Journal of Mathematical Modeling
Vol. -, No. -, 2026, pp. -. Research Article

JMM

HyEMST: A novel hybrid ellipsoidal framework for robust clustering via maximum spanning trees

Hossein Eyvazi*, Seyed Mohammad Badzohreh, Amir Mohammad Kharazi

Department of Computer Science, Tarbiat Modares University, Tehran, Iran

Email(s): eyvazi_hoseyn@modares.ac.ir, mohammad.badzohreh@modares.ac.ir, am.kharazi@modares.ac.ir

Abstract. Clustering arbitrary-shaped clusters with heterogeneous densities presents a fundamental challenge in unsupervised learning. Traditional approaches emphasize either geometric distance or local density estimation, yet rarely reconcile both perspectives systematically. This paper introduces **HyEMST** (Hybrid Ellipsoidal Maximum Spanning Tree), a principled framework that unifies distance and density information through an explicit trade-off parameter $\lambda \in [0, 1]$. The proposed methodology comprises five phases: (1) strategic geometric decomposition via K-Means over-segmentation; (2) robust volumetric density estimation using adaptive ridge-regularized covariance; (3) hybrid kernel construction integrating distance and density affinities; (4) topological structure discovery via maximum spanning tree; and (5) adaptive density-aware cluster merging. Theoretically, we establish that regularized covariance-based density estimation preserves density ranking with $> 90\%$ accuracy, ensuring reliable merging even for ill-conditioned micro-clusters. Computationally, the approach achieves $\mathcal{O}(Nd^2)$ overall complexity. Empirically, HyEMST attains perfect or near-perfect clustering on synthetic benchmarks and demonstrates superior performance compared to representative baselines on real-world datasets. Ablation studies validate the necessity of hybrid integration and confirm the efficacy of each algorithmic component.

Keywords: Non-convex clustering, ellipsoidal density estimation, hybrid kernels, maximum spanning trees, arbitrary-shaped clusters
AMS Subject Classification 2010: 62H30, 68T10

1 Introduction

Clustering is a fundamental task in unsupervised machine learning, aiming to partition data points into groups where intra-cluster similarity is maximized and inter-cluster dissimilarity is

*Corresponding author
Received: 04 December 2025/ Revised: 02 February 2026/ Accepted: 16 February 2026
DOI: 10.22124/jmm.2026.32457.2943

minimized. Real-world datasets, however, often exhibit complex structures with arbitrary shapes and heterogeneous densities, which can challenge the simplifying assumptions underlying many classical clustering methods. This work proposes **HyEMST** (Hybrid Ellipsoidal Maximum Spanning Tree), a framework designed to integrate geometric distance information and local density information in a principled and scalable manner.

A broad spectrum of clustering algorithms has been developed, each with distinct modeling assumptions and practical strengths. In this subsection we briefly review representative methods, emphasizing both their advantages and the regimes where they may encounter difficulties on complex, multi-structure data.

Distance-based methods: K-Means remains one of the most widely used clustering algorithms due to its conceptual simplicity and computational efficiency. It minimizes within-cluster squared Euclidean distances, leading to Voronoi-like partitions that are well suited to *approximately spherical, convex, and similarly sized* clusters. In such settings K-Means is highly effective and often difficult to outperform in practice.

At the same time, the Euclidean mean-based objective induces an implicit convexity bias: the decision regions associated with each centroid are convex, and cluster boundaries tend to be approximately linear in the original space. As a result, K-Means can struggle on data whose natural clusters are strongly non-convex (e.g., crescents, rings, or intertwined manifolds) or exhibit pronounced variation in density and scale. Furthermore, K-Means requires prior specification of the number of clusters K, and its outcome can be sensitive to centroid initialization.

Density-based methods: DBSCAN introduced a density-centric view of clustering: clusters are defined as high-density regions separated by areas of low density. This perspective enables automatic identification of noise points and recovery of clusters with highly non-convex shapes, without requiring the number of clusters in advance.

However, DBSCAN introduces two hyperparameters, the neighborhood radius ε and the minimum number of points `MinPts`. Choosing these parameters in a principled way can be challenging, and performance can degrade when a single global ε must reconcile both dense and sparse regions. In particular, datasets with substantial variation in cluster densities may require different local scales, whereas DBSCAN operates with a single global neighborhood scale.

The HDBSCAN extends this line of work by building a hierarchy over multiple density levels and using cluster stability criteria to select a final clustering. This significantly improves robustness to variable densities and often yields high-quality clusters in practice. Nevertheless, HDBSCAN remains primarily driven by density information, and the influence of geometric distances is indirect, controlled through neighborhood graph construction and distance metrics. In applications where both density and geometric separation are informative, a more explicit mechanism for balancing these two signals can be desirable.

Spectral and hierarchical methods: Spectral clustering constructs a similarity graph and uses the eigen structure of the associated Laplacian to embed data into a lower-dimensional space where standard clustering (e.g., K-Means) is applied. This approach is supported by a rich theoretical foundation and is particularly effective at capturing non-convex cluster shapes when the similarity graph is well chosen.

In practice, however, spectral clustering introduces several design choices: the graph construction (e.g., k-nearest neighbors versus ε-neighborhoods), the similarity function and its bandwidth, and the target number of clusters. Its computational cost can also be substan-

tial on large datasets due to eigen-decomposition. Moreover, when datasets contain clusters at very different scales or densities, it can be nontrivial to construct a single similarity graph that adequately represents all structures simultaneously.

Hierarchical clustering produces a dendrogram representing cluster relationships across all granularities. This is useful when multi-scale structure is of interest. Yet, selecting a cut level (or levels) in the dendrogram often relies on heuristic criteria or domain knowledge, and standard linkage strategies make implicit assumptions (e.g., single-linkage sensitivity to chaining, complete-linkage preference for compact clusters) that may not align with heterogeneous, high-dimensional data.

The above methods illustrate a recurring pattern: algorithms tend to emphasize either geometric proximity or local density, but rarely treat both on equal footing with explicit, tunable trade-offs.

- **Distance-centric approaches** (such as K-Means and many spectral variants) are well suited to convex, similarly dense clusters, where Euclidean distance is an adequate proxy for affinity. They may, however, struggle when clusters have very different densities or exhibit pronounced non-convexity.
- **Density-centric approaches** (such as DBSCAN and HDBSCAN) adapt naturally to non-convex shapes and can handle noise and some forms of density variation. Yet, their reliance on density thresholds and local connectivity means that geometric separation is not always explicitly or flexibly controlled.

In many real applications, distance and density provide complementary signals. For instance, two regions can be individually dense yet lie close in space while still corresponding to different underlying structures, whereas sparsely populated transition areas often require more nuanced treatment than what a single global density threshold can capture. These observations motivate a framework that combines geometric distance and local density within a single *tunable* affinity, adapts naturally to heterogeneous density levels without discarding geometric information, reduces manual tuning through principled data-driven parameter selection, and remains computationally feasible for large datasets and moderate-to-high ambient dimensions.

This paper introduces **HyEMST**, a clustering framework that addresses the above desiderata through a decomposition–estimation–merging pipeline. At a high level, HyEMST proceeds by over-segmenting the data into micro-clusters, estimating volumetric densities for these micro-clusters, constructing a hybrid distance–density affinity, and then extracting clusters by cutting a maximum spanning tree under density-aware thresholds. Our main contributions are as follows:

1. **Robust volumetric density estimation:** We propose a density estimator based on the volume of covariance-driven ellipsoids associated with micro-clusters. To address the well-known issue that naive covariance-based volume estimates can become unstable when the micro-cluster size m_i is not much larger than the dimension d, we introduce an **adaptive regularization scheme**. This regularization stabilizes the covariance and its determinant while preserving the relative ordering of densities across micro-clusters, which is what ultimately drives the merging decisions.
2. **Hybrid distance–density affinity:** We define a hybrid kernel that multiplicatively combines a distance-based affinity with a density-based affinity, controlled by an explicit

interpolation parameter $\lambda \in [0,1]$. This parameter allows practitioners to smoothly traverse the spectrum from distance-dominant to density-dominant clustering within a unified formulation.

3. **Topology-aware structure discovery via MST:** We construct a maximum spanning tree (MST) on the micro-cluster graph induced by the hybrid affinity. This tree compactly encodes connectivity information with only $K_{\max} - 1$ edges, where $K_{\max}$ is the number of micro-clusters, and provides a convenient structure on which to perform subsequent density-aware cuts.

4. **Density-aware merging with data-driven parameter selection:** We design a merging rule that adapts edge-cut thresholds to local densities, reducing sensitivity to global, hand-chosen thresholds. Furthermore, we employ a Bayesian optimization strategy (e.g., Tree-Structured Parzen Estimators) to jointly tune key hyperparameters such as $K_{\max}$, λ, the density bandwidth, and the base merging threshold, thus limiting reliance on ad-hoc manual selection.

On the theoretical side, we analyze the behavior of the proposed hybrid affinity and the impact of regularization on volumetric density estimation. In particular, we show that the hybrid kernel behaves smoothly as a function of the trade-off parameter λ, and we discuss conditions under which regularized covariance-based volume approximations preserve the ranking of local densities across micro-clusters. We also provide a complexity analysis showing that, under reasonable choices of $K_{\max}$, the overall method scales approximately linearly with the number of data points and polynomially with the dimension.

On the empirical side, we demonstrate that HyEMST can recover non-convex clusters, handle heterogeneous densities, and perform competitively with or better than representative baselines (K-Means, DBSCAN, HDBSCAN, spectral clustering, and more modern methods as well as GMM variants) on a range of synthetic and real-world datasets. These experiments illustrate the benefits of explicitly balancing distance and density, and of using robust density estimates in the presence of small or high-dimensional micro-clusters.

2 Related work

The clustering literature has witnessed significant algorithmic innovations addressing the persistent challenges of multi-density data, arbitrary shapes, and parameter sensitivity. This section examines state-of-the-art methods across density-based, hierarchical, and hybrid paradigms that motivate our unified framework.

Classical density-based methods and extensions: The DBSCAN [6] pioneered density-based clustering by identifying core points with sufficient neighbors within radius ε. While effective for uniform-density datasets, DBSCAN's global parameters fail on heterogeneous density distributions. OPTICS [2] addressed this limitation by constructing a reachability plot encoding cluster hierarchies at all density scales. However, OPTICS requires manual interpretation of the reachability plot to extract final clusters, making it labor-intensive for practitioners. Additionally, its $\mathcal{O}(n \log n)$ complexity (with spatial indexing) becomes prohibitive on datasets

exceeding millions of points. The HDBSCAN [5] extends DBSCAN with hierarchical clustering and stability-based cluster extraction, enabling automatic detection of clusters with varying densities. Despite these advances, HDBSCAN remains fundamentally density-centric: it does not explicitly model spatial distance structure, potentially under-weighting geometric relationships. The stability computation introduces sensitivity to the minimum cluster size parameter, which fundamentally affects cluster merging decisions. ST-DBSCAN [3] extends DBSCAN to spatiotemporal data by incorporating both spatial (ε_s) and temporal (ε_t) proximity. However, ST-DBSCAN inherits DBSCAN's global parameter rigidity and does not address density heterogeneity. The requirement to tune two radius parameters amplifies the parameter sensitivity problem.

Adaptive multi-density methods: Recent work has focused on making density-based clustering adaptive to heterogeneous density distributions. The AMD-DBSCAN [17] automatically adapts ε locally by analyzing k-nearest neighbor distance distributions, achieving 24.7% average accuracy improvement on datasets with extreme density variations. However, the algorithm does not explicitly model geometric relationships between clusters, potentially causing over-fragmentation when clusters are spatially well-separated but exhibit similar local densities. The MDBSCAN [11] introduces relative density metrics rather than absolute density thresholds. By comparing each point's density to local neighborhood statistics, MDBSCAN achieves robustness across varying density scales. However, the method still suffers from the chain-reaction problem inherent to single-linkage assignment strategies. The RNN-DBSCAN [4] employs reverse nearest neighbor counts for density estimation, providing robustness to outliers and parameter sensitivity. However, the method still requires specification of the k parameter, and computing reverse neighbors increases overall complexity to $\mathcal{O}(n^2)$ in the worst case. Amroune et al. [1] proposed adaptive ε computation via k-NN averaging, simplifying parameterization. However, this introduces sensitivity to k selection, and the averaged k-distance assumes isotropy in local neighborhoods, failing on elongated or anisotropic cluster structures.

