GAS-DBSCAN: A Grid-based Adaptive Sampling Method for DBSCAN Clustering under Skewed Data Distribution

Yu Wang, Junhua Fang(✉), Jiayi Li, and Pingfu Chao

Department of Computer Science and Technology, Soochow University, Suzhou, China
20224227074@stu.suda.edu.cn, jhfang@suda.edu.cn,
20235227131@stu.suda.edu.cn, pfchao@suda.edu.cn

Abstract. Clustering is one of the important methods for knowledge acquisition, which is the process of analyzing data by grouping instances based on their similarities and dissimilarities. DBSCAN is a notable density-based clustering algorithm, but its reliance on implicit neighborhood queries causes significant computational overhead, particularly in challenging scenarios. While there have been proposals to alleviate this overhead through sampling techniques, traditional sampling approaches tend to be random. Randomness in sampling can overlook smaller, less dense clusters in skewed distributions, ultimately compromising clustering precision and reliability. However, more sophisticated sampling methods, while introducing additional computation, may not always enhance efficiency. We propose Grid-based Adaptive Sampling for DBSCAN, named GAS-DBSCAN, a method specifically designed to address sampling imbalance in density-based clustering. Through grid-based adaptive mechanisms, our approach significantly improves the detection of smaller clusters. Experimental results on various datasets demonstrate a threefold improvement in sampling accuracy and 267% in clustering accuracy compared to existing methods under skewed distributions, with theoretical analysis further validating its efficiency.

Keywords: DBSCAN · Grid mapping · Adaptive sampling · Grid-based clustering · Skewed distribution

1 Introduction

Background. In the digital era, data accumulation continues to accelerate at a staggering pace [12,18], driving the need to extract meaningful patterns from vast information repositories [14,16]. Clustering, which groups similar objects into distinct clusters [10], plays a crucial role in uncovering latent structures within complex datasets. This approach has demonstrated its versatility across diverse domains, from customer segmentation [11] and computer vision to bioinformatics, establishing itself as a fundamental tool for deriving insights from unstructured data. Among clustering techniques, density-based methods excel in identifying arbitrary-shaped clusters within large datasets. DBSCAN [9], a

prominent example, forms clusters by identifying core points through density evaluation and constructing neighborhood graphs to determine cluster membership.

Motivations. The main computational overhead of DBSCAN arises from the need to compute the ε-neighborhood for each point, which necessitates traversing the entire dataset. Consequently, the time complexity of DBSCAN is $O(n^2)$ in the worst case, where n represents the dataset size. To mitigate such expensive overhead, some methods have proposed sampling strategies. For instance, DBSCAN++ [12] proposes to sample a subset of size m from the entire dataset (i.e., the size is n) as a candidate set that potentially includes core points. This approach offers a notable enhancement in the runtime efficiency, reducing it from $O(n^2)$ to $O(mn)$. However, in skewed datasets with limited sampling, points tend to concentrate in larger clusters, overlooking smaller ones (Figure 1). For instance, with clusters of 95 and 5 points respectively, uniform sampling of 5 points yields a mere $0.06^5 = 7.776 \times 10^{-7}$ probability of detecting the smaller cluster (Figure 1 (a)). Our grid-based approach (Figure 1 (c)) addresses this by distributing samples across grids, ensuring comprehensive cluster detection while maintaining $O(mn)$ complexity.

Challenges. To achieve an efficient clustering method, the system needs to have lightweight computational methods for handling massive amounts of data. At the same time, the system also needs to address the unreliability of clustering results in the case of skewed data distribution. The specific description is as follows.

❶ **The massive input data poses challenges to the effectiveness of clustering algorithms.** The computational bottleneck of naive DBSCAN lies in neighborhood searches, and each neighborhood search for a single point requires $O(n)$ time. While optimization techniques like KD-tree have been proposed, these algorithms still maintain a worst-case overhead of $O(n)$, resulting in an inevitable $O(n^2)$ worst-case complexity.

❷ **Skewed data distribution challenges the effectiveness of clustering algorithms.** Existing sampling approaches such as uniform sampling [7] and K-center attempt to reduce computational overhead through dataset subsets, yet face significant challenges with skewed data. While uniform sampling offers computational efficiency, it often fails to detect smaller clusters at low sampling rates. Although K-center provides better accuracy, its $O(mn)$ initialization complexity limits its practical advantages. As illustrated in Figure 1, both methods struggle with accuracy in skewed distributions, especially for small clusters.

Solution. To address the above challenges, we propose a Grid-based Adaptive Sampling method based on DBSCAN (GAS-DBSCAN), which not only enhances clustering quality but also provides guarantees for data consistency. It addresses the above challenges in clustering and offers the following contributions:

1. We propose a grid-based adaptive sampling method to improve clustering accuracy in skewed data, which efficiently performs clustering while maintaining a time complexity of $O(mn)$.

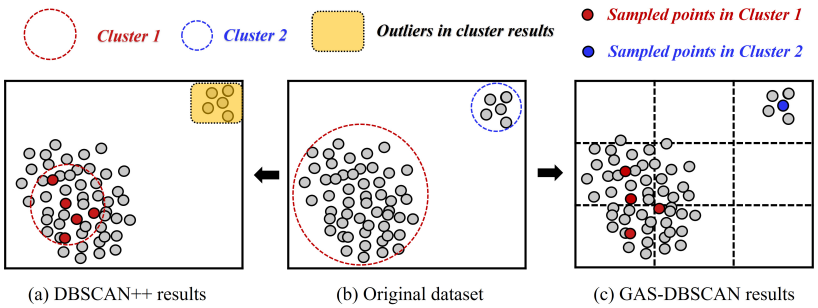


Fig. 1. Comparison of sampling methods in skewed data distribution: (a) uniform sampling fails to detect the smaller cluster, (b) original data with two clusters, and (c) proposed grid-based sampling achieves complete cluster detection through evenly distributed sampling points.

