UNIVERSIDADE FEDERAL DE SÃO CARLOS
CAMPUS SÃO CARLOS

Bruno Leonel Nunes

Data Bubbles para Algoritmos de Fluxo de Dados Hierárquicos Baseados em Densidade

BRUNO LEONEL NUNES

Data Bubbles para Algoritmos de Fluxo de Dados Hierárquicos Baseados em Densidade

Trabalho de Conclusão de Curso submetido à Universidade Federal de São Carlos, como requisito necessário para obtenção do grau de Bacharel em Engenharia de Computação

São Carlos, julho de 2025

A jornada de mil milhas começa com um único passo.- Lao Tsé

Agradecimentos

Toda história de sucesso é composta por diferentes pilares, e a minha certamente não é exceção. Acredito firmemente que nossas conquistas não são resultado apenas de nossas próprias ações, mas sim de uma rede de apoio formada por amigos, familiares, instituições e entidades. Portanto, gostaria de expressar meus sinceros agradecimentos a todos os envolvidos nesta jornada, sem os quais nada disso seria possível.

Quero começar expressando um agradecimento especial ao Luan, uma das pessoas mais importantes ao longo deste trajeto. Ele esteve presente antes, durante e após todas as etapas deste percurso. Acredito sinceramente que minha experiência na universidade teria sido muito mais desafiadora sem os conselhos e experiências que ele compartilhou comigo. Obrigado, amigo, por ser tão fundamental nesta jornada.

Também gostaria de expressar minha gratidão pelo apoio dos meus professores, especialmente do meu orientador, que ofereceu orientação e compartilhou sua valiosa experiência ao longo de todo o projeto.

Não posso deixar de agradecer principalmente aos meus pais, que estiveram ao meu lado desde o início, oferecendo apoio emocional e financeiro incondicional.

Além disso, sou grato a mim mesmo por perseverar diante das dificuldades. Enfrentei diversos obstáculos ao longo do caminho, mas nunca desisti. Obrigado por nunca deixar de acreditar em mim e por sempre me encorajar a seguir em frente.

Por trás de cada história de sucesso, há uma série de pessoas que ajudaram a escrevê-la.

Resumo

O agrupamento de dados em Fluxos de Dados (FD) apresenta desafios devido ao volume, velocidade e natureza evolutiva da informação. Algoritmos hierárquicos baseados em densidade, como o HASTREAM, são promissores para identificar clusters de formas arbitrárias, mas sua eficácia depende crucialmente da qualidade da sumarização online dos dados. A abordagem padrão, utilizando Micro-Clusters (MCs), tende a perder informações sobre a distribuição interna dos dados, distorcendo as estimativas de densidade e impactando negativamente a qualidade do agrupamento final.

Este trabalho propõe uma nova abordagem de sumarização para algoritmos de agrupamento hierárquico baseados em densidade em FD, utilizando Data Bubbles (DBs) adaptadas. As DBs, que originalmente incluem estimativas de densidade interna (`nnDist`), foram modificadas para operar sob o modelo de janela amortecida, com novas formulações para suas propriedades (como `extent` e `nnDist`) baseadas em estatísticas ponderadas. Além disso, propomos uma modificação no cálculo da distância core para DBs, visando alinhar-se melhor com os princípios do HDBSCAN*.

A metodologia envolve a integração dessas DBs adaptadas e suas métricas de distância revisadas (Distância Core modificada e Distância de Alcançabilidade Mútua) ao algoritmo HASTREAM, resultando no HASTREAM-DB. Descrevemos os processos de manutenção online das DBs adaptadas (incluindo a gestão de bubbles potenciais e outliers) e a subsequente fase offline de construção da hierarquia de clusters sobre essas estruturas.

O objetivo é demonstrar que a utilização de Data Bubbles adaptadas, por preservarem melhor a informação espacial e de densidade local, pode mitigar as distorções inerentes à sumarização por MCs, levando a uma identificação de clusters mais precisa e robusta em ambientes de Fluxo de Dados.

Palavras-chave: Sumarização de Dados, Data Bubbles, Fluxo de Dados, Agrupamento Baseado em Densidade, Agrupamento Hierárquico, HASTREAM.

Abstract

Clustering data in Data Streams (DS) presents challenges due to the volume, velocity, and evolving nature of information. Hierarchical density-based clustering algorithms, such as HASTREAM, are promising for identifying arbitrarily shaped clusters, but their effectiveness critically depends on the quality of online data summarization. The standard approach, using Micro-Clusters (MCs), tends to lose information about the internal data distribution, distorting density estimates and negatively impacting the quality of the final clustering.

This work proposes a novel summarization approach for hierarchical density-based clustering algorithms in DS, utilizing adapted Data Bubbles (DBs). DBs, which originally include internal density estimates (`nnDist`), have been modified to operate under the damped window model, with new formulations for their properties (such as `extent` and `nnDist`) based on weighted statistics. Furthermore, we propose a modification to the core distance calculation for DBs, aiming for better alignment with HDBSCAN* principles.

The methodology involves integrating these adapted DBs and their revised distance metrics (modified Core Distance and Mutual Reachability Distance) into the HASTREAM algorithm, resulting in HASTREAM-DB. We describe the online maintenance processes for the adapted DBs (including the management of potential and outlier bubbles) and the subsequent offline phase of constructing the cluster hierarchy upon these structures.

The objective is to demonstrate that the use of adapted Data Bubbles, by better preserving local spatial and density information, can mitigate the distortions inherent in MC-based summarization, leading to more accurate and robust cluster identification in Data Stream environments.

Keywords: Data Summarization, Data Bubbles, Data Streams, Density-Based Clustering, Hierarchical Clustering, HASTREAM.

CF *Clustering Feature*

DBSCAN Density-Based Spatial Clustering of Applications with Noise

FD Fluxo de Dados

HDBSCAN* Hierarchical DBSCAN*

MST Árvore Geradora Mínima

MC *Micro Cluster*

MRD Distância de Alcançabilidade Mútua

o-DB *Outlier Data Bubble*

o-MC *Outlier Micro-Cluster*

p-DB *Potential Data Bubble*

p-MC *Potential Micro-Cluster*

DB *Data Bubble*

Lista de ilustrações

Lista de tabelas

Sumário

1 Introdução

A geração contínua e massiva de Fluxos de Dados (Fluxos de Dados (FDs)) em diversas aplicações modernas impõe desafios significativos à análise de dados tradicional [Gama 2010]. O agrupamento, uma técnica essencial para descobrir padrões, quando aplicado a FDs, exige estratégias de sumarização online para lidar com o volume e a velocidade dos dados. A abordagem padrão para essa sumarização é o uso de **Micro-Clusters (MCs)** [Aggarwal et al. 2003, Cao et al. 2006]. Algoritmos de agrupamento baseados em densidade, como o DBSCAN [**?**] e suas variantes hierárquicas como o Hierarchical DBSCAN* (HDBSCAN*)* [Campello et al. 2015], são particularmente valorizados neste contexto por sua capacidade de identificar grupos de formas arbitrárias e ruído sem a necessidade de pré-definir o número de grupos. Contudo, sua aplicação em fluxos depende fundamentalmente da qualidade da sumarização via *Micro Clusters* (*MCs*).

O problema central que este trabalho aborda reside na limitação dos *MCs* no contexto de algoritmos baseados em densidade. Ao representarem regiões de dados apenas por estatísticas agregadas (como centroide e raio), os *MCs* perdem informações cruciais sobre a distribuição espacial interna e as distâncias reais entre os pontos. Essa omissão gera uma **representação sumarizada imprecisa**, um fenômeno que causa "distorção estrutural"e distorce as estimativas de densidade local. Isso é particularmente problemático para algoritmos hierárquicos como o HASTREAM, que constroem suas hierarquias sobre essa representação falha, comprometendo a qualidade dos grupos identificados.