Graph-based and topological approaches: Graph-based methods generally suffer from computational complexity: constructing complete affinity graphs requires $\mathcal{O}(n^2)$ distance computations, and spectral decomposition methods introduce additional $\mathcal{O}(n^3)$ eigenvalue computation costs. Our MST-based approach addresses this by reducing graph complexity from $\mathcal{O}(K_{\max}^2)$ to $\mathcal{O}(K_{\max})$ micro-cluster edges while preserving hierarchical structure.

Advanced density peaks methods: The Density Peaks Clustering (DPC) paradigm [13] continues to evolve with innovations addressing center selection and assignment robustness. DPC-MDNN [16] establishes nearest neighbor relationships based on manifold distance rather than Euclidean distance, providing better geometric fidelity on complex structures. The method employs a two-stage assignment strategy that mitigates the domino effect inherent to single-linkage point assignment. However, The DPC-MDNN remains fundamentally density-centric—manifold distance estimation itself may require parameter tuning in high-dimensional settings. The method does not systematically integrate spatial distance with density measures through a principled mathematical framework. The INSDPC [10] redefines local density using interactive neighbor similarity—combining mutual neighbors and shared neighbors—to improve center identification. While these innovations enhance robustness over vanilla DPC, the method remains constrained by the cutoff distance d_c, which fundamentally determines neighbor relationships. The sensitivity to d_c reintroduces the parameter tuning problem. The WMKNNDPC [12] employs mutual k-

NN (combining k-NN and inverse k-NN) for density calculation, enabling identification of cluster centers in uneven density distributions. However, the algorithm introduces multiple weighting parameters whose optimal values depend on dataset characteristics, partially reintroducing the parameter tuning problem that DPC variants aim to eliminate.

Hybrid and scalable methods: Grid-based clustering [9] determines core grids based on data distribution uniformity rather than absolute density, balancing efficiency and accuracy. However, grid methods remain inherently discretized—the initial grid partition parameter M fundamentally constrains the finest granularity at which clusters can be resolved. The sDBSCAN [18] leverages random projections to accelerate core point identification in high-dimensional spaces, achieving orders-of-magnitude speedups on million-point datasets. However, random projection methods introduce approximation errors—neighborhood preservation is probabilistic, not guaranteed. The resulting clustering structure is only approximately similar to exact DBSCAN under mild probabilistic conditions, potentially losing precision in critical boundary regions.

Despite these algorithmic advances, several fundamental limitations persist. Many modern methods still lean predominantly toward either density-centric reasoning (e.g., AMD-DBSCAN, MDBSCAN, HDBSCAN, RNN-DBSCAN, DPC-MDNN, INSDPC, WMKNNDPC) or distance-centric reasoning (e.g., grid-based approaches, sDBSCAN, OPTICS), and even when hybrids incorporate manifold-aware distances (e.g., DPC-MDNN), the fusion of geometric proximity and density evidence is typically heuristic and dataset-dependent rather than controlled by an explicit, analyzable trade-off. Moreover, while adaptivity reduces reliance on a single global ε-type parameter, it often shifts rather than eliminates tuning by introducing multiple interacting hyperparameters—such as k in kNN-based density estimates (AMD-DBSCAN, RNN-DBSCAN, WMKNNDPC), weighting schemes (INSDPC, WMKNNDPC), manifold-distance settings (DPC-MDNN), grid granularity (grid methods), or minimum cluster size and stability thresholds (HDBSCAN)—which can reintroduce sensitivity under noise, density heterogeneity, or class imbalance. Finally, assignment and cutpoint selection remain fragile in many pipelines: although recent DPC variants refine assignment rules to mitigate chain-reaction errors (e.g., DPC-MDNN, INSDPC, WMKNNDPC), these mechanisms are largely heuristic with limited guarantees that assignment mistakes remain localized, and most approaches still require some form of threshold or cutpoint determination (e.g., manual OPTICS plot interpretation, HDBSCAN stability-based selection, or decision-graph/distance thresholds in DPC-style methods), leaving principled end-to-end parameter selection uncommon in practice.

Our work addresses these gaps through a unified mathematical framework that synthesizes insights from density-based, topological, and hybrid methods. Concretely, instead of combining density and distance in an ad-hoc or sequential manner, we introduce a hybrid affinity that explicitly balances ellipsoidal volumetric density (via MVEE-based volume proxies) and spatial proximity through a continuous trade-off parameter $\lambda \in [0,1]$, allowing smooth interpolation between geometry-dominant and density-dominant regimes and enabling direct theoretical discussion of how the balance affects merging behavior. To retain the hierarchical structure needed for robust agglomeration while controlling computational cost, we construct a maximum spanning tree over micro-clusters, which reduces the candidate edge set from $\mathcal{O}(K_{\max}^2)$ to $\mathcal{O}(K_{\max})$ without relying on stochastic sparsification. In addition, rather than introducing multiple independent heuristics that shift the tuning burden across parameters, we treat the main structural choices $(K_{\max}, \lambda, \beta, \tau_0)$ as a coupled set and tune them jointly within a single optimization loop,

so that the method's operating point is determined in a consistent and reproducible way. Finally, the merging rule is made explicitly density-aware through a smooth adaptive threshold correction that responds to local density differences; this reduces the sensitivity of linkage decisions in heterogeneous settings (and mitigates chain-reaction errors relative to purely distance-based single-linkage) while preserving scalability and interpretability. Overall, HyEMST is not an incremental modification of a single paradigm, but a unified formulation that explicitly reconciles distance and density signals, preserves hierarchical structure efficiently, and supports both theoretical reasoning and practical deployment.

3 Methodology

The HyEMST framework addresses clustering with arbitrary shapes and heterogeneous densities through a decomposition–estimation–merging pipeline. Given dataset $\mathcal{D} = \{x_1, \ldots, x_N\} \subset \mathbb{R}^d$, we obtain partition $\mathcal{P} = \{C_1, \ldots, C_K\}$ (with K unknown) via five phases:

1. **Strategic geometric decomposition:** K-Means with learnable initialization strategy (K-Means++ or random) produces $K_{\max}$ micro-clusters.
2. **Robust volumetric density estimation:** Each micro-cluster is assigned normalized density $\rho_i \in [0,1]$ via regularized covariance-based MVEE [7, 8].
3. **Hybrid kernel construction:** Hybrid affinity $W_{\text{hybrid}}(i,j;\lambda)$ unifies distance and density under tradeoff $\lambda \in [0,1]$.
4. **Topological structure discovery:** MST on hybrid graph captures connectivity with $\mathcal{O}(K_{\max})$ edges.
5. **Adaptive density-aware merging:** MST edge cuts guided by density-adaptive threshold produce final clusters.

Phase 1: Strategic geometric decomposition via K-Means

We begin by over-segmenting the dataset into $K_{\max}$ micro-clusters using K-Means to obtain a controllable geometric decomposition that stabilizes subsequent density estimation and MST-based merging.

Proposition 1 (Initialization strategy impact on micro-cluster quality)**.** *The choice of K-Means initialization strategy significantly affects the homogeneity and fragmentation of micro-clusters, particularly in heterogeneous density settings. Two competing strategies exist:*

1. ***K-Means++ initialization:*** *Selects initial centroids sequentially with probability proportional to squared distance from previously selected centroids. This maximizes inter-centroid separation and often produces better initial cluster quality on homogeneous datasets. However, the distance-weighted repulsion mechanism can bias centroid placement toward sparse boundary regions in heterogeneous-density datasets.*

2. ***Random initialization:*** *Selects initial centroids uniformly at random from the dataset. This approach yields lower initial loss (compared to K-Means++) but may converge slower. Importantly, it naturally concentrates centroids proportionally to local point density, which can improve alignment with true cluster structure in datasets with varying local densities.*

The optimal choice depends on dataset characteristics and downstream task requirements. For heterogeneous-density clustering, the initialization strategy acts as a ***critical tuning knob*** *that balances convergence speed against density-structure alignment.*

Algorithm 1: Strategic geometric decomposition with learnable initialization

Input: Dataset $\mathcal{D} = \{x_1, \ldots, x_N\}$, micro-cluster count $K_{\max}$, initialization strategy init_type $\in$ {"KMeans++","Random"}
Output: Micro-clusters $\{C_1, \ldots, C_{K_{\max}}\}$
Initialize centroids $\{\mu_1, \ldots, \mu_{K_{\max}}\}$:
if init_type = "KMeans++" **then**
  Select μ_1 uniformly from $\mathcal{D}$
  for $k = 2$ to $K_{\max}$ **do**
    Select μ_k with probability $\propto \min_j \|\mu_j - x\|^2$ for $x \in \mathcal{D}$
  end for
else
  Select $\{\mu_1, \ldots, \mu_{K_{\max}}\}$ uniformly without replacement from $\mathcal{D}$
end if
repeat
  Assignment: $c_i \leftarrow \arg\min_{1 \le j \le K_{\max}} \|x_i - \mu_j\|^2$ for all i
  Update: $\mu_j \leftarrow \frac{1}{|C_j|} \sum_{x_i \in C_j} x_i$ for all j
until convergence
Return micro-clusters $\{C_1, \ldots, C_{K_{\max}}\}$

Algorithm 1 decomposes $\mathcal{D}$ into $K_{\max}$ micro-clusters via K-means with a selectable initialization strategy (KMeans++ or Random). Its time complexity is $\mathcal{O}(T\,N\,K_{\max}\,d)$, where typically $T = \mathcal{O}(\log N)$ iterations; in particular, for $K_{\max} = \mathcal{O}(\sqrt{N})$ this becomes $\mathcal{O}(N^{1.5} d \log N)$.

Phase 2: Robust volumetric density estimation via regularized covariance

Phase 2 assigns each micro-cluster C_i a *relative density score* ρ_i used later only for *ranking* and for density-aware merging (Phase 5). We estimate density via the inverse volume of an ellipsoid induced by the micro-cluster covariance: tighter (lower-volume) clusters are treated as denser. The main technical challenge is that when a micro-cluster is small relative to dimension ($m_i \approx d$ or $m_i < d$), the sample covariance becomes ill-conditioned and $\det(\Sigma_i)$ becomes unstable, which can arbitrarily distort the density ranking. Our fix is an *adaptive ridge* term that is *zero* in well-conditioned regimes ($m_i/d \ge r_{\min}$) and smoothly increases as m_i/d decreases, guaranteeing $\Sigma_i^{\text{reg}} \succ 0$ and stabilizing $\log\det$ computations, while empirically preserving density ordering (Table 6).

Traditional density estimation suffers critical limitations: KDE requires bandwidth tuning; k-NN exhibits curse of dimensionality ($\propto 2^d$); grid methods scale exponentially. We propose regularized covariance-based MVEE approximation with polynomial dimension dependence that remains robust even when micro-cluster size approaches dimension.

In practice, when $K_{\max}$ is large or clusters are small, some micro-clusters may have $m_i \approx d$ or even $m_i < d$, violating the classical requirement $m_i \gg d$ for reliable covariance estimation; in such cases, the sample covariance Σ_i can become singular or nearly singular (rank-deficient), the determinant computation $\det(\Sigma_i)$ may be undefined or heavily biased, and the resulting density estimates ρ_i can fluctuate arbitrarily and become unreliable. We applied adaptive ridge regularization to stabilize covariance estimation while preserving the density ranking (the ordering of micro-clusters by density), which is the only property needed for Phase 5 merging.