2. We theoretically demonstrate that the grid-based adaptive sampling method could balance the sampling preference at cluster levels, which further validates the accuracy of clustering.
3. We propose the Unbiased Level of Cluster Sampling (ULCS) as an evaluation metric, which allows us to measure the degree of sampling balance at the cluster level. ULCS innovatively provides a means to assess the clustering quality from the perspective of sampling.
4. Our comprehensive experiments conducted on both synthetic and real-world datasets have demonstrated that our proposed method surpasses current leading methods across various metrics, including ARI, AMI, etc.

2 Related Work

This section provides a comprehensive review of the performance optimization strategies for enhancing the clustering efficiency of DBSCAN.

Grid-based DBSCAN. A grid-based method proposed by Boonchoo et al. [3] introduces the Hierarchical Grid-Based (HGB) structure with an index for non-empty grids to improve neighboring grid query efficiency. A grid-based method proposed by Wang et al. [6] reveals that traditional grid techniques in Fast-DBSCAN and ρ-approximate DBSCAN variants suffer from redundant distance computations in high-dimensional spaces. SW-DBSCAN [17] employs grid-based partitioning to identify dense regions. It uses sliding windows to merge local clustering results across varying density ranges.

Sampling-based DBSCAN. A sampling-based method proposed by Aggarwal et al. [1] demonstrates that adaptively sampling $O(k)$ centers can achieve a constant factor bi-criteria approximation for k-means clustering, extending the $O(\log k)$ approximation of k-means++. Jang et al. [12] suggest a method to obtain reliable clustering results without performing neighborhood searches on each

data point in the dataset. Additionally, Jiang et al. [14] propose that subsampling some edges of the ε-neighborhood graph can significantly improve speed and reduce memory consumption while maintaining competitive performance.

Parallel-based DBSCAN. There are also some methods focusing on parallel computing. He et al. [11] utilize MapReduce and reasonable data partitioning to reduce I/O frequency and enhance operational efficiency. Andrade et al. [2] leverage GPU-based parallel technology, achieving over a 100-fold speedup in DBSCAN clustering. A locality-sensitive hashing (LSH) based method proposed by Lovas et al. [15] integrates LSH indexing with DBSCAN clustering, achieving significant speedup for high-dimensional data through parallel processing while maintaining clustering accuracy.

3 Preliminaries

3.1 Basic Concepts

Given an i.i.d. sample set $\mathcal{X} = \{x_1, ..., x_n\}$, where n is the data size, ε is defined as the neighborhood radius ($\varepsilon > 0$), and $minPts$ is the threshold for determining the core points ($minPts > 0$).

Definition 1 (ε-radius neighborhood). *ε-radius neighborhood is defined as a ball neighborhood $B(x, \varepsilon) \cap \mathcal{X}$ centered at point x with radius ε, where $B(x, \varepsilon) := \{x' : |x - x'| \leq \varepsilon\}$.*

Definition 2 (Core point). *The point $x \in \mathcal{X}$ is a core point if $|B(x, \varepsilon) \cap \mathcal{X}| \geq minPts$. In other words, a core point is a point where there are at least $minPts$ points within its ε-radius neighborhood.*

Definition 3 (Border point). *A point $x \in \mathcal{X}$ is a border point if $|B(x, \varepsilon) \cap \mathcal{X}| < minPts$ and it is within the ε-radius neighborhood of a core point.*

Definition 4 (Outliers or noise points). *Points that are neither core points nor border points are defined as outliers or noise points.*

Definition 5 (Directly density-reachable). *Given a point x_i, it is directly density-reachable from x_j if x_j is a core point and x_i is in the ε-radius neighborhood of x_j. A core point is density-reachable to every point in its ε-neighborhood.*

Definition 6 (Density-connected). *Point x_i and x_j are density-connected if x_i and x_j are both density-reachable from a point x_k.*

3.2 Problem Statement

Based on the above definitions, the problem can be stated as: Given a dataset in D-dimension $\mathcal{X}^{(D)}$, ε, and $minPts$, we perform GAS-DBSCAN to identify dense regions and assign all the points into clusters or outliers. This issue can be formally described as:

Table 1. Notations used in this paper.

Symbol	Description	Symbol	Description
$\mathcal{X}$	The dataset	n	# of points in $\mathcal{X}$
D	The dimension of the dataset	m	Sample size
λ	A specific density threshold	ε	Neighborhood radius
$L_f(\lambda)$	$L_f(\lambda) := \{x \in \mathcal{X} : f(x) \geq \lambda\}$	$f(x)$	Density at point x
$\lvert x - x' \rvert$	Euclidean distance between x, x'	$d(x, \mathcal{X})$	$\inf_{x' \in \mathcal{X}} \lvert x - x' \rvert$
β	Boundary density smoothness of level-sets	δ	Confidence degree

$$\begin{aligned} & \text{GAS-DBSCAN}(\mathcal{X}^{(D)}) \leftarrow \mathcal{C} = \{C_1, C_2, ..., Outliers\} \\ s.t., \quad & \forall x_i, x_j \in C_k, \quad x_i \text{ and } x_j \text{ are density-connected.} \\ & \forall C_i, C_j (i \neq j), \quad C_i \cap C_j = \emptyset \end{aligned} \tag{1}$$

The two conditions delineated in the aforementioned description indicate that the computation of clustering must ensure accuracy. Building upon this foundation, the computation of clustering must also possess efficiency. In other words, even facing skewed data, namely the necessary ε-neighborhood query is only performed on part of the data, the clustering quality should still be guaranteed.