Como uma alternativa promissora, as **Data Bubbles (DBs)** [Breunig et al. 2001] foram propostas para oferecer uma sumarização que preserva melhor essas informações espaciais. A característica distintiva das *Data Bubbles* (*DBs*) é a inclusão de uma estimativa da *nnDist* (*nearest neighbor distance*), que quantifica a distância média ao k-vizinho mais próximo *dentro* da própria *bubble*. Essencialmente, enquanto um *MC* descreve a localização e a dispersão geral, a *nnDist* de uma *DB* fornece uma medida direta da densidade interna da região sumarizada. Esta informação sobre o adensamento local é vital para cálculos de densidade mais precisos.

Portanto, este trabalho explora o potencial das *DBs* como uma estrutura de sumarização mais adequada para o agrupamento hierárquico baseado em densidade em FDs. A abordagem proposta envolve adaptar as *Data Bubbles* ao ambiente dinâmico dos fluxos, incorporando decaimento temporal e reformulando suas propriedades, e integrá-las ao algoritmo HASTREAM em substituição aos *MCs*. O objetivo é investigar se essa integração, ao fornecer uma representação sumarizada mais fiel à densidade local, pode mitigar as distorções inerentes aos *MCs* e, assim, aprimorar a qualidade e a robustez na

identificação de grupos em cenários de dados contínuos e evolutivos.

1.1 Objetivos

1.1.1 Objetivo Principal

Propor, implementar e avaliar uma abordagem de sumarização de dados, baseada em *Data Bubble* (*DB*) adaptadas para Fluxo de Dados (FD), integrando-a ao algoritmo de agrupamento hierárquico baseado em densidade HASTREAM [Hassani, Spaus e Seidl 2014]. O propósito central é melhorar a qualidade e a fidelidade da identificação de grupos em ambientes dinâmicos, superando as limitações de distorção de densidade impostas pela sumarização tradicional via *MC*.

1.1.2 Objetivos Secundários

Para alcançar o objetivo principal, os seguintes passos foram seguidos:

- Estudar e adaptar as *DBs* para o contexto de Fluxo de Dados (FD). Como foram criadas para dados estáticos, suas variáveis chave (*extent*, *nnDist*) e estatísticas precisaram ser reformuladas para incorporar o decaimento temporal, garantindo que continuem a representar a densidade local de forma precisa mesmo com a evolução do fluxo.
- Consequentemente, as métricas de distância (Distância Core, Distância de Alcançabilidade Mútua (MRD)), essenciais para algoritmos como o Hierarchical DBSCAN* (HDBSCAN*)*, precisaram ser redefinidas para operar corretamente sobre as *DBs* adaptadas, preservando a semântica de densidade e vizinhança.
- Desenvolver os mecanismos para a manutenção online das *DBs* adaptadas, incluindo a incorporação de novos pontos, a aplicação do decaimento e o gerenciamento do ciclo de vida das *DBs* (classificação como *Potential Data Bubble* (*p-DB*) ou *Outlier Data Bubble* (*o-DB*) e remoção).
- Integrar a representação sumarizada por *p-DB*s à fase offline do HASTREAM, utilizando as métricas de distância adaptadas para construir a hierarquia de grupos.
- Finalmente, conduzir uma avaliação experimental comparativa para validar a abordagem, comparando o HASTREAM-DB com o HASTREAM original (baseado em *MCs*) usando métricas de qualidade para demonstrar os benefícios da sumarização por *DBs* adaptadas.

2 Fundamentação Teórica

Esta seção apresenta os conceitos fundamentais necessários para a compreensão do problema abordado e da solução proposta neste trabalho. Abordaremos o que são Fluxos de Dados (FDs), a técnica de sumarização via **Micro-Clusters (MCs)**, o algoritmo de referência HASTREAM e a estrutura alternativa que propomos adaptar, as **Data Bubbles (DBs)**.

2.1 Fluxo de Dados (Data Streams)

Um Fluxo de Dados (FD), ou *Data Stream*, pode ser formalmente definido como uma sequência ordenada $S = \{x_1, x_2, x_3, \ldots, x_k, \ldots\}$ de instâncias (ou objetos), onde x_i é a instância que chega no i-ésimo passo de tempo, e o comprimento da sequência k pode ser potencialmente infinito. Essa definição captura a essência dos cenários modernos onde os dados são gerados de forma contínua e ininterrupta, como em redes de sensores, monitoramento de tráfego de rede, transações financeiras ou atividade em redes sociais [Miccio e Schwartz 2021, Zerhari, Lahcen e Mouline 2015, Djonlagic et al. 2021]. A Figura 1 ilustra a natureza temporal dos fluxos de dados, mostrando como as instâncias chegam sequencialmente, com diferentes níveis de relevância dependendo de sua idade.

O processamento de FDs apresenta desafios únicos que os distinguem da análise de dados estática tradicional [**?**]:

1. *Volume e Infinitude:* Os fluxos podem ser massivos e teoricamente infinitos, tornando o armazenamento persistente de todos os dados recebidos impraticável.

2. *Velocidade:* Os dados frequentemente chegam em alta velocidade, exigindo que o processamento seja realizado rapidamente, muitas vezes sob restrições de tempo real.

3. *Evolução (Concept Drift):* A distribuição estatística subjacente aos dados pode mudar ao longo do tempo. Os padrões identificados em um momento podem se tornar obsoletos, exigindo que os modelos de análise se adaptem continuamente.

4. *Acesso Limitado:* Geralmente, cada instância do fluxo só pode ser examinada uma única vez e depois deve ser descartada ou sumarizada, devido às restrições de armazenamento e processamento.

Diante desses desafios, algoritmos projetados para FDs operam de forma incremental, utilizando memória e tempo computacional limitados. Crucialmente, eles dependem de

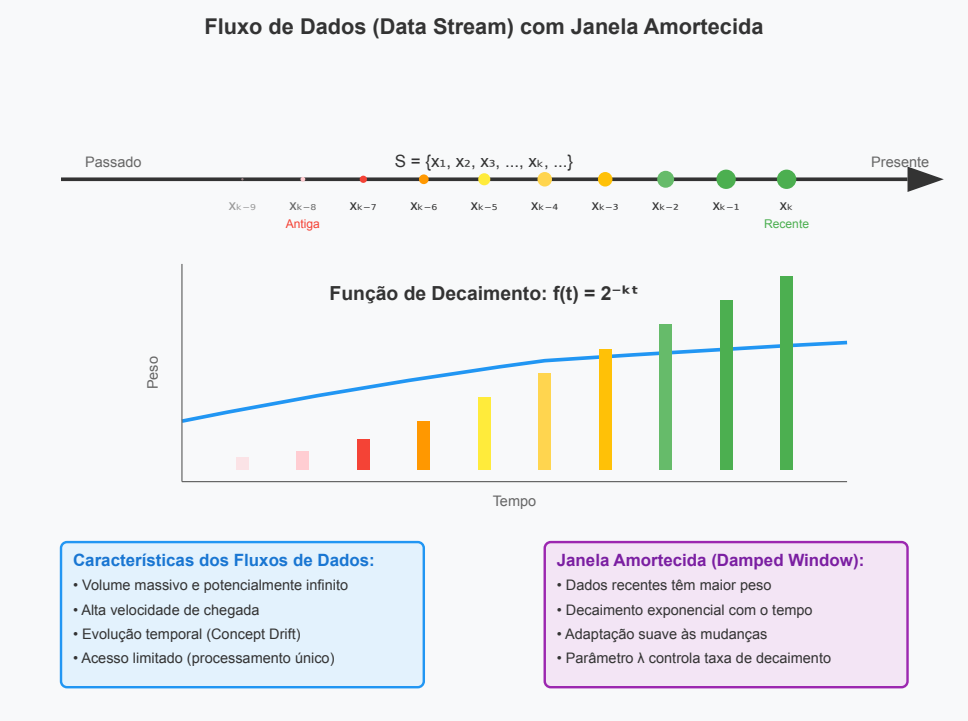

Figura 1 – Representação de um fluxo de dados com janela amortecida. Os dados mais recentes (em verde) possuem maior peso, enquanto os dados antigos (em vermelho) têm sua influência reduzida exponencialmente através da função de decaimento $f(t) = 2^{-\lambda t}$. (Fonte: O autor).

técnicas de sumarização para manter uma representação compacta e atualizada do estado do fluxo.

