que existam mais de dois grupos filhos, e a fórmula é aplicada dinamicamente de acordo com o número real de grupos filhos. A Figura 5 ilustra o funcionamento do FOSC. Os nós representam os grupos $\mathbf{C}_2$ a $\mathbf{C}_{23}$, e os valores de estabilidade calculados para cada grupo são exibidos dentro dos nós. Nós de cor branca representam aqueles escolhidos pelo FOSC para compor a solução final. Partindo dos nós folha, $\mathbf{C}_{14}$ e $\mathbf{C}_{17}$ são inicialmente selecionados porque, de acordo com a Equação 5, $S(\mathbf{C}_{14}) > \hat{S}(\mathbf{C}_{19}) + \hat{S}(\mathbf{C}_{20}) + \hat{S}(\mathbf{C}_{21})$ e $S(\mathbf{C}_{17}) > \hat{S}(\mathbf{C}_{22}) + \hat{S}(\mathbf{C}_{23})$. Em seguida, $\mathbf{C}_{14}$, $\mathbf{C}_{15}$, e $\mathbf{C}_{16}$ são selecionados porque sua estabilidade é maior do que a estabilidade de $\mathbf{C}_8$; $\mathbf{C}_{17}$ e $\mathbf{C}_{18}$ são selecionados porque a soma de suas estabilidades é maior do que a estabilidade de $\mathbf{C}_{12}$. Por fim, $\mathbf{C}_2$ é selecionado pois sua estabilidade é maior que a soma das estabilidades de $\mathbf{C}_5$ e $\mathbf{C}_6$, enquanto $\mathbf{C}_3$ e $\mathbf{C}_4$ são descartados pelo mesmo motivo, produzindo a solução final $\{\mathbf{C}_2, \mathbf{C}_9, \mathbf{C}_{10}, \mathbf{C}_{11}, \mathbf{C}_{13}, \mathbf{C}_{14}, \mathbf{C}_{15}, \mathbf{C}_{16}, \mathbf{C}_{17}, \mathbf{C}_{18}\}$, cuja soma total da estabilidade é máxima.

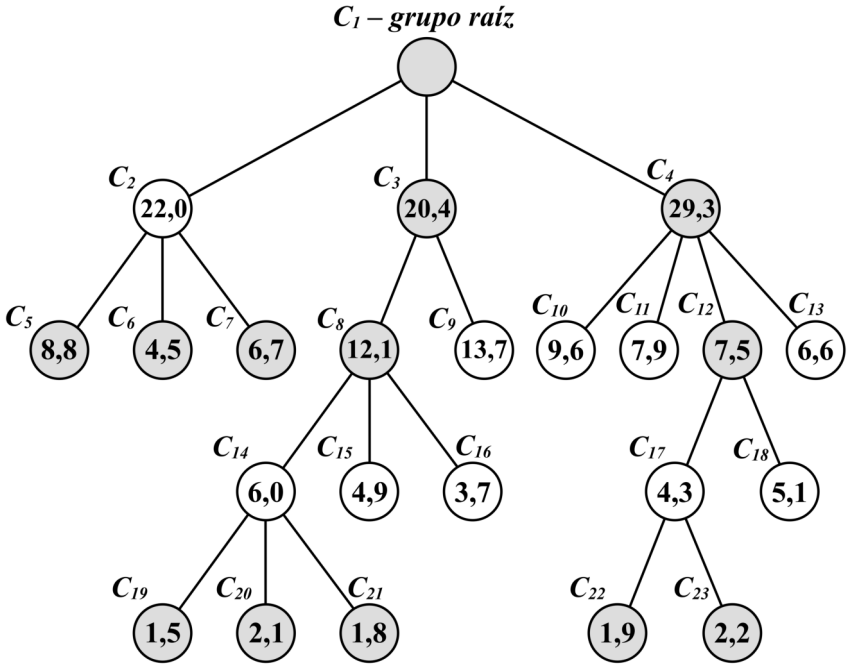


Figura 5 – Aplicação do FOSC à hierarquia HDBSCAN*.

Após o processo do FOSC percorrer completamente a árvore hierárquica, pode ocorrer que alguns objetos não pertençam a nenhum dos grupos escolhidos para compor a solução final. De acordo com Campello, Moulavi e Sander (2013), estes objetos são considerados ruído do ponto de vista da densidade e, em conformidade com Breunig et al. (2000), nós os interpretamos como anomalias neste trabalho.

O FOSC aceita um parâmetro opcional denominado m_{clSize}, que estabelece a quantidade mínima de objetos que definem um grupo durante o processamento da hierarquia HDBSCAN*. Segundo Campello et al. (2015), trata-se de um parâmetro adicional de suavização, além de m_{pts}, que tem o efeito de simplificar a hierarquia HDBSCAN* e, consequentemente, afeta o resultado do agrupamento produzido pelo FOSC. Se um grupo da hierarquia for dividido em menos objetos que m_{clSize} durante uma determinada etapa, esses objetos são então marcados como ruído (ou anomalias). Como m_{pts} já estabelece um limite inferior para o número de vizinhos que definem um objeto núcleo (e, consequentemente, o número de vizinhos que definem ruído), Campello et al. (2015) sugerem $m_{clSize} = m_{pts}$, tornando assim o FOSC um processo essencialmente não paramétrico.

A capacidade do FOSC de identificar anomalias baseadas em densidade de forma não supervisionada serviu como inspiração para nossa investigação sobre a adição de capacidades de detecção de anomalias ao HDBSCAN*(cd,-) e kNN_LDP, justificada pelos seguintes argumentos: a) o FOSC é não paramétrico, dependendo indiretamente apenas de m_{pts}, o mesmo parâmetro utilizado pelo HDBSCAN*(cd,-); como o parâmetro k também pode ser visto como equivalente a m_{pts} sob a perspectiva de densidade (GøTTCKE; ZIMEK; CAMPELLO, 2025), o FOSC pode ser combinado com HDBSCAN*(cd,-) e kNN_LDP sem adicionar novas configurações ou parâmetros; b) o FOSC requer inicialmente o cálculo de uma árvore geradora mínima e, em seguida, da hierarquia HDBSCAN*; o cálculo da árvore geradora mínima pode ser reutilizado a partir do HDBSCAN*(cd,-), e o grafo kNN necessário para o kNN_LDP já é computado durante a construção da AGM, o que significa que o FOSC pode ser eficientemente integrado com ambos os métodos; c) o FOSC possui baixa complexidade temporal e espacial, exigindo apenas $O(\kappa)$ para ambos, onde κ é o número de grupos da hierarquia HDBSCAN*, tipicamente muito menor que n; e d) embora existam trabalhos voltados à identificação de anomalias de forma não supervisionada por meio do *Global-Local Outlier Scores from Hierarchies* (GLOSH) (GHOSH et al., 2024), que é outra etapa de pós-processamento do HDBSCAN* proposta por Campello et al. (2015), constatamos que há uma escassez de pesquisas na literatura sobre as capacidades nativas do FOSC em detectar anomalias.

3.2 HDBSCAN*(cd,-) + Fo

O primeiro método que propomos é a combinação do HDBSCAN*(cd,–) com o HDBSCAN* e o FOSC. Nomeamos o método resultante como HDBSCAN*(cd,–) + Fo, onde Fo significa *FOSC outliers*. Desta forma, neste documento, referir-nos-emos ao método pelo seu nome abreviado. Conforme discutido nas seções 2.1 e 3.1, tanto o HDBSCAN*(cd,-) quanto o HDBSCAN* (e, consequentemente, o FOSC) dependem do cálculo de uma árvore geradora mínima no espaço transformado das distâncias de alcançabilidade mútua. Portanto, como passo inicial do método combinado, calculamos $G_{m_{pts}}$ e a árvore

geradora mínima uma única vez. As etapas seguintes do HDBSCAN*(cd,-) e HDBSCAN* com FOSC consomem a mesma árvore geradora mínima como entrada e podem ser executadas de forma independente, possivelmente em paralelo. As partições planas produzidas como resultado de ambos os métodos são então combinadas de modo que os objetos rotulados como ruído pelo FOSC tornem-se anomalias na partição final, enquanto os objetos atribuídos a grupos pelo FOSC recebem seus rótulos do HDBSCAN*(cd,–). O Algoritmo 1 detalha as etapas do método proposto. Na listagem do algoritmo, a notação $classe(\mathbf{x}, \mathbf{Y})$ é utilizada para representar a classe atribuída a $\mathbf{x}$ na partição $\mathbf{Y}$, a notação $classe(ruído)$ é utilizada para representar a classe especial utilizada pelo FOSC para marcar observações como ruído no contexto de agrupamento por densidade, e a notação $classe(anomalia)$ é utilizada para representar a classe específica que designa uma observação como anomalia no momento de produzir a partição final. As Figuras 6(a)–6(c) exibem representações em alto nível do fluxo de trabalho do HDBSCAN* com FOSC, HDBSCAN*(cd,-) e HDBSCAN*(cd,-) + Fo, respectivamente.

Algoritmo 1 HDBSCAN*(cd,-) + Fo

Entrada: Conjunto de dados $\mathbf{X}$
Saída: Partição plana $\mathbf{Y}$ com rótulos e anomalias marcadas

1. Calcular $G_{m_{pts}}$ e a árvore geradora mínima de $G_{m_{pts}}$ {etapa 1 do HDBSCAN*}
2. A partir da árvore geradora mínima obtida no passo 1:
   2.1 Calcular a hierarquia HDBSCAN* {etapa 2 do HDBSCAN*}
   2.2 Propagar rótulos e gerar a partição plana $\mathbf{Y}_A$ {etapa de propagação de rótulos do HDBSCAN*(cd,-)}
3. Extrair a partição plana $\mathbf{Y}_B$ a partir da hierarquia HDBSCAN obtida no passo 2.1 {etapa de pós-processamento FOSC}
4. Combinar $\mathbf{Y}_A$ e $\mathbf{Y}_B$, aplicando a seguinte regra para cada objeto $\mathbf{x}_i \in \mathbf{X}$:

   Se $classe(\mathbf{x}_i, \mathbf{Y}_B) \neq classe(ruído)$, então $classe(\mathbf{x}_i) = classe(\mathbf{x}_i, \mathbf{Y}_A)$;
   Senão $classe(x_i) = classe(anomalia)$.

3.3 *k*NN_LDP* + Fo

O segundo método proposto neste trabalho é a combinação do classificador kNN_LDP com o HDBSCAN* e FOSC para a detecção de anomalias. Conforme já discutido na Seção 2.2, o kNN_LDP original (GØTTCKE; ZIMEK; CAMPELLO, 2025) pode deixar de rotular objetos quando nenhum de seus k vizinhos contribui com probabilidades de distribuição de rótulos. Isto significa que não podemos aplicar diretamente a mesma lógica de combinação de partições descrita na Seção 3.2, pois há o risco de objetos reconhe-

cidos como normais pelo FOSC não receberem um rótulo do kNN_LDP. Para eliminar este risco, nossa abordagem consiste em adicionar uma etapa de pós-processamento ao kNN_LDP original que propaga rótulos para todos os objetos. Com a certeza de que todos os objetos possuirão um rótulo atribuído, o FOSC pode então ser combinado com segurança com o kNN_LDP. Nomeamos o kNN_LDP com esta etapa adicional como kNN_LDP* e utilizaremos esta notação ao longo deste documento.

A etapa de pós-processamento proposta funciona da seguinte forma: após a execução normal do kNN_LDP, é realizada uma segunda etapa de expansão de rótulos. Para cada objeto não rotulado, é realizado um processo de expansão iterativa de sua vizinhança além do parametro k original, até o ponto em que exatamente k vizinhos forneçam probabilidades de distribuição de rótulos não nulas. A vizinhança expandida garante que todos os objetos não rotulados recebam rótulos bem definidos, calculados com base nos k vizinhos mais próximos que tenham probabilidades de distribuição de rótulos não nulas. A Figura 6(e) apresenta um fluxograma simplificado do kNN_LDP*, e o Algoritmo 2 detalha as principais etapas envolvidas.

Algoritmo 2 Etapa de pós-processamento do kNN_LDP
Entrada: Conjunto de dados $\mathbf{X}$, probabilidades de distribuição de rótulos propagadas $\mathbf{P}$
Saída: Partição plana $\mathbf{Y}$ com todos os objetos rotulados

1. Para cada objeto $\mathbf{x}_i \in \mathbf{X} \mid P(\mathbf{x}_i) = \emptyset$
   1.1 Definir $k' = 2k$
   1.2 Enquanto $|\{\mathbf{x}_j \in N_{k'}(\mathbf{x}_i) \mid P(\mathbf{x}_j) \neq \emptyset\}| < k$
      1.2.1 Definir $k' = k' + 1$
   1.3 Propagar probabilidades de $\{\mathbf{x}_j \in N_{k'}(\mathbf{x}_i) \mid P(\mathbf{x}_j) \neq \emptyset\}$ para $\mathbf{x}_i$
   1.4 Propagar probabilidades de $\mathbf{x}_i$ para seus k vizinhos mais próximos e para seus k vizinhos reversos mais próximos
2. Com base nas probabilidades de distribuição de rótulos dos objetos, extrair partição plana $\mathbf{Y}$, aplicando a seguinte regra para cada objeto $\mathbf{x}_i \in \mathbf{X}$: {etapa de conversão de probabilidades em rótulos do kNN_LDP original}

   $$classe(\mathbf{x}_i) = classe(max(P(\mathbf{x}_i))) \mid classe(P(\mathbf{x}_i)) \neq classe(desconhecido)$$

Uma vez que garantimos que todos os objetos recebam um rótulo a partir do kNN_LDP*, podemos, então, integrá-lo ao HDBSCAN* e ao FOSC por meio da mesma lógica de combinação anterior: objetos marcados como ruído pelo FOSC tornam-se anomalias, e os demais objetos herdam os rótulos do kNN_LDP*. A integração eficiente do kNN_LDP* com o HDBSCAN* e FOSC se dá por meio da reutilização do kNNG gerado pelo HDBSCAN* durante o cálculo das distâncias núcleo (conforme detalhado na seção 3.1) como entrada para o kNN_LDP*. Desta forma, o cálculo do RkNNG e consultas ao kNNG

realizadas pelo kNN_LDP* fazem uso do mesmo grafo computado pelo HDBSCAN* uma única vez. Ao combinar kNN_LDP* com o HDBSCAN* e FOSC, consideramos os mesmos valores para k e m_{pts}, de modo que o mesmo tamanho de vizinhança seja utilizado nas estimativas de densidade de ambos os algoritmos, seguindo o trabalho de Gøttcke, Zimek e Campello (2025). Nomeamos o método híbrido resultante kNN_LDP* + Fo, e, neste trabalho, o método será referido por este nome. A Figura 6(f) ilustra o macro fluxo de execução do kNN_LDP* + Fo, e o Algoritmo 3 apresenta as etapas do método final em maior detalhe.

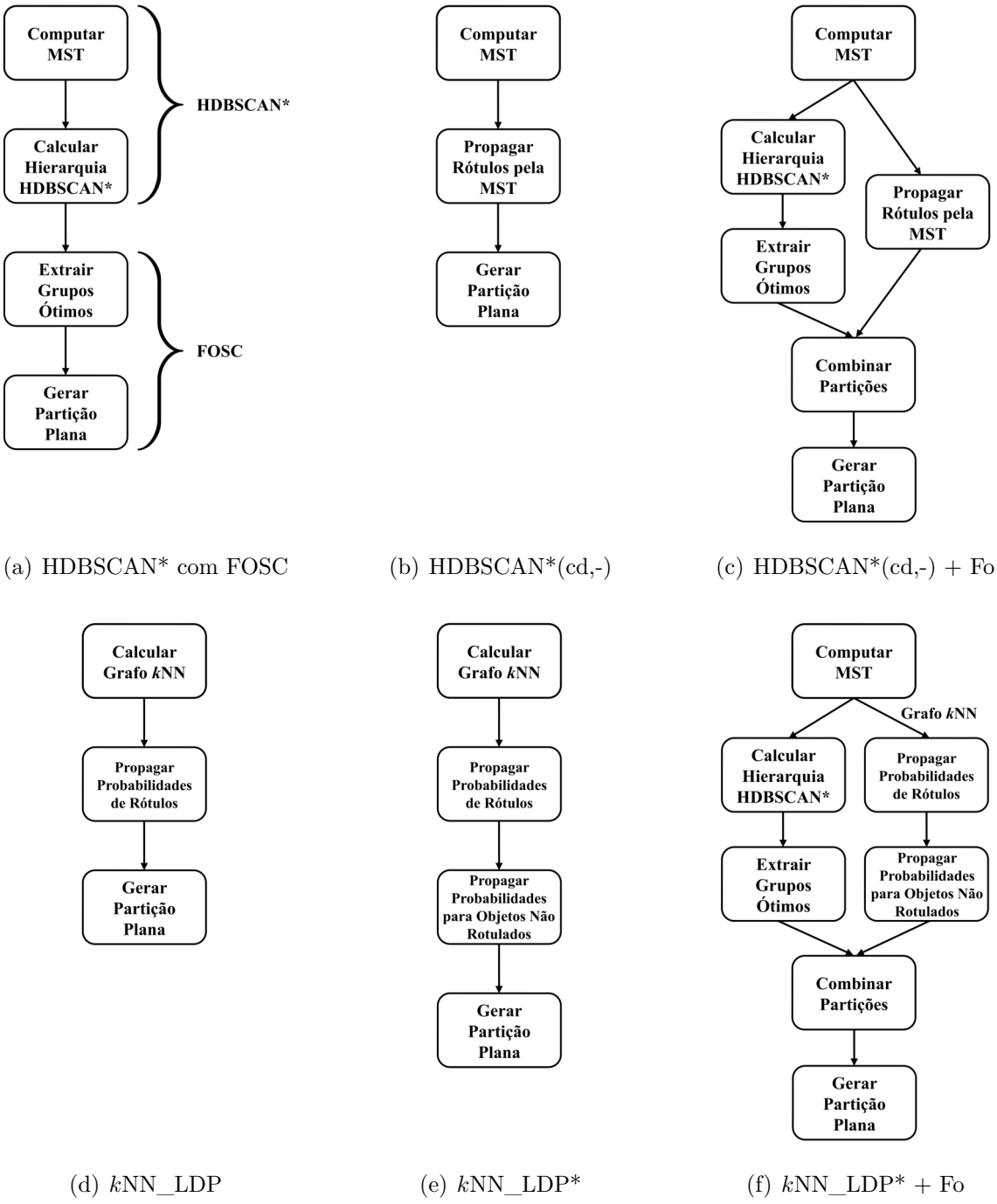
Algoritmo 3 kNN_LDP* + Fo
Entrada: Conjunto de dados $\mathbf{X}$
Saída: Partição plana $\mathbf{Y}$ com rótulos e anomalias marcadas

1. Calcular kNNG($\mathbf{X}$) e a árvore geradora mínima de $\mathbf{X}$ no espaço transformado das distâncias de alcançabilidade mútua {etapa 1 do HDBSCAN*}
2. A partir do kNNG e da árvore geradora mínima obtidos no passo 1:
   2.1. Calcular a hierarquia HDBSCAN* {etapa 2 do HDBSCAN*}
   2.2. Propagar probabilidades de distribuição de rótulos para todos os $\mathbf{x}_i \in \mathbf{X}_U$ {versão original do kNN_LDP}
3. A partir da hierarquia HDBSCAN* e das probabilidades de distribuição de rótulos calculadas no passo 2:
   3.1. Extrair a partição plana $\mathbf{Y}_A$ a partir da hierarquia obtida no passo 2.1 {etapa de pós-processamento FOSC}
   3.2. Propagar rótulos para todos os $\mathbf{x}_i$ ainda não rotulados, extraindo a partição $\mathbf{Y}_B$ {Algoritmo 2}
4. Combinar $\mathbf{Y}_A$ e $\mathbf{Y}_B$, aplicando a seguinte regra para cada objeto $\mathbf{x}_i \in \mathbf{X}$:

   Se $classe(\mathbf{x}_i, \mathbf{Y}_A) \neq classe(ruído)$, então $classe(\mathbf{x}_i) = classe(\mathbf{x}_i, \mathbf{Y}_B)$;
   Senão $classe(\mathbf{x}_i) = classe(anomalia)$.

3.4 Análise de Complexidade

Em termos de complexidade de tempo, o limite superior do HDBSCAN*(cd,-) + Fo é determinado pela etapa de construção da árvore geradora mínima, que é compartilhada pelo HDBSCAN* e HDBSCAN*(cd,-) e calculada uma única vez. A complexidade temporal equivale a $O(dn^2)$ se o algoritmo receber como entrada um conjunto de dados com n instâncias e d dimensões (já que há esforço adicional para calcular as distâncias para o kNNG), e $O(n^2)$ se for fornecida uma matriz de distâncias como entrada (CAMPELLO et al., 2015). A complexidade de espaço também é dominada pelo mesmo passo de cál-



(a) HDBSCAN* com FOSC (b) HDBSCAN*(cd,-) (c) HDBSCAN*(cd,-) + Fo

(d) kNN_LDP (e) kNN_LDP* (f) kNN_LDP* + Fo

Figura 6 – Fluxogramas dos algoritmos envolvidos neste trabalho

culo da árvore geradora mínima e é no máximo $O(dn)$ para um conjunto de dados com d dimensões como entrada e $O(n^2)$ se for utilizada uma matriz de distâncias (CAMPELLO et al., 2015). O kNN_LDP* + Fo também requer no máximo complexidade temporal equivalente a $O(dn^2)$, se for utilizado um conjunto de dados com n instâncias e d dimensões como entrada, e $O(n^2)$ se uma matriz de distâncias for utilizada, já que o passo que realiza o cálculo da árvore geradora mínima do HDBSCAN* (que inclui o kNNG, que é computado uma única vez e reutilizado pelos passos de propagação de probabilidades de distribuição de rótulos do kNN_LDP*) domina o processo (CAMPELLO et al., 2015). De forma análoga ao HDBSCAN*(cd,-) + Fo, a complexidade de espaço do kNN_LDP* + Fo possui um limite superior $O(dn)$ se um conjunto de dados d-dimensional for utilizado como entrada e $O(n^2)$ se for usada uma matriz de distâncias (CAMPELLO et al., 2015).

Capítulo 4

Experimentos

Neste capítulo, apresentamos: a) a lista de conjuntos de dados reais que serviram como base para a geração dos conjuntos semissintéticos com anomalias globais e locais utilizados em nossos experimentos; b) a metodologia e os parâmetros de pré-processamento que empregamos no processo de geração destes conjuntos de dados semissintéticos; c) os diferentes tratamentos que aplicamos e as composições dos experimentos; d) as medidas de validação que utilizamos para avaliar a qualidade das tarefas de classificação e de detecção de anomalias, assim como a forma como sumarizamos estas medidas de validação; e e) os resultados dos experimentos divididos em detecção de anomalias e classificação semissupervisionada e uma breve discussão dos resultados de cada tarefa.

4.1 Conjuntos de Dados

Para os experimentos, utilizamos 19 conjuntos de dados comumente empregados em tarefas de classificação (FERNÁNDEZ-DELGADO et al., 2014; GERTRUDES et al., 2019; GøTTCKE; ZIMEK; CAMPELLO, 2025), a partir dos quais geramos 42 conjuntos de dados semissintéticos, sendo 21 com anomalias locais e 21 com anomalias globais, seguindo a metodologia utilizada por Ghosh et al. (2024), Han et al. (2022). A Tabela 1 lista os conjuntos de dados e suas características. Dois dos conjuntos de dados, *cardiotocography* e *vertebral*, possuem duas possíveis colunas objetivo, resultando em duas variações cada.

Tabela 1 – Conjuntos de dados semissintéticos utilizados nos experimentos

Conjunto	n	# Atrib.	# Classes	# Anomalias	% Anomalias
autoPrice	166	15	2	7	4,22
banknote-authentication	1.440	4	2	68	4,72
cardiotocography_class	1.909	21	4	79	4,14
cardiotocography_nsp	2.099	21	2	106	5,05
chscase_geyser1	233	2	2	11	4,72
diggle_table	308	8	6	15	4,87
fault	1.911	27	3	97	5,08
glass	221	9	4	10	4,52
iris	157	4	2	7	4,46
letter	19.975	16	8	712	3,56
lymphography	155	18	2	7	4,52
mfeat-karhunen	2.100	64	2	100	4,76
seeds	220	7	3	10	4,55
segmentation-normcols	2.183	19	7	83	3,80
stock	997	9	2	47	4,71
transplant	137	3	2	6	4,38
vertebral_2C	325	6	2	15	4,62
vertebral_3C	325	6	2	15	4,62
wdbc	597	30	2	28	4,69
wine	186	13	2	8	4,30
yeast	1.523	8	3	74	4,86

4.2 Pré-processamento

Seguindo a mesma metodologia que os trabalhos de Ghosh et al. (2024), Han et al. (2022), treinamos um Modelo de Mistura Gaussiana (MMG) apenas com amostras normais dos conjuntos de dados reais, e em seguida, geramos novas observações normais sintéticas a partir do modelo treinado. Para as anomalias locais, multiplicamos a matriz de covariância Σ do MMG treinado por uma constante $\alpha = 5$ para produzir uma matriz distorcida $\hat{\Sigma} = 5\Sigma$, conforme as publicações de Ghosh et al. (2024), Han et al. (2022). O MMG com a matriz modificada foi então utilizado para gerar anomalias sintéticas locais em uma proporção de aproximadamente 5% do total de objetos. Para as anomalias globais aplicamos uma distribuição uniforme com limites $(\alpha \cdot min(\mathbf{X}^k), \alpha \cdot max(\mathbf{X}^k))$, mutiplicados por $\alpha = 1.1$ (também conforme Ghosh et al. (2024), Han et al. (2022)), onde $\mathbf{X}^k$ é o k-ésimo atributo de $\mathbf{X}$. Em alinhamento com o trabalho de Ghosh et al. (2024), utilizamos a técnica *tomek link* (KRAWCZYK et al., 2019) para evitar que as anomalias globais não fossem geradas demasiado próximos de observações normais. Desta forma, geramos anomalias globais também em quantidade aproximada a 5% do total de objetos. Em alguns casos específicos, removemos classes com menos objetos do que anomalias com o objetivo de evitar a inclusão de classes mais raras do que as próprias anomalias. O código-fonte para a geração dos conjuntos de dados semissintéticos foi obtido do reposi-

tório público fornecido por Han et al. (2022) e adaptado para uso em conjuntos de dados contendo múltiplas classes.