4 GAS-DBSCAN

In this section, we first present the framework of GAS-DBSCAN in Section 4.1, which explains how the method performs sampling and clustering with our grid-based adaptive sampling strategy. Then, we provide the detailed procedure of our algorithm in Section 4.2. Additionally, theoretical analysis in Sections 4.3 and 4.4 demonstrates that GAS-DBSCAN is a consistent estimator, which ensures the precision and reliability of our proposal. Table 1 shows the symbols mentioned in this paper and their corresponding definitions. Notably, the expression $\inf_{x' \in \mathcal{X}} \lvert x - x' \rvert$ represents the infimum (the greatest lower bound) of the Euclidean distances between the point x and all points in set $\mathcal{X}$.

4.1 Overview

Figure 2 depicts the GAS-DBSCAN framework. Taking three parameters and a dataset as inputs, it unfolds in three stages: ① **Grid Mapping**: Partition data space into grids by distribution and input parameters, structuring data for later operations. ② **Grid-based Sampling**: Use an adaptive strategy combining fixed and distribution-sensitive methods to ensure full cluster coverage. ③ **Clustering**: Detect core points in the sampled subset and apply DBSCAN. This reduces overhead while maintaining result integrity. This approach streamlines clustering, enhancing efficiency without quality loss.

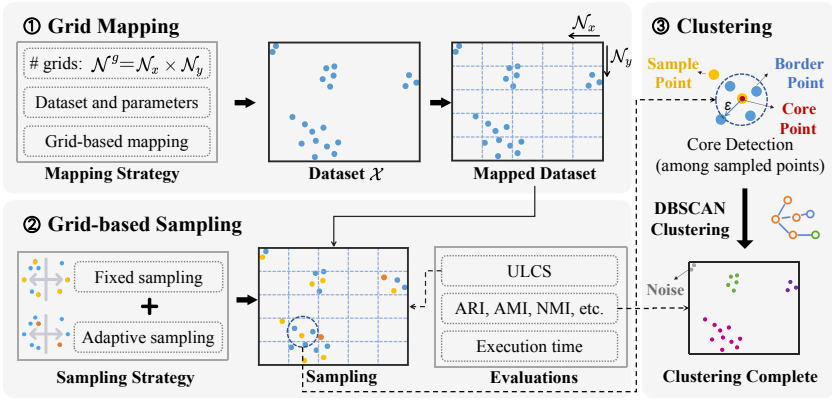


Fig. 2. The framework of GAS-DBSCAN.

As illustrated in Figure 3, GAS-DBSCAN comprises three phases. Initially, it partitions the dataset into grids to ensure a more uniform distribution of sampling points, facilitating efficient point access. Next, GAS-DBSCAN performs adaptive sampling within each grid, ensuring that even smaller clusters obtain sampling points, thus increasing their detection probability. Denoting the sampled subset of the entire dataset as S, GAS-DBSCAN then identifies core points by calculating ε-radius neighborhoods in S, then constructs a graph G connecting density-connected core points, and extracts its connected components as clusters. The detailed procedural description is as follows:

1. **Configuration.** Given input parameters ε, $minPts$, and the number of grids $\mathcal{N}^g$, the blue points in Figure 3 (a) represent the original dataset, while the yellow points are the samples obtained after applying the sampling strategy. The parameter ε is empirically set to balance clustering results. A large ε may merge distinct clusters, while a small ε may over-partition them. Input parameters are manually adjusted based on cluster distribution and performance metrics. Further details are provided in Section 5.
2. **Grid Mapping.** Subsequently, the dataset is distributed to corresponding grids based on their location and the input $\mathcal{N}^g$. This data mapping is accomplished using the novel strategy proposed in this paper, effectively completing the grid mapping process, as illustrated in Figure 3 (b). For instance, given the dataset shown in Figure 3 (a), we set $minPts = 3$, $\mathcal{N}^g$, and the value of ε is marked in the figure. Following the aforementioned method, a 5×5 grid is generated based on these parameters, as shown in Figure 3 (b). The data points are then mapped to their corresponding grid cells, thus completing the data-to-grid mapping process.
3. **Sampling.** Following the example in Figure 3, grid-based adaptive sampling is performed. As shown in Figure 3 (c), the number of points to sample from each non-empty grid is calculated based on the point counts and parameters. The final sampling size, determined by global sampling size m, grid point

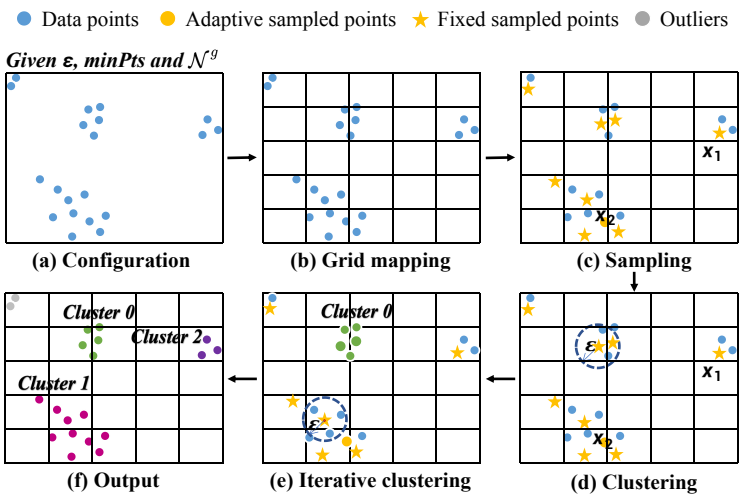


Fig. 3. The workflow of GAS-DBSCAN.

count, and total data size n, combines a fixed value with an adaptive size. First, a fixed number of points are sampled within each grid to ensure cluster coverage. For example, the point x_1 in Figure 3 (c) is one such sampled point. The calculation strategy is detailed in Algorithm 1. Then, points are randomly sampled from each grid and added to the sample set, and this process is repeated for all grids to complete the global sampling, such as point x_2 in Figure 3 (c).