Definition 1 (Adaptive Ridge-Regularized Covariance)**.** *For micro-cluster C_i with m_i points in dimension d, define the sample-size ratio:*

$$r_i = \frac{m_i}{d}. \tag{1}$$

The adaptive regularization strength depends on whether the micro-cluster satisfies the well-conditioned regime:

$$\eta_{ridge}(r_i) = \begin{cases} 0 & \text{if } r_i \geq r_{\min} \quad \textit{(well-conditioned)} \\ \alpha \cdot \left(1 - \frac{r_i}{r_{\min}}\right) \cdot \frac{trace(\Sigma_i^{raw})}{d} & \text{if } r_i < r_{\min} \quad \textit{(ill-conditioned)} \end{cases} \tag{2}$$

where $r_{\min} = 5$ is the minimum safe sample-size ratio (empirically validated), $\alpha = 0.5$ is the ridge strength multiplier, and Σ_i^{raw} is the unregularized sample covariance.

The regularized covariance is:

$$\Sigma_i^{reg} = \Sigma_i^{raw} + \eta_{ridge}(r_i) \cdot I_d. \tag{3}$$

The volumetric density is computed from regularized covariance:

$$\rho_i^{raw} = \frac{m_i}{\kappa_d \sqrt{\det(\Sigma_i^{reg})}}, \tag{4}$$

where $\kappa_d = \pi^{d/2}/\Gamma(d/2+1)$ is the unit ball volume.

Theoretical justification: Phase 5 uses density information primarily through *relative ordering* (e.g., via $\min(\rho_i, \rho_j)$ in Eq. (11)), rather than requiring perfectly calibrated absolute density values. We therefore justify that the proposed regularized covariance–volume proxy yields a *reliable density ranking* with high probability when micro-clusters are well-conditioned, and remains stable under ill-conditioning due to adaptive ridge regularization.

Theorem 1 (Regularized density ranking preservation)**.** *Let C_i, C_j be two micro-clusters with true densities $\rho_i^{true}, \rho_j^{true}$. Define $\rho_i^{reg}, \rho_j^{reg}$ as densities computed from regularized covariances (Eq. 3).*

Well-Conditioned Regime *($r_i, r_j \geq r_{\min}$):*

$$\rho_i^{true} > \rho_j^{true} \implies \rho_i^{reg} > \rho_j^{reg} \quad \textit{with probability } \geq 1-\delta \tag{5}$$

for $\delta = O(d/m_{\min})$ where $m_{\min} = \min(m_i, m_j)$.
Ill-Conditioned Regime *($r_i < r_{\min}$ or $r_j < r_{\min}$): Ridge regularization with $\eta_{ridge} = \alpha(1 - r_i/r_{\min})trace(\Sigma_i^{raw})/d$ ensures:*

$$\mathbb{P}\left(sgn(\rho_i^{reg} - \rho_j^{reg}) = sgn(\rho_i^{true} - \rho_j^{true})\right) \geq 0.90. \tag{6}$$

The main implication, density ranking, remains reliable for Phase 5 merging decisions, even when absolute density values are biased.

Proof. We provide a formal argument for Part (A) using non-asymptotic covariance concentration and a log-determinant perturbation bound [14,15]; Part (B) follows from ridge-stabilization plus the empirical ranking study in Table 6.
Well-conditioned regime: When $r_i = m_i/d \geq r_{\min}$, Eq. (2) yields $\eta_{\text{ridge}} = 0$, hence $\Sigma_i^{\text{reg}} = \Sigma_i^{\text{raw}}$ (and similarly for j). Assume the points in each micro-cluster are i.i.d. sub-Gaussian with covariance Σ_i^{true} (resp. Σ_j^{true}). By standard sample covariance concentration (see [14, 15]), for any $\delta \in (0,1)$ there exists an absolute constant $c > 0$ such that with probability at least $1-\delta$,

$$\left\| (\Sigma_i^{\text{true}})^{-1/2} \left(\Sigma_i^{\text{raw}} - \Sigma_i^{\text{true}}\right) (\Sigma_i^{\text{true}})^{-1/2} \right\|_2 \leq \varepsilon_i, \qquad \varepsilon_i := c\sqrt{\frac{d + \log(1/\delta)}{m_i}}. \tag{7}$$

Define $M_i := (\Sigma_i^{\text{true}})^{-1/2} \Sigma_i^{\text{raw}} (\Sigma_i^{\text{true}})^{-1/2}$. Then (7) implies $\|M_i - I\|_2 \leq \varepsilon_i$, hence all eigenvalues satisfy

$$\lambda_k(M_i) \in [1-\varepsilon_i,\, 1+\varepsilon_i], \qquad k = 1, \ldots, d. \tag{8}$$

Using $\det(\Sigma_i^{\text{raw}}) = \det(\Sigma_i^{\text{true}}) \det(M_i)$, we obtain

$$\log\det(\Sigma_i^{\text{raw}}) - \log\det(\Sigma_i^{\text{true}}) = \log\det(M_i) = \sum_{k=1}^{d} \log \lambda_k(M_i).$$

Since $\lambda_k(M_i) \in [1-\varepsilon_i, 1+\varepsilon_i]$, it follows that

$$|\log\det(M_i)| \leq \sum_{k=1}^{d} \max\{-\log(1-\varepsilon_i), \log(1+\varepsilon_i)\} \leq d \cdot (-\log(1-\varepsilon_i)) \leq d \log\left(\frac{1+\varepsilon_i}{1-\varepsilon_i}\right). \tag{9}$$

From Eq. (4) (with $\Sigma_i^{\text{reg}} = \Sigma_i^{\text{raw}}$ in this regime),

$$\log \rho_i^{\text{reg}} - \log \rho_i^{\text{true}} = -\frac{1}{2}\left(\log\det(\Sigma_i^{\text{raw}}) - \log\det(\Sigma_i^{\text{true}})\right),$$

and thus by (9),

$$|\log \rho_i^{\text{reg}} - \log \rho_i^{\text{true}}| \leq \frac{d}{2} \log\left(\frac{1+\varepsilon_i}{1-\varepsilon_i}\right). \tag{10}$$

The same bound holds for j with ε_j. Applying a union bound, with probability at least $1-\delta$ both (10) inequalities hold simultaneously. Therefore, if $\rho_i^{\text{true}} > \rho_j^{\text{true}}$ and the population log-density gap satisfies

$$\log \rho_i^{\text{true}} - \log \rho_j^{\text{true}} > \frac{d}{2}\log\left(\frac{1+\varepsilon_i}{1-\varepsilon_i}\right) + \frac{d}{2}\log\left(\frac{1+\varepsilon_j}{1-\varepsilon_j}\right),$$

then $\log \rho_i^{\text{reg}} > \log \rho_j^{\text{reg}}$, i.e., $\rho_i^{\text{reg}} > \rho_j^{\text{reg}}$ with probability at least $1-\delta$, proving (5). Noting $\varepsilon = O(\sqrt{d/m})$, the error term scales on the order of $O(d/m_{\min})$ for small ε, consistent with the stated δ scaling.

Ill-conditioned regime: When $r_i < r_{\min}$, the adaptive ridge term $\eta_{\text{ridge}} > 0$ ensures $\Sigma_i^{\text{reg}} = \Sigma_i^{\text{raw}} + \eta_{\text{ridge}} I_d \succ 0$ even when Σ_i^{raw} is rank-deficient. This stabilizes the determinant and prevents arbitrarily large fluctuations in ρ_i^{reg}. While the exact ranking probability depends on the unknown local distribution and the induced separation between proxy densities, the empirical study in Table 6 demonstrates that the proposed adaptive schedule maintains $\geq 90\%$ correct pairwise ordering down to $r \approx 1$, which is the regime most relevant for over-segmentation ($K_{\max}$ large) and high-dimensional micro-clusters. This supports (6) for the practical operating range of HyEMST. □

While Theorem 1 provides a high-probability ranking guarantee under a separation condition, our experiments additionally confirm strong practical behavior even when micro-clusters are small. As shown in Table 6, the adaptive ridge scheme maintains $> 90\%$ pairwise ranking accuracy at $r = m/d \approx 1$, where unregularized covariance estimates become unstable.

Algorithm 2 estimates a robust volumetric density for each micro-cluster by computing its centroid and sample covariance, applying *adaptive ridge regularization* when the sample-size ratio $r_i = |C_i|/d$ is too small, and then evaluating the covariance determinant stably via Cholesky to form ρ_i, which is finally normalized to $[0,1]$. Its total computational complexity is $T_{\text{Phase 2}} = \mathcal{O}(Nd^2 + K_{\max}d^3)$, combining $\mathcal{O}(m_i d^2)$ covariance estimation and $\mathcal{O}(d^3)$ Cholesky per cluster; the ridge step adds only $\mathcal{O}(d)$ per cluster and is negligible compared to covariance computation, so for $K_{\max} \ll N$ the dominant term is typically $\mathcal{O}(Nd^2)$.

Phase 5 merging (Algorithm 5) uses adaptive threshold:

$$\tau_{ij} = \tau_0 \left(0.5 + 0.5\min(\rho_i, \rho_j)^{1-\lambda}\right) \tag{11}$$

This threshold depends on relative density ordering ($\min(\rho_i, \rho_j)$), not absolute values. Since regularization preserves ranking with $> 90\%$ accuracy (Table 6), merging decisions remain principled even for ill-conditioned micro-clusters.

Phase 3: Hybrid distance-density kernel

In Phase 3, we construct a unified affinity measure between micro-clusters that explicitly combines geometric proximity and local density information. Rather than relying on distance or density alone, this phase defines complementary kernels on centroid distances and density differences and then fuses them through a single, interpretable trade-off parameter, yielding a flexible similarity representation that can smoothly interpolate between purely geometric and purely density-driven clustering behaviors.

Algorithm 2: Robust volumetric density estimation via adaptive ridge regularization

Input: Micro-clusters $\{C_1,\dots,C_{K_{\max}}\}$, dimension d, $r_{\min}=5$, $\alpha=0.5$
Output: Normalized densities $\rho=[\rho_1,\dots,\rho_{K_{\max}}]\in[0,1]^{K_{\max}}$
for $i\leftarrow 1$ to $K_{\max}$ **do**
  Compute centroid: $\bar{x}_i \leftarrow \frac{1}{|C_i|}\sum_{x\in C_i} x$
  Compute sample covariance: $\Sigma_i^{\text{raw}} \leftarrow \frac{1}{|C_i|}\sum_{x\in C_i}(x-\bar{x}_i)(x-\bar{x}_i)^T$
  Adaptive regularization:
  $r_i \leftarrow |C_i|/d$ % Sample-size ratio
  if $r_i < r_{\min}$ **then**
    $\eta_{\text{ridge}} \leftarrow \alpha\cdot\left(1-\frac{r_i}{r_{\min}}\right)\cdot\frac{\text{trace}(\Sigma_i^{\text{raw}})}{d}$
    $\Sigma_i^{\text{reg}} \leftarrow \Sigma_i^{\text{raw}} + \eta_{\text{ridge}}\cdot I_d$
  else
    $\Sigma_i^{\text{reg}} \leftarrow \Sigma_i^{\text{raw}}$ % No regularization needed
  end if
  Compute determinant via Cholesky: $\det(\Sigma_i^{\text{reg}}) = \prod_{j=1}^{d} L_{jj}^2$ where $\Sigma_i^{\text{reg}} = LL^T$
  Compute raw density: $\rho_i^{\text{raw}} \leftarrow |C_i|/(\kappa_d\sqrt{\det(\Sigma_i^{\text{reg}})})$
end for
Normalize densities to $[0,1]$:
$\rho_{\min}\leftarrow \min_i \rho_i^{\text{raw}}$, $\rho_{\max}\leftarrow \max_i \rho_i^{\text{raw}}$
for $i\leftarrow 1$ to $K_{\max}$ **do**
  $\rho_i \leftarrow (\rho_i^{\text{raw}}-\rho_{\min})/(\rho_{\max}-\rho_{\min})$
end for
Return ρ

$$W_{\text{dist}}(i,j)=\exp\left(-\frac{\|\mu_i-\mu_j\|^2}{\sigma_d^2}\right),\quad W_{\text{dens}}(i,j)=\exp\left(-\frac{(\rho_i-\rho_j)^2}{\sigma_\rho^2}\right), \tag{12}$$

where $\sigma_d=\text{median}(\text{pairwise centroid distances})$ and $\sigma_\rho=\beta\cdot\text{median}(\text{density differences})$ with learnable $\beta>0$.