4.3 Composição dos Experimentos

Comparamos os dois métodos propostos com o kNN_LDP original (GøTTCKE; ZIMEK; CAMPELLO, 2025) e com o SSDBSCAN* (GERTRUDES et al., 2019), que é uma versão mais eficiente do SSDBSCAN original (LELIS; SANDER, 2009), que faz uso de uma única árvore geradora mínima. Para o SSDBSCAN*, utilizamos a implementação pública disponibilizada por Gertrudes et al. (2019) e para o kNN_LDP utilizamos a implementação disponibilizada publicamente pelos próprios autores. Para as estimativas de densidade, os parâmetros m_{pts} e k foram explorados no intervalo $[2, 5]$. Conforme mencionado na Seção 3.3, em nossos experimentos e durante o processo de combinação dos métodos propostos, consideramos $m_{pts} = k$, com base nos argumentos apresentados por Gøttcke, Zimek e Campello (2025). Especificamente para o componente FOSC de nossos métodos propostos, o parâmetro m_{clSize} foi fixado em 2. Quatro diferentes níveis de rótulos semissupervisionados foram testados em nossos experimentos: 2%, 5%, 8% e 10% dos objetos rotulados. Para cada conjunto de dados, subconjuntos de observações foram selecionados aleatoriamente em 20 ensaios para servir de entrada para os algoritmos, garantindo que pelo menos um objeto fosse selecionado para cada classe existente. Os 20 ensaios foram então sumarizados por meio da média aritmética simples, com o objetivo de evitar qualquer viés oriundo de resultados individuais.

4.4 Medidas de Validação

Para avaliar a capacidade de detecção de anomalias dos métodos, consideramos o conjunto de objetos marcados como anomalias como uma classe especial: calculamos a *F-Measure* com base nas anomalias da partição verdade e sumarizamos os resultados através da média aritmética de todos os ensaios. Para determinar a qualidade dos resultados da tarefa de classificação semissupervisionada, utilizamos a *F-Measure* ponderada de todas as classes (excluindo do cálculo a classe especial de anomalias), o que significa que o valor da *F-Measure* de cada classe foi multiplicado pela proporção de objetos que a representam. A *F-Measure* ponderada também foi sumarizada por meio da média aritmética dos 20 ensaios em todas as comparações.

4.5 Resultados

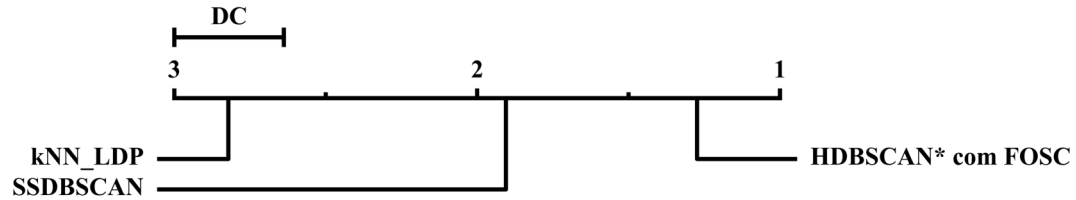
Nesta seção, apresentamos os resultados dos experimentos realizados, divididos em detecção de anomalias e classificação semissupervisionada. Para a tarefa de detecção

de anomalias, comparamos o HDBSCAN* e FOSC com dois algoritmos de classificação semissupervisionada baseados em densidade: o SSDBSCAN* e kNN_LDP. Para a tarefa de classificação semissupervisionada, comparamos os dois métodos propostos HDBSCAN*(cd,-) + Fo e kNN_LDP* + Fo com seus equivalentes originais, sem a integração com o HDBSCAN* e FOSC, e também com os mesmos métodos SSDBSCAN* e kNN_LDP originais mencionados anteriormente.

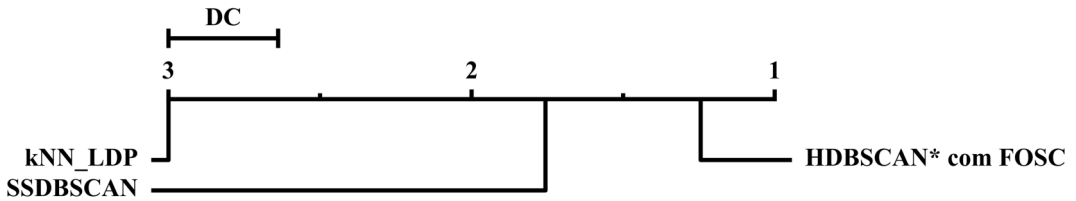
4.5.1 Detecção de Anomalias

Para ambos os métodos propostos neste trabalho, o mecanismo que determina se uma observação é uma anomalia ou não é o HDBSCAN* com a etapa de pós-processamento FOSC. Logo, comparamos a qualidade da tarefa de detecção de anomalias do HDBSCAN* e FOSC com os algoritmos kNN_LDP e SSDBSCAN*, algoritmos similares que também são capazes de identificar anomalias do ponto de vista da densidade. Para cada algoritmo e para cada combinação de percentual de rótulos semissupervisionados, conjunto de dados e tipo de anomalias foi selecionada a maior pontuação média da *F-Measure* da classe especial de anomalias entre todos os valores de k ou m_{pts}. Para determinar se há diferenças entre os resultados dos algoritmos, aplicamos o teste de Friedman aos três competidores, considerando a hipótese nula que os algoritmos são equivalentes, logo suas colocações em termos de qualidade dos resultados nos conjuntos de dados são iguais. Os resultados obtidos indicaram que a hipótese nula deveria ser rejeitada, o que significa que os algoritmos comparados produziram resultados diferentes entre si. Para determinar as diferenças entre os resultados dos três competidores, seguindo a metodologia publicada por Demšar (2006), prosseguimos para realizar o teste posterior de Nemenyi. Nossos resultados mostram que o método HDBSCAN* juntamente com o FOSC obteve o melhor desempenho na tarefa de reconhecimento de anomalias locais e globais, seguido pelo SSDBSCAN* na segunda posição e o kNN_LDP original com o pior desempenho. A Figura 7 exibe as colocações médias e a distância crítica entre os três métodos para conjuntos com anomalias globais e locais. Segundo o teste aplicado com significância de 95%, é possível afirmar que os três algoritmos produziram resultados diferentes entre si, independentemente do tipo de anomalias existentes nos dados.

A Figura 8 mostra as melhores pontuações da *F-Measure* da classe de anomalias de cada algoritmo entre todos os níveis de k e m_{pts}, considerando todos os conjuntos de dados. Embora os resultados do HDBSCAN* com FOSC não variem em relação aos diferentes níveis de rótulos semissupervisionados, observamos que o aumento do percentual de rótulos fornecidos ao kNN_LDP e ao SSDBSCAN* tende a causar uma leve diminuição na pontuação máxima da *F-Measure* da classe de anomalias. Este comportamento emerge através do próprio funcionamento natural e esperado dos algoritmos kNN_LDP e SSDBSCAN*: Conforme mais objetos rotulados são fornecidos como entrada, menos objetos são deixados sem um rótulo pelo kNN_LDP pois as regiões do espaço sem um objeto rotulado



(a) Anomalias globais



(b) Anomalias locais

Figura 7 – Colocações médias da *F-Measure* da classe especial de anomalias para $\alpha = 0.05$.

próximo (de acordo com k) tendem a diminuir. Da mesma forma, conforme a quantidade de objetos rotulados fornecidos ao SSDBSCAN* aumenta, a probabilidade do algoritmo encontrar um objeto rotulado de classe diferente do objeto rotulado de origem durante a construção da AGM diminui. Consequentemente a quantidade de objetos não rotulados pelo algoritmo tende a reduzir. Os comportamentos citados podem ser observados por conjunto de dados contendo anomalias globais por meio da Figura 9 e por conjuntos de dados contendo anomalias locais por meio da Figura 9. Muito embora a diminuição da *F-Measure* conforme o aumento de objetos rotulados não ocorra para todos os conjuntos de dados, os conjuntos *autoPrice*, *fault*, *transplant*, *vertebral_2C*, *vertebral_3C*, e *wine* exemplificam de forma clara este comportamento, tanto para anomalias globais quanto para anomalias locais. A queda de pontuação da *F-Measure*, quando ocorre, se mostra mais acentuada em conjuntos com anomalias globais.

Outra tendência clara observada nos resultados é que os três métodos apresentaram melhor desempenho em identificar anomalias globais do que em identificar anomalias locais. A queda de desempenho foi maior para o kNN_LDP e HDBSCAN* com FOSC do que para o SSDBSCAN* quando comparamos os resultados de detecção de anomalias locais com os de anomalias globais. Com exceção de dois conjuntos de dados contendo anomalias globais (conjuntos *glass* e yeast, *cujos* resultados são exibidos na Figura 9), o algoritmo kNN_LDP original apresentou baixo desempenho, de forma geral, tanto para o reconhecimento de anomalias globais quanto para o reconhecimento de anomalias locais. Isso nos leva a concluir que os objetos deixados sem rótulo pelo kNN_LDP não são, em geral, bons candidatos a anomalias globais ou locais, enquanto os objetos rotulados como ruído pelo HDBSCAN* e FOSC apresentaram melhor correspondência com as anomalias

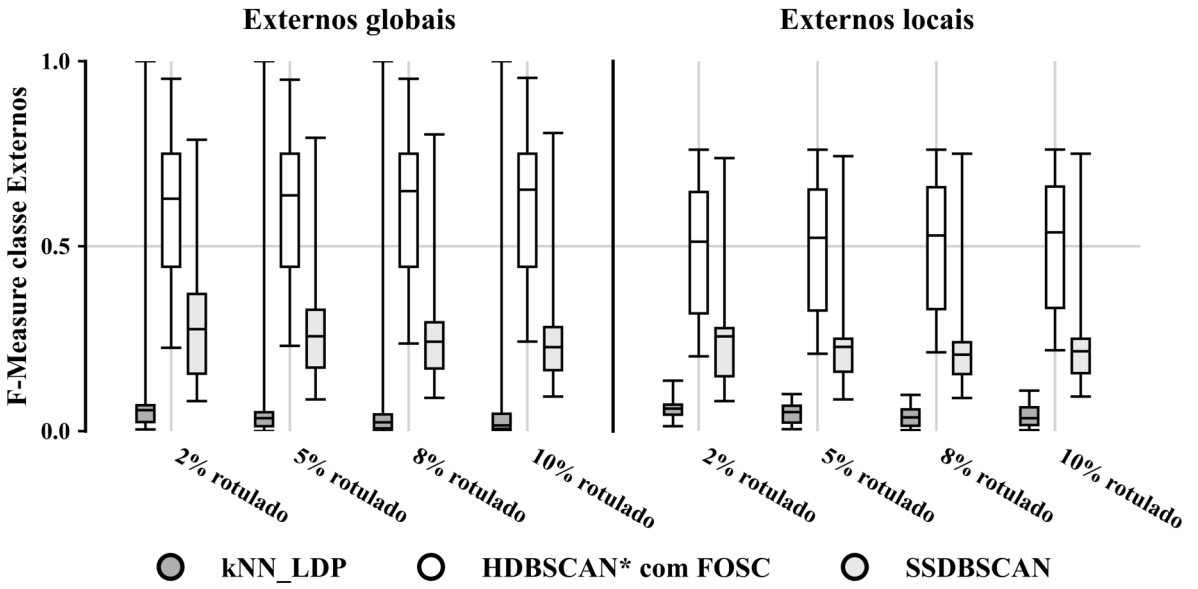


Figura 8 – Comparação da *F-Measure* da classe de anomalias

globais e locais verdadeiras de forma geral. Por fim, o algoritmo SSDBSCAN* apresentou desempenho intermediário entre o kNN_LDP e HDBSCAN* com FOSC. Nos conjuntos de dados com anomalias globais, com exceção dos conjuntos *iris* e *stock*, os melhores resultados da *F-Measure* produzidos pelo SSDBSCAN* ficaram abaixo de 0,5. Já nos conjuntos com anomalias locais, com exceção dos conjuntos *diggle_table*, *lymphography*, e *stock*, os melhores resultados da *F-Measure* produzidos pelo SSDBSCAN* também ficaram em sua maiora abaixo de 0,5.

4.5.2 Classificação Semissupervisionada

Avaliamos o HDBSCAN(cd,-) e o kNN_LDP* juntamente com os novos métodos híbridos propostos HDBSCAN*(cd,-) + Fo e kNN_LDP* + Fo, além dos competidores kNN_LDP e SSDBSCAN*, seguindo novamente a metodologia proposta por (DEMšAR, 2006). Para cada método, selecionamos a maior pontuação média da *F-Measure* ponderada (excluindo a classe especial de anomalias) entre todos os níveis de m_{pts} ou k e aplicamos o teste de Friedman, considerando significância de 95%. A hipótese nula foi rejeitada e novamente o teste posterior de Nemenyi foi aplicado aos competidores. Conforme ilustrado pela Figura 11 o kNN_LDP* e o kNN_LDP original apresentaram os melhores desempenhos gerais na tarefa de classificação, sem diferenças significativas entre ambos os métodos. Os métodos kNN_LDP* + Fo e HDBSCAN*(cd,-) apresentaram desempenho intermediário, também sem diferenças significativas entre ambos, enquanto o SSDBSCAN* e HDBSCAN*(cd,-) + Fo apresentaram o pior desempenho entre todos os métodos. O kNN_LDP, kNN_LDP*, SSDBSCAN* e HDBSCAN*(cd,-) apresentaram desempenho ligeiramente melhor em conjuntos de dados com anomalias locais do que em

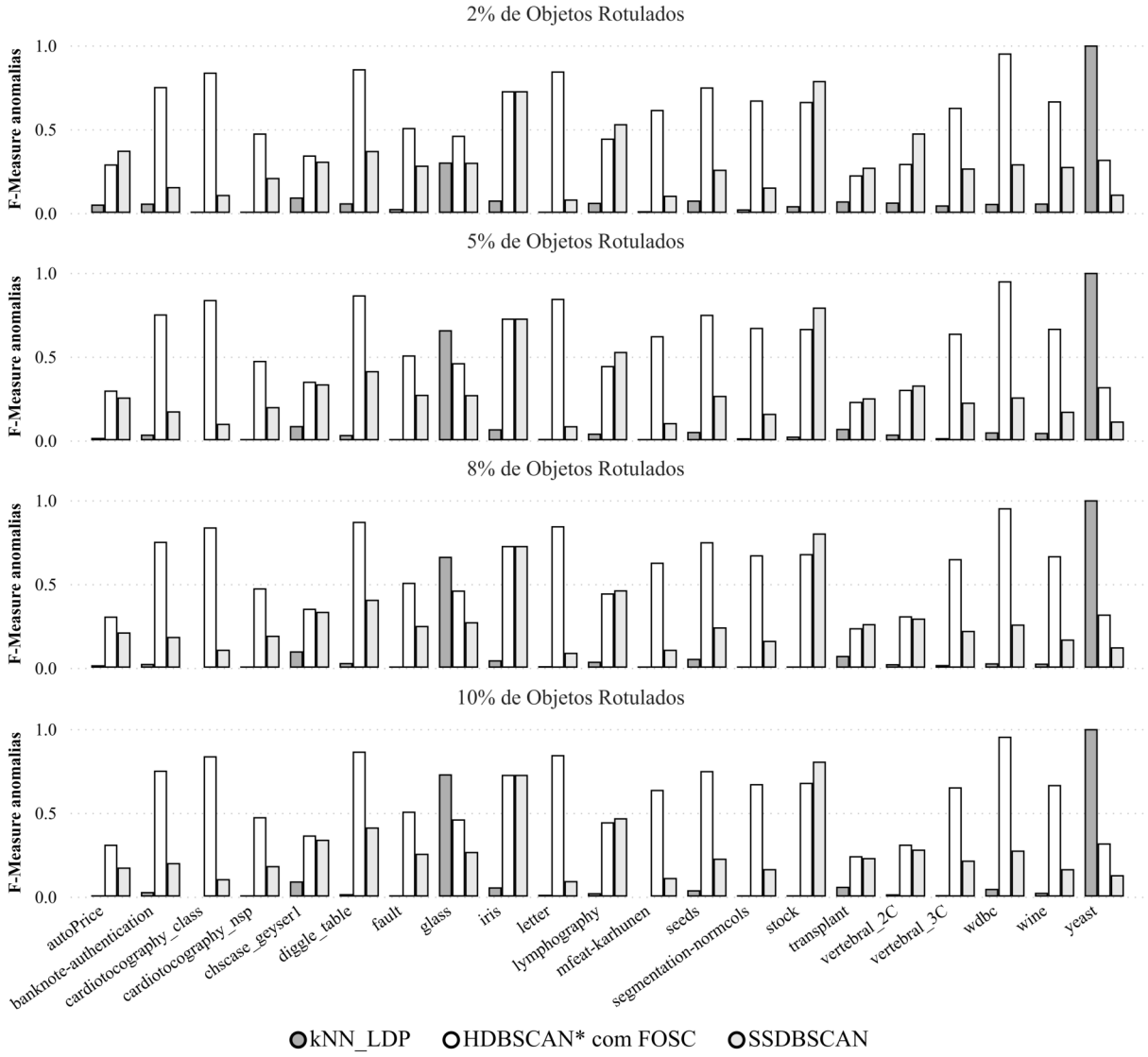


Figura 9 – Comparação da *F-Measure* da classe de anomalias por conjunto de dados com anomalias globais

conjuntos de dados com anomalias globais, ao mesmo tempo em que o kNN_LDP* + Fo e HDBSCAN*(cd,-) + Fo apresentaram desempenho levemente pior em conjuntos com anomalias locais.

Os resultados mostram que há uma pequena degradação na qualidade da tarefa de classificação quando o HDBSCAN*(cd,-) e kNN_LDP* são combinados com o HDBSCAN* e FOSC. A Figura 12 exibe a diferença absoluta das maiores pontuações médias da *F-Measure* ponderada entre HDBSCAN*(cd,-) + Fo e HDBSCAN*(cd,-) e entre kNN_LDP* + Fo e kNN_LDP*, respectivamente. Como pode ser observado pelos gráficos, esta degradação tende a ser menor em conjuntos de dados com anomalias globais. Para o método HDBSCAN*(cd,-) + Fo, a degradação com relação ao método original HDBSCAN*(cd,-) aumenta levemente conforme o percentual de rótulos semissupervisionados cresce. Este

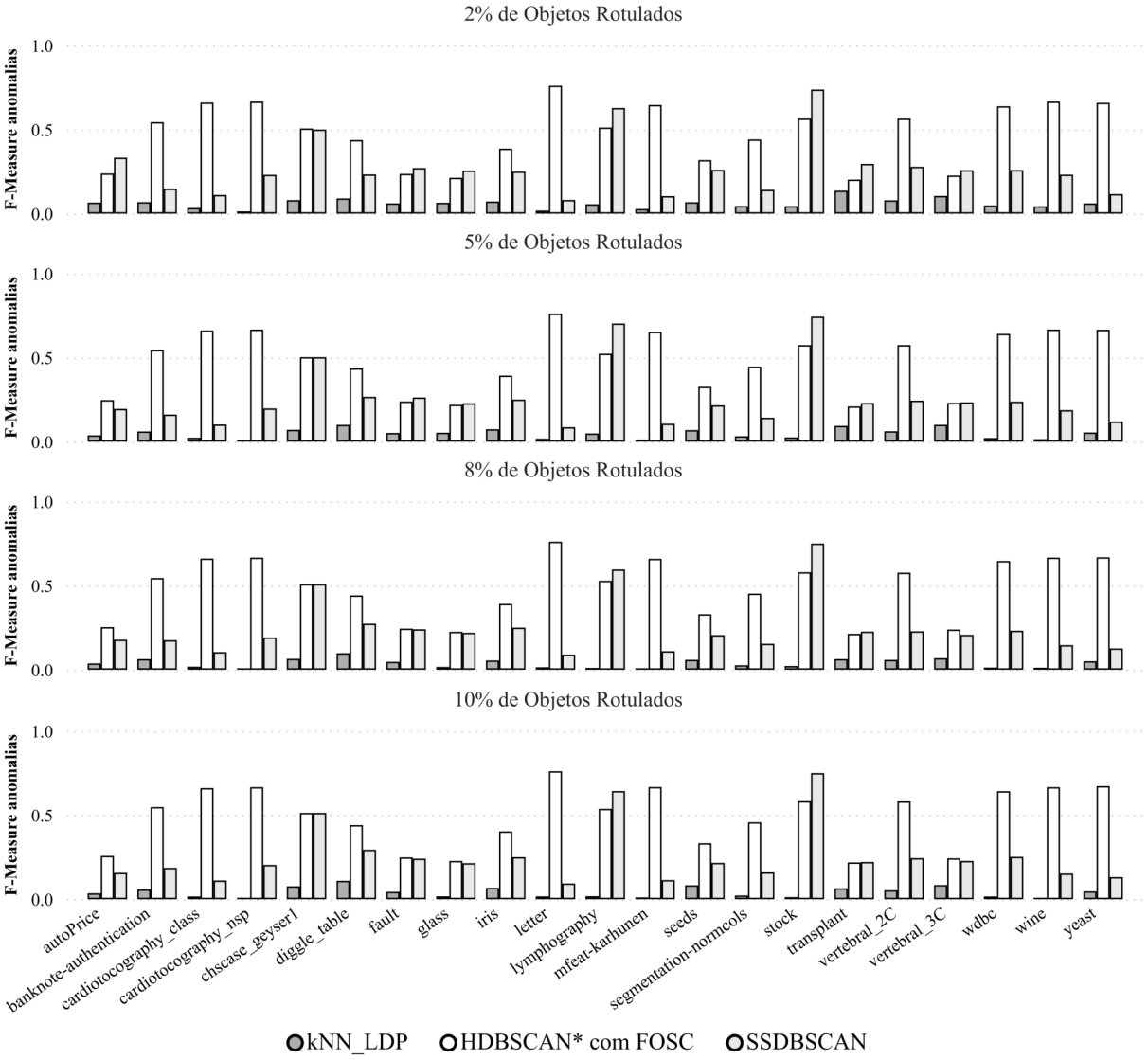
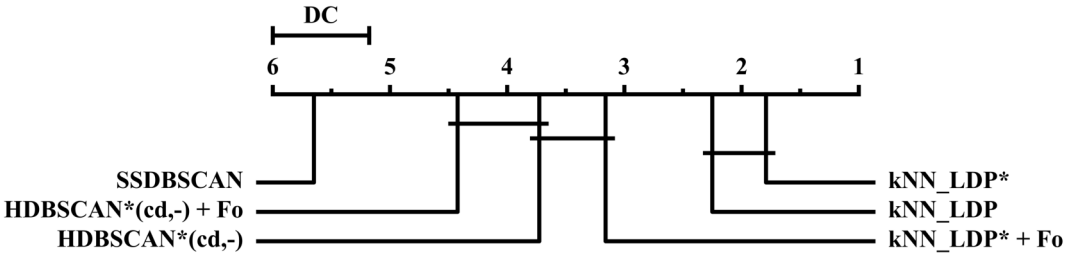


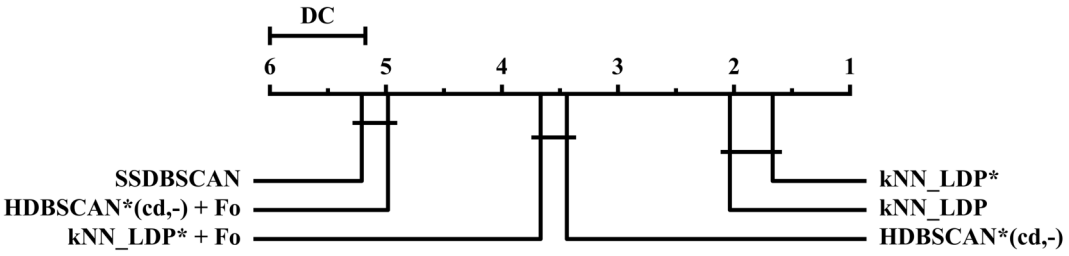
Figura 10 – Comparação da *F-Measure* da classe de anomalias por conjunto de dados com anomalias locais

comportamento pode ser explicado através da melhora do desempenho do componente HDBSCAN*(cd,-) conforme mais objetos rotulados são fornecidos, já que o componente HDBSCAN* com FOSC é não supervisionado e portanto independe dos objetos rotulados.