4. **Clustering.** Following the sampling process, core points that have at least $minPts$ neighbors within its ε-neighborhood, while other points are marked as border points. All these points form a graph G where core points serve as vertices connected by edges, as illustrated in Figure 3 (d) to (f), demonstrating the sequential process of cluster formation.
5. **Output.** According to the parameters and sampling results, one of the two data points in the upper left corner of Figure 3 (e) is sampled. Since the sampled yellow point is not a core point, no edges are added in G with other points, and the two points do not form connected components or clusters. After processing, each connected component is marked as a cluster, with Cluster 0 in green and Cluster 1 in purple in Figure 3 (f). Points not in any connected component are considered outliers, as shown in Figure 3 (f).

4.2 Algorithm Procedure

Given a dataset in D-dimension $\mathcal{X}^{(D)}$, along with the sampling size m, ε, $minPts$, and grid control parameters (Input of Algorithm 1), we use two-dimensional data to explain the algorithm procedure. The method first maps each point to corresponding two-dimensional rectangular grids based on the input parameters

(line 1). Based on the specified parameter $\mathcal{N}^g$, we compute the number of grids for each dimension. Specifically, we take the two greatest common factors of $\mathcal{N}^g$ as the grid sizes of x-dimension and y-dimension, denoted as $\mathcal{N}_x \times \mathcal{N}_y$. For simplicity, we take the x-dimension as an example. For each dimension (e.g., x-dimension), we calculate the grid side length $\mathcal{L}_x = \frac{x_{max} - x_{min}}{\mathcal{N}_x}$. The algorithm determines sampling points per grid by combining fixed and adaptive sampling sizes, capped by the grid's point count (lines 3∼6). Random sampling is then performed within non-empty grids to obtain set S (line 7), followed by a neighborhood search to identify core points (line 9). Finally, a graph G is constructed by connecting core points with their border points (lines 10∼13), with connected components forming the final clusters (line 14).

Algorithm 1 GAS-DBSCAN

Input: dataset $\mathcal{X}$, $\mathcal{N}^g$, $\mathcal{N}^{\text{fixed}}$, m, $\varepsilon, minPts$
Output: connected components of G
1: Map the dataset $\mathcal{X}$ to the grids of $\mathcal{L}_x \times \mathcal{L}_y$
2: $S \leftarrow \emptyset$
3: **for** each grid **do**
4: $size \leftarrow$ the number of points in this grid
5: $\mathcal{N}^{\text{adaptive}} \leftarrow size \times \lceil \frac{m}{n} \rceil$
/** Under the constraints in Equation (5) **/
6: $\mathcal{N} \leftarrow \min(\mathcal{N}^{\text{fixed}} + \mathcal{N}^{\text{adaptive}}, size)$
7: S += sample $\mathcal{N}$ points from the current grid
8: **end for**
9: $C \leftarrow$all core-points in S
10: $G \leftarrow$ empty graph
11: **for** $c \in C$ **do**
12: Build an edge in G from c to each point in $\mathcal{X} \cap B(c, \varepsilon)$
13: **end for**
14: **return** connected components of G

4.3 Statistical Consistency Estimation

To ensure fair and comprehensive sampling in GAS-DBSCAN, the sampling threshold is designed based on probabilistic principles. An estimation is considered statistically consistent if the probability of $|p_m(x_1, \cdots, x_m) - p_0| < \epsilon$ approaches 1 as the sample size m increases to infinity for any parameter value p_0 within the range, where $\epsilon > 0$ [8]. Consistency implies that accurate clustering results can theoretically be achieved using a subset of sampling points, establishing the algorithm's theoretical reliability. We further demonstrate that GAS-DBSCAN is a consistent estimator of density level-sets. The final grid-based adaptive sampling result combines uniform and adaptive grid sampling:
❶ **Uniform sampling.** Sample m_1 points uniformly to obtain the sampled subset S_1, where these sampled points are distributed within the global grids

where there exist data points. This sampling procedure is identical to that of the uniform sampling in DBSCAN++.

❷ **Grid-based sampling.** Sample m_2 points from the grids that do not contain sampled points in S_1 to obtain another sampled subset S_2 (ensuring at least one point is sampled per grid). The maximum number of these additional sampled points is $\mathcal{N}^g$, which is equal to the number of grids.

Corresponding to Algorithm 1, m_1 represents the $\mathcal{N}^{\text{adaptive}}$ value calculated in line 5, while m_2 is derived from $\mathcal{N}^{\text{fixed}}$. The subsequent proof demonstrates that the clustering result obtained using these sampling parameters achieves the minimax optimal estimation. To further analyze, we have the following definition of *Hausdorff distance*, where A represents a dataset:

$$d_H(A, A') = \max\{\max_{x \in A} d(x, A'), \max_{x' \in A'} d(x', A)\} \tag{2}$$

, and we have a constant C_1 sufficiently large that,

$$d_H(\widehat{L_f(\lambda)}, L_f(\lambda)) \leq C_1 \cdot (\max_{x \in \widehat{L_f(\lambda)}} d(x, L_f(\lambda)) + \max_{x \in L_f(\lambda)} d(x, \widehat{L_f(\lambda)})). \tag{3}$$

Based on upper bounds in Equations (2) and (3), we now demonstrate that GAS-DBSCAN achieves the optimal Hausdorff error bound:

❶ The first part of the expression, $\max_{x \in \widehat{L_f(\lambda)}} d(x, L_f(\lambda))$, guarantees that our estimated results are generated from S_1 and S_2. The level set $L_f(\lambda)$ obtained by original DBSCAN and the core point set $\widehat{L_f(\lambda)}$ identified through grid initialization are bounded by $O(C_{\delta,n}^{2/\beta} \cdot minPts^{-1/2\beta})$, as demonstrated in [13].