2.1.1 Modelos de Janela e Decaimento Temporal

Para lidar com a evolução temporal, são empregados Modelos de Janela (*Window Models*). Este trabalho, assim como o algoritmo HASTREAM, utiliza o modelo de **Janela Amortecida** (*damped window*). Nele, a contribuição de cada instância para uma estrutura de sumarização recebe um peso que decai exponencialmente com o tempo. Uma função de decaimento comum é dada pela Equação 2.1:

$$f(t) = 2^{-\lambda t} \tag{2.1}$$

onde t é o tempo decorrido e λ $(0 < \lambda < 1)$ é a taxa de decaimento. Quanto maior λ, mais rapidamente os dados antigos perdem relevância. Essa função é aplicada diretamente às estatísticas das estruturas de sumarização, permitindo que o modelo se adapte suavemente às mudanças na distribuição dos dados.

2.1.2 Micro-Clusters como Estruturas de Sumarização

Para realizar a sumarização eficiente, a abordagem mais comum utiliza estruturas conhecidas como **Micro-Clusters (MCs)**. Um *MC* é uma representação compacta de

um conjunto de pontos de dados similares, armazenando apenas as estatísticas essenciais. A Figura 2 ilustra como conjuntos de pontos são sumarizados em *MCs*.

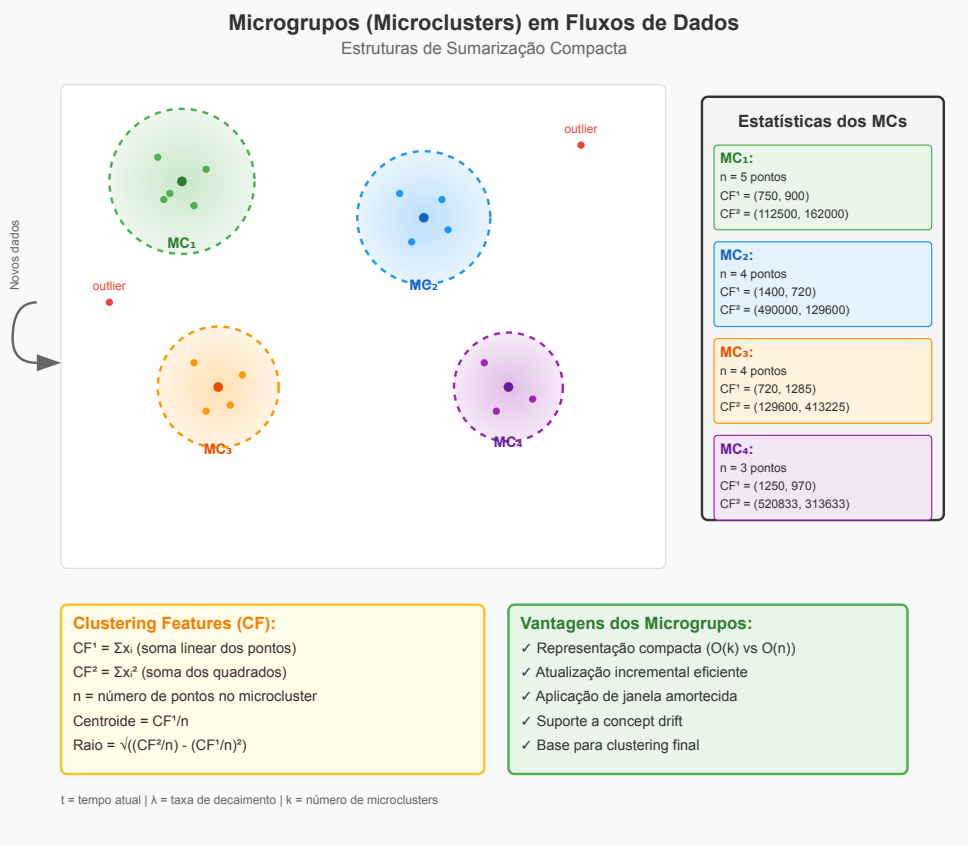

Figura 2 – Representação de Micro-Clusters (MCs) em um fluxo de dados. Cada MC sumariza um conjunto de pontos similares através de estatísticas compactas, permitindo processamento eficiente. Os centroides dos MCs são indicados pelos pontos escuros, enquanto as áreas circulares mostram a extensão de cada agrupamento. (Fonte: O autor).

Matematicamente, um *MC* é definido por suas **Clustering Features (CFs)**, um conceito introduzido pelo algoritmo BIRCH [**?**], que consistem em:

1. n: o número de pontos de dados atribuídos ao MC;
2. $CF^1 = \sum_{i=1}^{n} \vec{x_i}$: a soma linear de todos os pontos (vetor);
3. $CF^2 = \sum_{i=1}^{n} \|\vec{x_i}\|^2$: a soma dos quadrados das normas de todos os pontos (escalar).

A partir dessas estatísticas, é possível calcular propriedades importantes do MC de forma eficiente:

- **Centroide:** $\vec{c} = \frac{CF^1}{n}$
- **Raio:** $r = \sqrt{\frac{CF^2}{n} - \left\|\frac{CF^1}{n}\right\|^2}$

Como demonstrado na Figura 2, cada *MC* mantém apenas essas estatísticas, representando dados com complexidade de espaço $O(k)$ (onde k é o número de *MCs*), em vez de $O(n)$. A aplicação da janela amortecida multiplica as CFs pelo fator de decaimento $f(t)$, garantindo a adaptabilidade temporal. Os *MCs* formam a base conceitual para muitos algoritmos de agrupamento em fluxo, incluindo o HASTREAM.

2.2 HASTREAM: Agrupamento Hierárquico em Fluxo de Dados

Dentre os algoritmos para agrupamento em FDs, o HASTREAM (*Hierarchical Anytime Stream Clustering*) [Hassani, Spaus e Seidl 2014] se destaca por adaptar a abordagem hierárquica e baseada em densidade para este cenário. Enquanto algoritmos como o DenStream [Cao et al. 2006] focam em encontrar uma partição plana de clusters, o HASTREAM, inspirado nos princípios do Hierarchical DBSCAN* (HDBSCAN*)* [Campello et al. 2015], visa identificar uma hierarquia de clusters com diferentes densidades e formas arbitrárias que evoluem ao longo do tempo.

O HASTREAM opera seguindo o modelo online-offline:

2.2.1 Fase Online

Nesta fase, o algoritmo mantém um conjunto atualizado de *MCs* que representam o estado corrente do fluxo. Utilizando um modelo de Janela Amortecida (Equação 2.1), as estatísticas ponderadas (w, $C\vec{F}^1$, CF^2) dos *MCs* são atualizadas para dar maior influência aos dados recentes. De forma análoga ao DenStream, o HASTREAM classifica os *MCs* com base em seu peso total w:

- **Potential Micro-Clusters (p-MCs):** *MCs* com peso suficiente para serem considerados parte de um cluster denso.
- **Outlier Micro-Clusters (o-MCs):** *MCs* com baixo peso, representando ruído ou clusters em formação/dissolução.

O objetivo desta fase é manter um conjunto representativo de *Potential Micro-Cluster*s (*p-MC*s) que reflita as áreas densas do fluxo no momento atual.