A degradação introduzida pelo método HDBSCAN*(cd,-) + Fo está detalhada por conjunto de dados nas Figuras 13(a) (conjuntos com anomalias globais) e 13(b) (conjuntos com anomalias locais). As Figuras 13(a), 13(b) citadas e as Figuras posteriores 14(a) e 14(b) seguem o mesmo padrão visual: Barras verticais representando diferenças positivas entre os métodos são exibidas crescendo para cima a partir da origem, enquanto barras verticais representando diferenças negativas são exibidas crescendo para baixo à partir da origem. As barras estão codificadas por cor de acordo com três intervalos: a) barras de cor branca representam diferenças pertencentes ao intervalo $[0, 1]$; b) barras de



(a) Anomalias globais

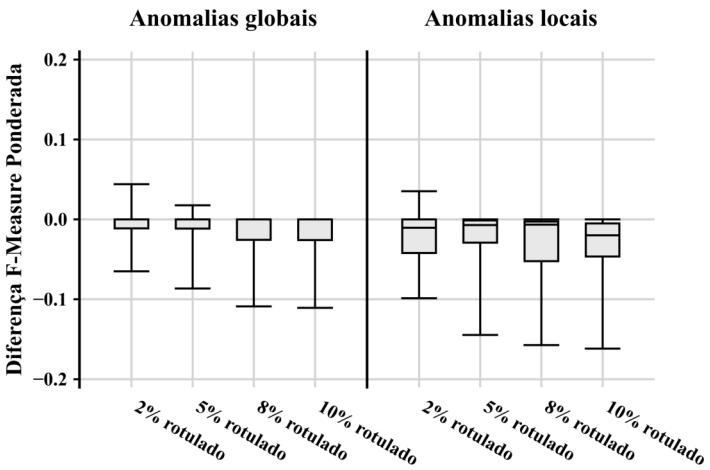


(b) Anomalias locais

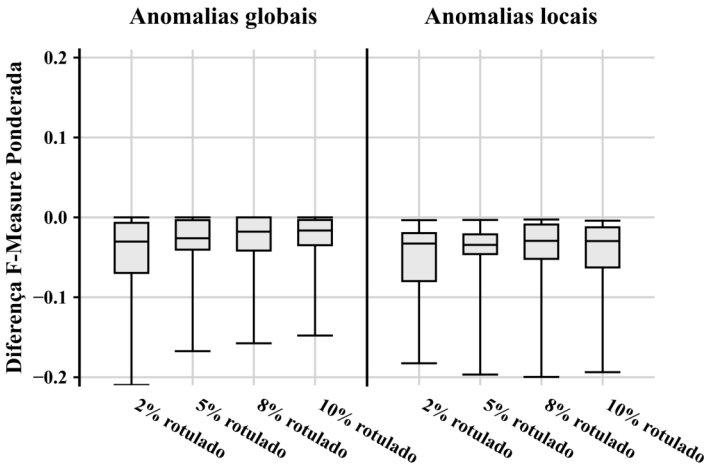
Figura 11 – Colocações médias da *F-Measure* ponderada para $\alpha = 0.05$.

cor cinza representam diferenças pertencentes ao intervalo $[-0,04, 0)$; c) barras de cor preta representam diferenças pertencentes ao intervalo $[-1, -0,04)$. A divisão do intervalo negativo das diferenças no ponto $0,04$ foi pensada de modo a possibilitar a análise do impacto na qualidade no contexto da quantidade relativa de anomalias presentes nos conjuntos de dados, que é em torno de 5%. O valor $0,04$ foi escolhido por representar percentualmente um valor relativo menor (com certa margem) que a quantidade de anomalias presentes nos conjuntos de dados. Desta forma, impactos negativos menores ou iguais a $0,04$ na *F-Measure* ponderada (desconsiderando a classe de anomalias) podem ser interpretados como casos em que a perda relativa de qualidade da tarefa de classificação é menor que a quantidade relativa de anomalias existentes nos dados, significando que o uso do método proposto é vantajoso. Da mesma forma, casos em que a degradação é maior que $0,04$ representam uma perda de qualidade de classificação relativamente maior que a quantidade de anomalias presentes, significando que o método proposto produziu resultados piores que o esperado.

Em contraste ao comportamento do HDBSCAN*(cd,-) + Fo com relação ao aumento dos rótulos semissupervisionados, o método kNN_LDP* + Fo, quando comparado com o kNN_LDP* original, apresenta degradação decrescente conforme o percentual de rótulos semissupervisionados aumenta. As Figuras 14(a) e 14(b) exibem este comportamento no nível dos conjuntos de dados, sendo alguns exemplos os conjuntos *cardiotocography_class*, *cardiotocography_nsp*, *diggle_table*, *fault*, *glass*, *letter*, *mfeat-karhunen*, e *wdbc*. Muito embora não ocorra para todos os conjuntos de dados, este comportamento é de certa forma



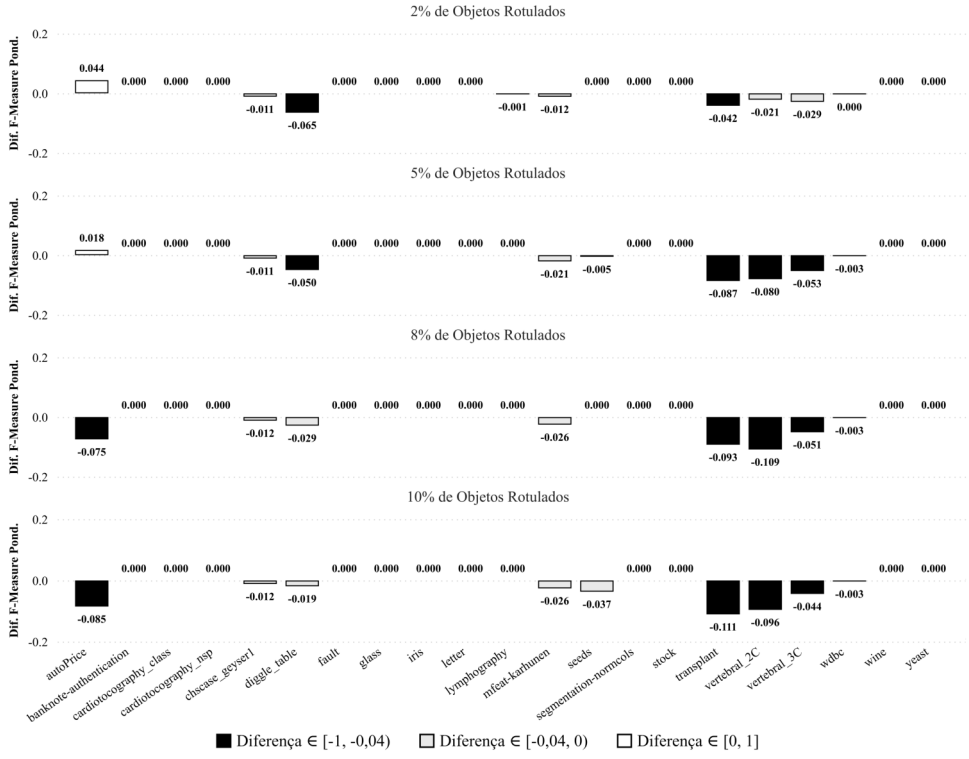
(a) HDBSCAN*(cd,-) + Fo e HDBSCAN*(cd,-)



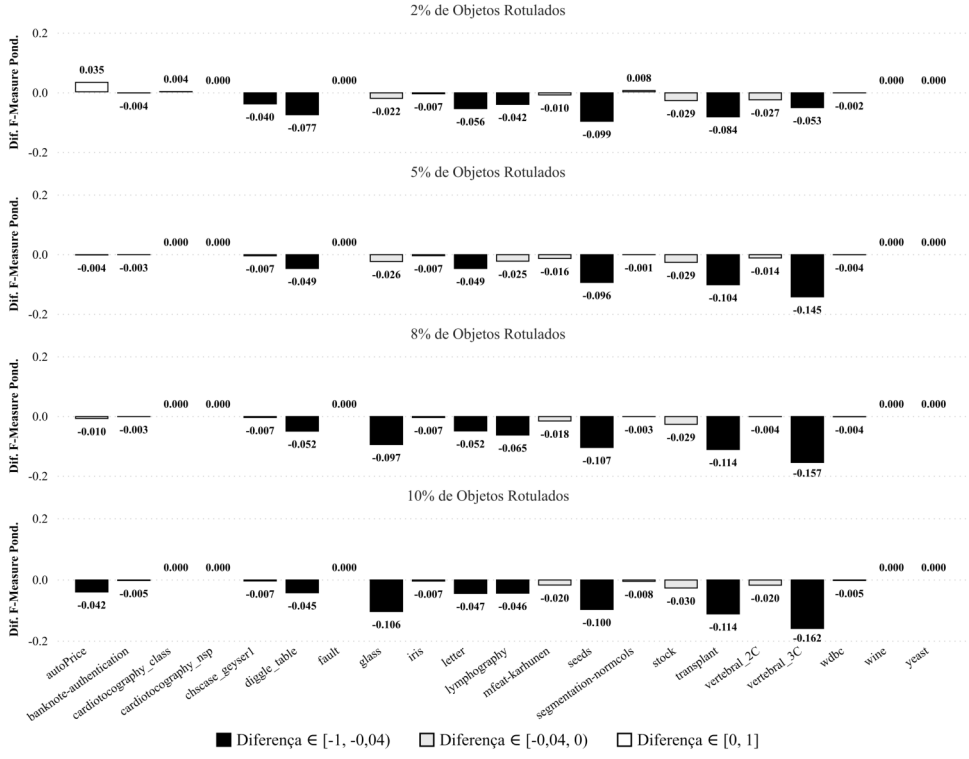
(b) kNN_LDP* + Fo e kNN_LDP*

Figura 12 – Diferença absoluta da *F-Measure* ponderada entre métodos

contraintuitivo, já que: a) o FOSC, que é o componente responsável pelo reconhecimento de externos do kNN_LDP* + Fo independe do percentual de rótulos semissupervisionados; e b) é esperado que o componente classificador do kNN_LDP* + Fo produza melhores resultados de *F-Measure* conforme o percentual de rótulos semissupervisiondos aumenta, por sua vez aumentando a degradação. A redução da degradação da *F-Measure* ponderada do kNN_LDP* + Fo conforme o aumento dos rótulos semissupervisionados acontece porque a combinação do kNN_LDP* com o HDBSCAN* e FOSC acaba por modificar o comportamento original do kNN_LDP* com relação a k e m_{pts}. Isoladamente, o kNN_LDP* tende a produzir melhores resultados da *F-Measure* ponderada quando $k = 5$, e a diferença de qualidade entre os níveis de rótulos semissupervisionados tende a ser maior para valores de k mais baixos (2 ou 3), sendo que para k mais alto (4 ou 5) os resultados dos diferentes níveis de rótulos semissupervisionados tendem a se aproximar. Já quando o kNN_LDP* é combinado com o HDBSCAN* e FOSC, os melhores resultados tendem a ser com $m_{pts} = k = 2$ ou $m_{pts} = k = 3$, com um comportamento que segue de



(a) Anomalias globais



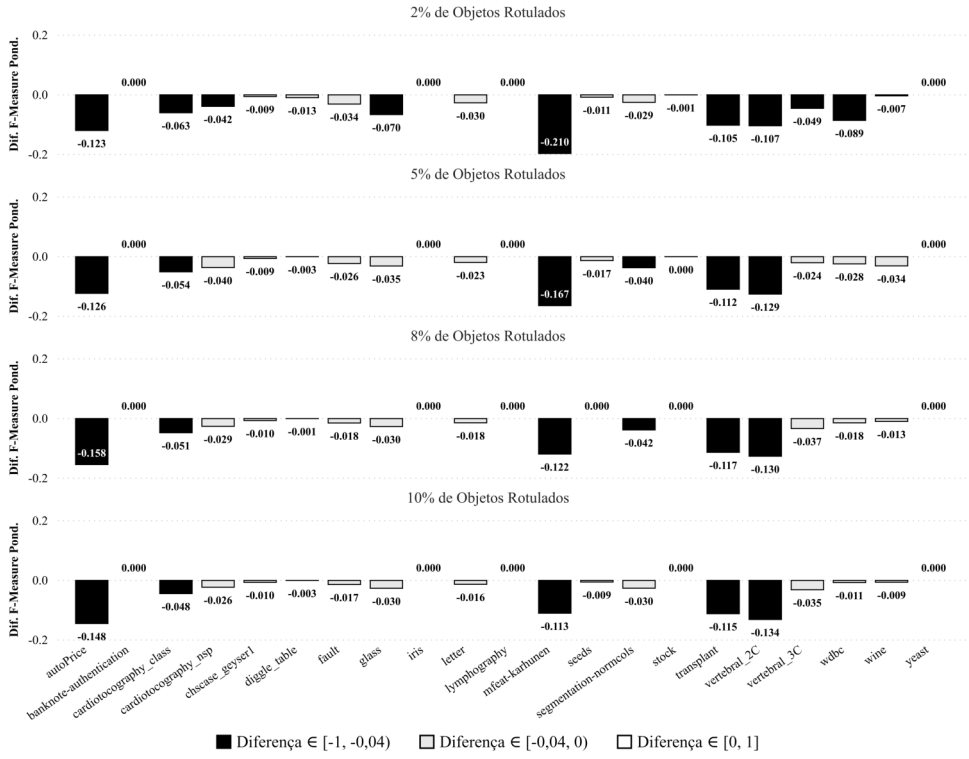
(b) Anomalias locais

Figura 13 – Diferença absoluta da *F-Measure* ponderada entre o HDBSCAN*(cd,-) + Fo e HDBSCAN*(cd,-) por conjunto de dados

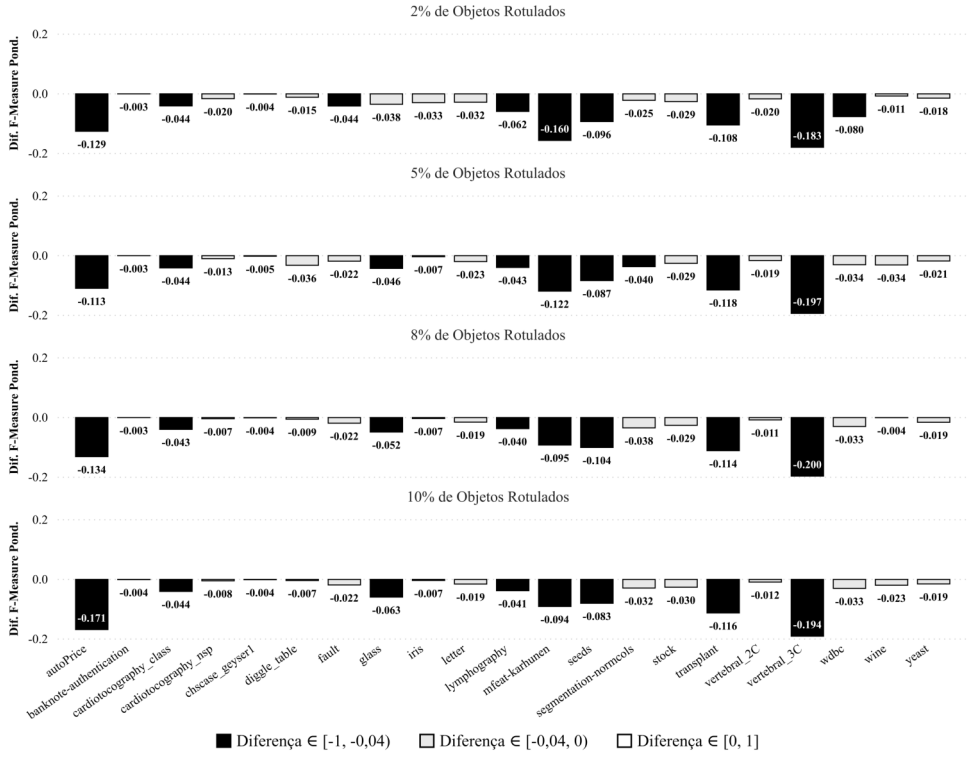
forma próxima o comportamento do kNN_LDP* original para estes valores mais baixos de k. Na prática, significa que os melhores resultados selecionados para os níveis de rótulos do método kNN_LDP* + Fo diferem em taxas relativas diferentes que os melhores resultados selecionados para os mesmos níveis de rótulos do método kNN_LDP*. A Figura 15 ilustra o comportamento descrito através de um exemplo do conjunto *letter* com anomalias globais. A Figura 15(a) exibe os resultados médios da F-Measure ponderada antes da seleção do melhor resultado final para o método kNN_LDP*, e a Figura 15(b) exibe os mesmos resultados para o método kNN_LDP* + Fo. O eixo **x** de ambos os gráficos representa os níveis de m_{pts} ou k e as linhas de cada gráfico representam os níveis de rótulos semissupervisionados. O eixo **y** representa a pontuação da F-Measure ponderada e foi definido propositalmente com escala $[0,65, 0,85]$ para que seja possível visualizar melhor as diferenças. A maior parte dos conjuntos de dados testados exibe este comportamento de degradação da *F-Measure* ponderada do método kNN_LDP* + Fo com relação ao kNN_LDP*, o que acaba por refletir nos resultados sumarizados, conforme ilustrado pela Figura 12(b).

Idealmente, a degradação da tarefa de classificação semissupervisionada introduzida pela combinação do HDBSCAN* e FOSC com os classificadores deveria ser ou zero ou muito baixa. De fato, em diversos casos a combinação do HDBSCAN* e FOSC com os classificadores resultou em zero impacto na qualidade da tarefa de classificação (diferença entre a *F-Measure* ponderada dos métodos é igual a zero). As figuras 13(a), 13(b), 14(a), e 14(b) exibem os impactos por métodos comparados, tipos de anomalias e conjuntos de dados. Alguns exemplos de casos onde não houve impacto na qualidade da classificação são os conjuntos contendo anomalias globais *banknote-authentication*, *cardiotocography_class*, *cardiotocography_nsp*, *fault*, *glass*, *iris*, *letter*, *lymphography*, *seeds*, *segmentation-normcols*, *stock*, *wine*, *yeast* quando comparamos o método HDBSCAN*(cd,-) + Fo com o HDBSCAN*(cd,-) original. Quando comparamos os mesmos métodos em conjuntos com anomalias locais, a quantidade de conjuntos de dados em que o impacto é zero é reduzida: somente os conjuntos *cardiotocography_class*, *cardiotocography_nsp*, *fault*, *wine*, e *yeast* demonstram este comportamento. Quando comparamos o método kNN_LDP* + Fo com o kNN_LDP* em conjuntos com anomalias globais, somente os cinco conjuntos *banknote-authentication*, *iris*, *lymphography*, *stock*, e *yeast* apresentam diferença zero entre a qualidade da tarefa de classificação. Já quando o kNN_LDP* + Fo é comparado com o kNN_LDP* em conjuntos de dados com anomalias locais, este comportamento não ocorreu em nenhum dos conjuntos de dados.

Em nossos resultados há alguns poucos casos envolvendo o método HDBSCAN*(cd,-) + Fo em que a *F-Measure* ponderada melhora com relação ao método original HDBSCAN*(cd,-) quando níveis mais baixos de rótulos semissupervisionados são fornecidos. O exemplo citado está contido nos gráficos de barras de 2% e 5% de objetos rotulados das Figuras 13(a) e 13(b). Este comportamento pode ser atribuído a erros de classificação do

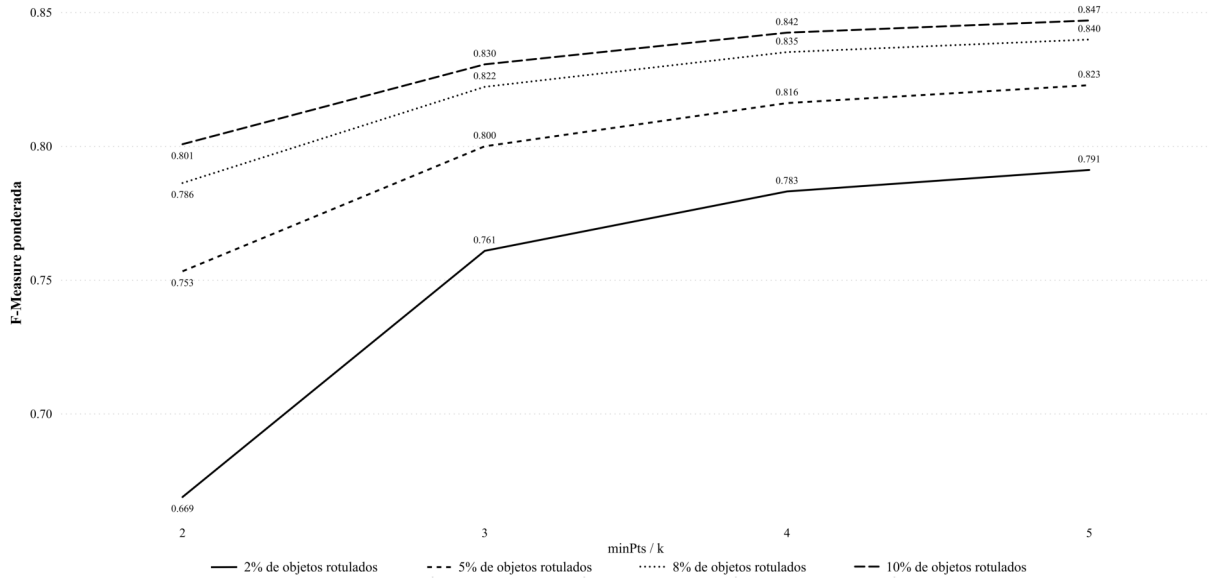


(a) Anomalias globais

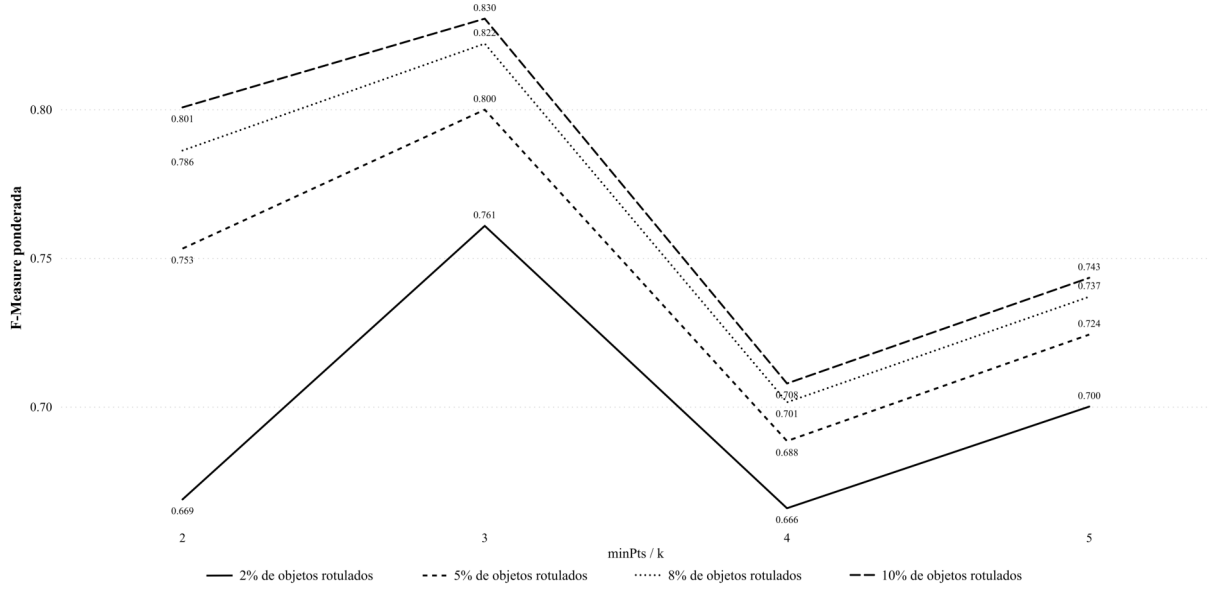


(b) Anomalias locais

Figura 14 – Diferença absoluta da *F-Measure* ponderada entre o *k*NN_LDP* + Fo e *k*NN_LDP* por conjunto de dados