❷ The second part, $\max_{x \in L_f(\lambda)} d(x, \widehat{L_f(\lambda)})$, ensures that our estimated clustering result obtained from grid-based adaptive sampling provides adequate or even superior coverage of the level-set. We have the conclusion proven in [5] that it's highly probable that there exists a sample point $x' \in S_1$ such that $|x - x'| \leq r_0$. Assuming some $k \geq dlogn$, $\sigma > 0$, there exists a constant such that with probability $> 1 - \sigma$, and $r_0 := \left(\frac{32 C_{\delta,n} \sqrt{D \log m}}{m v_D \cdot \lambda}\right)^{1/D}$.

Within a distance of r_0 from x, points are ensured to be sampled, which are denoted as x' (attributed to the sufficiently high probability density in the neighborhood of x). Subsequently, it is imperative to prove that x' meets the density criterion for the density level set, signifying that x' is a core point. The result follows Lemma 8 in the proof of Jiang et al. [13], which demonstrates that any sampled point $x \in L_f(\lambda)$ satisfies $|B(x, \varepsilon) \cap \mathcal{X}| \geq minPts$. Therefore, $x' \in \widehat{L_f(\lambda)}$, which means that x' is a core point. Consequently, $\max_{x \in L_f(\lambda)} d(x, \widehat{L_f(\lambda)}) \leq r_0$, fulfilling the desired condition.

$$d_H(\widehat{L_f(\lambda)}, L_f(\lambda)) \leq C \cdot \left(C_{\delta,n}^{2/\beta} \cdot minPts^{-1/2\beta} + C_{\delta,n}^{1/D} \cdot \left(\frac{\sqrt{\log m}}{m}\right)^{1/D}\right). \tag{4}$$

If we take $minPts \approx n^{2\beta/(2\beta+D)}$ and let the first expression dominate,

$$n^{-1/(2\beta+D)} \gtrsim m^{-1/D}. \tag{5}$$

Based Equation (4) and Equation (5), we have $m \gtrsim n^{D/(2\beta+D)}$. Let $\widehat{L_f(\lambda)}\prime$ denote the clustering result obtained using grid-based adaptive sampling. The improvement lies in our even distribution of a constant number of sampling points obtained from S_2 across all grids. Then we have,

$$d_H(\widehat{L_f(\lambda)}\prime, L_f(\lambda)) \lesssim n^{-1/(2\beta+D)}. \tag{6}$$

For example, if we take $\beta = 0.9, n = 10,000, D = 2$, then we should at least sample 128 points to satisfy the statistical consistency. Based on this, GAS-DBSCAN has a time complexity of $O(mn)$, and it can achieve a minimax optimal estimation in sub-quadratic runtime.

4.4 Unbiased Level of Cluster Sampling (ULCS)

To present a more intuitive comparison on the improvement of the balanced sampling performance, we now define the probability of sampling the smallest cluster of a dataset as the Unbiased Level of Cluster Sampling, i.e., ULCS. As previously mentioned, the original uniform sampling method is more prone to inadequately covering small clusters in unevenly distributed datasets. This section therefore analyzes how our method enhances the sampling of small clusters.

Consider a dataset with k clusters C_1, C_2, C_3,$\cdots$, C_k, where C_3 represents the smallest cluster occupying a proportion π of the dataset. To evaluate GAS-DBSCAN's sampling performance on unbalanced data, we consider a scenario where cluster C_3 occupies an entire grid. Here, ULCS achieves unity as our sampling strategy ensures at least one sampled point per grid. If C_3 is co-located with other clusters within a single grid, and we define π_g as the proportion of C_3 within that grid, it follows that $\pi_g \geq \pi$. This assertion holds because C_3 is fully encompassed by the grid, and the grid's maximum capacity for points is n.

The improvement of ULCS from grid-based sampling over uniform sampling is defined as the ratio of ULCS by GAS-DBSCAN to that by uniform sampling, with the same sampling size. Given both methods have m globally sampled points, let $\pi_g = \tau \cdot \pi$ and $m_g = \eta \cdot m$. Here, π_g is the proportion of C_3 points in the grid, τ is the ratio of C_3 point proportion in the grid to that in the whole dataset ($\tau \geq 1$), and η is the ratio of sampled points in the grid to globally sampled points ($\eta \leq 1$). The simplified improvement rate is as follows:

$$\frac{1-(1-\pi_g)^{m_g}}{1-(1-\pi)^m} = \frac{1-(1-\tau\cdot\pi)^{\eta\cdot m}}{1-(1-\pi)^m}. \tag{7}$$

Given that C_3 comprises 80% of the grid points, our analysis, supported by Figure 4 and Equation (7), indicates that the proposed GAS-DBSCAN method consistently achieves robust ULCS improvement across various data distribution scenarios. Moreover, as the number of sampling points decreases, this method

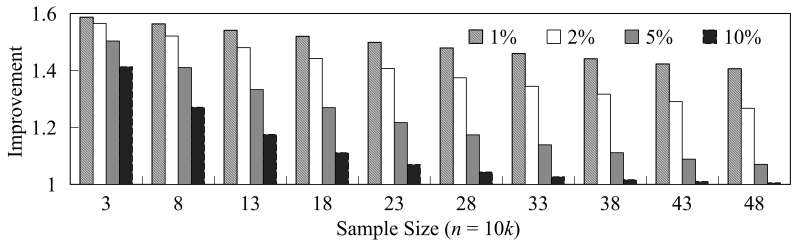


Fig. 4. Improvement of ULCS. The proportion in the legend represents π. And the more uneven the data distribution is, as the number of sampling points increases, the improvement in ULCS becomes less significant.

demonstrates greater improvements in ULCS across varying proportions of the small cluster. When the sample size increases, the ULCS of both methods will increase, and the improvement of ULCS by GAS-DBSCAN will slightly decrease. Under unbalanced data distribution, the sample size has less impact on the ULCS improvement performance, which can be observed from the drop in the histogram. In other words, the method yields a more competitive sampling quality when dealing with skewed datasets.