2.2.2 Fase Offline

Acionada sob demanda, esta fase aplica uma variante do HDBSCAN** sobre o conjunto de *p-MC*s. Cada *MC* é tratado como um pseudo-ponto localizado em seu centroide $\vec{c}$. O processo envolve:

1. **Cálculo das Distâncias Baseadas em Densidade entre *MCs*:**

- *Distância Core para MCs:* Para um dado parâmetro $MinPts$, a distância core de um *MC* mc_p é a distância euclidiana até seu k-ésimo vizinho mais próximo, onde k é o menor número tal que a soma dos pesos w dos k vizinhos mais próximos (incluindo mc_p) atinja o limiar $MinPts$.
- *Distância de Alcançabilidade Mútua (MRD) para MCs:* A MRD entre dois *MCs*, mc_p e mc_q, é calculada com base em suas distâncias core e na distância euclidiana entre seus centroides $\vec{c}_p$ e $\vec{c}_q$, conforme a Equação **??**.

2. **Construção da Hierarquia:** Um grafo ponderado pela MRD é construído, a Árvore Geradora Mínima (MST) é extraída, e a hierarquia de clusters é gerada ao remover arestas em ordem decrescente de peso.

3. **Extração de Clusters Estáveis (Opcional):** Um critério como o FOSC [Campello et al. 2013] pode ser aplicado para selecionar uma partição "plana"de clusters.

A força do HASTREAM é sua capacidade de gerar agrupamentos hierárquicos em fluxos. Sua principal limitação, no entanto, é tratar *MCs* como pseudo-pontos, baseando os cálculos de densidade unicamente nas distâncias entre centroides. Isso é uma **sumarização imprecisa** porque ignora a extensão real e a distribuição interna dos pontos dentro de cada *MC*, podendo levar a hierarquias que não refletem fielmente a estrutura de densidade dos dados originais. Esta limitação justifica a busca por estruturas de sumarização alternativas.

2.3 Data Bubbles (DB): Uma Alternativa para Sumarização

A sumarização via *MCs* pode levar a uma "distorção estrutural", como ilustrado na Figura 3. Embora a distância entre centroides seja a mesma em cenários distintos (a e b), a densidade e a disposição dos dados são diferentes, uma nuance que a representação por *MCs* (c) não captura.

Para superar essa limitação, as **Data Bubbles (DBs)** foram propostas por [Breunig et al. 2001] para preservar melhor as informações de distância e densidade locais. Originalmente, uma *DB* B_i que sumariza um conjunto de dados estático X_i é definida pela tupla:

$$B_i = (\vec{rep}_i, n_i, \text{extent}_i, \text{nnDist}_i) \tag{2.2}$$

onde:

- $\vec{rep}_i$: O representante da *bubble* (centroide).
- n_i: A cardinalidade (número de pontos).

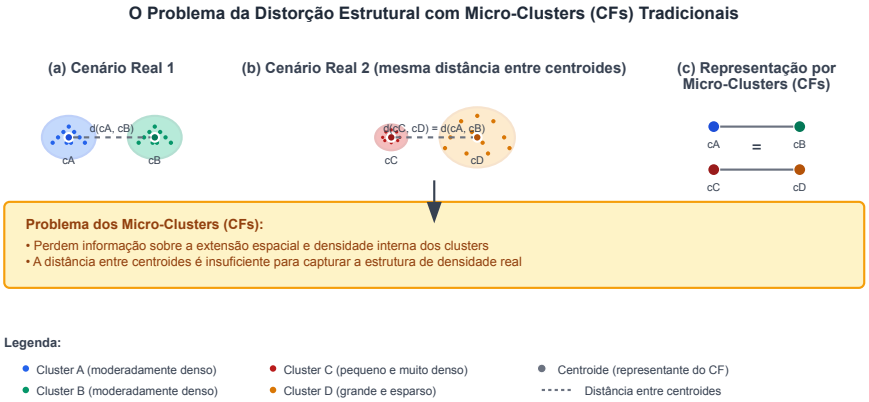

Figura 3 – O Problema da Distorção Estrutural com Micro-Clusters (MCs). Mesmo que a distância entre centroides ($d(c_A, c_B)$ e $d(c_C, c_D)$) seja a mesma nos cenários (a) e (b), a estrutura de densidade é distinta. A representação por MCs (c) perde essa informação, levando a uma interpretação equivocada. (Fonte: O autor).

- extent_i: A extensão espacial da *bubble*, calculada como duas vezes o **Raio RMS (Root Mean Square)**, que mede a dispersão dos pontos:

$$\text{extent}_i = 2 \cdot \sqrt{\frac{SS_i}{n_i} - \|r\vec{e}p_i\|^2} \tag{2.3}$$

- nnDist_i: A distância média estimada ao k-vizinho mais próximo (*nearest neighbor distance*) *dentro* da *bubble*, que estima sua densidade interna. É calculada como:

$$\text{nnDist}_i(k) = \left(\frac{k}{n_i}\right)^{\frac{1}{d}} \cdot \text{extent}_i \tag{2.4}$$

  para um k pequeno (e.g., $k = 1$) e dimensionalidade d.

A inclusão dos termos extent_i e nnDist_i permite que as *DBs* capturem a dispersão e a densidade interna, superando a distorção dos *MCs*, como ilustrado na Figura 4.

Com base nessa estrutura enriquecida, métricas de distância mais refinadas podem ser definidas para operar sobre as *DBs*.

2.3.1 Distância entre DBs Estáticas

A distância direta $\text{dist}(B_i, B_j)$ entre duas *DBs* estáticas estima a distância entre seus pontos mais próximos, considerando a separação entre representantes, as extensões e as densidades internas (via nnDist). A lógica da Equação **??** é que, se as *bubbles* se sobrepõem, a distância é governada pela mais densa delas.

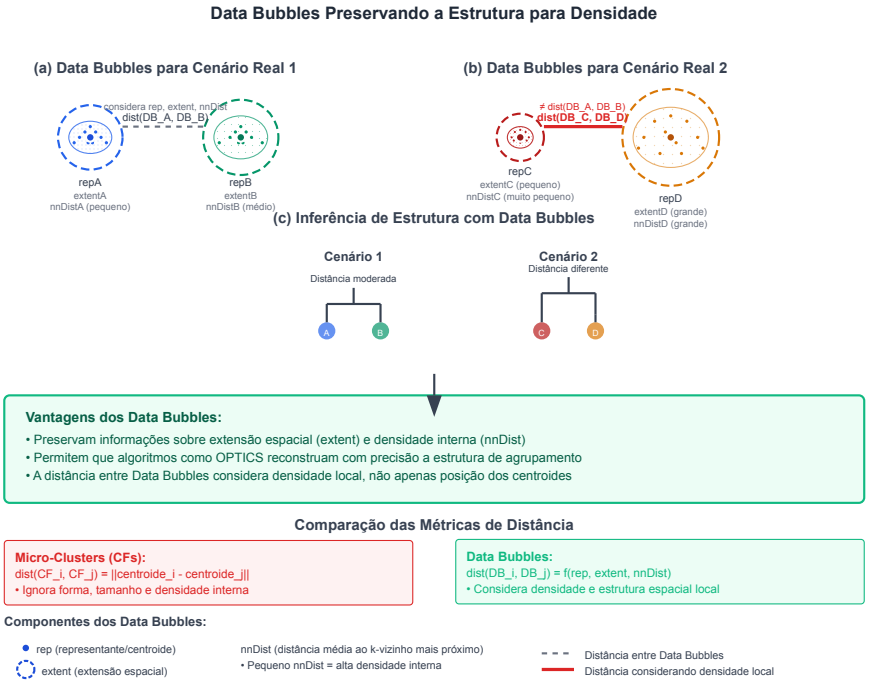

Figura 4 – Data Bubbles Preservando a Estrutura para Densidade. Os painéis (a) e (b) mostram como as DBs representam os cenários da Figura 3, considerando a extensão espacial (*extent*) e a densidade interna (*nnDist*). (c) demonstra que a inferência da estrutura se torna mais precisa, pois a distância entre DBs reflete melhor as relações de densidade reais. (Fonte: O autor).