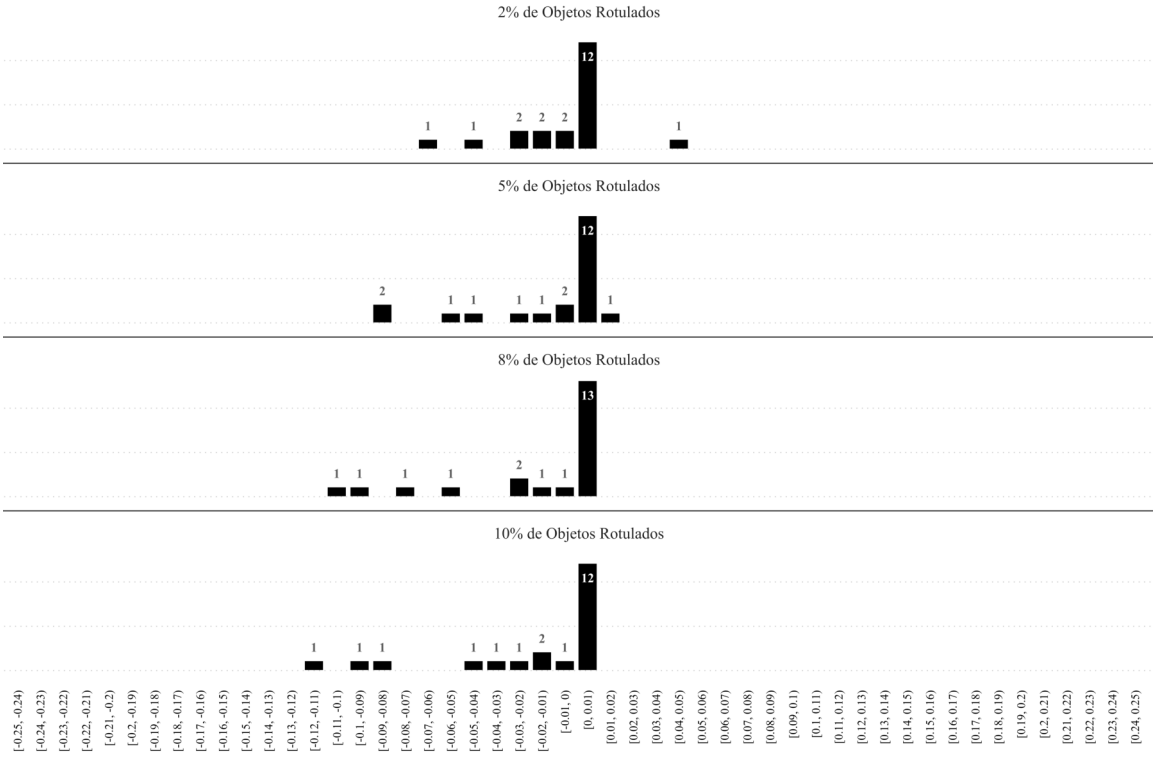
(a) kNN_LDP*



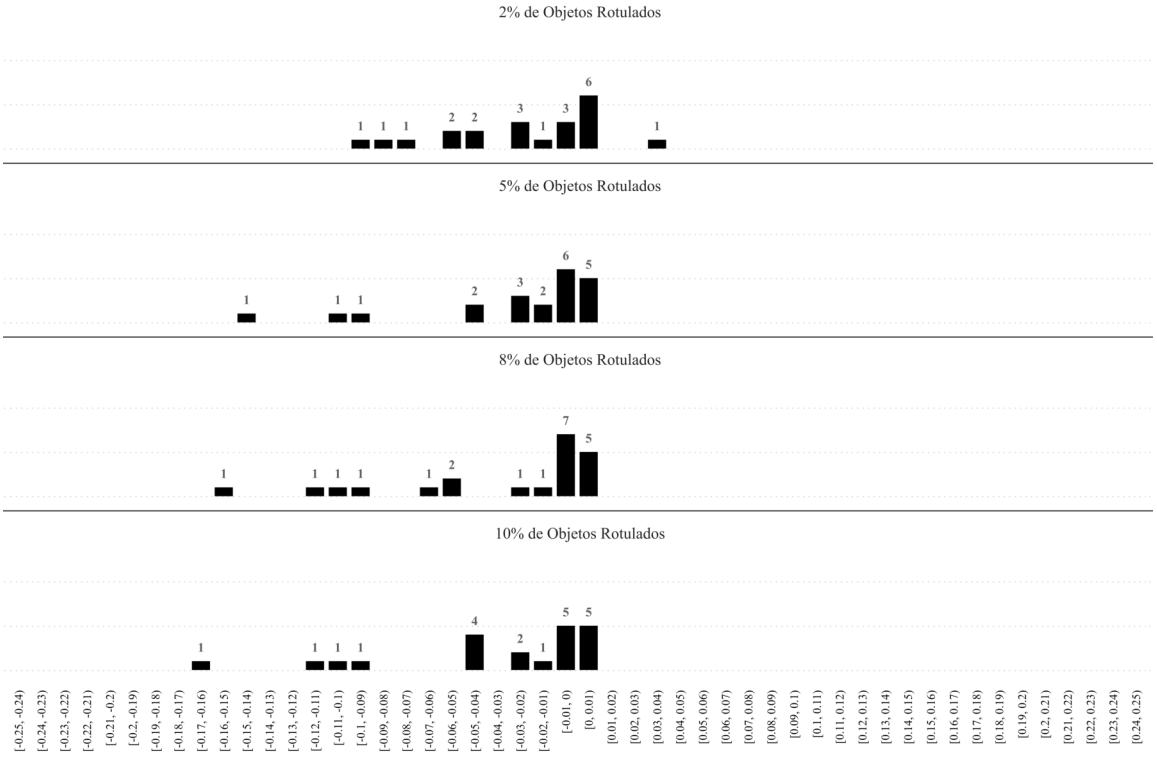
(b) kNN_LDP* + Fo

Figura 15 – *F-Measure* ponderada por níveis percentuais de rótulos semissupervisionados e m_{pts} e k para o conjunto *letter* com anomalias globais

HDBSCAN*(cd,-) que são eliminados por serem marcados como anomalias pelo componente FOSC do método HDBSCAN*(cd,-) + Fo. Estes erros tendem a não ocorrer quando níveis mais altos de rótulos semissupervisionados são fornecidos ao HDBSCAN*(cd,-), resultando em diferenças negativas (degradação) da tarefa de classificação entre o HDBSCAN*(cd,-) + Fo e HDBSCAN*(cd,-) original. O desempenho dos métodos propostos em termos da degradação da *F-Measure* ponderada pode ser observado em maior detalhe através dos histogramas contidos nas Figuras 16, e 17, que exibem a distribuição da diferença absoluta da F-Measure ponderada entre o HDBSCAN*(cd,-) + Fo e HDBSCAN*(cd,-) e entre o kNN_LDP* + Fo e kNN_LDP*, respectivamente. O intervalo total exibido no eixo $\mathbf{x}$ de todos os gráficos das duas figuras equivale a $[-0,25, 0,25)$ com centro na origem, já que as diferenças se concentram em um subconjunto deste intervalo. O intervalo individual para cada barra dos histogramas é de $0,01$. Observando os histogramas é possível determinar que, apesar das variações de desempenho entre os métodos comparados e tipos de anomalias presentes, em todas as configurações mais da metade dos conjuntos de dados apresenta degradação absoluta da *F-Measure* ponderada igual ou menor que $0,04$. A Tabela 2 exibe um resumo das distribuições da diferença absoluta da F-Measure, com agrupamento das distribuições pelos intervalos $[-1, -0,04)$ (resultados razoáveis a ruins) e $[-0,04, 1]$ (resultados razoáveis a bons). É possível determinar através dos números que o HDBSCAN*(cd,-) + Fo apresentou o melhor desempenho geral em termos de degradação, e que a degradação introduzida pelo método foi menor em conjuntos de dados com anomalias globais.

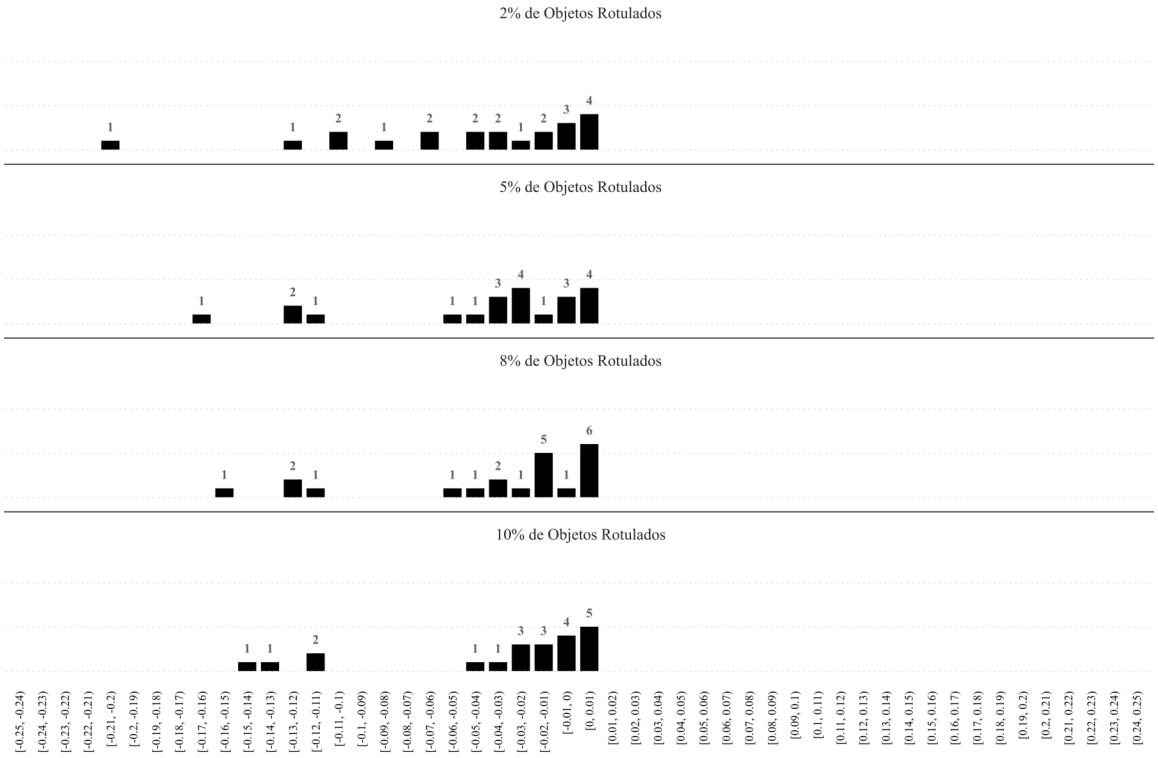


(a) Anomalias globais

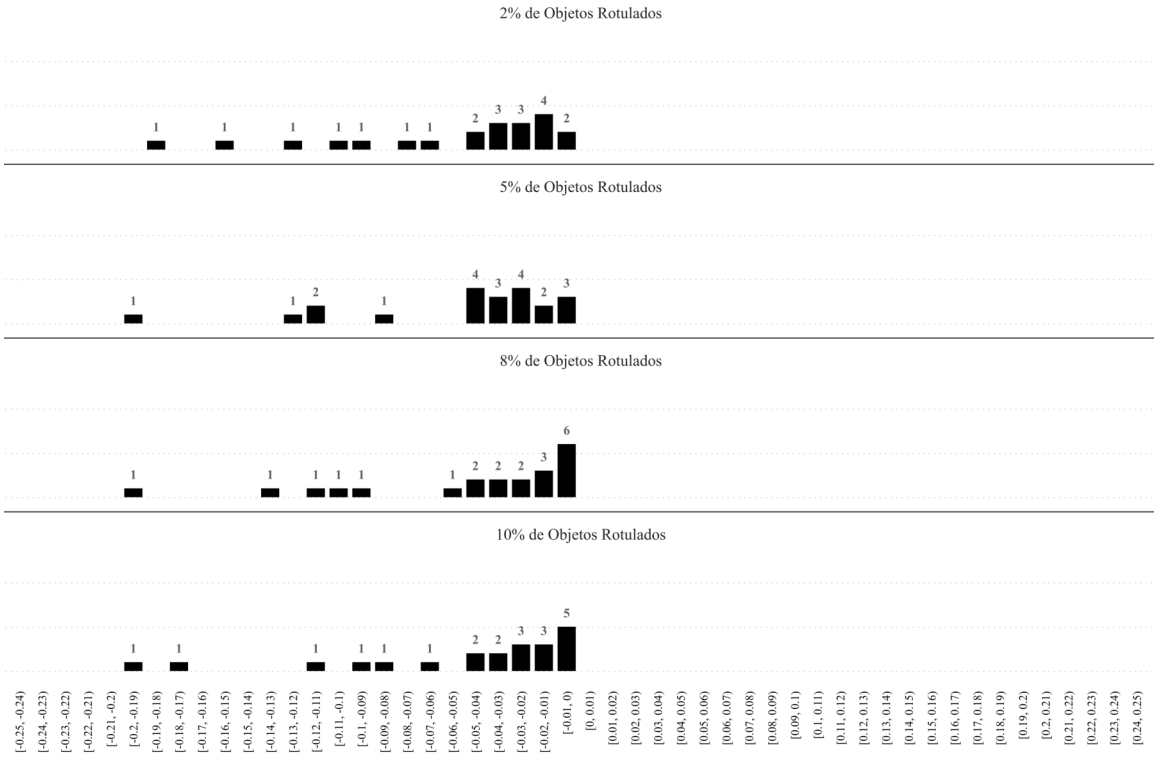


(b) Anomalias locais

Figura 16 – Histograma da diferença absoluta da *F-Measure* ponderada entre o HDBSCAN*(cd,-) + Fo e HDBSCAN*(cd,-) por conjunto de dados



(a) Anomalias globais



(b) Anomalias locais

Figura 17 – Histograma da diferença absoluta da *F-Measure* ponderada entre o kNN_LDP* + Fo e kNN_LDP* por conjunto de dados

Tabela 2 – Resumo da distribuição da diferença absoluta da *F-Measure* ponderada entre métodos

Métodos comparados	**Tipo de anomalias**	**% de objetos rotulados**	**Diferença absoluta da F-Measure ponderada**						
			$[-1, -0,04)$			$[-0,04, 1]$			**Total**
			Dif. Média	**# Conjuntos**	**% Conjuntos**	**Dif. Média**	**# Conjuntos**	**% Conjuntos**	**Dif. Média**
HDBSCAN*(cd,-) + Fo e HDBSCAN*(cd,-)	globais	2	-0,0533	2	9,5	-0,0015	19	90,5	-0,0065
		5	-0,0674	4	19,0	-0,0013	17	81,0	-0,0139
		8	-0,0819	4	19,0	-0,0041	17	81,0	-0,0189
		10	-0,0840	4	19,0	-0,0056	17	81,0	-0,0206
	locais	2	-0,0644	7	33,3	-0,0038	14	66,7	-0,0240
		5	-0,0888	5	23,8	-0,0085	16	76,2	-0,0276
		8	-0,0921	7	33,3	-0,0060	14	66,7	-0,0347
		10	-0,0829	8	38,1	-0,0078	13	61,9	-0,0364
kNN_LDP* + Fo e kNN_LDP*	globais	2	-0,0953	9	42,9	-0,0112	12	57,1	-0,0472
		5	-0,1049	6	28,6	-0,0159	15	71,4	-0,0413
		8	-0,1031	6	28,6	-0,0116	15	71,4	-0,0378
		10	-0,1116	5	23,8	-0,0122	16	76,2	-0,0359
	locais	2	-0,1005	9	42,9	-0,0206	12	57,1	-0,0549
		5	-0,0899	9	42,9	-0,0206	12	57,1	-0,0503
		8	-0,0979	8	38,1	-0,0158	13	61,9	-0,0471
		10	-0,1007	8	38,1	-0,0170	13	61,9	-0,0489

Capítulo 5

Conclusão

Neste trabalho, propomos a combinação de dois proeminentes algoritmos de classificação semissupervisionada baseada em densidade, HDBSCAN*(cd,-) e kNN_LDP, com o algoritmo de agrupamento hierárquico HDBSCAN* e a etapa de pós processamento FOSC, com o objetivo de acrescentar capacidades de detecção de anomalias a ambos os algoritmos. Para isso, apresentamos o kNN_LDP*, uma extensão do algoritmo kNN_LDP original que acrescenta uma etapa de pós-processamento que propaga rótulos para todos os objetos, independentemente do valor de k utilizado. A etapa de pós-processamento proposta reaproveita as estruturas de dados já calculadas pelo kNN_LDP e não acrescenta impacto significativo à complexidade computacional original do algoritmo. Apesar de ter sido concebido para o fim específico de nosso trabalho, o kNN_LDP* por si só pode ser uma alternativa útil ao kNN_LDP original em cenários de classificação tradicionais nos quais se deseja uma propagação explícita de rótulos para todos os objetos.

Demonstramos que o HDBSCAN* com FOSC é muito superior a algoritmos similares baseados em densidade na tarefa de identificação de anomalias globais e locais. Também demonstramos que o HDBSCAN* com FOSC pode ser integrado de forma eficiente ao HDBSCAN*(cd,-) e ao kNN_LDP* através do reaproveitamento de estruturas de dados comuns, sem impacto considerável à complexidade de ambos os algoritmos. Por fim, comprovamos que o HDBSCAN* com FOSC é uma técnica viável de identificação de anomalias baseada em densidade, produzindo apenas um pequeno impacto negativo na qualidade da classificação quando combinado com os classificadores citados. Considerando a F-Measure ponderada de todas as classes, demonstramos que este impacto negativo não passa de 0,04 para a maior parte dos conjuntos de dados, sendo que para inúmeros conjuntos o impacto é zero, dependendo do tipo de anomalias presentes e método utilizado.

Com relação a pesquisas futuras, uma possível linha de investigação é o uso das pon-

tuações GLOSH do HDBSCAN* em conjunto com o FOSC para aprimorar a qualidade da tarefa de detecção de anomalias. Como o GLOSH atribui uma pontuação de 0 a 1 para as observações, indicando o nível de anormalidade das mesmas, estas pontuações podem ser utilizadas para ponderar os valores de estabilidade, assim influenciando a decisão final do FOSC de rotular uma observação como membro de um grupo ou como ruído (ou anomalia). Também para melhorar a qualidade da tarefa de detecção de anomalias, uma segunda linha de pesquisa ainda não explorada é a modificação da função de estabilidade do FOSC. Considerando a detecção de anomalias no contexto específico da classificação semi-supervisionada, a função de estabilidade original não supervisionada pode ser substituída por variações que levem em consideração as informações de rótulos semissupervisionados disponíveis. Um benefício óbvio desta abordagem seria a não marcação de objetos previamente rotulados como anomalias, eliminando este tipo específico de erro na tarefa de reconhecimento de anomalias e conseqüentemente produzindo resultados de qualidade superior em termos de reconhecimento de anomalias. Por fim, uma terceira linha de pesquisa possível seria modificar a função de estabilidade do FOSC para que passe a otimizar o reconhecimento de anomalias, e não necessariamente a determinação dos grupos aos quais os objetos percentem. Esta abordagem é possível pois no contexto de nosso trabalho, como os agrupamentos dos objetos obtidos pelo FOSC não são utilizados nem para a tarefa de classificação nem para a de detecção de anomalias, as informações sobre os grupos dos objetos podem ser efetivamente descartadas. Desta forma, uma sugestão de modificação seria a substituição da função original por uma que considerasse os espectros de densidade em que os objetos de interesse (anomalias) não pertençam a nenhum grupo, selecionando os objetos cujo espectro é maior.

Submissões

- MASS, B.;NALDI, M. C.;GERTRUDES, J. C. Density-based Semi-supervised Classification with Anomaly Detection. **IEEE Transactions on Artificial Intelligence (TAI)**. Institute of Electrical and Electronics Engineers (IEEE). Submissão: out. 2025. ISSN 2691-4581.

Referências

AGYEMANG, M.; BARKER, K.; ALHAJJ, R. A comprehensive survey of numeric and symbolic outlier mining techniques. **Intelligent Data Analysis**, v. 10, n. 6, p. 521–538, nov. 2006. ISSN 15714128, 1088467X.

ANKERST, M. et al. OPTICS: ordering points to identify the clustering structure. In: **Proceedings of the 1999 ACM SIGMOD international conference on Management of data**. Philadelphia Pennsylvania USA: ACM, 1999. p. 49–60. ISBN 978-1-58113-084-3.

BREUNIG, M. M. et al. LOF: identifying density-based local outliers. **ACM SIGMOD Record**, v. 29, n. 2, p. 93–104, jun. 2000. ISSN 0163-5808.

BÖHM, C.; PLANT, C. HISSCLU: a hierarchical density-based method for semi-supervised clustering. In: **Proceedings of the 11th international conference on Extending database technology: Advances in database technology**. Nantes France: ACM, 2008. p. 440–451. ISBN 978-1-59593-926-5.

CAMPELLO, R. J. G. B.; MOULAVI, D.; SANDER, J. Density-Based Clustering Based on Hierarchical Density Estimates. In: **Advances in Knowledge Discovery and Data Mining**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013. v. 7819, p. 160–172. ISBN 978-3-642-37455-5 978-3-642-37456-2. Series Title: Lecture Notes in Computer Science.

CAMPELLO, R. J. G. B. et al. Hierarchical Density Estimates for Data Clustering, Visualization, and Outlier Detection. **ACM Transactions on Knowledge Discovery from Data**, v. 10, n. 1, p. 1–51, jul. 2015.

CHANDOLA, V.; BANERJEE, A.; KUMAR, V. Anomaly detection: A survey. **ACM Computing Surveys**, v. 41, n. 3, p. 1–58, jul. 2009. ISSN 0360-0300, 1557-7341.

CHAPELLE, O.; SCHöLKOPF, B.; ZIEN, A. **Introduction to Semi-Supervised Learning**. Cambridge, Massachusetts: MIT Press, 2006. 1-12 p. OCLC: 1170021034. ISBN 978-0-262-25589-9.

DEMšAR, J. Statistical comparisons of classifiers over multiple data sets. **J. Mach. Learn. Res.**, JMLR.org, v. 7, p. 1–30, dez. 2006. ISSN 1532-4435.

DONG, W.; MOSES, C.; LI, K. Efficient k-nearest neighbor graph construction for generic similarity measures. In: **Proceedings of the 20th International Conference**

on World Wide Web. New York, NY, USA: Association for Computing Machinery, 2011. (WWW '11), p. 577–586. ISBN 9781450306324.

ESTER, M. et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In: **Proceedings of the Second International Conference on Knowledge Discovery and Data Mining**. [S.l.]: AAAI Press, 1996. (KDD'96), p. 226–231.

FERNÁNDEZ-DELGADO, M. et al. Do we need hundreds of classifiers to solve real world classification problems? **J. Mach. Learn. Res.**, v. 15, n. 1, p. 3133–3181, jan. 2014. ISSN 1532-4435. Publisher: JMLR.org.

FILIPOVYCH, R.; DAVATZIKOS, C. Semi-supervised pattern classification of medical images: Application to mild cognitive impairment (MCI). **NeuroImage**, v. 55, n. 3, p. 1109–1119, abr. 2011. ISSN 10538119.

FRäNTI, P.; SIERANOJA, S. K-means properties on six clustering benchmark datasets. **Applied Intelligence**, v. 48, n. 12, p. 4743–4759, dez. 2018. ISSN 0924-669X, 1573-7497.

GERTRUDES, J. C. et al. A unified view of density-based methods for semi-supervised clustering and classification. **Data Mining and Knowledge Discovery**, v. 33, n. 6, p. 1894–1952, nov. 2019. ISSN 1384-5810, 1573-756X.

GHOSH, K. et al. Unsupervised Parameter-free Outlier Detection using HDBSCAN* Outlier Profiles. In: **2024 IEEE International Conference on Big Data (BigData)**. Washington, DC, USA: IEEE, 2024. p. 7021–7030.

GOLALIPOUR, K. et al. From clustering to clustering ensemble selection: A review. **Engineering Applications of Artificial Intelligence**, v. 104, p. 104388, set. 2021. ISSN 09521976.

GøTTCKE, J. M.; ZIMEK, A.; CAMPELLO, R. J. Bayesian label distribution propagation: A semi-supervised probabilistic k nearest neighbor classifier. **Information Systems**, v. 129, p. 102507, mar. 2025. ISSN 03064379.

HAN, S. et al. **ADBench: Anomaly Detection Benchmark**. [S.l.]: arXiv, 2022. ArXiv:2206.09426 [cs].

HANDL, J. **cluster generators**. 2017. <https://personalpages.manchester.ac.uk/staff/julia.handl/generators.html>. Acessado em: (12/01/2026).

HAWKINS, D. M. **Identification of Outliers**. Dordrecht: Springer, 1980. (Monographs on Applied Probability and Statistics). ISBN 978-94-015-3996-8 978-94-015-3994-4.

HODGE, V.; AUSTIN, J. A Survey of Outlier Detection Methodologies. **Artificial Intelligence Review**, v. 22, n. 2, p. 85–126, out. 2004. ISSN 0269-2821, 1573-7462.

IGLESIAS, F. et al. MDCGen: Multidimensional Dataset Generator for Clustering. **Journal of Classification**, v. 36, n. 3, p. 599–618, out. 2019. ISSN 0176-4268, 1432-1343.

JOACHIMS, T. et al. Transductive inference for text classification using support vector machines. In: **Icml**. [S.l.: s.n.], 1999. v. 99, p. 200–209.

KNORR, E. M.; NG, R. T. Algorithms for mining distance-based outliers in large datasets. In: **Proceedings of the 24rd International Conference on Very Large Data Bases**. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1998. (VLDB '98), p. 392–403. ISBN 1558605665.

KRAWCZYK, B. et al. Instance reduction for one-class classification. **Knowledge and Information Systems**, v. 59, n. 3, p. 601–628, jun. 2019. ISSN 0219-1377, 0219-3116.

LELIS, L.; SANDER, J. Semi-supervised Density-Based Clustering. In: **2009 Ninth IEEE International Conference on Data Mining**. Miami Beach, FL, USA: IEEE, 2009. p. 842–847. ISBN 978-1-4244-5242-2.

NETO, A. C. A. et al. CORE-SG: Efficient Computation of Multiple MSTs for Density-Based Methods. In: **2022 IEEE 38th International Conference on Data Engineering (ICDE)**. Kuala Lumpur, Malaysia: IEEE, 2022. p. 951–964.