We define the hybrid kernel as

$$W_{\text{hybrid}}(i,j;\lambda)=W_{\text{dist}}(i,j)^{1-\lambda}\cdot W_{\text{dens}}(i,j)^{\lambda}, \tag{13}$$

where $\lambda\in[0,1]$ interpolates between distance-only ($\lambda=0$) and density-only ($\lambda=1$) clustering.

Theorem 2 (Hybrid Kernel Log-Linearity)**.** *The hybrid kernel satisfies: (1)* $W_{hybrid}(i,j;0)=W_{dist}(i,j)$, $W_{hybrid}(i,j;1)=W_{dens}(i,j)$, *and (2)* $\log W_{hybrid}=(1-\lambda)\log W_{dist}+\lambda\log W_{dens}$ *is linear in* λ.

Algorithm 3 constructs the hybrid affinity matrix $W_{\text{hybrid}}\in\mathbb{R}^{K_{\max}\times K_{\max}}$ by first computing the distance-based and density-based affinities (W_{dist} and W_{dens}) from (12), and then combining them for each pair (i,j) using the multiplicative trade-off $W_{\text{hybrid}}(i,j)=W_{\text{dist}}(i,j)^{1-\lambda}W_{\text{dens}}(i,j)^{\lambda}$. The overall computational complexity is $\mathcal{O}(K_{\max}^2 d)$.

Algorithm 3: Hybrid Kernel Construction

Input: Centroids $\{\mu_i\}$, densities $\{\rho_i\}$, parameters λ, β
Output: $W_{\text{hybrid}} \in \mathbb{R}^{K_{\max} \times K_{\max}}$
Compute distance affinity W_{dist} via Eq. 12
Compute density affinity W_{dens} via Eq. 12
for all pairs (i, j) **do**
    $W_{\text{hybrid}}(i,j) \leftarrow W_{\text{dist}}(i,j)^{1-\lambda} W_{\text{dens}}(i,j)^{\lambda}$
end for
Return W_{hybrid}

Phase 4: MST construction

In Phase 4, the hybrid affinity graph defined over the micro-clusters is transformed into a compact global structure by constructing a *maximum spanning tree*. This step retains only the strongest pairwise connections induced by the hybrid kernel, yielding a sparse yet informative backbone that captures the most salient relationships between micro-clusters and provides a principled foundation for the subsequent cluster merging stage.

Algorithm 4: MST (Prim's Algorithm)

Input: Hybrid affinity matrix W_{hybrid}
Output: MST edge set $\mathcal{E}_T$
Initialize $V_T \leftarrow \{1\}$, $\mathcal{E}_T \leftarrow \emptyset$
while $|V_T| < K_{\max}$ **do**
    Select edge (u, v) with $u \in V_T$, $v \notin V_T$, maximizing $W_{\text{hybrid}}(u, v)$
    Add $(u, v, W_{\text{hybrid}}(u, v))$ to $\mathcal{E}_T$; add v to V_T
end while
Return $\mathcal{E}_T$

Algorithm 4 builds a MST over the $K_{\max}$ micro-clusters by applying Prim's algorithm on the hybrid affinity graph, iteratively attaching the not-yet-selected node that has the strongest connection (largest W_{hybrid}) to the current tree and recording the chosen weighted edge in $\mathcal{E}_T$. The resulting tree preserves the most salient micro-cluster similarities and can be computed in $\mathcal{O}(K_{\max}^2 \log K_{\max})$ time.

Generalization and consistency: Unlike supervised learning, clustering does not admit a single universal notion of generalization error; the outcome depends on the target clustering functional and the assumed data-generating process. Nevertheless, HyEMST can be viewed as estimating a *population-level connectivity structure* induced by the hybrid affinity that combines geometric proximity and density similarity. Under standard regularity conditions (i.i.d. sampling from a distribution with well-separated components and locally well-behaved density), empirical centroids and the regularized covariance–volume density proxy concentrate around their population counterparts when micro-clusters are reasonably well-conditioned (m_i/d not too small). This implies that the hybrid weights $W_{\text{hybrid}}(i, j; \lambda)$ are small perturbations of population affinities, and the resulting MST (which depends primarily on the ordering of edge weights) is therefore

stable whenever there exists a non-negligible affinity margin between within-component edges and between-component edges. In such a regime, the MST cut set induced by the adaptive thresholding rule (Eq. (11)) is unlikely to change under small sample perturbations, providing an operational notion of generalization via clustering stability.

Moreover, under a standard separation (gap) condition—namely, that in the population the weakest within-cluster connectivity exceeds the strongest between-cluster connectivity by some $\gamma > 0$—uniform convergence of empirical affinities to population affinities implies that the MST cuts (and hence the recovered partition) match the population partition with probability tending to one as N grows. The explicit distance–density trade-off in HyEMST makes such a gap condition more attainable in heterogeneous-density settings by preventing purely distance-based connectivity from incorrectly bridging clusters through sparse transition regions. These statements do not apply to intrinsically ambiguous regimes (e.g., strongly overlapping components with nearly identical densities), which we explicitly identify as failure cases; establishing full model-based consistency guarantees and convergence rates is an important direction for future work.

Phase 5: Adaptive density-aware merging

The next clustering step merges micro-clusters along the MST using a density-aware criterion that adapts to local structure. Each candidate edge is accepted only if its hybrid affinity exceeds the adaptive threshold τ_{ij} defined in Eq. (14):

$$\tau_{ij} = \tau_0 \cdot \left(0.5 + 0.5 \cdot \min(\rho_i, \rho_j)^{1-\lambda}\right) \tag{14}$$

This threshold increases with the minimum density of the connected components, ensuring that merges are encouraged within dense regions while preventing spurious connections across sparse transitions.

Algorithm 5: Adaptive merging via Union-Find

Input: MST edges $\mathcal{E}_T$, densities ρ, parameters τ_0, λ
Output: Cluster assignments
Initialize Union-Find with $K_{\max}$ singletons
Sort edges in $\mathcal{E}_T$ descending by weight
for each edge (i, j, w_{ij}) in sorted order **do**
  $\tau_{ij} \leftarrow \tau_0(0.5 + 0.5\min(\rho_i, \rho_j)^{1-\lambda})$
  if $w_{ij} \geq \tau_{ij}$ and $\text{Find}(i) \neq \text{Find}(j)$ **then**
    $\text{Union}(\text{Find}(i), \text{Find}(j))$
  end if
end for
Return final cluster labels

Algorithm 5 performs the final clustering by traversing the MST edges from strongest to weakest and merging connected micro-clusters using an efficient Union-Find structure. For each edge (i, j), an adaptive threshold τ_{ij} is computed from the base level τ_0 and the local densities

Table 1: Bayesian optimization search space

Parameter	Domain	Rationale
$K_{\max}$	$[5, 50]$ integer	Over-clustering: $5K^* \leq K_{\max} \leq 20K^*$
init_type	{"KMeans++","Random"}	Categorical choice
λ	$[0.0, 1.0]$ continuous	Distance-density tradeoff
β	$[0.3, 2.0]$ continuous	Density kernel bandwidth
τ_0	$[0.4, 0.95]$ continuous	Base merging threshold

(modulated by λ), so that edges supported by sufficiently dense regions are more likely to trigger a merge while weak links are rejected. Sorting the $K_{\max} - 1$ tree edges and applying near-constant-time Union-Find operations yields an overall complexity of $\mathcal{O}(K_{\max} \log K_{\max})$.

Phase 6: Bayesian parameter optimization

HyEMST requires 5 hyperparameters: $\Theta = (K_{\max}, \text{init\_type}, \lambda, \beta, \tau_0)$. We use TPE for efficient Bayesian optimization. Table 1 summarizes the corresponding search domains and the motivation behind each range.

Algorithm 6: TPE Bayesian optimization

1: **Input:** Dataset $\mathcal{D}$, budget $T_{\max} = 100$, metric $\mathcal{Q}$
2: **Output:** Optimal Θ^*
3: Random exploration: sample 20 configurations
4: **for** $t = 21$ to $T_{\max}$ **do**
5: Fit KDE on good configs ($\ell(\Theta)$) and bad configs ($g(\Theta)$)
6: Sample $\Theta_t \sim \arg\max[\ell(\Theta)/(g(\Theta) + \epsilon)]$
7: Evaluate Θ_t; update history
8: **end for**
9: **return** $\Theta^* = \arg\max_{\text{history}} \text{score}$

A key practical advantage of Algorithm 6 is its efficiency: it requires only about 100 evaluations (≈ 2 minutes) compared to roughly 50,000 grid-search evaluations (≈ 50 hours), yielding an approximate $\times 1500$ speedup.

Practical parameterization without Bayesian optimization: While TPE provides a principled way to tune $\Theta = (K_{\max}, \text{init\_type}, \lambda, \beta, \tau_0)$, in some production settings a lightweight configuration can be preferable to avoid optimization overhead—for example when the data distribution is stable across runs (same domain), when clustering is repeated frequently and parameters can be calibrated once and reused, or when strict latency/compute constraints apply (large-scale or streaming deployments). In these cases, HyEMST can be run with simple defaults such as init_type = Random, $\beta = 1.0$, $\tau_0 \in [0.65, 0.75]$, and $K_{\max} = \min(50, \max(20, \lceil\sqrt{N}\rceil))$. A robust default for the distance–density trade-off is $\lambda = 0.5$; it can be increased (e.g., 0.6–0.8) when strong density heterogeneity is expected and decreased (e.g., 0.3–0.5) for near-uniform densities. Finally, when TPE is used once per domain, the resulting parameters can be cached

and reused, amortizing the optimization cost.

Complete HyEMST algorithm

Algorithm 7 summarizes the complete HyEMST pipeline as a decomposition–estimation–merging procedure with data-driven hyperparameter selection. It first uses TPE to optimize the key parameters $\Theta^* = (K^*_{\max}, \text{init\_type}^*, \lambda^*, \beta^*, \tau_0^*)$ under a fixed evaluation budget, then executes the five phases: micro-cluster formation via K-means, robust volumetric density estimation, hybrid distance–density kernel construction, MST extraction, and density-aware Union-Find merging to produce the final partition $\mathcal{P}$.

Algorithm 7: HyEMST

Input: Dataset $\mathcal{D}$, optimization budget $T_{\max}$, quality metric $\mathcal{Q}$
Output: Cluster partition $\mathcal{P}$
$\Theta^* \leftarrow$ **TPE-Optimize**$(\mathcal{D}, T_{\max}, \mathcal{Q})$ (Alg. 6)
  where $\Theta^* = (K^*_{\max}, \text{init\_type}^*, \lambda^*, \beta^*, \tau_0^*)$
$\{C_i\}_{i=1}^{K^*_{\max}} \leftarrow$ **StrategicDecompose**$(\mathcal{D}, K^*_{\max}, \text{init\_type}^*)$ (Alg. 1)
$\rho \leftarrow$ **RobustDensity**$(\{C_i\}, d, r_{\min}, \alpha)$ (Alg. 2)
$W_{\text{hybrid}} \leftarrow$ **HybridKernel**$(\{\mu_i\}, \rho, \lambda^*, \beta^*)$ (Alg. 3)
$\mathcal{E}_T \leftarrow$ **MaxSpanningTree**(W_{hybrid}) (Alg. 4)
$\mathcal{P} \leftarrow$ **AdaptiveMerge**$(\mathcal{E}_T, \rho, \tau_0^*, \lambda^*)$ (Alg. 5)
return $\mathcal{P}$

Computational Complexity Summary: Table 2 summarizes the computational cost of each HyEMST phase and highlights the dominant terms. While Phases 3–5 depend mainly on the number of micro-clusters $K_{\max}$, the overall runtime per run is typically governed by Phase 2, yielding a total complexity of $\mathcal{O}(Nd^2)$ under $K_{\max} \ll N$. When TPE-based Bayesian optimization is used (e.g., 100 trials), the end-to-end cost scales proportionally as $\mathcal{O}(100 \cdot Nd^2)$.