2.3.2 Distância Core para Data Bubbles Estáticas

A distância core, $core_{MinPts}(B_i)$, de uma *DB* estática B_i estima o quão "adensada"é sua vizinhança. Conforme [Breunig et al. 2001], se B_i e suas vizinhas não somam $MinPts$ pontos, a distância é infinita. Caso contrário, é a distância até a vizinha mais distante necessária para atingir $MinPts$, ajustada pela densidade interna dessa vizinha (Equação **??**). O termo k' na equação original representa o número de pontos que a *DB* vizinha B_C precisa "contribuir"para que a vizinhança de B_i alcance o limiar $MinPts$. Se a própria B_i já tem $n_i \geq MinPts$, sua distância core é simplesmente sua nnDist interna (Equação **??**).

2.3.3 MRD para Data Bubbles Estáticas

Com as distâncias core e direta definidas, a Distância de Alcançabilidade Mútua (MRD) entre duas *DBs* estáticas (Equação **??**) é calculada de forma análoga à dos pontos e *MCs*. Ela representa o raio mínimo para que as duas *bubbles* se tornem mutuamente alcançáveis em termos de densidade. Esta métrica, que utiliza a informação extra das *DBs*, é a base para a construção da hierarquia no algoritmo proposto.

Em resumo, as *DBs* oferecem uma representação sumarizada mais rica que os *MCs*. As métricas de distância construídas sobre elas têm o potencial de refletir as relações de

densidade de forma mais acurada do que aquelas baseadas apenas em centroides. Isso as torna candidatas ideais para aprimorar algoritmos hierárquicos como o HASTREAM, motivando as adaptações propostas neste trabalho.

3 Metodologia: HASTREAM-DB - Agrupamento com Data Bubbles

Com base na fundamentação teórica, que expôs as limitações dos *MCs* e o potencial das *DBs* estáticas (Seção 2.3), esta seção detalha a metodologia proposta: o HASTREAM-DB. Trata-se de uma adaptação do algoritmo HASTREAM que substitui a sumarização baseada em *MCs* por uma nova abordagem utilizando *DBs* adaptadas para FD. A metodologia mantém a estrutura geral online-offline, mas modifica as estruturas de sumarização e os cálculos de distância.

3.1 Fase Online: Manutenção das Data Bubbles Adaptadas

A fase online do HASTREAM-DB processa continuamente os dados de entrada e mantém um conjunto atualizado de *DBs* que sumarizam o estado do fluxo. Isso requer adaptar a definição estática da *DB* (Equação 2.2) para o modelo de Janela Amortecida (Equação 2.1).

3.1.1 Definição da Data Bubble para Fluxo de Dados

Propomos uma *DB* adaptada para FD, $B_i(t)$, definida no tempo t pela seguinte tupla:

$$B_i(t) = (\vec{CF}_i^1(t), CF_i^2(t), w_i(t), \vec{rep}_i(t), \text{extent}_i(t), \text{nnDist}_i(t), t_{0,i})$$

onde:

- $\vec{CF}_i^1(t), CF_i^2(t), w_i(t)$: São as estatísticas *Clustering Feature* (CF) ponderadas (soma linear vetorial, soma dos quadrados escalar e peso total), análogas às dos *MCs*.
- $t_{0,i}$: O timestamp de criação da *bubble*.
- $\vec{rep}_i(t)$: O representante (centroide) ponderado, $\vec{rep}_i(t) = \vec{CF}_i^1(t)/w_i(t)$.
- $\text{extent}_i(t)$: A extensão ponderada. O cálculo original de extent para *DBs* estáticas (Equação 2.3) depende da contagem de pontos n_i, que não é semanticamente compatível com o modelo de decaimento temporal. Para resolver isso, propomos uma nova formulação para o *extent* baseada nas estatísticas ponderadas, análoga ao cálculo do raio dos *MCs* (descrito na Seção 2.1.2). Mantendo a definição original de ser **duas**

vezes o raio RMS, a fórmula adaptada é:

$$\text{extent}_i(t) = 2 \cdot \sqrt{\frac{CF_i^2(t)}{w_i(t)} - \left\| \frac{\vec{CF}_i^1(t)}{w_i(t)} \right\|^2} = 2 \cdot \sqrt{\frac{CF_i^2(t)}{w_i(t)} - \|r\vec{e}p_i(t)\|^2} \tag{3.1}$$

- $\text{nnDist}_i(t)$: A distância interna estimada ponderada. Utilizamos a lógica da Equação 2.4, mas aplicada sobre as quantidades ponderadas $w_i(t)$ e $\text{extent}_i(t)$. Para $k = 1$ e dimensionalidade d:

$$\text{nnDist}_i(t, k = 1) = \left(\frac{1}{w_i(t)} \right)^{\frac{1}{d}} \cdot \text{extent}_i(t) \tag{3.2}$$

  Esta adaptação mantém a estimativa de densidade interna mesmo com o peso $w_i(t)$ variando continuamente.

3.1.2 Processo de Atualização Online

O fluxo é processado ponto a ponto. Para cada novo ponto $\vec{p}$ que chega no tempo t_{atual}:

1. *Aplicar Decaimento*: As estatísticas ponderadas de todas as *DBs* existentes são multiplicadas pelo fator de decaimento $f(\Delta t)$.

2. *Encontrar Melhor DB*: O ponto $\vec{p}$ é atribuído à *DB* existente mais "próxima"que possa absorvê-lo.

3. *Atualizar ou Criar DB*: Se uma *DB* B_i é encontrada, suas estatísticas são atualizadas pela adição do ponto $\vec{p}$ (com peso inicial 1): $w_i \leftarrow w_i + 1$, $\vec{CF}_i^1 \leftarrow \vec{CF}_i^1 + \vec{p}$, $CF_i^2 \leftarrow CF_i^2 + \|\vec{p}\|^2$. Caso contrário, uma nova *DB* é criada contendo apenas $\vec{p}$.

4. *Recalcular Propriedades*: As propriedades da *DB* modificada ($r\vec{e}p_i(t)$, $\text{extent}_i(t)$, e $\text{nnDist}_i(t)$) são recalculadas usando as Equações 3.1 e 3.2.

3.1.3 Gerenciamento de p-DBs e o-DBs

Para focar nas regiões relevantes, adotamos a distinção entre *Potential Data Bubble* (*p-DB*) e *Outlier Data Bubble* (*o-DB*), inspirada no DenStream [Cao et al. 2006]. Dados os parâmetros μ (limiar de peso potencial) e β (fator de outlier):

- *Potential Data Bubble* (*p-DB*): Uma *DB* com peso $w_i(t) \geq \beta\mu$, representando uma região densa.

- *Outlier Data Bubble* (*o-DB*): Uma *DB* com peso $w_i(t) < \beta\mu$, representando ruído ou uma região esparsa.

Periodicamente, o status das *DBs* é verificado, e as *o-DB*s obsoletas (com peso abaixo de um limiar ξ) são removidas, mantendo o modelo online compacto e focado.

3.2 Fase Offline: Agrupamento Hierárquico sobre DBs Adaptadas

Quando solicitada, a fase offline opera sobre o conjunto de *p-DB*s para construir a hierarquia de grupos, seguindo os passos do HDBSCAN** adaptados para as *DBs* de fluxo.

1. *Cálculo da Distância Core para DBs Adaptadas*: A definição de distância core para *DBs* estáticas foi apresentada na Seção 2.3.2. Para o HASTREAM-DB, buscando alinhar o conceito a um raio que englobe $MinPts$ em peso, propomos a seguinte formulação para $core_{MinPts}(B_i(t))$:

   Dado $MinPts$ e $SNN(B_i(t), MinPts) = k$ o índice da vizinha necessária para que $B_i(t)$ e suas k primeiras vizinhas somem pelo menos $MinPts$ em peso, a distância core é:

   $$core_{MinPts}(B_i(t)) = \begin{cases} nnDist_i(t, MinPts), \text{se } SNN(B_i(t), MinPts) = 0 \\ dist(r\vec{ep}_i(t), r\vec{ep}_k(t)) - \text{extent}_k(t) + \\ nnDist_k\left(t, MinPts - \sum_{l=0}^{k-1} w_{NN(B_i(t),l)}(t)\right), \text{caso contrário.} \end{cases} \tag{3.3}$$

   Onde no "caso contrário"($k > 0$), $B_k(t)$ é a k-ésima vizinha adaptada. A distância core é a distância até essa vizinha, ajustada por sua extensão e densidade interna para o peso restante necessário para atingir $MinPts$.