NGAI, E. et al. The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature. **Decision Support Systems**, v. 50, n. 3, p. 559–569, fev. 2011. ISSN 01679236.

PIETRASZEK, T. On the use of ROC analysis for the optimization of abstaining classifiers. **Machine Learning**, v. 68, n. 2, p. 137–169, jul. 2007. ISSN 0885-6125, 1573-0565.

SCHLEGL, T. et al. **Unsupervised Anomaly Detection with Generative Adversarial Networks to Guide Marker Discovery**. [S.l.]: arXiv, 2017. ArXiv:1703.05921 [cs].

VEGA-PONS, S.; RUIZ-SHULCLOPER, J. A SURVEY OF CLUSTERING ENSEMBLE ALGORITHMS. **International Journal of Pattern Recognition and Artificial Intelligence**, v. 25, n. 03, p. 337–372, maio 2011. ISSN 0218-0014, 1793-6381.

VINCES, B. V. S. et al. A comparative evaluation of clustering-based outlier detection. **Data Mining and Knowledge Discovery**, v. 39, n. 2, p. 13, mar. 2025. ISSN 1384-5810, 1573-756X.

XIANG, S. et al. Semi-supervised Credit Card Fraud Detection via Attribute-Driven Graph Representation. **Proceedings of the AAAI Conference on Artificial Intelligence**, v. 37, n. 12, p. 14557–14565, jun. 2023. ISSN 2374-3468, 2159-5399.

ZHU, X.; GOLDBERG, A. B. **Introduction to Semi-Supervised Learning**. Cham: Springer International Publishing, 2009. (Synthesis Lectures on Artificial Intelligence and Machine Learning). ISBN 978-3-031-00420-9 978-3-031-01548-9.

Apêndices

APÊNDICE A

Extensão do Arcabouço Unificado com o CoreSG

A proposta inicial de pesquisa investigada durante parte do curso de mestrado consistiu em estender o arcabouço unificado para classificação e agrupamento semissupervisionados por densidade proposto por Gertrudes et al. (2019) para que seja incorporada uma versão do HDBSCAN* com a substituição do grafo completo pelo grafo simplificado *Core-distance based Spanning Graph* (CoreSG) proposta por Neto et al. (2022). A expectativa era que a extensão do arcabouço desta forma possibilitasse a geração eficiente de múltiplas hieraquias para amplas faixas de valores de m_{pts} para realização de tarefas de agrupamento e classificação semissupervisionadas, desta forma atualizando o arcabouço em termos de eficiência. De acordo com a literatura pesquisada até então, a combinação do arcabouço unificado com o algoritmo HDBSCAN* modificado com o grafo CoreSG nunca havia sido realizada e as melhorias esperadas em termos de desempenho eram significativas, a ponto de possibilitar a aplicação de estratégias antes proibitivas devido à complexidade computacional do algoritmo HDBSCAN*. Logo, considerando que o objetivo principal de pesquisa viabilizaria a geração eficiente de múltiplos resultados de agrupamentos para uma faixa toda de valores de m_{pts}, os seguintes objetivos secundários foram definidos:

- Estender o arcabouço unificado, acrescentando um processo de seleção do melhor resultado individual em termos de qualidade, entre todos os resultados base gerados para uma faixa de valores de m_{pts};
- Estender o arcabouço unificado, acrescentando um processo de combinação, ou *cluster ensemble* (GOLALIPOUR et al., 2021), (VEGA-PONS; RUIZ-SHULCLOPER,

2011) dos múltiplos resultados base para geração de um novo resultado, de qualidade igual ou superior aos resultados individuais;

- ❏ Comparar, em termos de qualidade e tempo de execução, a abordagem *all-points-core-distance* com a abordagem original de distância núcleo juntamente com o processo de geração eficiente de múltiplas hierarquias através do CoreSG;
- ❏ Investigar diferentes possibilidades para avaliação da qualidade de resultados de agrupamentos hierárquicos por densidade, assim como estratégias de seleção e agregação destes resultados.

O ponto central da proposta de pesquisa dependia da combinação do arcabouço unificado com o grafo CoreSG, e implicou na implementação de uma nova macro etapa de cálculo do CoreSG à seqüência de passos do algoritmo HDBSCAN* modificado proposto pela publicação original do arcabouço. O código fonte do arcabouço unificado em linguagem Java foi obtido através do repositório público disponibilizado por Gertrudes et al. (2019) e todas as modificações necessárias para a construção do grafo CoreSG aplicadas ao projeto original. A nova etapa foi inserida antes dos passos de cálculo de distâncias núcleo e extração da árvore geradora mínima. É importante notar que a utilização do CoreSG só faz sentido para as instâncias do arcabouço unificado em que é utilizada a definição de distância núcleo conforme a publicação original do algoritmo HDBSCAN*, já que nestes casos a distância núcleo é definida justamente em função do parâmetro m_{pts}. Por definição o processo de cálculo do CoreSG se dá com base na escolha do limite superior de um intervalo de valores de m_{pts} como entrada. Portanto, a implementação realizada substituiu o parâmetro m_{pts} do HDBSCAN* original pelos parâmetros m_{pts_min} e m_{pts_max} representando os limites (fechados) do intervalo de possíveis valores de m_{pts}.

Com a nova etapa de cálculo do CoreSG, múltiplas distâncias núcleo passaram a ser calculadas, uma para cada valor individual de $m_{pts} \in [m_{pts\_min}, m_{pts\_min} + 1, m_{pts\_min} + 2, ..., m_{pts\_max}]$. Consequentemente, também passaram a ser geradas múltiplas árvores geradoras mínimas (em contraste a uma única distância núcleo e uma única árvore geradora mínima referentes a um único valor de m_{pts} conforme o arcabouço original). Naturalmente, as duas macro etapas posteriores do HDBSCAN* propostas pela publicação original do arcabouço (etapa de atribuição de pesos às distâncias e etapa de expansão de rótulos) também foram modificadas para passar a receber múltiplos resultados como entrada e produzir múltiplos resultados como saída. A figura 18 exibe as seis possíveis variações do HDBSCAN* propostas pelos autores do arcabouço unificado porém com suas macro etapas atualizadas conforme a proposta de extensão do arcabouço unificado: Com a nova etapa de cálculo do CoreSG e funcionalidade das etapas seguintes ajustada para acomodar múltiplos resultados. A etapa nova está representada com fundo cinza escuro, enquanto etapas que sofreram modificações são apresentadas em fundo cinza claro. Etapas que permaneceram iguais às do arcabouço original são exibidas em fundo branco.

Vale reforçar que as variações 1, 3 e 5 do HDBSCAN* utilizam a medida *all-points core distance* em lugar da distância núcleo, logo o CoreSG não se aplica a estas variações, e consequentemente nenhuma das três sofreu modificações.

Após a finalização da implementação técnica do CoreSG no projeto original do arcabouço unificado, foram realizados testes de desempenho para mensurar os ganhos em termos de tempo de execução em comparação ao HDBSCAN* original. Para os testes foram utilizados conjuntos de dados conforme a lista de conjuntos envolvidos nos experimentos de classificação semissupervisionada da publicação do arcabouço unificado, cuja lista completa está exibida na Tabela 3. Além dos conjuntos de dados desta lista, foram utilizados dois conjuntos de dados obtidos através de amostragens do conjunto de dados *Credit Card Fraud Detection* disponível publicamente no sítio de internet *Kaggle* (https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud): a variação de nome *creditcard_10* representa as primeiras dez mil linhas do conjunto original, e de forma análoga, a variação de nome *creditcard_35* representa as primeiras trinta e cinco mil linhas do conjunto original.

Para avaliar o desempenho do HDBSCAN* com o grafo CoreSG em comparação ao HDBSCAN* original, utilizamos o tempo total de geração das árvores geradoras mínimas, conforme um intervalo de $m_{pts} \in [2, 30]$, já que o passo de geração das AGMs domina a complexidade do HDBSCAN*. Uma árvore geradora mínima foi gerada para cada valor individual de m_{pts}. Também calculamos o tempo médio necessário para geração de uma árvore geradora mínima no mesmo intervalo de m_{pts}. Conforme esperado para o HDBSCAN* com CoreSG, a primeira árvore geradora mínima (para $m_{pts} = 30$) leva substancialmente mais tempo que as seguintes pois envolve o cálculo do grafo CoreSG como passo inicial. Este tempo maior necessário para construir o CoreSG foi contabilizado normalmente nos cálculos de tempo total e naturalmente diluído ao ser calculado o tempo médio para geração de uma AGM no intervalo $m_{pts} \in [2, 30]$.

Para facilidade da análise e processamento dos resultados, foram construídos alguns painéis analíticos com os indicadores mencionados. A Figura 19 exibe dois exemplos destes painéis, para o conjunto de dados *yeast_Galactose* (Figura 19(a)) e o conjunto de dados *semeion* (Figura 19(b)). O eixo **X** de ambos os gráficos exibe o nível de m_{pts} e o eixo **Y** mensura o tempo em segundos necessário para geração das árvores geradoras mínimas. As linhas representam os dois métodos: HDBSCAN* com CoreSG e HDBSCAN* original. Através dos gráficos é possível identificar: a) O tempo maior necessário para a construção do CoreSG e extração da primeira AGM; e b) Uma considerável diferença entre o desempenho dos dois conjuntos de dados. Conforme veremos mais adiante, esta diferença decorre do fato de que conjuntos de dados maiores tendem a se beneficiar mais das melhorias de eficiência trazidas pelo uso do CoreSG: O conjunto de dados *yeast_Galactose* consiste de somente 205 observações e 81 atributos. Já o conjunto de dados *semeion* é composto de 1.593 observações e 256 atributos, requerendo uma quantidade consideravelmente maior

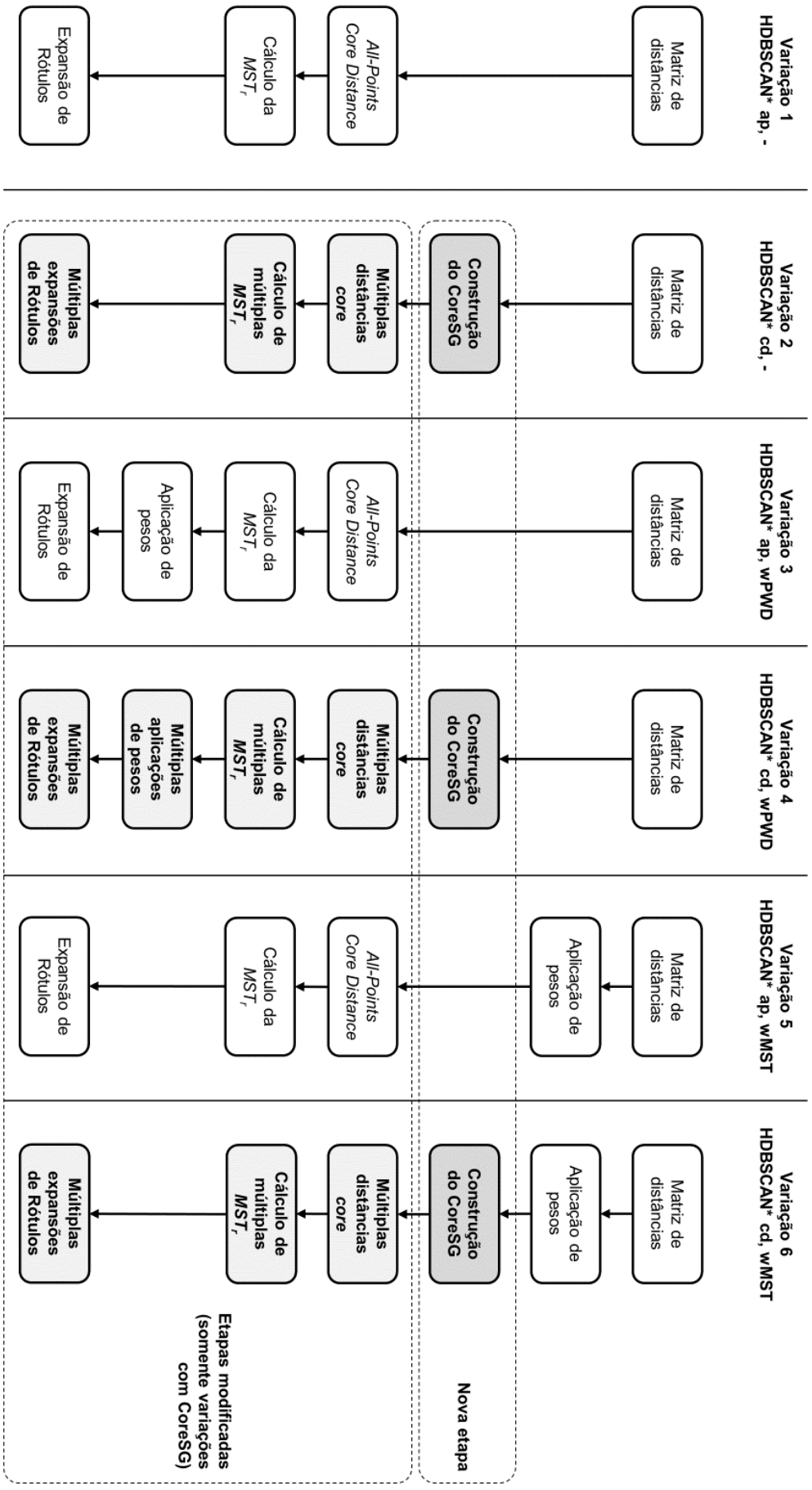


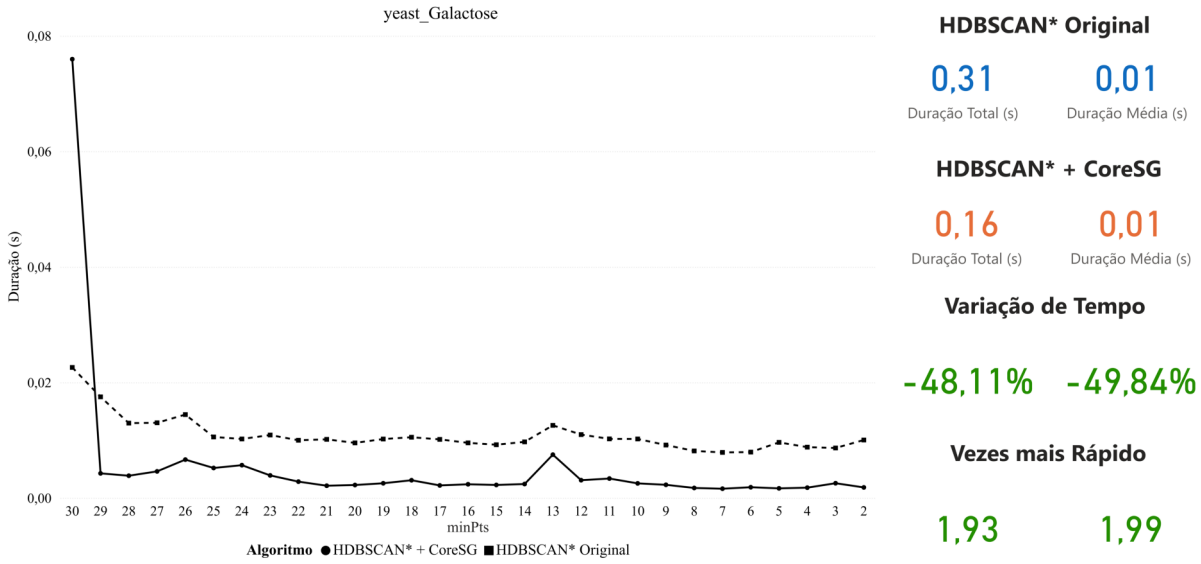
Figura 18 – Extensão do arcabouço unificado, com a nova etapa de cálculo do grafo CoreSG.

Dataset	*#obj*	*#att*	*#cl*	Distance
ACE ECFP4 (Sutherland et al. 2004)	114	1025	2	Tanimoto
ACE ECFP6 (Sutherland et al. 2004)	114	1025	2	Tanimoto
Analcatdata authorship (Vanschoren et al. 2013)	841	70	4	Cosine
Armstrong-v1 (de Souto et al. 2008)	72	1082	2	Cosine
Auto price (Vanschoren et al. 2013)	159	16	2	Euclidean
Bank note–Authentication (Vanschoren et al. 2013)	1372	5	2	Euclidean
Cardiotocography (Vanschoren et al. 2013)	2126	36	10	Euclidean
Chowdary (de Souto et al. 2008)	104	183	2	Cosine
Chcase Geyser1 (Vanschoren et al. 2013)	222	2	2	Euclidean
COX2 ECFP6 (Sutherland et al. 2004)	322	1025	2	Tanimoto
DHFR ECFP4 (Sutherland et al. 2004)	397	1025	2	Tanimoto
DHFR ECFP6 (Sutherland et al. 2004)	397	1025	2	Tanimoto
Diggle table (Vanschoren et al. 2013)	310	8	9	Euclidean
Fontaine ECFP4 (Fontaine et al. 2005)	435	1024	2	Tanimoto
Fontaine ECFP6 (Fontaine et al. 2005)	435	1024	2	Tanimoto
Gordon (de Souto et al. 2008)	181	1627	2	Cosine
Iris (Lichman 2013)	150	5	3	Euclidean
M1 ECFP4 (Gaulton et al. 2017)	769	1025	2	Tanimoto
M1 ECFP6 (Gaulton et al. 2017)	769	1025	2	Tanimoto
Mfeat-factors (Vanschoren et al. 2013)	2000	216	10	Euclidean
Mfeat-Karhunen (Vanschoren et al. 2013)	2000	65	10	Euclidean
Seeds (Lichman 2013)	210	8	3	Euclidean
Segmentation (Vanschoren et al. 2013)	2100	20	7	Euclidean
Semeion (Vanschoren et al. 2013)	1593	256	10	Cosine
Stock (Vanschoren et al. 2013)	950	10	2	Euclidean
Transplant (Vanschoren et al. 2013)	131	4	2	Euclidean
WDBC (Lichman 2013)	569	32	2	Euclidean
Wine (Lichman 2013)	178	13	3	Euclidean
Yeast galactose (Yeung et al. 2003)	205	81	4	Euclidean

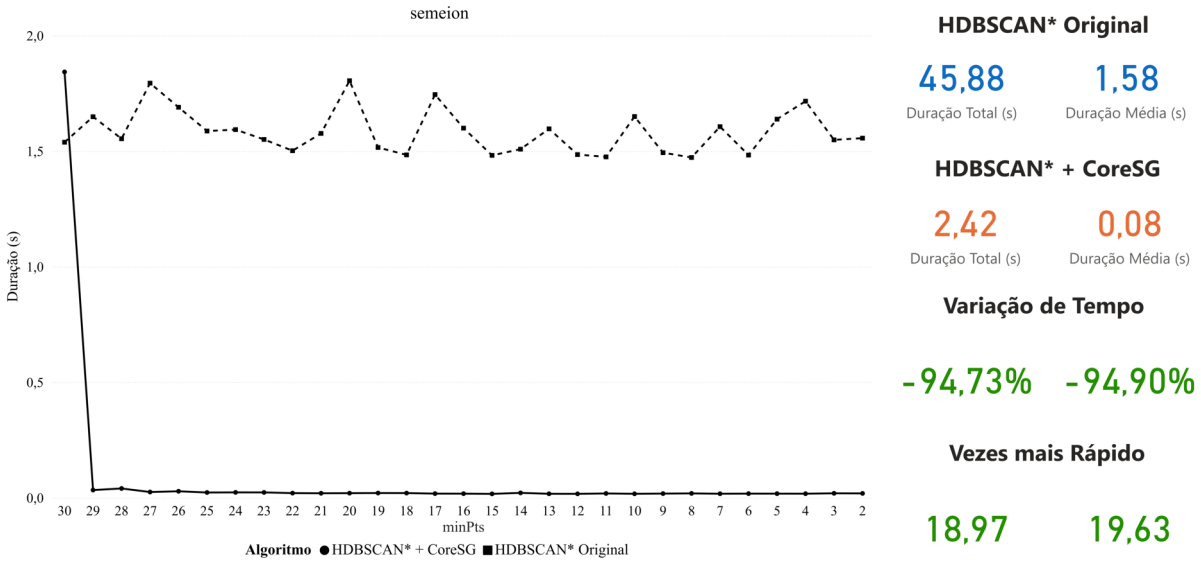
Tabela 3 – Conjuntos de dados utilizados nos experimentos de classificação semi-supervisionada conforme publicação do arcabouço original. Fonte: (GERTRUDES et al., 2019).

de cálculos, considerando a natureza quadrática do HDBSCAN*, e conseqüentemente se beneficiando proporcionalmente mais da redução de arestas trazida pelo CoreSG que o conjunto *yeast_Galactose* e outros conjuntos menores.

A Figura 20 exibe um gráfico contendo um resumo do aumento de velocidade de geração das árvores geradoras mínimas por conjunto de dados quando o CoreSG é utilizado,



(a) yeast_Galactose



(b) semeion

Figura 19 – Painéis de análise exploratória comparando o tempo em segundos necessário para geração de árvores geradoras mínimas entre o HDBSCAN* com CoreSG e o HDBSCAN* original

em comparação ao grafo completo do HDBSCAN* original. No gráfico citado e em nossas análises, o aumento **A** da velocidade de geração de uma AGM (que pode ser igualmente entendido como a redução do tempo necessário para geração de uma AGM) é dado por:

$$A = \frac{1}{(\frac{\bar{T}_{mst\_HDBSCAN^*\_CoreSG}}{\bar{T}_{mst\_HDBSCAN^*}})} \tag{6}$$

onde $\bar{T}_{mst\_HDBSCAN^*\_CoreSG}$ é o tempo médio de geração de uma árvore geradora mínima produzida pelo HDBSCAN* com CoreSG e $\bar{T}_{mst\_HDBSCAN^*}$ é o tempo médio de geração de uma árvore geradora mínima produzida pelo HDBSCAN* original. O aumento de velocidade deve ser interpretado como a quantidade (média) de vezes mais rápido que o HDBSCAN* com CoreSG é capaz de gerar uma AGM quando comparado ao HDBSCAN* original. O gráfico da Figura 20 demonstra novamente que conjuntos de dados maiores tendem a produzir os maiores aumentos, enquanto conjuntos menores tendem a apresentar relativamente pouco aumento de velocidade ou até redução da velocidade, como nos casos dos conjuntos *transplant*, *autoPrice*, e *iris*. Este comportamento pode ser visto de forma mais clara através da Figura 21, onde estão exibidos os mesmos conjuntos de dados em um gráfico de dispersão. O eixo **Y** do gráfico de dispersão representa a quantidade de atributos dos conjuntos de dados, e o eixo **X** representa a quantidade de observações. Os eixos foram propositalmente configurados para crescer de forma exponencial para facilitar a visualização. O tamanho e cor das bolhas representam o aumento da velocidade de geração de árvores geradoras mínimas do HDBSCAN* com o CoreSG com relação ao HDBSCAN* original. O gráfico confirma as observações anteriores e demonstra visualmente que, conforme a quantidade de atributos ou observações de um conjunto de dados aumenta, o CoreSG tende a ser mais eficiente e consequentemente a produzir resultados melhores (com maiores aumentos de velocidade).

Os resultados confirmaram que o grafo CoreSG proposto por Neto et al. (2022) havia sido implementado com sucesso no arcabouço de Gertrudes et al. (2019), atendendo o objetivo principal de pesquisa definido inicialmente. Em seguida, passamos a investigar os múltiplos resultados das partições produzidas a partir de cada nível de m_{pts}.