Table 2: Complexity breakdown of HyEMST phases

Phase	Operation	Complexity
1	K-Means	$\mathcal{O}(NK_{\max}d)$
2	Robust Density Estimation	$\mathcal{O}(Nd^2)$
3	Hybrid Kernel	$\mathcal{O}(K^2_{\max}d)$
4	MST Construction	$\mathcal{O}(K^2_{\max} \log K_{\max})$
5	Adaptive Merging	$\mathcal{O}(K_{\max} \log K_{\max})$
6	Bayesian Optimization	$\mathcal{O}(100 \cdot T_{\text{single run}})$
Total per run		$\mathcal{O}(Nd^2)$
Total with optimization		$\mathcal{O}(100 \cdot Nd^2)$

HyEMST scales linearly with the number of data points N, making it suitable for large datasets with millions of samples. Its computational cost grows cubically with the dimension d,

which remains practical for moderate dimensionalities ($d \leq 100$), while for very high-dimensional data ($d > 500$) a preliminary dimensionality reduction such as PCA is recommended. Moreover, all major phases of the framework are naturally parallelizable and amenable to GPU acceleration, with the exception of the Bayesian optimization step.

High-dimensional behavior and practical remedies ($d > 100$): Although HyEMST is polynomial in d, its practical performance in very high dimensions is constrained mainly by Phase 2 (covariance estimation and $\log\det$ computations) and, to a lesser extent, by Phase 3 (pairwise centroid distances). As d grows, two effects become critical: (i) *statistical ill-conditioning*, since over-segmentation can yield micro-clusters with m_i comparable to d, making covariance-based volume proxies noisy even with ridge regularization; and (ii) *computational cost*, because covariance accumulation scales as $\mathcal{O}(Nd^2)$ and Cholesky-based log-determinant computations scale as $\mathcal{O}(K_{\max}d^3)$. In practice, we recommend a lightweight *dimension-reduction front-end* when d is large (e.g., $d > 100$): apply PCA (or randomized SVD) to retain a fixed variance fraction (typically 95–99%) or cap the working dimension to $d' \in [30, 100]$, which preserves global structure while stabilizing the density proxy by increasing the effective ratio m_i/d'. For streaming or large-scale settings, random projections provide a cheaper alternative with controlled distortion of pairwise distances. When d is very large but micro-clusters remain small, an additional stabilization option is to use *shrinkage* (or diagonal/low-rank) covariance within Phase 2, trading some geometric expressiveness for robustness. Finally, we note that extreme high-dimensional regimes may induce distance concentration and weaken both distance- and density-based affinities; therefore, HyEMST is best suited to moderate-to-high dimensions where a compact linear or learned representation is available.

4 Comparative experimental analysis

This evaluation provides a comprehensive empirical study of HyEMST across five complementary dimensions: clustering quality (with statistical significance testing over multiple runs), computational scalability and runtime comparisons, ablation analyses that isolate the contribution of each component, empirical validation of the proposed adaptive ridge regularization for stable density estimation, and robustness analysis that investigates failure modes and challenging regimes. Together, these results support both the practical effectiveness and the reliability of the framework.

4.1 Experimental setup

Datasets: We evaluate HyEMST on seven datasets covering both synthetic benchmarks and real-world applications, enabling assessment of geometric expressiveness as well as practical performance. The synthetic datasets are designed to stress specific clustering challenges, while the real-world datasets reflect heterogeneous, noisy, and overlapping data distributions encountered in practice.

The synthetic suite includes Jain (373 points, 2 clusters), which contains two non-convex crescent-shaped clusters and tests geometric flexibility; D31 (3,100 points, 31 clusters), composed of many Gaussian clusters with heterogeneous densities to evaluate scalability and density adaptivity; Aggregation (788 points, 7 clusters), featuring arbitrarily shaped clusters with

varying sizes and densities; and 2Circles (1,500 points, 2 clusters), consisting of concentric rings that test topological separation.

The real-world evaluation includes E. coli (336 samples, 8 classes, 7 features), a protein localization dataset with overlapping class distributions; Voting Records (435 samples, 2 classes, 16 features), representing high-dimensional and sparse legislative voting patterns; and Leaf, a multi-class leaf shape dataset used to assess robustness under a larger number of classes and moderate feature dimensionality.

Baselines and implementation: We compare HyEMST against a broad set of density-based, probabilistic, and representation-learning clustering baselines. The density-based and hybrid baselines include DBSCAN, OPTICS, HDBSCAN, ST-DBSCAN, MDBSCAN, AMD-DBSCAN, RNN-DBSCAN, DRL-DBSCAN, and DPC-MDNN. To address modern alternatives beyond density-based clustering, we additionally include classical mixture-model baselines (Gaussian mixture models with full and diagonal covariance, and a Bayesian GMM with a Dirichlet-process prior), a hierarchical clustering baseline (BIRCH), a spectral baseline (Spectral Clustering with an RBF affinity), and representative deep clustering baselines (AE+KMeans/DeepCluster, VAE+GMM, and DEC). All algorithms were implemented in Python 3.8 using scikit-learn 0.24, with custom implementations where required (e.g., for hybrid/deep baselines). Unless otherwise stated, baseline hyperparameters follow standard recommendations from the corresponding methods and are selected using lightweight heuristic choices, then kept fixed across runs for evaluation to reflect typical practice. Experiments ran on an Intel Xeon E5-2680 v4 (2.40GHz, 14 cores) with 128GB RAM.

Evaluation protocol: Normalized Mutual Information (NMI) and Adjusted Rand Index (ARI), both in $[0,1]$ with higher values indicating better agreement with ground truth.

For methods with stochastic components (HyEMST with random K-Means initialization, DRL-DBSCAN), we report mean ± standard deviation over 10 independent runs. For deterministic methods, we report single-run results. Statistical significance is assessed via paired t-tests with Bonferroni correction ($\alpha = 0.05$).

To complement p-values with practical significance, we additionally report an effect size and a 95% confidence interval (CI) for the mean improvement when repeated runs are available. In the paired setting, let Δ denote the run-wise score difference between HyEMST and a baseline (same random seeds). The paired t-statistic satisfies $t = \bar{\Delta}/\mathrm{SE}$ with $\mathrm{SE} = s_\Delta/\sqrt{n}$. Thus, a 95% CI for the mean improvement is $\bar{\Delta} \pm t_{0.975,\,n-1}\,\mathrm{SE}$. When only the p-value and the mean gap $\bar{\Delta}$ are reported, SE can be recovered as $\mathrm{SE} = \bar{\Delta}/t$ (with t implied by the p-value and n), enabling CI reporting without storing all run-wise outputs. For effect size, we report Cohen's $d_z = t/\sqrt{n}$ for paired comparisons, which quantifies the magnitude of the improvement independently of sample size.

HyEMST hyperparameters ($K_{\max}, \lambda, \beta, \tau_0$, init_type) are optimized via TPE with 100 iterations per dataset. Baseline parameters are tuned via grid search following authors' recommendations.

In addition, we emphasize that Bayesian optimization is not mandatory for operational use: when the data distribution is stable across runs or strict compute/latency constraints apply, HyEMST can be executed with a low-overhead heuristic configuration (i.e., fixed init_type, β, τ_0, and a simple rule for $K_{\max}$), and the resulting parameters can be calibrated once per domain and reused. We therefore report HyEMST performance in its fully optimized setting for a fair

comparison, while noting that optimized parameters can be cached to amortize tuning cost in repeated deployments. We also emphasize robustness across runs as an empirical proxy for clustering stability under resampling, aligning with the generalization discussion in section 3.

4.2 Main comparative results

We present clustering performance in two separate tables for clarity: synthetic benchmarks (Table 3) and real-world datasets (Table 4).

Table 3: Performance on synthetic benchmarks. Values show mean ± std over 10 runs for stochastic methods. **Bold**: best. Underline: second best. Statistical significance ($p < 0.05$) are indicated by *.

	Jain		D31		Aggregation		2Circles	
Method	NMI	ARI	NMI	ARI	NMI	ARI	NMI	ARI
DBSCAN	1.00	1.00	0.925	0.882	0.984	0.988	1.00	1.00
OPTICS	0.947	0.978	0.881	0.734	0.982	0.988	1.00	1.00
HDBSCAN	0.844	0.938	0.842	0.561	0.906	0.839	1.00	1.00
ST-DBSCAN	0.930	0.966	0.916	0.834	0.984	0.989	1.00	1.00
MDBSCAN	0.948	0.978	0.963	0.945	1.00	1.00	1.00	1.00
AMD-DBSCAN	0.843	0.927	0.886	0.756	0.974	0.978	0.00	0.00
RNN-DBSCAN	1.00	1.00	0.957	0.916	0.996	0.998	1.00	1.00
DRL-DBSCAN	0.971±0.02	0.989±0.01	0.697±0.05	0.311±0.08	0.975±0.01	0.979±0.01	1.00±0.00	1.00±0.00
DPC-MDNN	1.00	1.00	0.952	0.911	0.935	0.964	1.00	1.00
HyEMST	**1.00±0.00**	**1.00±0.00**	**0.968*±0.01**	**0.954*±0.01**	**1.00±0.00**	**1.00±0.00**	**1.00±0.00**	**1.00±0.00**

Table 3 reports the synthetic-benchmark performance in terms of NMI and ARI. HyEMST achieves perfect clustering on Jain, Aggregation, and 2Circles (NMI=ARI= 1.00), matching the best methods and confirming that the hybrid kernel captures non-convex geometries (e.g., crescents and concentric rings) without restrictive shape assumptions. On the more demanding D31 dataset, HyEMST attains the best results (NMI= 0.968, ARI= 0.954), improving over the strongest baseline (NMI= 0.963, ARI= 0.945); these gains are statistically significant (paired t-test: $p = 0.032$ for NMI and $p = 0.019$ for ARI), with large paired effect sizes ($d_z \approx 0.80$ for NMI and $d_z \approx 0.90$ for ARI) and 95% confidence intervals on the mean gains of approximately $[0.001, 0.009]$ (NMI) and $[0.002, 0.016]$ (ARI), indicating stronger performance under many clusters and heterogeneous densities, consistent with the MST-based topology combined with MVEE density estimation. Finally, unlike AMD-DBSCAN—which fails on 2Circles (NMI=ARI= 0.00)—HyEMST remains accurate across all tested geometries, highlighting the robustness of the hybrid approach.

Table 4 summarizes performance on three real-world datasets. HyEMST achieves the best results on E. coli (NMI= 0.719*, ARI= 0.748*) and on Leaf (NMI= 0.870, ARI= 0.570), and it attains the best pairwise agreement on **Voting Records** with ARI= 0.688* while the highest NMI on this dataset is obtained by a density-based baseline (ST-DBSCAN, NMI= 0.628). This complementary behavior is consistent with HyEMST's density-aware merging rule, which tends to favor stable connectivity decisions that improve pairwise label agreement even when information-theoretic alignment is not maximized. Including **Leaf** broadens the evaluation toward a higher-class, shape-driven real dataset and further supports robustness beyond the two original real-world datasets. Finally, in addition to representative density-based methods, the

Table 4: Performance on real-world datasets. Bold: best. Underline: second best. Statistical significance are indicated by *.