2. *Cálculo da MRD entre DBs Adaptadas*: A MRD entre duas *DBs* adaptadas $B_i(t)$ e $B_j(t)$ é calculada com base na distância core proposta e na distância direta entre elas, $dist_{fluxo}$, que é análoga à Equação **??** mas utiliza as propriedades da *DB* adaptada.

   $$MRD_{MinPts}(B_i(t), B_j(t)) = \max\{core_{MinPts}(B_i(t)), core_{MinPts}(B_j(t)), dist_{fluxo}(B_i(t), B_j(t))\} \tag{3.4}$$

3. *Construção da Hierarquia*: Um grafo completo é formado com as *p-DB*s como vértices e arestas ponderadas pela MRD (Eq. 3.4). A MST deste grafo é extraída e, ao remover suas arestas em ordem decrescente de peso, a hierarquia de grupos é gerada.

4. *Extração de Grupos (Opcional)*: Um critério de estabilidade (como o FOSC adaptado) pode ser usado para extrair uma partição "plana"da hierarquia.

Ao utilizar as *DBs* adaptadas e as métricas de distância propostas, esperamos que o HASTREAM-DB construa uma hierarquia de grupos que reflita de forma mais precisa e robusta a estrutura de densidade subjacente ao fluxo, superando as limitações do HASTREAM original.

4 Experimentos e Resultados

Esta seção apresenta a avaliação experimental da abordagem proposta, HASTREAM-DB, em comparação com o algoritmo HASTREAM original. O foco é analisar a qualidade das partições de grupos geradas, demonstrando a eficácia da sumarização via *DBs* na preservação da estrutura de densidade dos dados.

4.1 Configuração Experimental

Para avaliar a qualidade do agrupamento, realizamos experimentos comparando o HASTREAM-DB com o HASTREAM sob um protocolo rigoroso e controlado.

4.1.1 Conjuntos de Dados

Utilizamos uma variedade de conjuntos de dados, tanto sintéticos quanto reais, para testar os algoritmos sob diferentes condições. As características principais destes conjuntos de dados estão resumidas na Tabela 1.

- **Dados Sintéticos:** Para avaliar os algoritmos em um ambiente controlado, três bases de dados bidimensionais foram geradas (28k2d, 40k2d, 100k15c). Os clusters foram gerados a partir de distribuições Gaussianas, com centroides, desvios-padrão e número de pontos variando para criar grupos com diferentes densidades e tamanhos. O fenômeno de *concept drift* foi simulado através da adição e remoção de clusters em diferentes *timestamps* (intervalos de processamento online). Por exemplo, um novo cluster poderia surgir no segundo *timestamp* enquanto outro, existente desde o início, deixava de ser gerado. Essa abordagem permite um controle preciso sobre a evolução dos grupos e facilita a avaliação visual e quantitativa do comportamento das estruturas de sumarização (*DBs* e *MCs*) ao longo do fluxo. O código-fonte para a geração destes dados será disponibilizado em apêndice para garantir a reprodutibilidade.
- **Dados Reais:** Quatro conjuntos de dados reais e amplamente utilizados na literatura foram selecionados:
  - KDD Cup'99 (kddcup): Contém dados de detecção de intrusão em redes, com 41 atributos (34 contínuos, nominais convertidos para ordinais) e 494.021 objetos. Link para o dataset: <https://www.kaggle.com/datasets/galaxyh/kdd-cup-1999-data/data>.
  - Forest Covertype (covertype): Descreve tipos de cobertura florestal, com 10 atributos contínuos e 581.012 objetos.

– Poker-Hand (Poker-Hand ou pk_250_5k): Contém dados de mãos de pôquer, com 10 atributos. Da base original de 900.000 objetos, utilizamos as primeiras 250.000 instâncias nos experimentos.

– Hyper Plane Stream (Hyper ou hyper_v): Um fluxo de dados sintético com *concept drift* contínuo, contendo 100.000 instâncias e 10 atributos. Link para o dataset: <https://www.cse.fau.edu/~xqzhu/Stream/hyperP.arff>.

Tabela 1 – Configurações e características das bases de dados para os experimentos.

Base de Dados	**Nº objetos por timestamps**[1]	$\lambda_{\text{HASTREAM-DB}}$	$\lambda_{\text{HASTREAM}}$	**Dimensões**	**Tamanho**	**Tipo**
28k2d	7000	0.035	0.001	2	28000	sintético
40k2d	10000	0.02	0.02	2	40000	sintético
100k15c	10000	0.001	0.001	2	100000	sintético
covertype	20000	0.01	0.003	10	581012	dados reais
kddcup	20000	0.0025	0.0025	41	494021	dados reais
Poker-Hand	5000	0.001	0.012	10	250000	dados reais
Hyper	2000	0.001	0.012	10	100000	dados reais

Os parâmetros dos algoritmos foram definidos para garantir uma comparação justa. O parâmetro de decaimento λ (Equação 2.1) foi ajustado individualmente para cada algoritmo e base de dados, conforme especificado na Tabela 1. Essa calibração foi necessária porque a natureza da sumarização (DBs vs. MCs) afeta a retenção de informação; valores diferentes de λ foram usados para buscar um equilíbrio funcional onde ambos os algoritmos mantivessem um modelo online de tamanho comparável (aproximadamente 10% de *p-DB*s ou *p-MC*s em relação aos objetos por *timestamp*). A velocidade do fluxo (v), definida como 100 objetos por unidade de tempo, serve como uma constante de normalização no cálculo do decaimento temporal, garantindo que o fator de decaimento $f(t)$ seja consistente entre as execuções. Os parâmetros $\beta = 0.75$ e $\mu = 2$ foram mantidos fixos, seguindo recomendações comuns na literatura para algoritmos como o DenStream.

4.1.2 Avaliação e Métricas

Para cada chamada da fase offline, tanto no HASTREAM-DB quanto no HASTREAM, são geradas múltiplas partições de grupos. Isso é feito variando-se o parâmetro $MinPts$ de um valor máximo ($MinPts_{max} = 200$ nestes experimentos) até 2, em passos decrescentes de 2. Para cada valor de $MinPts$, uma Árvore Geradora Mínima (MST) é extraída do grafo de alcançabilidade mútua, e uma partição de grupos é obtida. Isso resulta em aproximadamente cem soluções de agrupamento por chamada offline para cada algoritmo, permitindo uma análise abrangente da hierarquia de densidade.

Para avaliar a qualidade dessas partições, utilizamos o Adjusted Rand Index (ARI) [Hubert e Arabie 1985]. O ARI mede a similaridade entre duas partições de dados, corrigindo para a concordância por acaso. Um valor de ARI próximo de 1 indica alta similaridade, enquanto valores próximos de 0 indicam similaridade aleatória.

O cerne deste trabalho é avaliar a fidelidade da estrutura de sumarização. Portanto, o *ground truth* (referência) para o cálculo do ARI não é um rótulo pré-definido, mas sim o resultado "ideal"que o próprio algoritmo hierárquico produziria se tivesse acesso aos dados brutos representados pela sumarização. Esta abordagem nos permite medir diretamente o quanto de informação estrutural foi perdida no processo de sumarização. O protocolo é o seguinte para cada chamada offline j e cada valor de $MinPts_i$:

1. Para HASTREAM-DB: Extraímos os objetos originais que foram sumarizados nas *p-DB*s ativas. Aplicamos HDBSCAN**($MinPts_i$) a esses objetos para obter a partição de referência ideal, $P_{HDBSCAN^*\_DB}(j, MinPts_i)$. Calculamos então o ARI entre a partição gerada pelo HASTREAM-DB e essa referência: $ARI(P_{HASTREAM-DB}(j, MinPts_i), P_{HD}$
2. Para HASTREAM: Repetimos o processo, extraindo os objetos originais das *p-MC*s ativas para gerar a referência $P_{HDBSCAN^*\_MC}(j, MinPts_i)$ e calculando $ARI(P_{HASTREAM}(j, Mi$

Após todas as chamadas offline ao longo do fluxo, calculamos a média e o desvio padrão dos ARIs obtidos para cada valor de $MinPts_i$ em cada base de dados.