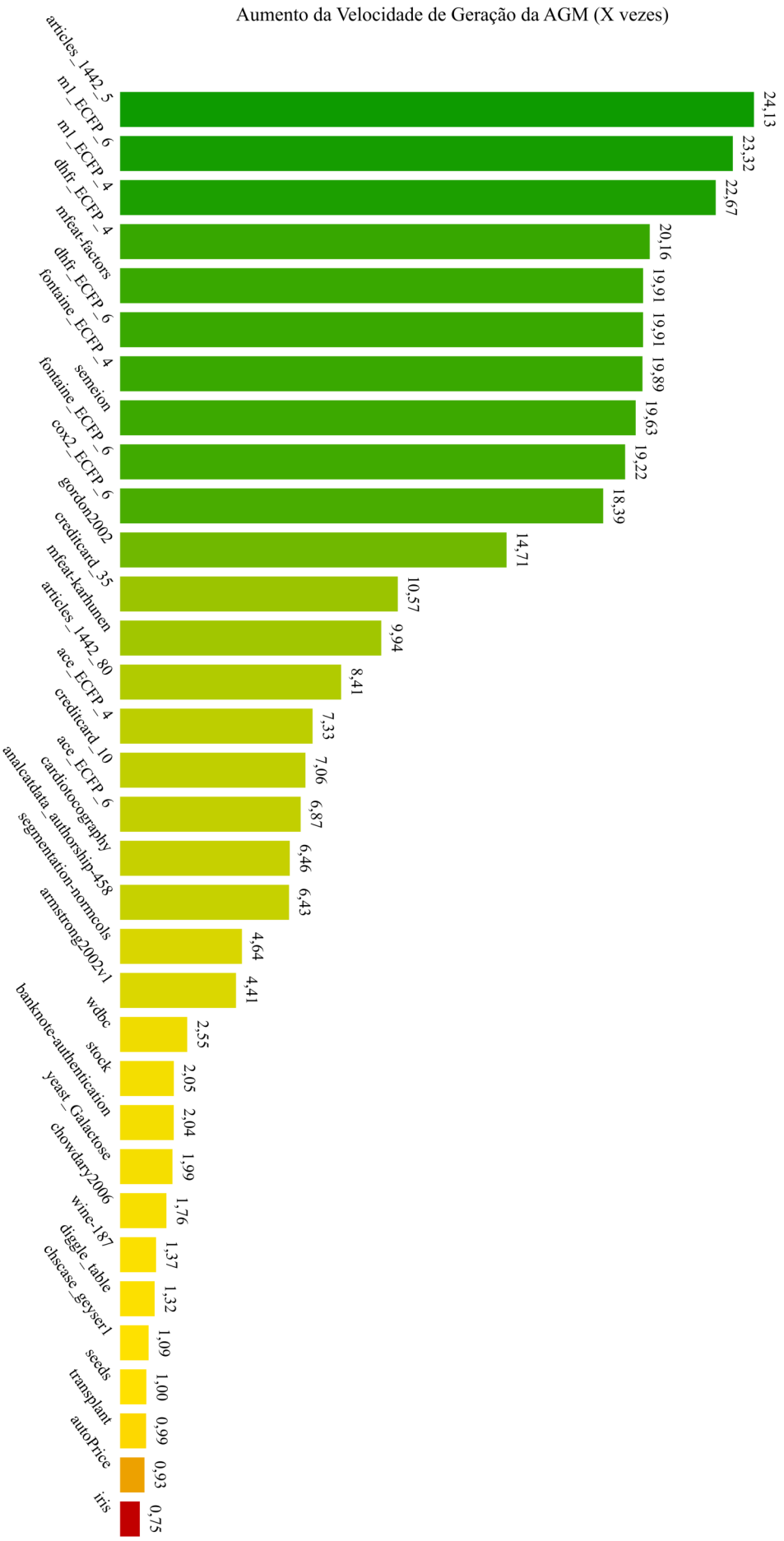


Figura 20 – Aumento da velocidade do tempo médio de geração das árvores geradoras mínimas por conjunto de dados, $m_{pts} \in [2, 30]$

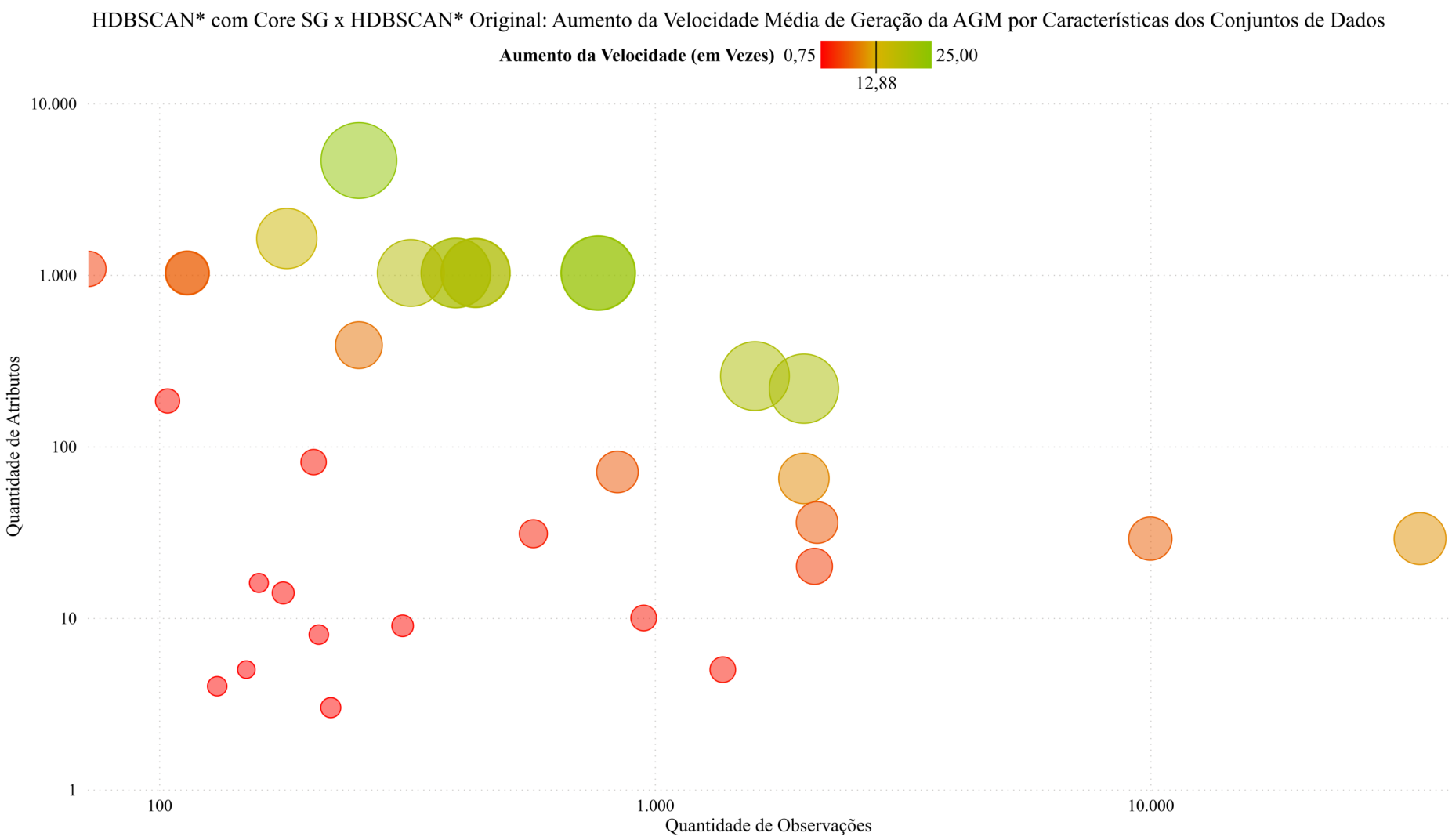


Figura 21 – Aumento da velocidade do tempo médio de geração das árvores geradoras mínimas por características (quantidade de atributos e quantidade de observações) dos conjunto de dados, $m_{pts} \in [2, 30]$

APÊNDICE B

Análise de Particões Planas de Múltiplos Níveis de Densidade

Com o grafo CoreSG devidamente implementado e integrado ao arcabouço unificado e o principal objetivo da proposta inicial de pesquisa atingido, os próximos passos de nossa pesquisa envolveram, naturalmente, a análise da qualidade das partições planas produzidas pelo HDBSCAN*(cd,-) e arcabouço estendido a partir de múltiplos níveis de m_{pts}. Iniciou-se então uma análise exploratória destas partições, sendo selecionados os conjuntos de dados *cardiotocography*, *mfeat-factors*, *mfeat-karhunen*, e *segmentation*, por serem os candidatos da lista de conjuntos de dados utilizados nos experimentos de classificação semissupervisionada da publicação do arcabouço unificado que possuíam uma quantidade mais alta de observações ($n \geq 2.000$). A escolha baseou-se no fato de que amplas faixas de valores de m_{pts} fazem sentido somente em contextos onde as estruturas de densidade são naturalmente mais densas e os objetos mais numerosos. A faixa de valores de m_{pts} foi definida em um primeiro momento como $m_{pts} \in [2, 30]$, assim como os experimentos de desempenho descritos no Apendice A. Para os níveis de rótulos semissupervisionados, foi decidido utilizar os mesmos níveis utilizados nos experimentos de Gertrudes et al. (2019): 2%, 5%, 8%, e 10%. Apesar dos experimentos serem de natureza de classificação e não de agrupamento, a medida de validação escolhida foi o índice externo *Adjusted Rand Index* (ARI), já que o objetivo destes experimentos iniciais era: a) entender a diversidade entre as partições geradas conforme o espectro de m_{pts} utilizado para gerar as partições; e b) investigar quais os níveis de m_{pts} eram responsáveis por gerar os melhores resultados em termos de similaridade com a partição verdade.

A estratégia adotada para a sumarização dos resultados tratou-se de um passo de pós-processamento dos resultados de classificação do HDBSCAN*(cd,-) e CoreSG que fez

uso do próprio HDBSCAN* original para agrupar as partições planas de saída de forma não supervisionada, utilizando o ARI como medida de distância entre as partições. O passo de pós-processamento também foi responsável por calcular o ARI de cada partição de saída (representando cada uma um resultado produzido por um valor diferente de m_{pts}) em comparação com a partição verdade, e uma matriz quadrada de dimensões $m_{pts\_max} - m_{pts\_min} + 1$ contendo os valores dos ARIs representando as comparações entre todas as partições.

Os resultados obtidos após a realização dos experimentos com as configurações citadas foram de certa forma inesperados: A principal descoberta foi que de forma geral, os diferentes níveis de m_{pts} geraram partições muito similares entre si, com diversisade muito baixa, representada por valores do ARI relativamente muito próximos entre todas as partições. Uma segunda descoberta foi que, apesar de todas as partições serem similares, as partições que se mostraram levemente mais similares (com pontuação do ARI levemente maior) à partição verdade foram aquelas geradas em geral por níveis mais baixos de m_{pts}, iguais ou próximos de 2, com redução progressiva leve do ARI conforme o valor de m_{pts} aumenta. A Figura 22 exibe um exemplo deste comportamento conforme os resultados do conjunto *mfeat-factors* com 2% de rótulos semissupervisionados. A figura contém uma matriz de similaridades codificada por cor com os valores do ARI entre as partições produzidas pelos diferentes níveis de m_{pts} e uma lista com os valores ARI das partições em comparação com a partição verdade. A cor verde representa valores do ARI próximos de 1, enquanto a cor amarela representa valores do ARI próximos do mínimo para o experimento em questão $(0,525)$. Através dos valores de ARI é possível concluir que a partição mais próxima da partição verdade é aquela gerada quando $m_{pts} = 2$, sendo que a similaridade com a verdade tende a cair progressivamente conforme m_{pts} aumenta. Além disso, a diferença entre os ARIs das partições produzidas pelos valores extremos $m_{pts} = 2$ e $m_{pts} = 30$ é de somente $(0,240)$.

Conforme citado anteriormente, as partições planas produzidas pelo HDBSCAN*(cd,-) e CoreSG para diferentes níveis de m_{pts} foram agrupadas em um passo de pós-processamento envolvendo o HDBSCAN* não supervisionado e o ARI como medida de distância interna entre as partições. O objetivo desta abordagem foi de detectar e entender os grupos de partições similares produzidas por níveis diferentes de densidade (de acordo com m_{pts}) para, em última instância, traçar uma estratégia para selecionar e combinar as partições. Entretanto, os grupos de partições produzidos pelo HDBSCAN* por meio desta análise exploratória inicial não levaram a nenhuma descoberta interessante: De forma geral, os grupos encontrados seguiram os níveis de m_{pts} adjacentes ou próximos. A Figura 23 ilustra o agrupamento de partições para o mesmo experimento (conjunto de dados *mfeat-factors* e 2% de rótulos semissupervisionados) cujos resultados foram exibidos previamente na Figura 22. Na Figura 23, cada cor representa um grupo encontrado pelo HDBSCAN*, e as linhas de mesma cor representam partições pertencentes ao mesmo grupo. Foram en-

min_pts	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	min_pts	Verdade
2	1,000	0,871	0,865	0,829	0,811	0,793	0,783	0,773	0,768	0,763	0,753	0,755	0,745	0,746	0,727	0,715	0,717	0,721	0,690	0,707	0,710	0,693	0,670	0,656	0,643	0,636	0,638	0,640	0,631	2	0,765
3	0,871	1,000	0,977	0,942	0,907	0,885	0,877	0,863	0,856	0,854	0,841	0,841	0,834	0,833	0,816	0,807	0,808	0,811	0,780	0,797	0,797	0,782	0,760	0,746	0,729	0,724	0,724	0,726	0,716	3	0,721
4	0,865	0,977	1,000	0,955	0,914	0,891	0,883	0,872	0,864	0,859	0,845	0,847	0,835	0,837	0,821	0,809	0,811	0,814	0,787	0,802	0,804	0,791	0,767	0,754	0,736	0,731	0,731	0,731	0,723	4	0,719
5	0,829	0,942	0,955	1,000	0,943	0,924	0,916	0,904	0,898	0,895	0,873	0,881	0,867	0,866	0,852	0,842	0,843	0,833	0,815	0,815	0,819	0,807	0,782	0,769	0,762	0,756	0,757	0,741	0,731	5	0,693
6	0,811	0,907	0,914	0,943	1,000	0,972	0,949	0,934	0,922	0,917	0,888	0,878	0,866	0,871	0,865	0,853	0,853	0,846	0,827	0,818	0,822	0,811	0,791	0,778	0,770	0,765	0,773	0,760	0,748	6	0,689
7	0,793	0,885	0,891	0,924	0,972	1,000	0,971	0,951	0,931	0,919	0,896	0,893	0,882	0,885	0,881	0,868	0,868	0,859	0,843	0,831	0,835	0,823	0,804	0,790	0,787	0,781	0,788	0,773	0,761	7	0,675
8	0,783	0,877	0,883	0,916	0,949	0,971	1,000	0,979	0,956	0,929	0,918	0,915	0,901	0,901	0,894	0,882	0,877	0,868	0,854	0,841	0,841	0,826	0,807	0,793	0,789	0,784	0,790	0,777	0,767	8	0,660
9	0,773	0,863	0,872	0,904	0,934	0,951	0,979	1,000	0,971	0,942	0,927	0,925	0,910	0,908	0,900	0,889	0,880	0,871	0,860	0,848	0,847	0,833	0,814	0,800	0,796	0,788	0,794	0,782	0,771	9	0,655
10	0,768	0,856	0,864	0,898	0,922	0,931	0,956	0,971	1,000	0,961	0,942	0,935	0,917	0,919	0,911	0,901	0,890	0,881	0,867	0,855	0,850	0,835	0,821	0,804	0,802	0,794	0,796	0,786	0,777	10	0,651
11	0,763	0,854	0,859	0,895	0,917	0,919	0,929	0,942	0,961	1,000	0,953	0,938	0,922	0,923	0,914	0,901	0,889	0,882	0,872	0,860	0,856	0,844	0,834	0,819	0,816	0,812	0,807	0,787	0,778	11	0,645
12	0,753	0,841	0,845	0,873	0,888	0,896	0,918	0,927	0,942	0,953	1,000	0,966	0,945	0,954	0,935	0,920	0,902	0,898	0,885	0,880	0,874	0,859	0,849	0,833	0,825	0,820	0,819	0,795	0,786	12	0,631
13	0,755	0,841	0,847	0,881	0,878	0,893	0,915	0,925	0,935	0,938	0,966	1,000	0,972	0,963	0,942	0,924	0,912	0,900	0,896	0,892	0,883	0,866	0,853	0,836	0,836	0,828	0,821	0,802	0,791	13	0,634
14	0,745	0,834	0,835	0,867	0,866	0,882	0,901	0,910	0,917	0,922	0,945	0,972	1,000	0,962	0,942	0,926	0,912	0,899	0,897	0,891	0,880	0,864	0,849	0,832	0,838	0,830	0,819	0,809	0,798	14	0,622
15	0,746	0,833	0,837	0,866	0,871	0,885	0,901	0,908	0,919	0,923	0,954	0,963	0,962	1,000	0,964	0,945	0,930	0,918	0,904	0,902	0,891	0,876	0,862	0,847	0,841	0,836	0,833	0,810	0,799	15	0,623
16	0,727	0,816	0,821	0,852	0,865	0,881	0,894	0,900	0,911	0,914	0,935	0,942	0,942	0,964	1,000	0,963	0,946	0,936	0,915	0,905	0,895	0,879	0,867	0,853	0,846	0,843	0,840	0,815	0,808	16	0,608
17	0,715	0,807	0,809	0,842	0,853	0,868	0,882	0,889	0,901	0,901	0,920	0,924	0,926	0,945	0,963	1,000	0,968	0,949	0,928	0,910	0,898	0,881	0,869	0,856	0,851	0,846	0,846	0,820	0,811	17	0,598
18	0,717	0,808	0,811	0,843	0,853	0,868	0,877	0,880	0,890	0,889	0,902	0,912	0,912	0,930	0,946	0,968	1,000	0,975	0,935	0,918	0,912	0,895	0,875	0,865	0,857	0,853	0,853	0,821	0,811	18	0,599
19	0,721	0,811	0,814	0,833	0,846	0,859	0,868	0,871	0,881	0,882	0,898	0,900	0,899	0,918	0,936	0,949	0,975	1,000	0,935	0,936	0,927	0,909	0,888	0,878	0,856	0,853	0,847	0,822	0,812	19	0,606
20	0,690	0,780	0,787	0,815	0,827	0,843	0,854	0,860	0,867	0,872	0,885	0,896	0,897	0,904	0,915	0,928	0,935	0,935	1,000	0,960	0,915	0,901	0,916	0,900	0,908	0,900	0,865	0,835	0,825	20	0,576
21	0,707	0,797	0,802	0,815	0,818	0,831	0,841	0,848	0,855	0,860	0,880	0,892	0,891	0,902	0,905	0,910	0,918	0,936	0,960	1,000	0,947	0,927	0,942	0,924	0,898	0,889	0,861	0,837	0,826	21	0,591
22	0,710	0,797	0,804	0,819	0,822	0,835	0,841	0,847	0,850	0,856	0,874	0,883	0,880	0,891	0,895	0,898	0,912	0,927	0,915	0,947	1,000	0,975	0,917	0,899	0,873	0,864	0,886	0,829	0,820	22	0,597
23	0,693	0,782	0,791	0,807	0,811	0,823	0,826	0,833	0,835	0,844	0,859	0,866	0,864	0,876	0,879	0,881	0,895	0,909	0,901	0,927	0,975	1,000	0,934	0,912	0,886	0,879	0,905	0,841	0,832	23	0,581
24	0,670	0,760	0,767	0,782	0,791	0,804	0,807	0,814	0,821	0,834	0,849	0,853	0,849	0,862	0,867	0,869	0,875	0,888	0,916	0,942	0,917	0,934	1,000	0,976	0,945	0,935	0,895	0,852	0,843	24	0,558
25	0,656	0,746	0,754	0,769	0,778	0,790	0,793	0,800	0,804	0,819	0,833	0,836	0,832	0,847	0,853	0,856	0,865	0,878	0,900	0,924	0,899	0,912	0,976	1,000	0,955	0,942	0,891	0,857	0,848	25	0,543
26	0,643	0,729	0,736	0,762	0,770	0,787	0,789	0,796	0,802	0,816	0,825	0,836	0,838	0,841	0,846	0,851	0,857	0,856	0,908	0,898	0,873	0,886	0,945	0,955	1,000	0,978	0,906	0,864	0,857	26	0,530
27	0,636	0,724	0,731	0,756	0,765	0,781	0,784	0,788	0,794	0,812	0,820	0,828	0,830	0,836	0,843	0,846	0,853	0,853	0,900	0,889	0,864	0,879	0,935	0,942	0,978	1,000	0,919	0,872	0,869	27	0,523
28	0,638	0,724	0,731	0,757	0,773	0,788	0,790	0,794	0,796	0,807	0,819	0,821	0,819	0,833	0,840	0,846	0,853	0,847	0,865	0,861	0,886	0,905	0,895	0,891	0,906	0,919	1,000	0,876	0,873	28	0,525
29	0,640	0,726	0,731	0,741	0,760	0,773	0,777	0,782	0,786	0,787	0,795	0,802	0,809	0,810	0,815	0,820	0,821	0,822	0,835	0,837	0,829	0,841	0,852	0,857	0,864	0,872	0,876	1,000	0,983	29	0,530
30	0,631	0,716	0,723	0,731	0,748	0,761	0,767	0,771	0,777	0,778	0,786	0,791	0,798	0,799	0,808	0,811	0,811	0,812	0,825	0,826	0,820	0,832	0,843	0,848	0,857	0,869	0,873	0,983	1,000	30	0,525

Figura 22 – Matriz ARI de similaridades entre partições e valores do ARI para cada nível de $m_{pts} \in [2, 30]$ para o conjunto *mfeat-factors* com 2% de rótulos semissupervisionados

contrados os seguintes grupos com relação aos níveis de m_{pts} que geraram as respectivas partições $C_1 = \{3, 4, 5\}, C_2 = \{6, 7, 8, 9, 10, 11\}, C_3 = \{12, 13, 14\}, C_4 = \{15, 16\}, C_5 = \{17, 18, 19\}, C_6 = \{20, 21\}, C_7 = \{22, 23\}, C_8 = \{24, 25\}, C_9 = \{26, 27\}, C_{10} = \{29, 30\}$. Além dos grupos citados, as duas partições referentes a $m_{pts} = 2$ e a $m_{pts} = 28$ (linhas de cor branca na Figura 23) foram reconhecidas como ruído pelo HDBSCAN*. Os demais conjuntos de dados que fizeram parte desta primeira etapa de análise exploratória apresentaram resultados similares, independente do percentual de rótulos semissupervisionados fornecidos ao HDBSCAN*(cd,-) com CoreSG.

Face aos resultados, e com o objetivo de investigar melhor o comportamento constatado, os experimentos da análise exploratória descritos foram então expandidos para incluir mais conjuntos de dados. Para este fim foram realizadas buscas na literatura por conjuntos de dados com características que aumentassem a probabilidade de obtenção de resultados interessantes, como aqueles com um n grande, aqueles com existência de grupos densos, e aqueles contendo grupos em presença de ruído. Encontramos certa dificuldade em encontrar conjuntos de dados reais disponíveis publicamente que atendessem as necessidades citadas e que ao mesmo tempo possuíssem as classes ou grupos verdadeiros disponíveis. Desta forma, para a segunda rodada de experimentos escolhemos 13 conjuntos sintéticos disponibilizados publicamente por Fränti e Sieranoja (2018), listados na Tabela 4. Novos experimentos foram então executados para os treze conjuntos com os mesmos limites definidos e utilizados nos experimentos anteriores para m_{pts} ($m_{pts} \in [2, 30]$) e percentuais de rótulos semissupervisionados (2%, 5%, 8%, 10%). Os resultados destes novos experimentos foram também agrupados pelo HDBSCAN* não supervisionado com o uso do ARI como medida de distância entre as partições.