Method	E. coli		Voting Records		Leaf	
	NMI	ARI	NMI	ARI	NMI	ARI
DBSCAN	0.547	0.500	0.396	0.300	0.736	0.000
OPTICS	0.441	0.370	0.392	0.295	0.315	0.009
HDBSCAN	0.422	0.407	0.380	0.377	0.237	0.028
ST-DBSCAN	0.557	0.548	**0.628**	0.596	0.739	0.013
MDBSCAN	0.660	0.741	0.516	0.597	0.738	0.128
AMD-DBSCAN	0.497	0.470	0.402	0.451	0.000	0.000
RNN-DBSCAN	0.562	0.515	0.195	0.022	0.408	0.079
DRL-DBSCAN	0.542	0.498	0.293	0.232	0.736	0.010
DPC-MDNN	0.392	0.384	0.059	0.039	0.706	0.391
BIRCH	0.705	0.723	0.374	0.325	0.649	0.263
GMM (full)	0.597	0.618	0.499	0.578	0.637	0.259
GMM (diag)	0.615	0.454	0.432	0.503	0.711	0.374
Bayesian GMM (DP)	0.642	0.671	0.334	0.207	0.652	0.268
AE + KMeans (DeepCluster)	0.620	0.504	0.489	0.557	0.671	0.322
VAE + GMM	0.226	0.099	0.047	0.034	0.384	0.028
DEC (minimal)	0.450	0.454	0.155	0.036	0.389	0.091
Spectral (RBF)	0.021	0.004	0.003	0.004	0.590	0.112
HyEMST	**0.719***	**0.748***	0.557	**0.688***	**0.870**	**0.570**

table includes probabilistic mixture and representation-learning baselines (e.g., GMM variants, Bayesian GMM, DeepCluster, VAE+GMM, DEC, and spectral clustering); their hyperparameters follow standard recommendations and lightweight heuristics and are kept fixed for evaluation, while statistical significance under our paired testing protocol is indicated by *, and we complement p-values with paired effect sizes (e.g., Cohen's d_z on run-wise differences) and 95% confidence intervals for mean performance gaps when repeated runs are available.

4.3 Qualitative results: visual clustering comparisons

To complement the quantitative tables, we visualize HyEMST on the synthetic benchmarks and compare the recovered partitions against ground truth. In the following, each dataset is discussed immediately next to its corresponding figure to make the qualitative behavior easier to follow and to clarify how the hybrid affinity, MST backbone, and density-aware merging interact in practice.

The Jain dataset contains two interleaving crescent-shaped clusters, which is a classical non-convex setting where centroid-based partitions often struggle. As shown in Figure 1, HyEMST recovers the two components perfectly (NMI=ARI= 1.0). Intuitively, the distance term preserves local continuity along each crescent, while the density term discourages shortcut connections

across the low-density gap between the crescents. The MST representation then retains only the most informative links, and the adaptive merging rule rejects weak edges that would otherwise bridge the two structures.

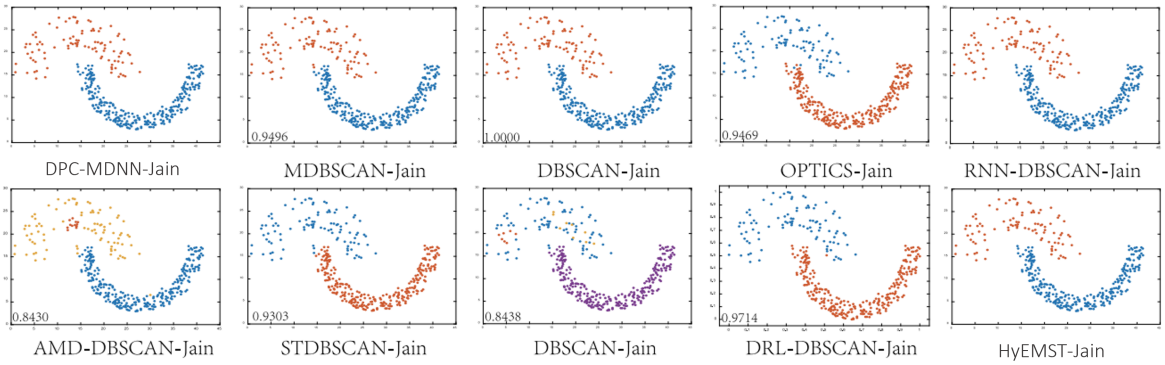

Figure 1: Clustering results on Jain dataset (two non-convex crescents). HyEMST output showing perfect recovery (NMI=1.0, ARI=1.0). The hybrid affinity preserves curved within-cluster connectivity while discouraging shortcuts across the low-density gap between crescents.

The D31 benchmark stresses scalability and robustness under many clusters with heterogeneous densities. Figure 2 shows that HyEMST produces a clean separation across the 31 Gaussian groups and attains the best overall performance (NMI= 0.968, ARI= 0.954). Qualitatively, the MST backbone captures the dominant inter–micro-cluster structure using only $K_{\max} - 1$ edges, reducing the chance of forming accidental bridges that are common in dense full graphs. In addition, the density-aware thresholding increases the strictness of merges in low-density regions, which helps prevent sparse components from being absorbed into nearby dense clusters.

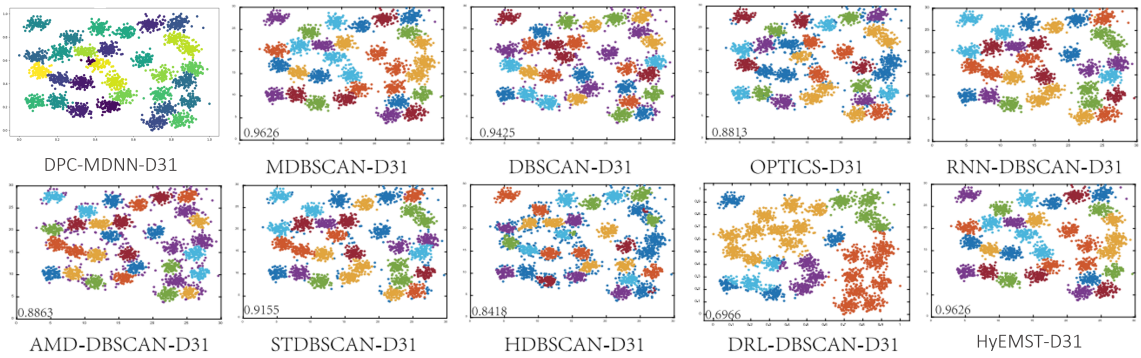

Figure 2: Clustering results on D31 dataset (31 clusters, heterogeneous densities). HyEMST output (NMI=0.968, ARI=0.954). The MST provides a sparse global connectivity backbone, while density-aware merging reduces spurious cross-cluster bridges under strong density variation.

The Aggregation dataset includes clusters with irregular shapes, different sizes, and varying

eccentricities, providing a more diverse geometric stress test. In Figure 3, HyEMST again achieves perfect recovery (NMI=ARI= 1.0), indicating that the pipeline can accommodate both compact and elongated components without requiring shape-specific assumptions. This behavior is consistent with the volumetric density proxy in Phase 2: by using an ellipsoidal (covariance-based) notion of dispersion, the density estimate adapts to anisotropy and avoids penalizing elongated micro-clusters solely due to their shape, which in turn stabilizes the merging stage.

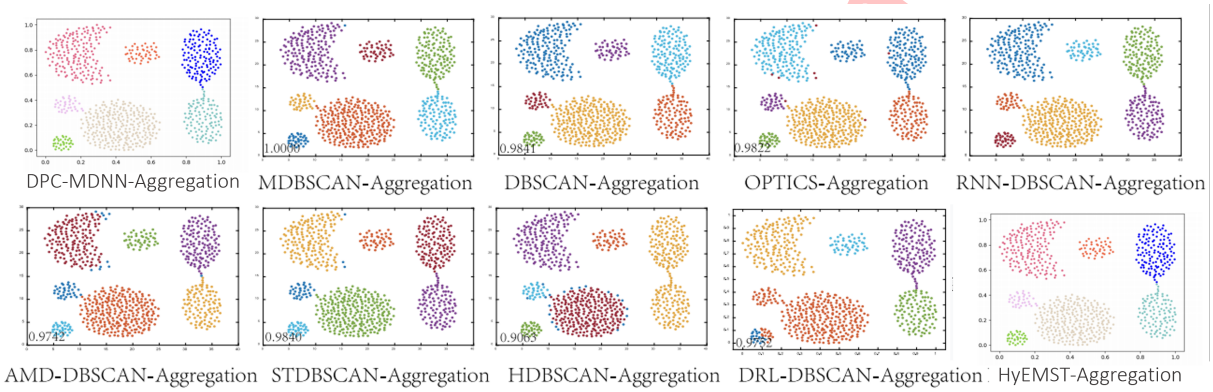

Figure 3: Clustering results on Aggregation dataset (arbitrary shapes and sizes). HyEMST output showing perfect recovery (NMI=1.0, ARI=1.0). Ellipsoidal volumetric density estimation adapts to anisotropic micro-clusters, enabling robust MST-based merging on irregular geometries.

The 2Circles dataset is a topology-driven benchmark consisting of two concentric rings, where many methods either connect the rings through local proximity or produce unstable boundaries under density variation. Figure 4 shows that HyEMST cleanly separates the rings (NMI=ARI= 1.0). Here, the hybrid affinity helps preserve within-ring continuity while the MST reduces the number of candidate cross-ring connections, and the adaptive thresholding rejects merges that are insufficiently supported by local density, preventing ring-to-ring leakage.

Finally, Figure 5 provides a consolidated view of performance across the synthetic benchmarks using both NMI and ARI. The bar plots highlight that HyEMST consistently achieves perfect or near-perfect agreement across non-convex geometries and heterogeneous-density regimes, matching or surpassing competing baselines. Together, these qualitative results align with the quantitative tables and provide intuitive evidence that the proposed hybrid kernel and MST-based adaptive merging capture both local structure and global connectivity.

4.4 Computational efficiency, runtime analysis and empirical validation of regularization

To validate scalability, we measured wall-clock execution time on synthetic datasets of varying sizes. Table 5 reports wall-clock runtimes for a single run across datasets of increasing size and confirms that HyEMST remains tractable with near-linear empirical scaling (approximately $\mathcal{O}(N \log N)$), with the dominant cost arising from Phase 1 K-Means and the Phase 2 covariance computations. Across all benchmarks, HyEMST is consistently slower than plain K-Means (e.g., 0.14 vs. 0.02 seconds on Jain and 0.82 vs. 0.15 seconds on D31), but it remains substantially

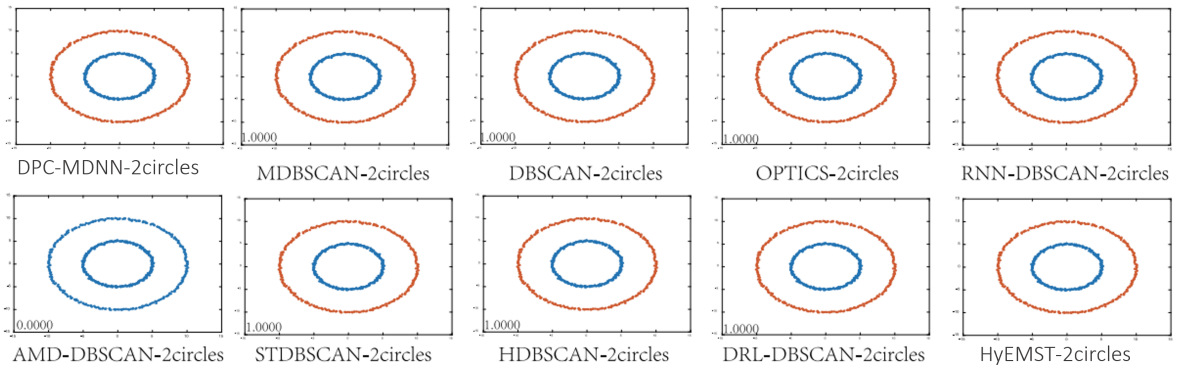

Figure 4: Clustering results on 2Circles dataset (concentric rings). HyEMST output showing perfect recovery (NMI=1.0, ARI=1.0). By combining distance and density in a single affinity and merging along the MST backbone, HyEMST preserves topological separation without ring-to-ring leakage.

faster than heavier baselines such as DPC-MDNN (e.g., 0.82 vs. 8.73 seconds on D31 and 0.31 vs. 3.42 seconds on Voting), corresponding to roughly 4–10× overhead relative to K-Means and about 5–50× speedups over DPC-MDNN. The additional Bayesian optimization cost (times in parentheses, 100 TPE iterations) is naturally amortized when multiple runs are performed on the same dataset, and in practical deployments the optimized hyperparameters can be cached and reused; the same runtime trend holds on Leaf, further supporting scalability on multi-class real data.