5 Experimental Evaluation

5.1 Experiment Objective

In this section, we demonstrate the effectiveness and clustering quality of GAS-DBSCAN by answering the following two research questions:

RQ1: How much does the GAS-DBSCAN improve sampling performance?

RQ2: How much does GAS-DBSCAN improve clustering quality compared to the baseline?

To answer the above questions, we give analysis in Section 5.3 along with the experiments under five accuracy metrics and the execution time comparisons.

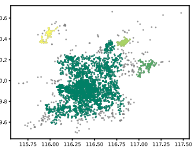
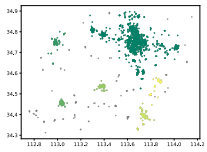
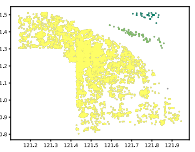
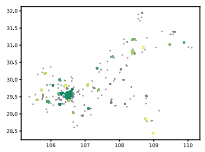
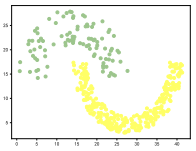
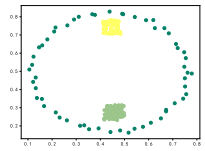
5.2 Experiment Settings

Datasets. The datasets used in this paper are extracted from publicly available datasets. Table 2 provides a clear representation of the characteristics of the datasets and the inputs of the experiments conducted on them. For simplicity, we only explain how to obtain dataset (A) below, and other real-world datasets are obtained in a similar way. All datasets are publicly available on GitHub [1]. Dataset (A) is extracted from Peking University's public POI dataset [4], comprising entries filtered from October 30, 2018 records where the attribute `pname` contains *Beijing* and `type` contains *company*. To determine clustering parameters, we first manually labeled and classified the data. Parameters were then

[1] https://github.com/mememelody/GAS-DBSCAN.git

selected based on achieving above 90% ARI and AMI scores using both DBSCAN and DRL-DBSCAN [19], ensuring minimal interference in method comparison.

Table 2. Clustering parameters and results.

Dataset Details	Distribution	Dataset Details	Distribution
Dataset (A): # points = 4902 # clusters = 4 $\varepsilon = 0.025$ $minPts = 20$		Dataset (B): # points = 1944 # clusters = 6 $\varepsilon = 0.05$ $minPts = 20$	
Dataset (C): # points = 10491 # clusters = 3 $\varepsilon = 0.05$ $minPts = 30$		Dataset (D): # points = 1651 # clusters = 25 $\varepsilon = 0.025$ $minPts = 15$	
Dataset (E): # points = 373 # clusters = 2 $\varepsilon = 2.1$ $minPts = 11$		Dataset (F): # points = 238 # clusters = 3 $\varepsilon = 0.012$ $minPts = 3$	

Baseline and the Ground-truth. Our approach draws inspiration from a uniform sampling method for DBSCAN. Naturally, we employ DBSCAN++ with uniform initialization as the baseline method. DBSCAN determines core points from the complete dataset, whereas our compared methods determine them only from the sampled subset. The ground-truth is obtained by running the standard DBSCAN algorithm on each dataset using the same parameters.

Metrics. We evaluate the performance of our proposed method using multiple evaluation metrics, including ULCS, Silhouette Coefficient, Adjusted Rand Index (ARI), Adjusted Mutual Information (AMI), Normalized Mutual Information (NMI), along with the number of outliers, and the execution time.

Experimental Environment. The experiments are performed on a standalone node featuring an Intel(R) Core(TM) i7-10700KF CPU @ 3.80GHz with a maximum turbo frequency of 3.79 GHz and 32GB of RAM.

5.3 Result Analysis

Unbiased Level of Cluster Sampling. To evaluate whether GAS-DBSCAN improves coverage when sampling smaller clusters, we conducted experiments on datasets (A) through (F) to assess the performance of ULCS, as shown in Figure 5. The x-axis represents the sampling rate, while the y-axis depicts ULCS. We can summarize the advantages of GAS-DBSCAN as follows:

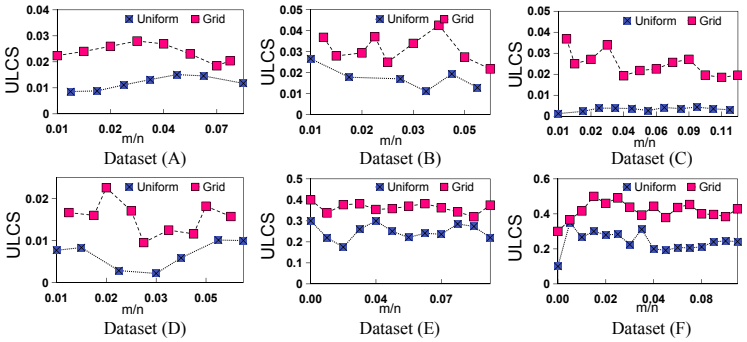


Fig. 5. ULCS performance. The pink line represents the result of grid initialization, and the blue one indicates that of uniform initialization.