4.2 Comparação entre as Partições HASTREAM-DB e HASTREAM

Os resultados da comparação da qualidade de agrupamento, medidos pelo ARI médio ao longo dos *timestamps* para cada valor de $MinPts_i$, são apresentados na Figura 5.

Conforme observado na Figura 5, as médias de ARI para o HASTREAM-DB (identificado como CoreStream nas legendas, em barras azuis) são consistentemente superiores às do HASTREAM (barras laranjas) em todos os conjuntos de dados analisados, com desvios padrão semelhantes. Isso indica que a qualidade dos grupos gerados pelo HASTREAM-DB, utilizando *DBs* para sumarização, foi superior à do HASTREAM, que utiliza *MCs*.

A análise detalhada revela que, com os cálculos de densidade baseados nas *DBs*, o HASTREAM-DB conseguiu identificar *grupos* mais internos e densos na hierarquia. Isso ocorre porque as densidades calculadas na hierarquia com *DBs* estão mais próximas das densidades reais entre os objetos, devido à preservação da informação interna pelas *DBs*. Em contraste, a hierarquia de *grupos* gerada pelo HASTREAM com *MCs* sofre com a distorção da densidade, pois os *MCs* são tratados como pseudo-pontos. Isso leva o HASTREAM a selecionar, na maioria das partições $MinPts$, os *grupos* do topo da hierarquia (macro *grupos*), falhando em identificar estruturas mais finas e densas.

Nos conjuntos de dados sintéticos bidimensionais (28k2d e 40k2d), onde os *grupos* foram projetados para ter diferentes densidades e evoluir no fluxo, a diferença nos ARIs médios é particularmente proeminente. O HASTREAM-DB demonstrou maior capacidade

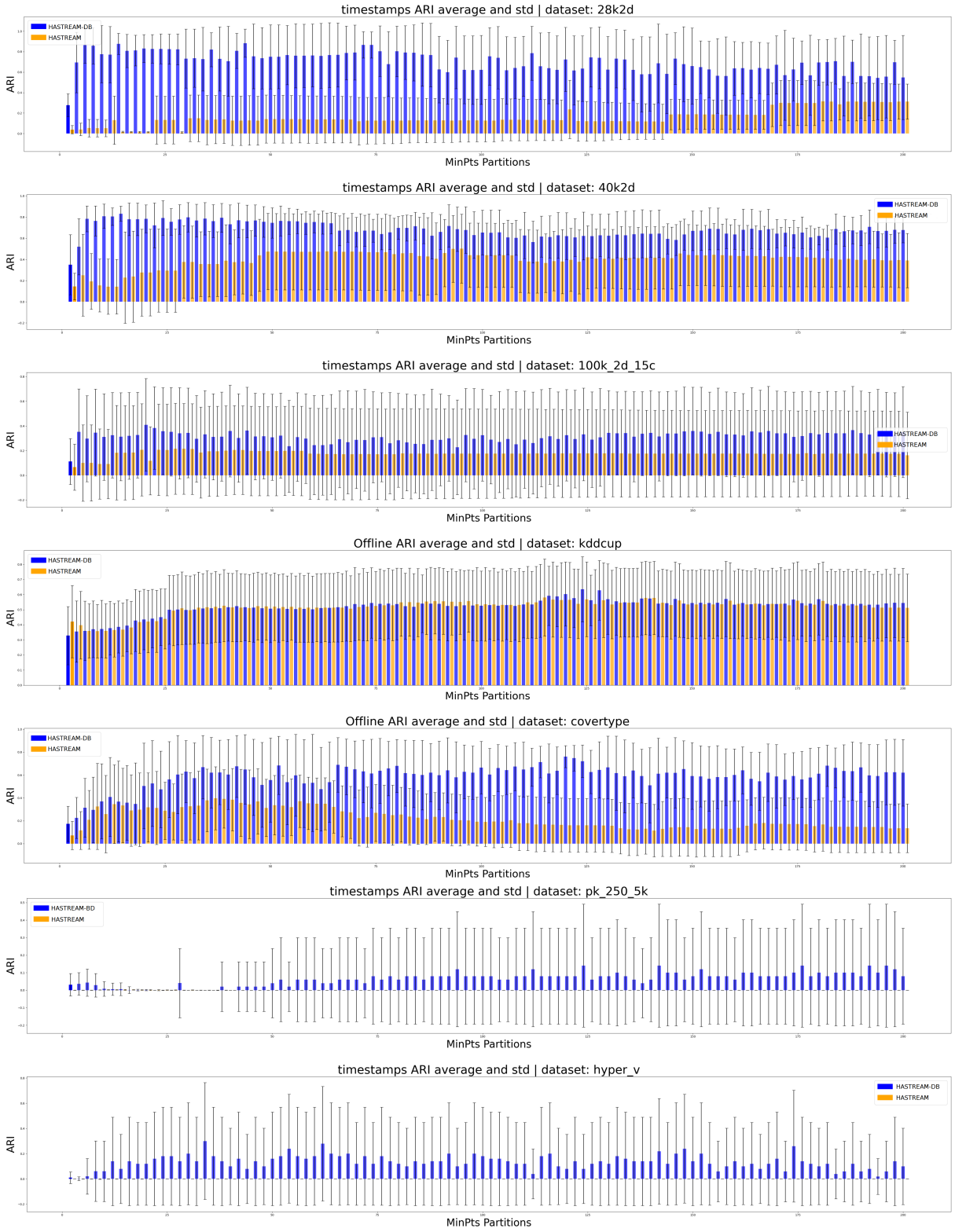

Figura 5 – Comparação das partições HASTREAM-DB (CoreStream) e HASTREAM. Média e Desvio Padrão do ARI aplicado aos *timestamps* para diferentes valores de $MinPts_i$ em cada conjunto de dados.

de encontrar os *grupos* internos e mais densos, refletindo melhor a estrutura hierárquica subjacente. O HASTREAM, por sua vez, devido à baixa densidade de conexão entre *MCs* para valores altos de $MinPts$, frequentemente identificava apenas os macro *grupos*.

Em conjuntos de dados reais, embora os ARIs médios de ambos os algoritmos tenham sido mais próximos em algumas chamadas (especialmente quando as densidades dos *grupos* não estavam bem distribuídas ou os *grupos* não estavam bem separados), o HASTREAM-DB ainda apresentou uma melhoria geral na qualidade dos agrupamentos finais. Notavelmente, nos conjuntos de dados Poker-Hand (PK) e Hyper Plane (Hyper), o HASTREAM obteve médias de ARI próximas de 0, indicando uma distorção quase total das densidades originais ao gerar as hierarquias com *MCs*. O HASTREAM-DB conseguiu melhorar significativamente os resultados para essas bases, aumentando a fidelidade da representação da densidade na hierarquia e, em algumas chamadas offline, gerando partições com alta similaridade às partições do HDBSCAN** aplicadas aos dados sumarizados.

Esses resultados sustentam a hipótese de que a sumarização com *DBs* adaptadas, ao preservar melhor a informação espacial e de densidade interna, permite que algoritmos hierárquicos como o HASTREAM-DB construam hierarquias de *grupos* mais representativas e de maior qualidade em comparação com a abordagem baseada em *MCs*.