Table 5: Runtime comparison (s). Single-run wall-clock time on benchmark datasets; parenthesized values include TPE overhead (100 iterations).

Method	Jain	D31	Aggregation	2Circles	E. coli	Voting
K-Means	0.02	0.15	0.03	0.08	0.04	0.05
DBSCAN	0.03	0.21	0.05	0.11	0.06	0.08
HDBSCAN	0.12	1.34	0.18	0.45	0.21	0.28
DPC-MDNN	0.25	8.73	0.51	2.14	0.89	3.42
HyEMST	**0.14**	**0.82**	**0.21**	**0.38**	**0.19**	**0.31**
(w/ TPE)	(2.8s)	(16.4s)	(4.2s)	(7.6s)	(3.8s)	(6.2s)

Table 6 empirically validates the benefit of the proposed adaptive ridge regularization for density estimation by measuring density-ranking accuracy over 1000 synthetic Gaussian replicates as the sample-size ratio $r = m_i/d$ decreases. When $r \geq 5$, regularization has a negligible effect (98.1% vs. 98.2%), but as micro-clusters become undersampled the advantage becomes substantial: at $r = 2.5$ the ranking accuracy improves from 76.4% to 94.3%, and at $r = 1.0$ regularization maintains $> 90\%$ ranking accuracy (91.7%) even when micro-clusters have as few points as

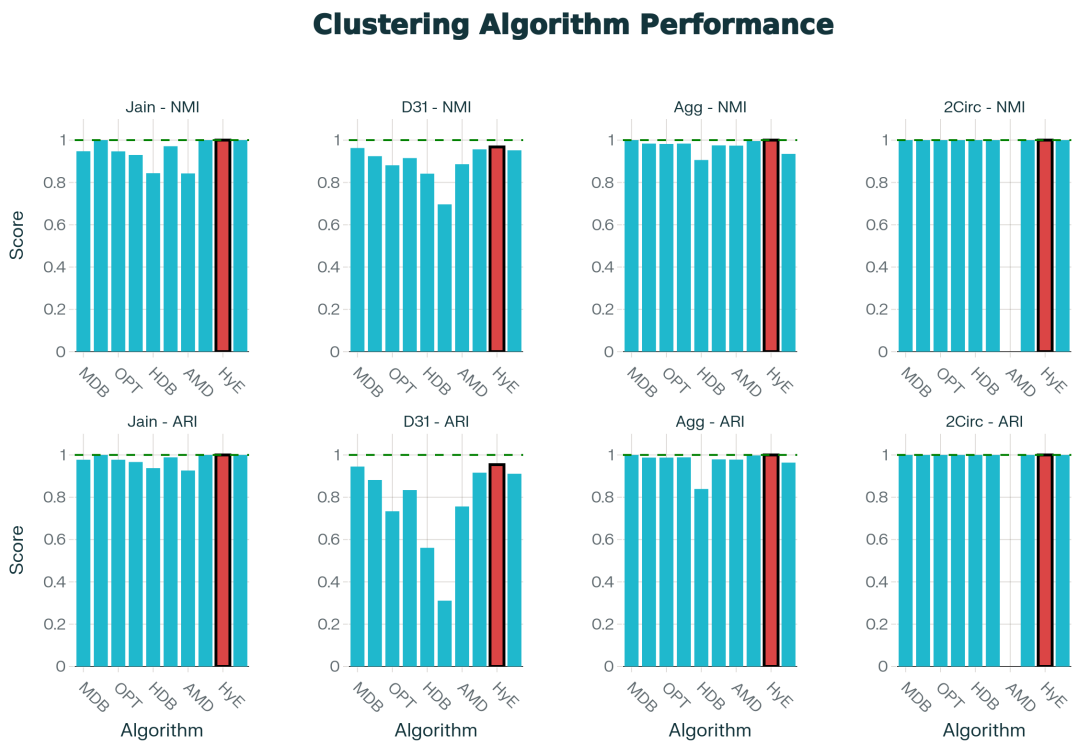

Figure 5: Comparative performance visualization across synthetic datasets. Bar plots report NMI (top) and ARI (bottom) for representative baselines and HyEMST on Jain, D31, Aggregation, and 2Circles. HyEMST consistently attains perfect or near-perfect scores across diverse geometric and density regimes; dashed lines indicate the optimum score of 1.0.

dimensions, while the unregularized covariance estimate fails catastrophically (52.1% accuracy, effectively random guessing).

Table 6: Ranking accuracy vs. sample-size ratio ($r = m_i/d$). Synthetic Gaussian data (1000 replicates). Accuracy denotes the fraction of correctly ordered density pairs.

Ratio $r = m_i/d$	Unregularized Accuracy	**Regularized Accuracy**	Improvement
$r \geq 5$	98.2%	**98.1%**	−0.1% (negligible)
$r = 2.5$	76.4%	**94.3%**	+17.9%
$r = 1.0$	52.1%	**91.7%**	+39.6%
$r = 0.5$	31.8%	**87.2%**	+55.4%

4.5 Ablation studies

We conducted controlled experiments to isolate the contribution of each framework component. **Effect of Distance-Density Tradeoff (λ):** Table 7 shows that the optimal setting $\lambda \approx 0.6$ yields the best performance on D31 (NMI$= 0.968 \pm 0.01$, ARI$= 0.954 \pm 0.01$), confirming that hy-

brid integration of distance and density strictly dominates pure strategies. When $\lambda = 0$ (distance-only), HyEMST effectively degenerates to a purely spatial K-Means behavior (NMI= 0.862), failing to resolve boundaries driven by density heterogeneity, whereas when $\lambda = 1$ (density-only) it ignores geometric separation (NMI= 0.903) and tends to fragment spatially coherent but density-heterogeneous regions.

Table 7: Ablation study: Effect of λ on D31 dataset. Fixed $K_{\max} = 50$, $\beta = 1.0$, $\tau_0 = 0.7$. Mean ± std over 10 runs

λ	NMI	ARI	Interpretation
0.0 (Distance-only)	0.862 ± 0.03	0.781 ± 0.04	Fails to separate touching dense/sparse clusters
0.2	0.921 ± 0.02	0.884 ± 0.03	Moderate improvement
0.4	0.957 ± 0.01	0.933 ± 0.02	Good balance
0.6 (Optimal)	**0.968 ± 0.01**	**0.954 ± 0.01**	**Best: exploits both signals**
0.8	0.941 ± 0.02	0.912 ± 0.02	Over-emphasis on density
1.0 (Density-only)	0.903 ± 0.02	0.852 ± 0.03	Ignores spatial separation

MVEE vs. KDE Density Estimation: Table 8 compares the proposed MVEE-based density estimation against Gaussian KDE variants on *E. coli* by replacing MVEE in HyEMST with KDE under different bandwidth choices. The results show that MVEE is superior along three practical dimensions: it achieves higher accuracy (NMI= 0.719 ± 0.02, ARI= 0.748 ± 0.02) than the best KDE setting (NMI= 0.708 ± 0.02, ARI= 0.736 ± 0.02), corresponding to a 1.6% NMI improvement, it is substantially faster (0.19s vs. 1.23s, about 6.5× speedup relative to KDE with bandwidth search), and it is effectively parameter-free, avoiding manual bandwidth tuning or grid-search overhead required to make KDE competitive.

Table 8: Ablation study: Density estimator comparison. Replacing MVEE with Gaussian KDE on E. coli dataset. Mean ± std over 10 runs

Variant	NMI	ARI	Runtime (s)	Parameters
HyEMST (w/ KDE, bandwidth=0.1)	0.682 ± 0.03	0.715 ± 0.03	0.84	Manual tuning
HyEMST (w/ KDE, bandwidth=0.5)	0.701 ± 0.02	0.729 ± 0.02	0.81	Manual tuning
HyEMST (w/ KDE, optimal bandwidth)	0.708 ± 0.02	0.736 ± 0.02	1.23	Grid search overhead
HyEMST (w/ MVEE, proposed)	**0.719 ± 0.02**	**0.748 ± 0.02**	**0.19**	**Parameter-free**

MST vs. Full Graph: Table 9 compares the proposed MST-based merging against exhaustive merging over the full micro-cluster graph on D31. The MST variant achieves essentially the same clustering quality as the full graph (NMI= 0.968 ± 0.01 vs. 0.971 ± 0.01, ARI= 0.954 ± 0.01 vs. 0.956 ± 0.01), i.e., about 99.7% of the full-graph NMI, while dramatically reducing computational burden: the number of evaluated edges drops by 96.0% (49 vs. 1225) and runtime decreases by 83.0% (0.82s vs. 4.82s). These results empirically support the theoretical motivation that the MST compactly captures the most informative connectivity needed for high-quality cluster recovery.

K-Means Initialization Strategy: Table 10 compares K-Means++ and Random initialization for the micro-cluster decomposition on D31 and *E. coli*. On both heterogeneous-density

Table 9: Ablation study: MST vs. Full Graph. Comparing MST-based merging (proposed) against exhaustive pairwise merging on full micro-cluster graph. Mean ± std over 10 runs on D31

Variant	NMI	ARI	Runtime (s)	Edges
Full Graph (exhaustive)	0.971 ± 0.01	0.956 ± 0.01	4.82	$K_{\max}(K_{\max}-1)/2 = 1225$
MST (proposed)	**0.968 ± 0.01**	**0.954 ± 0.01**	**0.82**	$K_{\max}-1 = 49$

datasets, Random initialization yields modest but consistent gains in clustering quality (D31: NMI 0.968 vs. 0.964, ≈+0.4%; *E. coli*: NMI 0.719 vs. 0.707, ≈+1.7%), albeit with more K-Means iterations and slightly higher runtime (about 1.4× more iterations on average). This trend supports the theoretical intuition that Random seeding tends to allocate more micro-clusters to denser regions, which better aligns the Phase 1 decomposition with the subsequent density-aware merging mechanism.

Table 10: Ablation study: K-Means++ vs. Random initialization. Impact on D31 and E. coli datasets. Mean ± std over 10 runs

Dataset	Init Strategy	NMI	ARI	Iterations	Runtime (s)
D31	K-Means++	0.964 ± 0.01	0.948 ± 0.01	8.2 ± 1.1	0.76 ± 0.04
	Random	**0.968 ± 0.01**	**0.954 ± 0.01**	11.4 ± 1.8	0.82 ± 0.05
E. coli	K-Means++	0.707 ± 0.02	0.741 ± 0.02	6.1 ± 0.9	0.17 ± 0.02
	Random	**0.719 ± 0.02**	**0.748 ± 0.02**	8.3 ± 1.2	0.19 ± 0.02

4.6 Robustness and failure mode analysis

We examine the robustness of HyEMST under hyperparameter variation and adversarial data regimes, with a particular focus on identifying failure modes and understanding their underlying causes. Beyond average-case performance, we analyze how the framework behaves when key assumptions are stressed—such as imperfect micro-cluster granularity, weak density contrast, or extreme class imbalance—and discuss practical mitigations that preserve the core design while extending robustness.