❶ **Effective sampling on heavily skewed datasets.** As the results indicate, GAS-DBSCAN outperforms the original uniform sampling method across six real-world and synthetic datasets. On average, ULCS is nearly doubled compared to the original method, and it can be up to four times higher for extremely skewed datasets like dataset (C). GAS-DBSCAN significantly improves the sampling rate of small clusters and ensures full representation of each grid. The method performs proportional sampling in each grid rather than global random sampling. Consequently, it avoids the problem of small clusters being ignored in global random sampling. By sampling proportionally within the local grid, we can more accurately capture the fine clustering structure in the dataset, thereby enhancing the accuracy and representativeness of the experimental results. This aligns with our previous analysis in Section 4.3 and Figure 4.

❷ **Notable improvement on sampling at lower sampling rates.** We observe that our method significantly enhances the ULCS at lower sampling rates and that the improvement is more pronounced when sampling rates are low. This suggests that the bias resulting from uniform sampling at lower sampling rates can be substantially reduced by employing the GAS-DBSCAN sampling method, thereby highlighting the superiority of our approach. The results above answer RQ1 and the superiority in sampling of our method.

ARI, AMI, NMI, Silhouette and Outliers. To validate the effectiveness of GAS-DBSCAN in enhancing the coverage of smaller clusters during sampling, we present the experimental outcomes depicted in Figure 6, which comprehensively compares the clustering quality across five distinct metrics, providing a holistic view of the performance. The horizontal axis represents the sampling rate, while the vertical axis signifies the ULCS, serving as a quantitative measure of the sampling quality. This analysis underscores the ability of GAS-DBSCAN to improve the clustering of smaller clusters. The advantages of GAS-DBSCAN demonstrated in Figure 6 are concluded as follows:

❶ **Higher clustering quality at lower** m/n **levels.** As sampling rates increase, GAS-DBSCAN consistently outperforms DBSCAN++, particularly at

Table 3. Execution time results (milliseconds).

Dataset	DBSCAN	Uniform	Grid	Dataset	DBSCAN	Uniform	Grid
(A)	16962	18962	**16863**	(E)	204	144	**123**
(B)	2634	2549	**2509**	(F)	117	62	**41**
(C)	87377	90880	**74738**	(X)	4.629E7	4.6989E7	**4.220E7**
(D)	1894	1886	**1799**				

lower m/n ratios. While uniform sampling shows vulnerability to skewed distributions, GAS-DBSCAN demonstrates robust performance across varying sampling ranges. For dataset (E), scores stabilize near 1 within rates (0.2, 0.3], showing up to 3-fold improvement. The superiority stems from GAS-DBSCAN's adequate sampling of small clusters in skewed distributions, as evidenced in datasets (A) and (C), where uniform sampling struggles with coverage issues.

❷ **Faster convergence in outlier detection.** Both methods' outlier detection converges to DBSCAN results as m/n increases, with GAS-DBSCAN showing faster convergence and fewer initial outliers. This advantage stems from the grid-based adequate coverage of potential core points, while uniform sampling struggles with noise classification, giving answers to RQ2.

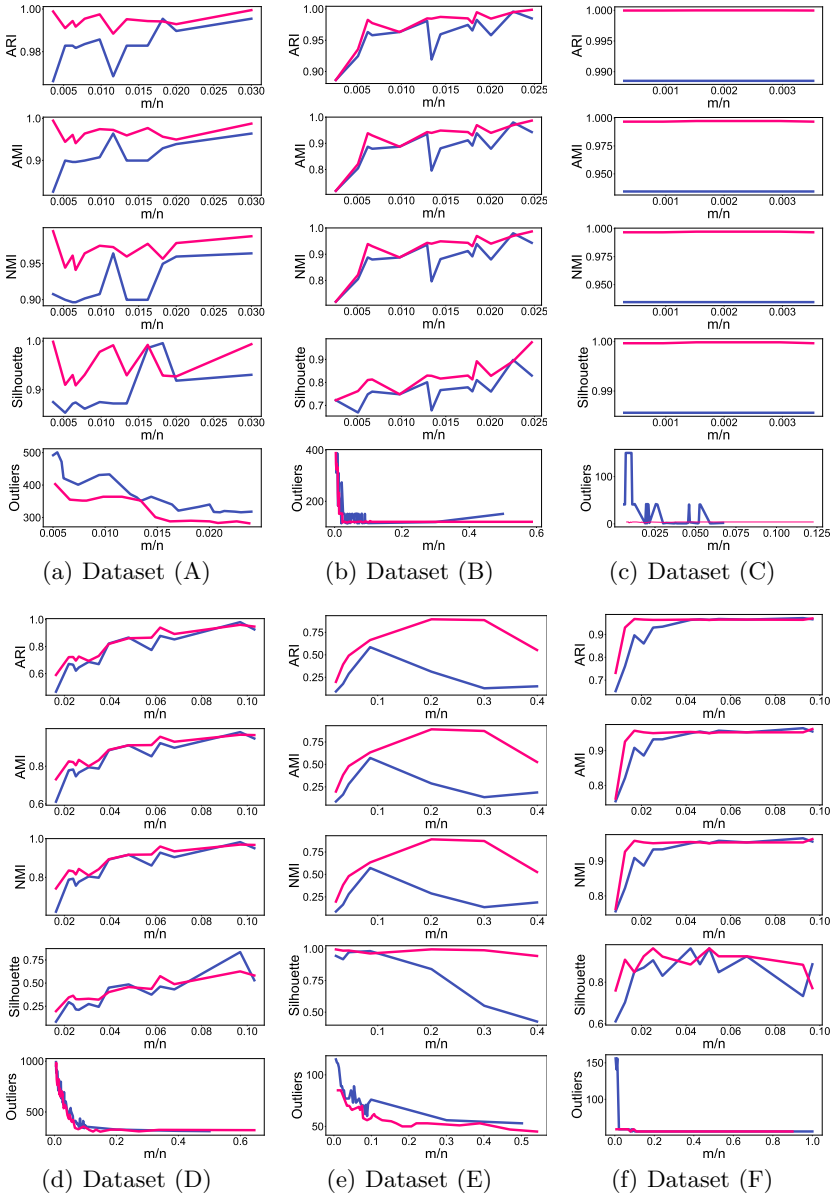
Execution Time. The runtime comparison in Table 3 demonstrates GAS-DBSCAN's superior computational efficiency, outperforming standard DBSCAN and uniform sampling methods at their optimal sampling rate, specifically when ARI reaches 90%, with results averaged across 10 experimental runs. To measure the algorithm's scalability, we use dataset (X) consisting of historical tropical storm paths from the past 50 years in various basins [2], totaling 227,500 points.