5 Conclusão

Este trabalho de conclusão de curso propôs e avaliou o HASTREAM-DB, uma nova abordagem para o agrupamento hierárquico baseado em densidade em Fluxos de Dados (Fluxo de Dados (FD)). O problema central combatido foi a distorção das informações de densidade e estrutura espacial causada pela sumarização via *MCs*, que compromete a qualidade de algoritmos como o HASTREAM. A solução desenvolvida demonstrou que a utilização de *DBs* adaptadas, por preservarem melhor a informação local, representa uma alternativa promissora e eficaz, contribuindo para a obtenção de resultados de agrupamento mais precisos e robustos em ambientes dinâmicos.

5.1 Alcance dos Objetivos

Para alcançar o resultado principal, foram estabelecidos objetivos secundários que guiaram o desenvolvimento do trabalho. A seguir, detalha-se como cada um foi atingido:

- **Estudar e adaptar as *DBs* para o contexto de Fluxo de Dados:** Este objetivo foi plenamente alcançado no Capítulo 3 (Seção 3.1.1). A estrutura da *DB* estática foi reformulada para operar sob o modelo de janela amortecida, com suas propriedades, como *extent* e *nnDist*, sendo redefinidas com base em estatísticas ponderadas (Equações 3.1 e 3.2), garantindo a representação fiel da densidade local ao longo da evolução do fluxo.

- **Redefinir as métricas de distância para as *DBs* adaptadas:** Conforme detalhado na Seção 3.2, as métricas de distância essenciais para algoritmos hierárquicos baseados em densidade foram redefinidas. Foi proposta uma nova formulação para a Distância Core (Equação 3.5) e, consequentemente, para a Distância de Alcançabilidade Mútua (MRD), preservando a semântica de densidade e vizinhança exigida pelo Hierarchical DBSCAN* (HDBSCAN*)* sobre as novas estruturas de sumarização.

- **Desenvolver mecanismos para a manutenção online das *DBs*:** O processo de manutenção online foi descrito na Seção 3.1. Ele inclui a incorporação incremental de novos pontos, a aplicação periódica do decaimento temporal e o gerenciamento do ciclo de vida das *DBs*, que são classificadas como *Potential Data Bubble* (p-DB) ou *Outlier Data Bubble* (o-DB), garantindo que o modelo online permaneça compacto e focado nas regiões de dados mais relevantes.

- **Integrar a representação sumarizada à fase offline do HASTREAM:** A integração foi realizada na fase offline do HASTREAM-DB (Seção 3.2), onde o

conjunto de p-DBs mantido pela fase online foi utilizado como base para a construção da hierarquia de grupos. As métricas de distância adaptadas foram empregadas para construir o grafo de alcançabilidade e a MST, a partir da qual a hierarquia é extraída.

- **Realizar uma avaliação experimental comparativa:** No Capítulo 4, foi conduzida uma avaliação experimental robusta, comparando o HASTREAM-DB com o HASTREAM original. Os resultados, medidos pelo Adjusted Rand Index (ARI), demonstraram consistentemente a superioridade da abordagem proposta em diversos conjuntos de dados, validando a hipótese de que a sumarização por *DBs* mitiga as distorções de densidade e leva a um agrupamento de maior qualidade.

5.2 Limitações do Trabalho

A principal limitação deste trabalho reside no foco exclusivo na qualidade do agrupamento, em detrimento de uma análise de desempenho computacional. A implementação do HASTREAM-DB foi desenvolvida como uma prova de conceito, e otimizações de performance não fizeram parte do escopo. As métricas de distância para *DBs* são inerentemente mais complexas de calcular do que a simples distância Euclidiana entre centroides de *MCs*, o que pode resultar em um tempo de execução maior. Uma análise formal da complexidade de tempo e memória do algoritmo proposto não foi realizada, constituindo uma limitação para uma avaliação completa da sua viabilidade em cenários com restrições de tempo extremamente rígidas.

5.3 Trabalhos Futuros

Com base nos resultados e nas limitações identificadas, diversas direções para trabalhos futuros podem ser exploradas:

- **Análise de Desempenho e Otimização:** Realizar uma análise de complexidade computacional formal para o HASTREAM-DB e desenvolver uma versão otimizada do algoritmo, visando reduzir o tempo de execução e o consumo de memória para torná-lo competitivo também em termos de performance.
- **Generalização da Abordagem:** Investigar a aplicação da sumarização por *DBs* adaptadas a outros algoritmos de agrupamento em fluxo de dados baseados em densidade, como o DenStream, para avaliar se os benefícios de qualidade se estendem a outras arquiteturas algorítmicas.
- **Estudo de Sensibilidade de Parâmetros:** Conduzir um estudo aprofundado sobre o impacto de parâmetros chave, como a taxa de decaimento λ e o número de

vizinhos k para o cálculo da *nnDist* (fixado em $k = 1$ neste trabalho), na qualidade e estabilidade dos resultados.

- **Análise de Dados com Múltiplas Densidades e Formatos:** Avaliar o desempenho do HASTREAM-DB em conjuntos de dados sintéticos projetados especificamente com *clusters* de formatos mais complexos (e.g., não convexos) e múltiplas escalas de densidade, a fim de explorar os limites da capacidade de representação das *DBs*.

Espera-se que este trabalho sirva como uma base sólida para futuras investigações sobre técnicas de sumarização que preservem a estrutura e a densidade em ambientes de fluxo de dados, impulsionando o desenvolvimento de algoritmos de agrupamento mais acurados e robustos.

Referências

AGGARWAL, C. C. et al. A framework for clustering evolving data streams. In: ELSEVIER. *Proceedings 2003 VLDB conference*. [S.l.], 2003. p. 81–92. Citado na página 12.

BREUNIG, M. M. et al. Data bubbles: Quality preserving performance boosting for hierarchical clustering. In: *Proceedings of the 2001 ACM SIGMOD international conference on Management of data*. [S.l.: s.n.], 2001. p. 79–90. Citado 3 vezes nas páginas 12, 18 e 20.

CAMPELLO, R. J. et al. A framework for semi-supervised and unsupervised optimal extraction of clusters from hierarchies. *Data Mining and Knowledge Discovery*, Springer, v. 27, p. 344–371, 2013. Citado na página 18.

CAMPELLO, R. J. G. B. et al. Hierarchical density estimates for data clustering, visualization, and outlier detection. *ACM Trans. Knowl. Discov. Data*, Association for Computing Machinery, New York, NY, USA, v. 10, n. 1, jul 2015. ISSN 1556-4681. Citado 2 vezes nas páginas 12 e 17.

CAO, F. et al. Density-based clustering over an evolving data stream with noise. In: SIAM. *Proceedings of the 2006 SIAM international conference on data mining*. [S.l.], 2006. p. 328–339. Citado 3 vezes nas páginas 12, 17 e 23.

DJONLAGIC, I. et al. Macro and micro sleep architecture and cognitive performance in older adults. *Nature human behaviour*, Nature Publishing Group UK London, v. 5, n. 1, p. 123–145, 2021. Citado na página 14.

GAMA, J. *Knowledge discovery from data streams*. [S.l.], 2010. Disponível em: <http://www.liaad.up.pt/area/jgama/DataStreamsCRC.pdf>. Acesso em: 14.10.2021. Citado na página 12.

HASSANI, M.; SPAUS, P.; SEIDL, T. Adaptive multiple-resolution stream clustering. In: SPRINGER. *Machine Learning and Data Mining in Pattern Recognition: 10th International Conference, MLDM 2014, St. Petersburg, Russia, July 21-24, 2014. Proceedings 10*. [S.l.], 2014. p. 134–148. Citado 2 vezes nas páginas 13 e 17.

HUBERT, L.; ARABIE, P. Comparing partitions. *Journal of classification*, Springer, v. 2, p. 193–218, 1985. Citado na página 27.

MICCIO, L. A.; SCHWARTZ, G. A. Mapping chemical structure–glass transition temperature relationship through artificial intelligence. *Macromolecules*, v. 54, n. 4, p. 1811–1817, 2021. Citado na página 14.

ZERHARI, B.; LAHCEN, A. A.; MOULINE, S. Big data clustering: Algorithms and challenges. In: . [S.l.: s.n.], 2015. Citado na página 14.