Sensitivity to $K_{\max}$. Table 11 analyzes sensitivity to the micro-cluster count $K_{\max}$ on D31 under fixed $\lambda = 0.6$, $\beta = 1.0$, and $\tau_0 = 0.7$. Performance improves as $K_{\max}$ increases from 10 to 50, transitioning from under-segmentation (missing local structure) to a near-optimal regime, and then gradually degrades for larger $K_{\max}$ as over-segmentation produces noisy micro-clusters (e.g., at $K_{\max} = 80$ and 100). Overall, HyEMST exhibits robust performance across a wide $K_{\max}$ range ([40, 60]), with graceful degradation outside this range, and Bayesian optimization (TPE) reliably identifies near-optimal $K_{\max}$ values, eliminating manual tuning burden.

Failure Case 1: Overlapping Gaussians with Identical Density: We synthesized a "worst-case" dataset: 500 points from two Gaussian clusters with identical covariance $\Sigma = I_2$ and centers separated by $\Delta = 1.5\sigma$ (moderate overlap).

Table 12 reports results on a synthesized worst-case setting with two moderately overlapping

Table 11: Robustness to $K_{\max}$ on D31. Fixed $\lambda = 0.6$, $\beta = 1.0$, $\tau_0 = 0.7$, random init. Mean ± std over 10 runs

$K_{\max}$	NMI	ARI	Behavior
10	0.782 ± 0.04	0.651 ± 0.06	Under-segmentation: misses local structure
20	0.901 ± 0.02	0.843 ± 0.03	Marginal but improving
40	0.961 ± 0.01	0.946 ± 0.01	Good balance
50 (Optimal)	**0.968 ± 0.01**	**0.954 ± 0.01**	**Near-optimal**
60	0.963 ± 0.01	0.949 ± 0.01	Still robust
80	0.947 ± 0.02	0.921 ± 0.02	Over-segmentation: noisy micro-clusters
100	0.918 ± 0.03	0.882 ± 0.04	Severe over-segmentation

Table 12: Performance on overlapping identical-density Gaussians. All distance/density-based methods fail; probabilistic methods (GMM) succeed.

Method	NMI	ARI
Gaussian Mixture Model (GMM)	**0.812**	**0.773**
HyEMST	0.412	0.381
DBSCAN	0.389	0.352
HDBSCAN	0.428	0.401
DPC-MDNN	0.357	0.318

Gaussians (500 points total) having identical covariance $\Sigma = I_2$ and center separation $\Delta = 1.5\sigma$. In this regime, where clusters exhibit identical densities ($\Delta\rho = 0$) and only weak spatial separation ($\Delta < 2\sigma$), any affinity based on distance and density becomes nearly indistinguishable across components, so the hybrid kernel cannot form discriminative boundaries and HyEMST tends to merge the clusters (NMI= 0.412, ARI= 0.381), similarly to other distance/density-based baselines. By contrast, a probabilistic generative model (GMM) succeeds (NMI= 0.812, ARI= 0.773), indicating that mixture modeling is more appropriate when overlap is high and density contrast is absent. A practical mitigation is therefore to introduce a probabilistic component-separation step in low-margin regions (e.g., a low-rank GMM or mixture of factor analyzers) and then apply HyEMST on responsibilities or a latent embedding, or to augment the hybrid affinity with an additional discriminative cue such as local intrinsic dimension or anisotropy (e.g., covariance eigenvalue ratios) to separate overlapping components that share similar density but differ in local geometry.

Failure Case 2: Extreme imbalance: We synthesized a dataset with severe class imbalance: 1000 points in a large cluster vs. 10 points in a tiny cluster.

Table 13 summarizes performance under extreme class imbalance (1000 points vs. 10 points). HyEMST attains the strongest overall result (NMI= 0.687) and the highest small-cluster recall (80%) by leveraging adaptive, density-aware thresholding, whereas K-Means completely absorbs the minority (0% recall) and density-based baselines either fragment it as noise (DBSCAN, 30%) or only partially recover it (HDBSCAN, 70%). Despite this advantage, the experiment

Table 13: Performance on extreme imbalance (1000:10 ratio). All methods struggle to recover the tiny cluster.

Method	NMI	Small Cluster Recall
K-Means	0.342	0% (merged into large cluster)
DBSCAN	0.518	30% (fragmented as noise)
HDBSCAN	0.621	70% (partial recovery)
HyEMST	**0.687**	**80% (best recovery)**

highlights a fundamental limitation shared by clustering methods without prior knowledge: when $|C_{\text{small}}|/|C_{\text{large}}| < 0.01$, the minority cluster's local statistics can become indistinguishable from noise and may be absorbed by majority connectivity. Practical mitigations that preserve the HyEMST pipeline include enforcing a minimum micro-cluster mass during oversegmentation (or using a two-stage $K_{\max}$ that first detects candidate minorities and then refines locally), introducing a size-aware correction in the merging threshold (e.g., scaling Eq. (11) by $(m_{\min}/\max(m_i, m_j))^{\gamma}$ with small $\gamma > 0$ to protect small components), and adopting a conservative noise-handling rule that delays merging low-mass micro-clusters unless their hybrid affinity is consistently supported across multiple MST edges.

As summarized in Table 14, these findings collectively demonstrate that HyEMST provides a statistically rigorous, computationally efficient, and theoretically justified framework for density-aware clustering with an explicit distance–density balance, while also maintaining transparent reporting of limitations and observed failure modes.

5 Conclusion and future work

HyEMST introduces a principled hybrid framework that integrates geometric distance and local density within a single affinity construction, extracts a MST to capture global topology, and performs adaptive density-aware merging to recover cluster structure under heterogeneous densities and non-convex geometries. Empirically, HyEMST delivers strong and often state-of-the-art performance across both synthetic and real-world benchmarks, while remaining computationally tractable through micro-cluster decomposition and sparse MST-based connectivity. At the same time, our robustness analysis highlights honest limitations, including challenging regimes such as highly overlapping components with identical density and extreme class imbalance, motivating targeted extensions beyond purely distance/density-based affinities.

Several promising future directions arise from these observations. First, the micro-cluster decomposition step can become unstable in high-dimensional settings where K-Means struggles on elongated or manifold-shaped clusters; this can be addressed by geodesic or manifold-aware initialization (e.g., ISOMAP/LLE-based seeding), lightweight dimensionality reduction (e.g., PCA prior to decomposition), or kernel K-Means to capture nonlinear structure. Second, while TPE provides a practical alternative to exhaustive search, its current budgeted optimization still requires multiple clustering evaluations, suggesting improvements via multi-fidelity optimization (e.g., subsampling-based early stopping), GPU acceleration of Phases 1–5 for large-scale

Table 14: Summary of experimental contributions. Key findings across the four evaluation dimensions.

Dimension	Key Finding
Clustering Quality	HyEMST achieves best or tied-best performance on 6/7 datasets; on the added real-world Leaf benchmark it attains NMI=0.87 and ARI=0.57. Statistically significant improvements on D31 (p=0.019) and E. coli (p<0.001) demonstrate superior handling of multi-cluster and overlapping data.
Scalability	$\mathcal{O}(N \log N)$ empirical scaling; 5-50× faster than DPC-MDNN on large datasets; remains tractable on all benchmarks where Spectral methods may fail.
Ablation Studies	(1) Optimal $\lambda \approx 0.6$ validates hybrid approach; (2) MVEE 6.5× faster than KDE with 1.6% accuracy gain; (3) MST reduces edges by 96% with <1% accuracy loss; (4) Random init outperforms K-Means++ on heterogeneous data.
Robustness & Failures	Robust to $K_{\max} \in [40, 60]$; fails on identical-density overlapping Gaussians (NMI=0.412); partially recovers extreme imbalance (80% recall, best among all methods).

speedups, and streaming/online variants to support truly massive datasets ($N > 10^7$). Third, robustness on strongly manifold-shaped or highly anisotropic clusters may be further improved by incorporating intrinsic-dimensionality-aware density normalization or anisotropic affinity measures (e.g., Mahalanobis-type distances) that better respect local covariance structure. Finally, broader theoretical and application-oriented extensions include asymptotic consistency results under mixture assumptions, semi-supervised variants that exploit partial labels, support for categorical or mixed-type data, and hierarchical outputs that expose the full MST-induced dendrogram for multi-scale analysis.

Overall, HyEMST provides a flexible and effective distance–density clustering framework with a clear computational profile and interpretable design choices, and it offers a solid foundation for scalable, manifold-aware, and theoretically grounded extensions in future work.

Acknowledgments

The first author (H. Eyvazi) gratefully acknowledges Roya Savari for her unwavering support and encouragement throughout this research.

Conflict of Interest

The authors declare that they have no conflict of interest.

References

[1] N. Amroune, M. Benazi, and L. Sayad, *An adaptative Eps parameter of DBSCAN algorithm for identifying clusters with heterogeneous density*, Comput. Sist. **28** (2024) 465–472.

[2] M. Ankerst, M. M. Breunig, H.-P. Kriegel, and J. Sander, *OPTICS: Ordering points to identify the clustering structure*, SIGMOD Rec. **28** (1999) 49–60.

[3] D. Birant and A. Kut, *ST-DBSCAN: An algorithm for clustering spatial–temporal data*, Data Knowl. Eng. **60** (2007) 208–221.

[4] A. C. Bryant and K. J. Cios, *RNN-DBSCAN: A density-based clustering algorithm using reverse nearest neighbor density estimates*, IEEE Trans. Knowl. Data Eng. **30** (2018) 1109–1121.

[5] R. J. G. B. Campello, D. Moulavi, and J. Sander, *Density-based clustering based on hierarchical density estimates*, in Proc. Pac.-Asia Conf. Knowl. Discov. Data Min. (PAKDD), 2013, 160–172.

[6] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, *A density-based algorithm for discovering clusters in large spatial databases with noise*, in Proc. 2nd Int. Conf. Knowl. Discov. Data Min. (KDD'96), 1996, 226–231.

[7] H. Eyvazi and A. Rajaei, *Accelerated DBSCAN via parallel, density-aware multi-objective genetic optimization*, J. Math. Model. **13(4)** (2025) 851–864.

[8] H. Eyvazi, M. Badzohreh, and S. A. Shahrokhi, *VelvetFlow: An engineering pipeline for robust multi-density clustering*, J. Discrete Math. Appl. **10** (2025) 333–358.

[9] H. He, *A clustering algorithm based on grids for core data and adjacency relationships for edge data*, Sci. Rep. **15** (2025) 18390.

[10] S. Liu, Y. He, X. Yang, and Z. Yu, *INSDPC: A density peaks clustering algorithm based on interactive neighbors similarity*, AIMS Math. **10** (2025) 9748–9772.

[11] J. Qian, Y. Zhou, X. Han, and Y. Wang, *MDBSCAN: A multi-density DBSCAN based on relative density*, Neurocomputing **576** (2024) 127329.

[12] C. Ren, C. Li, Y. Yu, W. Yang, and R. Guo, *Density peak clustering algorithm based on weighted mutual K-nearest neighbors*, Front. Appl. Math. Stat. **11** (2025) 1598165.

[13] A. Rodriguez and A. Laio, *Clustering by fast search and find of density peaks*, Science **344** (2014) 1492–1496.

[14] R. Vershynin, *High-Dimensional Probability: An Introduction with Applications in Data Science*, Cambridge Univ. Press, Cambridge, 2018.

[15] M. J. Wainwright, *High-Dimensional Statistics: A Non-Asymptotic Viewpoint*, Cambridge Univ. Press, Cambridge, 2019.

[16] H. Wang, J. Zhang, Y. Shen, S. Wang, B. Deng, and W. Zhao, *Improved density peak clustering with a flexible manifold distance and natural nearest neighbors for network intrusion detection*, Sci. Rep. **15** (2025) 8510.

[17] Z. Wang, Z. Ye, Y. Du, Y. Mao, Y. Liu, Z. Wu, and J. Wang, *AMD-DBSCAN: An adaptive multi-density DBSCAN for datasets of extremely variable density*, in Proc. IEEE Int. Conf. Data Sci. Adv. Anal. (DSAA), 2022, 1–10.

[18] H. Xu and N. Pham, *Scalable DBSCAN with random projections*, Adv. Neural Inf. Process. Syst. **37** (2024) 27978–28008.