Among all the datasets, GAS-DBSCAN exhibits a lower average running time compared to the other two methods. As the number of sampled points m increases, the number of candidate core objects to be detected also rises, leading to an increase in clustering time. The overall time complexity of GAS-DBSCAN is $O(mn)$, which is consistent with the upper bound we derived in (6) of Section 4.3. Also, GAS-DBSCAN provides a higher sampling coverage hit rate, achieving high clustering quality with fewer sampling points and reducing redundant core point detection. These results thus suggest that GAS-DBSCAN is not only faster, but can also achieve better clustering results, thus achieving a trade-off between clustering quality and efficiency.

6 Conclusion

This paper presents a grid-based adaptive sampling method for DBSCAN clustering, which enhances sampling efficiency in skewed data distributions while reducing computational overhead. We demonstrate both theoretically and empirically that our method maintains high cluster coverage and statistical consistency, showing superior performance over conventional uniform-based DBSCAN

[2] https://data.amerigeoss.org/tl/dataset/asia-pacific-storm-tracks-1956-to-2018



(a) Dataset (A) (b) Dataset (B) (c) Dataset (C)

(d) Dataset (D) (e) Dataset (E) (f) Dataset (F)

Fig. 6. Clustering quality evaluations. The pink line represents the result of grid initialization, and the blue one indicates that of uniform initialization.

on synthetic and real-world datasets. Future work will explore distributed parallel processing for massive datasets, focusing on lightweight stateful operations with load balancing, and the integration of historical information through summary indexes to optimize neighborhood calculations.

Acknowledgments. This work was supported by the Natural Science Foundation of the National Natural Science Foundation of China under grant (No. 61802273), Jiangsu Higher Education Institutions of China (No. 23KJA520011), China Science and Technology Plan Project of Suzhou (No. SYG202139).

References

1. Aggarwal, A., Deshpande, A., Kannan, R.: Adaptive sampling for k-means clustering. In: Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (2009)
2. Andrade, G., Ramos, G., et al.: G-dbscan: A gpu accelerated algorithm for density-based clustering. Procedia Computer Science (2013)
3. Boonchoo, T., Ao, X., Liu, Y., Zhao, W., Zhuang, F., He, Q.: Grid-based dbscan: Indexing and inference. PR (2019)
4. Center, S.: Map poi (point of interest) data. Peking University Open Research Data Platform (2017)
5. Chaudhuri, K., Dasgupta, S.: Rates of convergence for the cluster tree. NeurIPS (2010)
6. Chen, Y., Zhou, L., Bouguila, N., Wang, C., Chen, Y., Du, J.: Block-dbscan: Fast clustering for large scale data. Pattern Recognition (2021)
7. Ding, H., Huang, J., Yu, H.: The effectiveness of uniform sampling for center-based clustering with outliers. arXiv preprint arXiv:1905.10143 (2019)
8. Doob, J.: Statistical estimation. Transactions of the American Mathematical Society (1936)
9. Ester, M., Kriegel, H.P., Sander, J., Xu, X., et al.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: KDD (1996)
10. Gowanlock, M., Rude, C.M., Blair, D.M., Li, J.D., Pankratius, V.: A hybrid approach for optimizing parallel clustering throughput using the gpu. TPDS (2018)
11. He, Y., Tan, H., Luo, W., Mao, H., Ma, D., Feng, S., Fan, J.: Mr-dbscan: an efficient parallel density-based clustering algorithm using mapreduce. In: ICPADS (2011)
12. Jang, J., Jiang, H.: Dbscan++: Towards fast and scalable density clustering. In: ICML. PMLR (2019)
13. Jiang, H.: Density level set estimation on manifolds with dbscan. In: International Conference on Machine Learning. PMLR (2017)
14. Jiang, H., Jang, J., Lacki, J.: Faster dbscan via subsampled similarity queries. NeurIPS (2020)
15. Keramatian, A., Gulisano, V., et al.: Ip. lsh. dbscan: Integrated parallel density-based clustering through locality-sensitive hashing. In: ICPP (2022)
16. Kim, C., Kim, J.: Spatial spillovers of sport industry clusters and community resilience: Bridging a spatial lens to building a smart tourism city. IPM (2023)
17. Ohadi, N., Kamandi, A., Shabankhah, M., et al.: Sw-dbscan: A grid-based dbscan algorithm for large datasets. In: ICWR (2020)
18. Wang, Y., Gu, Y., Shun, J.: Theoretically-efficient and practical parallel dbscan. In: SIGMOD (2020)
19. Zhang, R., Peng, H., Dou, Y., Wu, J., Sun, Q., Li, Y., Zhang, J., Yu, P.S.: Automating dbscan via deep reinforcement learning. In: CIKM (2022)