Tabela 4 – Conjuntos de dados sintéticos utilizados na análise exploratória de partições de múltiplos níveis de densidade

Conjunto de Dados	n	# Atrib.	# Classes
asymmetric	1.000	2	5
birch1	100.000	2	100
Compound	399	2	6
g2-2-10	2.048	2	2
jain	373	2	2
overlap	1.000	2	6
s1	5.000	2	15
s2	5.000	2	15
s3	5.000	2	15
s4	5.000	2	15
skewed	1.000	2	6
unbalance2	6.500	2	8
worms_2d	105.600	2	35

Os novos resultados obtidos não se mostraram fundamentalmente diferentes dos re-

min_pts	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
2	1,000	0,871	0,865	0,829	0,811	0,793	0,783	0,773	0,768	0,763	0,753	0,755	0,745	0,746	0,727	0,715	0,717	0,721	0,690	0,707	0,710	0,693	0,670	0,656	0,643	0,636	0,638	0,640	0,631
3	0,871	1,000	0,977	0,942	0,907	0,885	0,877	0,863	0,856	0,854	0,841	0,841	0,834	0,833	0,816	0,807	0,808	0,811	0,780	0,797	0,797	0,782	0,760	0,746	0,729	0,724	0,724	0,726	0,716
4	0,865	0,977	1,000	0,955	0,914	0,891	0,883	0,872	0,864	0,859	0,845	0,847	0,835	0,837	0,821	0,809	0,811	0,814	0,787	0,802	0,804	0,791	0,767	0,754	0,736	0,731	0,731	0,731	0,723
5	0,829	0,942	0,955	1,000	0,943	0,924	0,916	0,904	0,898	0,895	0,873	0,881	0,867	0,866	0,852	0,842	0,843	0,833	0,815	0,815	0,819	0,807	0,782	0,769	0,762	0,756	0,757	0,741	0,731
6	0,811	0,907	0,914	0,943	1,000	0,972	0,949	0,934	0,922	0,917	0,888	0,878	0,866	0,871	0,865	0,853	0,853	0,846	0,827	0,818	0,822	0,811	0,791	0,778	0,770	0,765	0,773	0,760	0,748
7	0,793	0,885	0,891	0,924	0,972	1,000	0,971	0,951	0,931	0,919	0,896	0,893	0,882	0,885	0,881	0,868	0,868	0,859	0,843	0,831	0,835	0,823	0,804	0,790	0,787	0,781	0,788	0,773	0,761
8	0,783	0,877	0,883	0,916	0,949	0,971	1,000	0,979	0,956	0,929	0,918	0,915	0,901	0,901	0,894	0,882	0,877	0,868	0,854	0,841	0,841	0,826	0,807	0,793	0,789	0,784	0,790	0,777	0,767
9	0,773	0,863	0,872	0,904	0,934	0,951	0,979	1,000	0,971	0,942	0,927	0,925	0,910	0,908	0,900	0,889	0,880	0,871	0,860	0,848	0,847	0,833	0,814	0,800	0,796	0,788	0,794	0,782	0,771
10	0,768	0,856	0,864	0,898	0,922	0,931	0,956	0,971	1,000	0,961	0,942	0,935	0,917	0,919	0,911	0,901	0,890	0,881	0,867	0,855	0,850	0,835	0,821	0,804	0,802	0,794	0,796	0,786	0,777
11	0,763	0,854	0,859	0,895	0,917	0,919	0,929	0,942	0,961	1,000	0,953	0,938	0,922	0,923	0,914	0,901	0,889	0,882	0,872	0,860	0,856	0,844	0,834	0,819	0,816	0,812	0,807	0,787	0,778
12	0,753	0,841	0,845	0,873	0,888	0,896	0,918	0,927	0,942	0,953	1,000	0,966	0,945	0,954	0,935	0,920	0,902	0,898	0,885	0,880	0,874	0,859	0,849	0,833	0,825	0,820	0,819	0,795	0,786
13	0,755	0,841	0,847	0,881	0,878	0,893	0,915	0,925	0,935	0,938	0,966	1,000	0,972	0,963	0,942	0,924	0,912	0,900	0,896	0,892	0,883	0,866	0,853	0,836	0,836	0,828	0,821	0,802	0,791
14	0,745	0,834	0,835	0,867	0,866	0,882	0,901	0,910	0,917	0,922	0,945	0,972	1,000	0,962	0,942	0,926	0,912	0,899	0,897	0,891	0,880	0,864	0,849	0,832	0,838	0,830	0,819	0,809	0,798
15	0,746	0,833	0,837	0,866	0,871	0,885	0,901	0,908	0,919	0,923	0,954	0,963	0,962	1,000	0,964	0,945	0,930	0,918	0,904	0,902	0,891	0,876	0,862	0,847	0,841	0,836	0,833	0,810	0,799
16	0,727	0,816	0,821	0,852	0,865	0,881	0,894	0,900	0,911	0,914	0,935	0,942	0,942	0,964	1,000	0,963	0,946	0,936	0,915	0,905	0,895	0,879	0,867	0,853	0,846	0,843	0,840	0,815	0,808
17	0,715	0,807	0,809	0,842	0,853	0,868	0,882	0,889	0,901	0,901	0,920	0,924	0,926	0,945	0,963	1,000	0,968	0,949	0,928	0,910	0,898	0,881	0,869	0,856	0,851	0,846	0,846	0,820	0,811
18	0,717	0,808	0,811	0,843	0,853	0,868	0,877	0,880	0,890	0,889	0,902	0,912	0,912	0,930	0,946	0,968	1,000	0,975	0,935	0,918	0,912	0,895	0,875	0,865	0,857	0,853	0,853	0,821	0,811
19	0,721	0,811	0,814	0,833	0,846	0,859	0,868	0,871	0,881	0,882	0,898	0,900	0,899	0,918	0,936	0,949	0,975	1,000	0,935	0,936	0,927	0,909	0,888	0,878	0,856	0,853	0,847	0,822	0,812
20	0,690	0,780	0,787	0,815	0,827	0,843	0,854	0,860	0,867	0,872	0,885	0,896	0,897	0,904	0,915	0,928	0,935	0,935	1,000	0,960	0,915	0,901	0,916	0,900	0,908	0,900	0,865	0,835	0,825
21	0,707	0,797	0,802	0,815	0,818	0,831	0,841	0,848	0,855	0,860	0,880	0,892	0,891	0,902	0,905	0,910	0,918	0,936	0,960	1,000	0,947	0,927	0,942	0,924	0,898	0,889	0,861	0,837	0,826
22	0,710	0,797	0,804	0,819	0,822	0,835	0,841	0,847	0,850	0,856	0,874	0,883	0,880	0,891	0,895	0,898	0,912	0,927	0,915	0,947	1,000	0,975	0,917	0,899	0,873	0,864	0,886	0,829	0,820
23	0,693	0,782	0,791	0,807	0,811	0,823	0,826	0,833	0,835	0,844	0,859	0,866	0,864	0,876	0,879	0,881	0,895	0,909	0,901	0,927	0,975	1,000	0,934	0,912	0,886	0,879	0,905	0,841	0,832
24	0,670	0,760	0,767	0,782	0,791	0,804	0,807	0,814	0,821	0,834	0,849	0,853	0,849	0,862	0,867	0,869	0,875	0,888	0,916	0,942	0,917	0,934	1,000	0,976	0,945	0,935	0,895	0,852	0,843
25	0,656	0,746	0,754	0,769	0,778	0,790	0,793	0,800	0,804	0,819	0,833	0,836	0,832	0,847	0,853	0,856	0,865	0,878	0,900	0,924	0,899	0,912	0,976	1,000	0,955	0,942	0,891	0,857	0,848
26	0,643	0,729	0,736	0,762	0,770	0,787	0,789	0,796	0,802	0,816	0,825	0,836	0,838	0,841	0,846	0,851	0,857	0,856	0,908	0,898	0,873	0,886	0,945	0,955	1,000	0,978	0,906	0,864	0,857
27	0,636	0,724	0,731	0,756	0,765	0,781	0,784	0,788	0,794	0,812	0,820	0,828	0,830	0,836	0,843	0,846	0,853	0,853	0,900	0,889	0,864	0,879	0,935	0,942	0,978	1,000	0,919	0,872	0,869
28	0,638	0,724	0,731	0,757	0,773	0,788	0,790	0,794	0,796	0,807	0,819	0,821	0,819	0,833	0,840	0,846	0,853	0,847	0,865	0,861	0,886	0,905	0,895	0,891	0,906	0,919	1,000	0,876	0,873
29	0,640	0,726	0,731	0,741	0,760	0,773	0,777	0,782	0,786	0,787	0,795	0,802	0,809	0,810	0,815	0,820	0,821	0,822	0,835	0,837	0,829	0,841	0,852	0,857	0,864	0,872	0,876	1,000	0,983
30	0,631	0,716	0,723	0,731	0,748	0,761	0,767	0,771	0,777	0,778	0,786	0,791	0,798	0,799	0,808	0,811	0,811	0,812	0,825	0,826	0,820	0,832	0,843	0,848	0,857	0,869	0,873	0,983	1,000

Figura 23 – Matriz ARI de similaridades entre partições com os grupos de partições codificadas por cor para o conjunto *mfeat-factors* com 2% de rótulos semissupervisionados

sultados iniciais, discutidos anteriormente. Grande parte dos treze conjuntos de dados explorados apresentou um ou ambos dos seguintes comportamentos: a) as partições responsáveis pelos melhores resultados de ARI foram aquelas produzidas com $m_{pts} = 2$ ou $m_{pts} = 3$; b) a diferença entre o menor ARI e o maior ARI produzidos por todas as partições do espectro inteiro de $m_{pts} \in [2, 30]$ se mostrou extremamente baixa. As Figuras 24, 25, 26, e 27 mostram os resultados do ARI de forma similar à Figura 22 para os conjuntos de dados *asymmetric*, *jain*, *overlap*, e *worms_2d*, respectivamente. Observando o comportamento específico dos conjuntos citados, é possível afirmar que: a) o conjunto *asymmetric* (Figura 24) produziu partições mais próximas da partição verdade quando $m_{pts} = 2$, e a amplitude de ARI para todas as partições do espectro $m_{pts} \in [2, 30]$ foi de $0,985 - 0,960 = 0,025$; b) o conjunto *jain* (Figura 25) produziu partições perfeitas quando $m_{pts} \in \{2, 3, 4, 5, 6\}$, com leve redução progressiva da similaridade conforme o aumento de m_{pts}, até aproximadamente o nível $m_{pts} = 20$, sendo que à partir de $m_{pts} = 21$ o ARI caiu bruscamente; c) o conjunto overlap (Figura 26) produziu a partição mais próxima da verdade quando $m_{pts} = 19$, entretanto a diferença entre o maior e menor ARI foi somente $0,502 - 0,371 = 0,131$; e d) o conjunto *worms_2d* (Figura 27) produziu a partição de melhor qualidade quando $m_{pts} = 17$, contudo com uma amplitude do ARI extremamente baixa de $0,486 - 0,454 = 0,032$. Os demais 9 conjuntos de dados (omitidos por questões de espaço) apresentaram comportamento igual ou similar aos 4 exemplos citados. Além disso, o percentual de rótulos semissupervisionados fornecidos ao HDBSCAN* e CoreSG não afetou de forma significativa a diversidade das partições dentro do espectro de densidade de $m_{pts} \in [2, 30]$.

Em resumo, os resultados dos novos experimentos se mostraram insatisfatórios para os objetivos de nossa pesquisa, devido aos dois motivos introduzidos no parágrafo anterior: Em primeiro lugar, em cerca de metade dos conjuntos de dados, as partições responsáveis por produzir os melhores resultados foram aquelas geradas com $m_{pts} = 2$ ou $m_{pts} = 3$. Este comportamento significa que um processo computacionalmente mais complexo de geração e seleção de partições com base em um intervalo de m_{pts} não seria justificado se os melhores resultados já poderiam ser obtidos através de uma única execução do HDBSCAN* com o menor valor possível do parâmetro m_{pts}. Em segundo lugar, para a maior parte dos conjuntos de dados (exceto os conjuntos *Compound*, *jain*, *overlap*, e *skewed*), a diferença entre o maior e o menor ARI de todas as partições geradas pelo intervalo $m_{pts} \in [2, 30]$ ficou abaixo de $0,100$. E mesmo para os casos onde esta diferença se mostrou maior, dois dos casos (conjuntos *jain* e *skewed*) apresentaram queda brusca do ARI em um dado nível de m_{pts}, mas mesmo assim geraram partições muito próximas nos níveis de m_{pts} anteriores à queda. Objetivamente, a melhoria em similaridade com relação à partição verdade introduzida não seria suficiente para justificar a maior complexidade computacional do processo de geração de múltiplas partições e posterior seleção da partição de melhor qualidade.

min_pts	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	min_pts	ARI
2	1,000	0,993	0,990	0,987	0,987	0,990	0,990	0,987	0,990	0,995	0,992	0,987	0,982	0,975	0,972	0,972	0,970	0,972	0,972	0,972	0,977	0,975	0,972	0,972	0,975	0,977	0,977	0,977	0,970	2	0,985
3	0,993	1,000	0,997	0,990	0,985	0,987	0,987	0,990	0,987	0,992	0,990	0,990	0,985	0,977	0,975	0,975	0,972	0,975	0,975	0,975	0,975	0,972	0,970	0,970	0,972	0,975	0,975	0,975	0,973	3	0,983
4	0,990	0,997	1,000	0,992	0,987	0,990	0,990	0,992	0,990	0,990	0,992	0,992	0,987	0,980	0,977	0,977	0,975	0,977	0,977	0,977	0,977	0,975	0,972	0,972	0,975	0,977	0,977	0,977	0,975	4	0,980
5	0,987	0,990	0,992	1,000	0,995	0,993	0,998	0,995	0,998	0,992	0,995	0,995	0,990	0,982	0,980	0,980	0,977	0,980	0,980	0,980	0,985	0,983	0,980	0,980	0,983	0,985	0,985	0,985	0,978	5	0,983
6	0,987	0,985	0,987	0,995	1,000	0,997	0,998	0,995	0,998	0,992	0,995	0,995	0,990	0,982	0,985	0,985	0,982	0,985	0,985	0,985	0,990	0,987	0,985	0,985	0,987	0,990	0,990	0,990	0,982	6	0,978
7	0,990	0,987	0,990	0,993	0,997	1,000	0,995	0,992	0,995	0,995	0,998	0,992	0,987	0,980	0,982	0,982	0,980	0,982	0,982	0,982	0,987	0,985	0,982	0,982	0,985	0,987	0,987	0,987	0,980	7	0,975
8	0,990	0,987	0,990	0,998	0,998	0,995	1,000	0,997	1,000	0,995	0,997	0,997	0,992	0,985	0,982	0,982	0,980	0,982	0,982	0,982	0,987	0,985	0,982	0,982	0,985	0,987	0,987	0,987	0,980	8	0,980
9	0,987	0,990	0,992	0,995	0,995	0,992	0,997	1,000	0,997	0,992	0,995	1,000	0,995	0,987	0,985	0,985	0,982	0,985	0,985	0,985	0,985	0,982	0,980	0,980	0,982	0,985	0,985	0,985	0,983	9	0,978
10	0,990	0,987	0,990	0,998	0,998	0,995	1,000	0,997	1,000	0,995	0,997	0,997	0,992	0,985	0,982	0,982	0,980	0,982	0,982	0,982	0,987	0,985	0,982	0,982	0,985	0,987	0,987	0,987	0,980	10	0,980
11	0,995	0,992	0,990	0,992	0,992	0,995	0,995	0,992	0,995	1,000	0,997	0,992	0,987	0,980	0,977	0,977	0,975	0,977	0,977	0,977	0,982	0,980	0,977	0,977	0,980	0,982	0,982	0,982	0,975	11	0,980
12	0,992	0,990	0,992	0,995	0,995	0,998	0,997	0,995	0,997	0,997	1,000	0,995	0,990	0,982	0,980	0,980	0,977	0,980	0,980	0,980	0,985	0,982	0,980	0,980	0,982	0,985	0,985	0,985	0,977	12	0,978
13	0,987	0,990	0,992	0,995	0,995	0,992	0,997	1,000	0,997	0,992	0,995	1,000	0,995	0,987	0,985	0,985	0,982	0,985	0,985	0,985	0,985	0,982	0,980	0,980	0,982	0,985	0,985	0,985	0,983	13	0,978
14	0,982	0,985	0,987	0,990	0,990	0,987	0,992	0,995	0,992	0,987	0,990	0,995	1,000	0,992	0,990	0,990	0,987	0,985	0,985	0,985	0,985	0,983	0,980	0,980	0,983	0,985	0,985	0,985	0,983	14	0,973
15	0,975	0,977	0,980	0,982	0,982	0,980	0,985	0,987	0,985	0,980	0,982	0,987	0,992	1,000	0,998	0,998	0,995	0,993	0,993	0,993	0,987	0,985	0,988	0,988	0,985	0,987	0,987	0,987	0,985	15	0,965
16	0,972	0,975	0,977	0,980	0,985	0,982	0,982	0,985	0,982	0,977	0,980	0,985	0,990	0,998	1,000	1,000	0,997	0,995	0,995	0,995	0,990	0,987	0,990	0,990	0,987	0,990	0,990	0,990	0,988	16	0,963
17	0,972	0,975	0,977	0,980	0,985	0,982	0,982	0,985	0,982	0,977	0,980	0,985	0,990	0,998	1,000	1,000	0,997	0,995	0,995	0,995	0,990	0,987	0,990	0,990	0,987	0,990	0,990	0,990	0,988	17	0,963
18	0,970	0,972	0,975	0,977	0,982	0,980	0,980	0,982	0,980	0,975	0,977	0,982	0,987	0,995	0,997	0,997	1,000	0,992	0,992	0,992	0,987	0,985	0,987	0,987	0,985	0,987	0,987	0,987	0,985	18	0,960
19	0,972	0,975	0,977	0,980	0,985	0,982	0,982	0,985	0,982	0,977	0,980	0,985	0,985	0,993	0,995	0,995	0,992	1,000	1,000	1,000	0,995	0,992	0,995	0,995	0,992	0,995	0,995	0,995	0,993	19	0,963
20	0,972	0,975	0,977	0,980	0,985	0,982	0,982	0,985	0,982	0,977	0,980	0,985	0,985	0,993	0,995	0,995	0,992	1,000	1,000	1,000	0,995	0,992	0,995	0,995	0,992	0,995	0,995	0,995	0,993	20	0,963
21	0,972	0,975	0,977	0,980	0,985	0,982	0,982	0,985	0,982	0,977	0,980	0,985	0,985	0,993	0,995	0,995	0,992	1,000	1,000	1,000	0,995	0,992	0,995	0,995	0,992	0,995	0,995	0,995	0,993	21	0,963
22	0,977	0,975	0,977	0,985	0,990	0,987	0,987	0,985	0,987	0,982	0,985	0,985	0,985	0,987	0,990	0,990	0,987	0,995	0,995	0,995	1,000	0,998	0,995	0,995	0,998	1,000	1,000	1,000	0,993	22	0,968
23	0,975	0,972	0,975	0,983	0,987	0,985	0,985	0,982	0,985	0,980	0,982	0,982	0,983	0,985	0,987	0,987	0,985	0,992	0,992	0,992	0,998	1,000	0,997	0,993	0,995	0,998	0,998	0,998	0,990	23	0,966
24	0,972	0,970	0,972	0,980	0,985	0,982	0,982	0,980	0,982	0,977	0,980	0,980	0,980	0,988	0,990	0,990	0,987	0,995	0,995	0,995	0,995	0,997	1,000	0,995	0,993	0,995	0,995	0,995	0,988	24	0,963
25	0,972	0,970	0,972	0,980	0,985	0,982	0,982	0,980	0,982	0,977	0,980	0,980	0,980	0,988	0,990	0,990	0,987	0,995	0,995	0,995	0,995	0,993	0,995	1,000	0,993	0,995	0,995	0,995	0,988	25	0,963
26	0,975	0,972	0,975	0,983	0,987	0,985	0,985	0,982	0,985	0,980	0,982	0,982	0,983	0,985	0,987	0,987	0,985	0,992	0,992	0,992	0,998	0,995	0,993	0,993	1,000	0,998	0,998	0,998	0,990	26	0,966
27	0,977	0,975	0,977	0,985	0,990	0,987	0,987	0,985	0,987	0,982	0,985	0,985	0,985	0,987	0,990	0,990	0,987	0,995	0,995	0,995	1,000	0,998	0,995	0,995	0,998	1,000	1,000	1,000	0,993	27	0,968
28	0,977	0,975	0,977	0,985	0,990	0,987	0,987	0,985	0,987	0,982	0,985	0,985	0,985	0,987	0,990	0,990	0,987	0,995	0,995	0,995	1,000	0,998	0,995	0,995	0,998	1,000	1,000	1,000	0,993	28	0,968
29	0,977	0,975	0,977	0,985	0,990	0,987	0,987	0,985	0,987	0,982	0,985	0,985	0,985	0,987	0,990	0,990	0,987	0,995	0,995	0,995	1,000	0,998	0,995	0,995	0,998	1,000	1,000	1,000	0,993	29	0,968
30	0,970	0,973	0,975	0,978	0,982	0,980	0,980	0,983	0,980	0,975	0,977	0,983	0,983	0,985	0,988	0,988	0,985	0,993	0,993	0,993	0,993	0,990	0,988	0,988	0,990	0,993	0,993	0,993	1,000	30	0,961

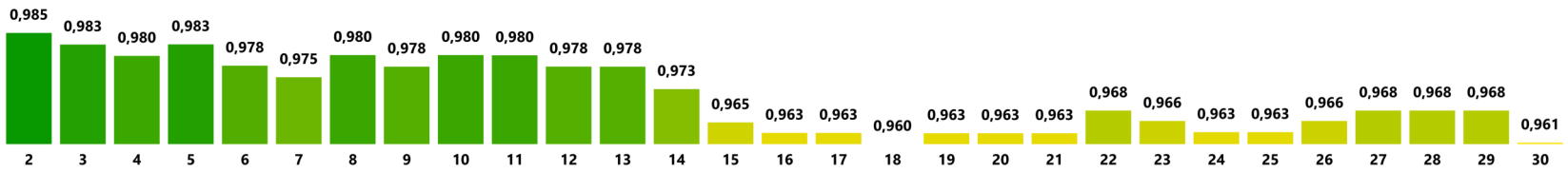


Figura 24 – Matriz ARI de similaridades entre partições e valores do ARI para cada nível de $m_{pts} \in [2, 30]$ para o conjunto *asymmetric* com 2% de rótulos semissupervisionados

min_pts	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	min_pts	ARI
2	1,000	1,000	1,000	1,000	1,000	0,989	0,977	0,989	0,977	0,989	0,977	0,966	0,910	0,866	0,366	0,822	0,406	0,406	0,779	0,069	0,118	0,020	0,020	0,020	0,020	0,020	0,020	0,020	0,020	2	1,000
3	1,000	1,000	1,000	1,000	1,000	0,989	0,977	0,989	0,977	0,989	0,977	0,966	0,910	0,866	0,366	0,822	0,406	0,406	0,779	0,069	0,118	0,020	0,020	0,020	0,020	0,020	0,020	0,020	0,020	3	1,000
4	1,000	1,000	1,000	1,000	1,000	0,989	0,977	0,989	0,977	0,989	0,977	0,966	0,910	0,866	0,366	0,822	0,406	0,406	0,779	0,069	0,118	0,020	0,020	0,020	0,020	0,020	0,020	0,020	0,020	4	1,000
5	1,000	1,000	1,000	1,000	1,000	0,989	0,977	0,989	0,977	0,989	0,977	0,966	0,910	0,866	0,366	0,822	0,406	0,406	0,779	0,069	0,118	0,020	0,020	0,020	0,020	0,020	0,020	0,020	0,020	5	1,000
6	1,000	1,000	1,000	1,000	1,000	0,989	0,977	0,989	0,977	0,989	0,977	0,966	0,910	0,866	0,366	0,822	0,406	0,406	0,779	0,069	0,118	0,020	0,020	0,020	0,020	0,020	0,020	0,020	0,020	6	1,000
7	0,989	0,989	0,989	0,989	0,989	1,000	0,989	1,000	0,989	1,000	0,989	0,977	0,921	0,877	0,372	0,833	0,412	0,412	0,789	0,070	0,121	0,020	0,020	0,020	0,020	0,020	0,020	0,020	0,020	7	0,989
8	0,977	0,977	0,977	0,977	0,977	0,989	1,000	0,989	1,000	0,989	1,000	0,989	0,932	0,887	0,378	0,843	0,419	0,419	0,799	0,072	0,123	0,021	0,021	0,021	0,021	0,021	0,021	0,021	0,021	8	0,977
9	0,989	0,989	0,989	0,989	0,989	1,000	0,989	1,000	0,989	1,000	0,989	0,977	0,921	0,877	0,372	0,833	0,412	0,412	0,789	0,070	0,121	0,020	0,020	0,020	0,020	0,020	0,020	0,020	0,020	9	0,989
10	0,977	0,977	0,977	0,977	0,977	0,989	1,000	0,989	1,000	0,989	1,000	0,989	0,932	0,887	0,378	0,843	0,419	0,419	0,799	0,072	0,123	0,021	0,021	0,021	0,021	0,021	0,021	0,021	0,021	10	0,977
11	0,989	0,989	0,989	0,989	0,989	1,000	0,989	1,000	0,989	1,000	0,989	0,977	0,921	0,877	0,372	0,833	0,412	0,412	0,789	0,070	0,121	0,020	0,020	0,020	0,020	0,020	0,020	0,020	0,020	11	0,989
12	0,977	0,977	0,977	0,977	0,977	0,989	1,000	0,989	1,000	0,989	1,000	0,989	0,932	0,887	0,378	0,843	0,419	0,419	0,799	0,072	0,123	0,021	0,021	0,021	0,021	0,021	0,021	0,021	0,021	12	0,977
13	0,966	0,966	0,966	0,966	0,966	0,977	0,989	0,977	0,989	0,977	0,989	1,000	0,943	0,898	0,384	0,832	0,408	0,408	0,788	0,073	0,125	0,021	0,021	0,021	0,021	0,021	0,021	0,021	0,021	13	0,966
14	0,910	0,910	0,910	0,910	0,910	0,921	0,932	0,921	0,932	0,921	0,932	0,943	1,000	0,931	0,399	0,863	0,443	0,443	0,818	0,081	0,138	0,023	0,023	0,023	0,023	0,023	0,023	0,023	0,023	14	0,910
15	0,866	0,866	0,866	0,866	0,866	0,877	0,887	0,877	0,887	0,877	0,887	0,898	0,931	1,000	0,446	0,907	0,472	0,472	0,839	0,088	0,150	0,025	0,025	0,025	0,025	0,025	0,025	0,025	0,025	15	0,866
16	0,366	0,366	0,366	0,366	0,366	0,372	0,378	0,372	0,378	0,372	0,378	0,384	0,399	0,446	1,000	0,415	0,795	0,795	0,403	0,269	0,428	0,084	0,084	0,084	0,084	0,084	0,084	0,084	0,084	16	0,366
17	0,822	0,822	0,822	0,822	0,822	0,833	0,843	0,833	0,843	0,833	0,843	0,832	0,863	0,907	0,415	1,000	0,524	0,524	0,906	0,096	0,162	0,028	0,028	0,028	0,028	0,028	0,028	0,028	0,028	17	0,822
18	0,406	0,406	0,406	0,406	0,406	0,412	0,419	0,412	0,419	0,412	0,419	0,408	0,443	0,472	0,795	0,524	1,000	0,966	0,536	0,240	0,386	0,074	0,074	0,074	0,074	0,074	0,074	0,074	0,074	18	0,406
19	0,406	0,406	0,406	0,406	0,406	0,412	0,419	0,412	0,419	0,412	0,419	0,408	0,443	0,472	0,795	0,524	0,966	1,000	0,536	0,240	0,386	0,074	0,074	0,074	0,074	0,074	0,074	0,074	0,074	19	0,406
20	0,779	0,779	0,779	0,779	0,779	0,789	0,799	0,789	0,799	0,789	0,799	0,788	0,818	0,839	0,403	0,906	0,536	0,536	1,000	0,104	0,175	0,030	0,030	0,030	0,030	0,030	0,030	0,030	0,030	20	0,779
21	0,069	0,069	0,069	0,069	0,069	0,070	0,072	0,070	0,072	0,070	0,072	0,073	0,081	0,088	0,269	0,096	0,240	0,240	0,104	1,000	0,720	0,434	0,434	0,434	0,434	0,434	0,434	0,434	0,434	21	0,069
22	0,118	0,118	0,118	0,118	0,118	0,121	0,123	0,121	0,123	0,121	0,123	0,125	0,138	0,150	0,428	0,162	0,386	0,386	0,175	0,720	1,000	0,271	0,271	0,271	0,271	0,271	0,271	0,271	0,271	22	0,118
23	0,020	0,020	0,020	0,020	0,020	0,020	0,021	0,020	0,021	0,020	0,021	0,021	0,023	0,025	0,084	0,028	0,074	0,074	0,030	0,434	0,271	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	23	0,020
24	0,020	0,020	0,020	0,020	0,020	0,020	0,021	0,020	0,021	0,020	0,021	0,021	0,023	0,025	0,084	0,028	0,074	0,074	0,030	0,434	0,271	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	24	0,020
25	0,020	0,020	0,020	0,020	0,020	0,020	0,021	0,020	0,021	0,020	0,021	0,021	0,023	0,025	0,084	0,028	0,074	0,074	0,030	0,434	0,271	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	25	0,020
26	0,020	0,020	0,020	0,020	0,020	0,020	0,021	0,020	0,021	0,020	0,021	0,021	0,023	0,025	0,084	0,028	0,074	0,074	0,030	0,434	0,271	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	26	0,020
27	0,020	0,020	0,020	0,020	0,020	0,020	0,021	0,020	0,021	0,020	0,021	0,021	0,023	0,025	0,084	0,028	0,074	0,074	0,030	0,434	0,271	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	27	0,020
28	0,020	0,020	0,020	0,020	0,020	0,020	0,021	0,020	0,021	0,020	0,021	0,021	0,023	0,025	0,084	0,028	0,074	0,074	0,030	0,434	0,271	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	28	0,020
29	0,020	0,020	0,020	0,020	0,020	0,020	0,021	0,020	0,021	0,020	0,021	0,021	0,023	0,025	0,084	0,028	0,074	0,074	0,030	0,434	0,271	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	29	0,020
30	0,020	0,020	0,020	0,020	0,020	0,020	0,021	0,020	0,021	0,020	0,021	0,021	0,023	0,025	0,084	0,028	0,074	0,074	0,030	0,434	0,271	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	30	0,020

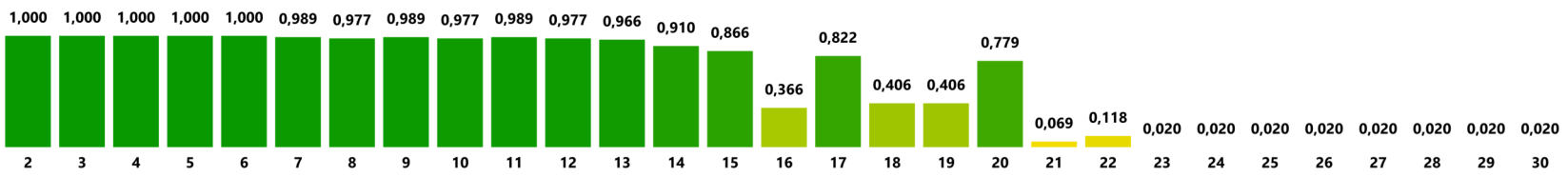


Figura 25 – Matriz ARI de similaridades entre partições e valores do ARI para cada nível de $m_{pts} \in [2, 30]$ para o conjunto *jain* com 2% de rótulos semissupervisionados

min_pts	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
2	1,000	0,989	0,952	0,927	0,947	0,920	0,926	0,913	0,909	0,907	0,915	0,908	0,926	0,916	0,920	0,892	0,886	0,813	0,890	0,885	0,878	0,867	0,794	0,788	0,853	0,731	0,845	0,834	0,846
3	0,989	1,000	0,954	0,932	0,948	0,926	0,931	0,918	0,915	0,913	0,915	0,909	0,926	0,921	0,922	0,893	0,892	0,817	0,890	0,881	0,883	0,869	0,799	0,793	0,855	0,733	0,846	0,835	0,851
4	0,952	0,954	1,000	0,956	0,985	0,952	0,934	0,946	0,946	0,930	0,932	0,928	0,947	0,946	0,951	0,924	0,924	0,844	0,922	0,917	0,915	0,904	0,833	0,827	0,893	0,764	0,884	0,856	0,883
5	0,927	0,932	0,956	1,000	0,951	0,934	0,916	0,923	0,930	0,914	0,944	0,931	0,935	0,935	0,935	0,927	0,917	0,829	0,909	0,895	0,897	0,883	0,814	0,805	0,869	0,761	0,873	0,850	0,874
6	0,947	0,948	0,985	0,951	1,000	0,967	0,945	0,961	0,958	0,937	0,944	0,936	0,955	0,955	0,960	0,933	0,933	0,849	0,928	0,923	0,921	0,910	0,838	0,832	0,899	0,768	0,890	0,857	0,892
7	0,920	0,926	0,952	0,934	0,967	1,000	0,967	0,966	0,956	0,959	0,934	0,926	0,940	0,949	0,939	0,918	0,908	0,835	0,903	0,894	0,896	0,882	0,816	0,810	0,871	0,742	0,860	0,836	0,872
8	0,926	0,931	0,934	0,916	0,945	0,967	1,000	0,952	0,949	0,966	0,925	0,926	0,939	0,937	0,928	0,903	0,893	0,820	0,887	0,878	0,881	0,870	0,805	0,799	0,860	0,737	0,851	0,834	0,862
9	0,913	0,918	0,946	0,923	0,961	0,966	0,952	1,000	0,972	0,956	0,945	0,938	0,952	0,953	0,940	0,921	0,916	0,843	0,911	0,901	0,905	0,898	0,831	0,825	0,887	0,760	0,875	0,850	0,880
10	0,909	0,915	0,946	0,930	0,958	0,956	0,949	0,972	1,000	0,967	0,957	0,939	0,954	0,956	0,948	0,928	0,928	0,846	0,913	0,904	0,908	0,901	0,830	0,821	0,886	0,763	0,879	0,848	0,883
11	0,907	0,913	0,930	0,914	0,937	0,959	0,966	0,956	0,967	1,000	0,943	0,925	0,949	0,946	0,934	0,914	0,909	0,829	0,899	0,890	0,894	0,883	0,814	0,805	0,869	0,747	0,855	0,830	0,866
12	0,915	0,915	0,932	0,944	0,944	0,934	0,925	0,945	0,957	0,943	1,000	0,954	0,963	0,946	0,939	0,934	0,929	0,844	0,923	0,909	0,912	0,901	0,824	0,815	0,884	0,770	0,874	0,859	0,880
13	0,908	0,909	0,928	0,931	0,936	0,926	0,926	0,938	0,939	0,925	0,954	1,000	0,976	0,954	0,942	0,936	0,937	0,860	0,936	0,933	0,931	0,920	0,846	0,837	0,902	0,778	0,892	0,870	0,903
14	0,926	0,926	0,947	0,935	0,955	0,940	0,939	0,952	0,954	0,949	0,963	0,976	1,000	0,978	0,966	0,938	0,937	0,856	0,937	0,927	0,925	0,910	0,836	0,827	0,892	0,768	0,882	0,855	0,893
15	0,916	0,921	0,946	0,935	0,955	0,949	0,937	0,953	0,956	0,946	0,946	0,954	0,978	1,000	0,983	0,955	0,950	0,864	0,940	0,931	0,930	0,911	0,837	0,828	0,894	0,768	0,887	0,853	0,902
16	0,920	0,922	0,951	0,935	0,960	0,939	0,928	0,940	0,948	0,934	0,939	0,942	0,966	0,983	1,000	0,959	0,949	0,866	0,939	0,934	0,932	0,921	0,847	0,838	0,904	0,778	0,897	0,869	0,906
17	0,892	0,893	0,924	0,927	0,933	0,918	0,903	0,921	0,928	0,914	0,934	0,936	0,938	0,955	0,959	1,000	0,980	0,876	0,956	0,952	0,947	0,926	0,850	0,847	0,911	0,793	0,912	0,878	0,931
18	0,886	0,892	0,924	0,917	0,933	0,908	0,893	0,916	0,928	0,909	0,929	0,937	0,937	0,950	0,949	0,980	1,000	0,878	0,966	0,958	0,962	0,936	0,852	0,849	0,922	0,799	0,922	0,883	0,932
19	0,813	0,817	0,844	0,829	0,849	0,835	0,820	0,843	0,846	0,829	0,844	0,860	0,856	0,864	0,866	0,876	0,878	1,000	0,900	0,887	0,884	0,864	0,940	0,937	0,846	0,721	0,840	0,794	0,851
20	0,890	0,890	0,922	0,909	0,928	0,903	0,887	0,911	0,913	0,899	0,923	0,936	0,937	0,940	0,939	0,956	0,966	0,900	1,000	0,977	0,970	0,944	0,858	0,854	0,924	0,794	0,917	0,876	0,932
21	0,885	0,881	0,917	0,895	0,923	0,894	0,878	0,901	0,904	0,890	0,909	0,933	0,927	0,931	0,934	0,952	0,958	0,887	0,977	1,000	0,973	0,952	0,861	0,864	0,941	0,800	0,924	0,879	0,931
22	0,878	0,883	0,915	0,897	0,921	0,896	0,881	0,905	0,908	0,894	0,912	0,931	0,925	0,930	0,932	0,947	0,962	0,884	0,970	0,973	1,000	0,973	0,877	0,871	0,949	0,819	0,937	0,899	0,937
23	0,867	0,869	0,904	0,883	0,910	0,882	0,870	0,898	0,901	0,883	0,901	0,920	0,910	0,911	0,921	0,926	0,936	0,864	0,944	0,952	0,973	1,000	0,895	0,889	0,968	0,836	0,950	0,896	0,923
24	0,794	0,799	0,833	0,814	0,838	0,816	0,805	0,831	0,830	0,814	0,824	0,846	0,836	0,837	0,847	0,850	0,852	0,940	0,858	0,861	0,877	0,895	1,000	0,976	0,886	0,750	0,863	0,810	0,845
25	0,788	0,793	0,827	0,805	0,832	0,810	0,799	0,825	0,821	0,805	0,815	0,837	0,827	0,828	0,838	0,847	0,849	0,937	0,854	0,864	0,871	0,889	0,976	1,000	0,902	0,753	0,869	0,807	0,851
26	0,853	0,855	0,893	0,869	0,899	0,871	0,860	0,887	0,886	0,869	0,884	0,902	0,892	0,894	0,904	0,911	0,922	0,846	0,924	0,941	0,949	0,968	0,886	0,902	1,000	0,830	0,953	0,890	0,915
27	0,731	0,733	0,764	0,761	0,768	0,742	0,737	0,760	0,763	0,747	0,770	0,778	0,768	0,768	0,778	0,793	0,799	0,721	0,794	0,800	0,819	0,836	0,750	0,753	0,830	1,000	0,841	0,798	0,791
28	0,845	0,846	0,884	0,873	0,890	0,860	0,851	0,875	0,879	0,855	0,874	0,892	0,882	0,887	0,897	0,912	0,922	0,840	0,917	0,924	0,937	0,950	0,863	0,869	0,953	0,841	1,000	0,899	0,933
29	0,834	0,835	0,856	0,850	0,857	0,836	0,834	0,850	0,848	0,830	0,859	0,870	0,855	0,853	0,869	0,878	0,883	0,794	0,876	0,879	0,899	0,896	0,810	0,807	0,890	0,798	0,899	1,000	0,893
30	0,846	0,851	0,883	0,874	0,892	0,872	0,862	0,880	0,883	0,866	0,880	0,903	0,893	0,902	0,906	0,931	0,932	0,851	0,932	0,931	0,937	0,923	0,845	0,851	0,915	0,791	0,933	0,893	1,000

min_pts	ARI
2	0,466
3	0,464
4	0,466
5	0,458
6	0,466
7	0,473
8	0,478
9	0,463
10	0,467
11	0,470
12	0,461
13	0,464
14	0,470
15	0,465
16	0,463
17	0,448
18	0,449
19	0,502
20	0,458
21	0,458
22	0,452
23	0,447
24	0,480
25	0,487
26	0,450
27	0,371
28	0,442
29	0,422
30	0,436

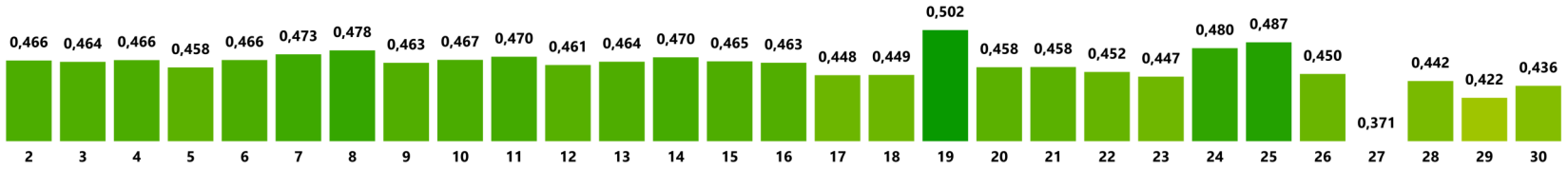


Figura 26 – Matriz ARI de similaridades entre partições e valores do ARI para cada nível de $m_{pts} \in [2, 30]$ para o conjunto *overlap* com 2% de rótulos semissupervisionados

min_pts	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
2	1,000	0,937	0,776	0,727	0,707	0,681	0,672	0,674	0,659	0,663	0,653	0,638	0,637	0,636	0,625	0,637	0,611	0,606	0,599	0,599	0,589	0,601	0,594	0,599	0,562	0,592	0,584	0,575	0,601
3	0,937	1,000	0,799	0,741	0,706	0,684	0,679	0,688	0,661	0,663	0,661	0,648	0,641	0,642	0,632	0,643	0,617	0,610	0,611	0,608	0,596	0,605	0,596	0,603	0,567	0,598	0,591	0,581	0,607
4	0,776	0,799	1,000	0,803	0,752	0,717	0,713	0,706	0,682	0,683	0,685	0,666	0,657	0,666	0,648	0,655	0,633	0,628	0,628	0,627	0,613	0,621	0,614	0,622	0,589	0,624	0,613	0,601	0,628
5	0,727	0,741	0,803	1,000	0,817	0,752	0,738	0,717	0,704	0,698	0,703	0,684	0,681	0,690	0,675	0,680	0,644	0,648	0,644	0,639	0,627	0,636	0,635	0,632	0,599	0,630	0,613	0,607	0,636
6	0,707	0,706	0,752	0,817	1,000	0,800	0,771	0,740	0,742	0,731	0,725	0,709	0,694	0,705	0,688	0,689	0,660	0,664	0,658	0,662	0,652	0,656	0,657	0,652	0,620	0,644	0,631	0,625	0,654
7	0,681	0,684	0,717	0,752	0,800	1,000	0,813	0,751	0,722	0,723	0,708	0,694	0,689	0,698	0,671	0,688	0,660	0,668	0,660	0,657	0,652	0,653	0,662	0,647	0,618	0,646	0,630	0,625	0,651
8	0,672	0,679	0,713	0,738	0,771	0,813	1,000	0,831	0,790	0,769	0,761	0,744	0,721	0,725	0,704	0,713	0,691	0,687	0,686	0,690	0,674	0,679	0,681	0,679	0,643	0,681	0,671	0,658	0,681
9	0,674	0,688	0,706	0,717	0,740	0,751	0,831	1,000	0,819	0,796	0,783	0,765	0,735	0,732	0,722	0,731	0,710	0,706	0,707	0,702	0,685	0,688	0,675	0,680	0,648	0,681	0,665	0,654	0,670
10	0,659	0,661	0,682	0,704	0,742	0,722	0,790	0,819	1,000	0,840	0,800	0,761	0,747	0,734	0,706	0,723	0,708	0,708	0,695	0,695	0,679	0,680	0,663	0,667	0,636	0,670	0,652	0,641	0,676
11	0,663	0,663	0,683	0,698	0,731	0,723	0,769	0,796	0,840	1,000	0,839	0,804	0,791	0,769	0,755	0,774	0,750	0,741	0,729	0,727	0,715	0,717	0,697	0,703	0,672	0,700	0,681	0,671	0,697
12	0,653	0,661	0,685	0,703	0,725	0,708	0,761	0,783	0,800	0,839	1,000	0,870	0,811	0,777	0,781	0,782	0,752	0,745	0,745	0,740	0,732	0,736	0,724	0,711	0,678	0,722	0,692	0,688	0,700
13	0,638	0,648	0,666	0,684	0,709	0,694	0,744	0,765	0,761	0,804	0,870	1,000	0,857	0,806	0,779	0,781	0,768	0,767	0,772	0,769	0,752	0,764	0,746	0,745	0,708	0,732	0,724	0,711	0,714
14	0,637	0,641	0,657	0,681	0,694	0,689	0,721	0,735	0,747	0,791	0,811	0,857	1,000	0,855	0,812	0,796	0,774	0,784	0,781	0,774	0,746	0,770	0,734	0,750	0,713	0,730	0,719	0,702	0,710
15	0,636	0,642	0,666	0,690	0,705	0,698	0,725	0,732	0,734	0,769	0,777	0,806	0,855	1,000	0,857	0,813	0,777	0,775	0,778	0,764	0,744	0,768	0,745	0,750	0,705	0,721	0,706	0,690	0,718
16	0,625	0,632	0,648	0,675	0,688	0,671	0,704	0,722	0,706	0,755	0,781	0,779	0,812	0,857	1,000	0,841	0,792	0,801	0,800	0,796	0,767	0,773	0,743	0,757	0,719	0,740	0,729	0,723	0,726
17	0,637	0,643	0,655	0,680	0,689	0,688	0,713	0,731	0,723	0,774	0,782	0,781	0,796	0,813	0,841	1,000	0,868	0,837	0,809	0,805	0,777	0,790	0,745	0,761	0,731	0,756	0,722	0,720	0,732
18	0,611	0,617	0,633	0,644	0,660	0,660	0,691	0,710	0,708	0,750	0,752	0,768	0,774	0,777	0,792	0,868	1,000	0,865	0,849	0,832	0,803	0,795	0,748	0,772	0,736	0,761	0,744	0,724	0,733
19	0,606	0,610	0,628	0,648	0,664	0,668	0,687	0,706	0,708	0,741	0,745	0,767	0,784	0,775	0,801	0,837	0,865	1,000	0,870	0,844	0,819	0,815	0,769	0,787	0,759	0,756	0,743	0,753	0,737
20	0,599	0,611	0,628	0,644	0,658	0,660	0,686	0,707	0,695	0,729	0,745	0,772	0,781	0,778	0,800	0,809	0,849	0,870	1,000	0,883	0,850	0,839	0,793	0,811	0,781	0,798	0,782	0,768	0,761
21	0,599	0,608	0,627	0,639	0,662	0,657	0,690	0,702	0,695	0,727	0,740	0,769	0,774	0,764	0,796	0,805	0,832	0,844	0,883	1,000	0,906	0,874	0,813	0,824	0,788	0,807	0,799	0,788	0,770
22	0,589	0,596	0,613	0,627	0,652	0,652	0,674	0,685	0,679	0,715	0,732	0,752	0,746	0,744	0,767	0,777	0,803	0,819	0,850	0,906	1,000	0,897	0,830	0,833	0,808	0,805	0,801	0,790	0,775
23	0,601	0,605	0,621	0,636	0,656	0,653	0,679	0,688	0,680	0,717	0,736	0,764	0,770	0,768	0,773	0,790	0,795	0,815	0,839	0,874	0,897	1,000	0,889	0,871	0,845	0,838	0,815	0,808	0,800
24	0,594	0,596	0,614	0,635	0,657	0,662	0,681	0,675	0,663	0,697	0,724	0,746	0,734	0,745	0,743	0,745	0,748	0,769	0,793	0,813	0,830	0,889	1,000	0,869	0,828	0,847	0,823	0,818	0,803
25	0,599	0,603	0,622	0,632	0,652	0,647	0,679	0,680	0,667	0,703	0,711	0,745	0,750	0,750	0,757	0,761	0,772	0,787	0,811	0,824	0,833	0,871	0,869	1,000	0,876	0,851	0,839	0,824	0,817
26	0,562	0,567	0,589	0,599	0,620	0,618	0,643	0,648	0,636	0,672	0,678	0,708	0,713	0,705	0,719	0,731	0,736	0,759	0,781	0,788	0,808	0,845	0,828	0,876	1,000	0,863	0,823	0,834	0,795
27	0,592	0,598	0,624	0,630	0,644	0,646	0,681	0,681	0,670	0,700	0,722	0,732	0,730	0,721	0,740	0,756	0,761	0,756	0,798	0,807	0,805	0,838	0,847	0,851	0,863	1,000	0,886	0,875	0,860
28	0,584	0,591	0,613	0,613	0,631	0,630	0,671	0,665	0,652	0,681	0,692	0,724	0,719	0,706	0,729	0,722	0,744	0,743	0,782	0,799	0,801	0,815	0,823	0,839	0,823	0,886	1,000	0,916	0,857
29	0,575	0,581	0,601	0,607	0,625	0,625	0,658	0,654	0,641	0,671	0,688	0,711	0,702	0,690	0,723	0,720	0,724	0,753	0,768	0,788	0,790	0,808	0,818	0,824	0,834	0,875	0,916	1,000	0,867
30	0,601	0,607	0,628	0,636	0,654	0,651	0,681	0,670	0,676	0,697	0,700	0,714	0,710	0,718	0,726	0,732	0,733	0,737	0,761	0,770	0,775	0,800	0,803	0,817	0,795	0,860	0,857	0,867	1,000

min_pts	ARI
2	0,456
3	0,456
4	0,460
5	0,465
6	0,478
7	0,466
8	0,476
9	0,471
10	0,470
11	0,482
12	0,483
13	0,475
14	0,474
15	0,478
16	0,479
17	0,486
18	0,473
19	0,480
20	0,471
21	0,479
22	0,472
23	0,481
24	0,481
25	0,474
26	0,454
27	0,476
28	0,469
29	0,465
30	0,479

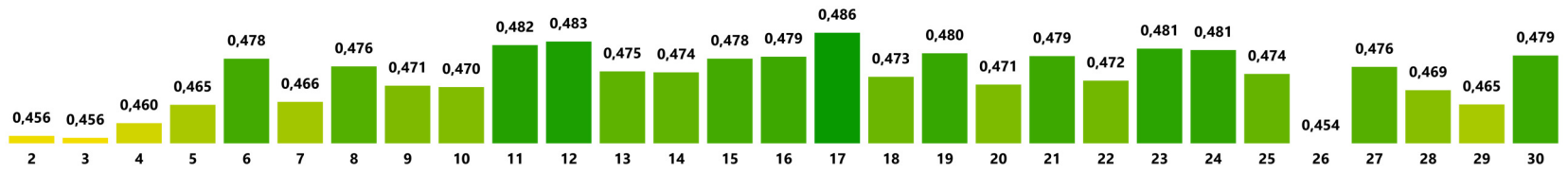


Figura 27 – Matriz ARI de similaridades entre partições e valores do ARI para cada nível de $m_{pts} \in [2, 30]$ para o conjunto *worms_2d* com 2% de rótulos semissupervisionados

Mesmo com dois ciclos de análise exploratória tendo produzido resultados negativos, foram realizados experimentos adicionais para ainda perseguir os objetivos secundários da proposta inicial de pesquisa. Uma das linhas investigadas consistiu da expansão do intervalo de m_{pts} para $[2, 100]$, mas não produziu resultados interessantes. Devido à falta de bases de dados reais públicas com as características desejadas, outra vertente de nossa investigação envolveu o uso de dois geradores de dados sintéticos, o gerador de grupos gaussianos disponibilizado por Handl (2017) e o gerador *MDCGenPy* (IGLESIAS et al., 2019). Os geradores foram utilizados para construir conjuntos de dados artificiais contendo grupos de objetos gaussianos e diferentes níveis de ruído, com o objetivo de produzir bases adequadas para a aplicação de nosso processo de geração e seleção de partições. Apesar de diversos conjuntos terem sido gerados e os experimentos correspondentes realizados, nenhum dos resultados se mostrou realmente satisfatório para nossa linha de pesquisa.

A conclusão a que chegamos é que o processo de propagação do HDBSCAN*(cd,-) proposto por Gertrudes et al. (2019) é extremamente robusto com relação ao parâmetro m_{pts}, e é capaz de produzir bons resultados com valores baixos de m_{pts} sem a necessidade de exploração de espectros maiores de densidade. Além disso, apesar do processo de geração e seleção de partições de níveis de densidade diferentes com o HDBSCAN*(cd,-) e CoreSG ser capaz de produzir resultados de qualidade levemente superior a uma única execução do HDBSCAN*, esta melhora não se mostrou significativa o suficiente para atingir o nível de contribuição científica necessário para o trabalho de pesquisa. Como conseqüência desta conclusão, acabamos por decidir abandonar a linha de pesquisa envolvendo a extensão do arcabouço unificado com o grafo CoreSG. Deste ponto em diante passamos a perseguir outra proposta de pesquisa, a de combinar a capacidade de reconhecimento de externos de métodos baseados em densidade com a tarefa de classificação semissupervisionada baseada em densidade. Esta segunda linha de pesquisa viria a se tornar o principal tema deste trabalho.