SPECTRAL DENSITY PEAKS CLUSTERING

Anonymous authors
Paper under double-blind review

ABSTRACT

Cluster analysis is still an open problem despite the wide range of techniques available in the literature. Each family of clustering algorithms has distinct strengths and limitations: *graph-theoretical* approaches benefit from a solid linear algebra foundation, but struggle with structures of varying size and density; *density-based* methods can detect arbitrarily-shaped clusters, yet they often rely on sensitive and method-dependent heuristics. We introduce a spectral formulation of Density Peaks clustering, based on a transition matrix that encodes the probability of a point being the nearest higher-density neighbor of another. This matrix naturally integrates with Perron Cluster Analysis to coarse-grain Markov chains with complex spectra, addressing the limitations of both the aforementioned paradigms. The proposed method, **Spectral Density Peaks**, is evaluated on synthetic and real-world datasets against state-of-the-art competitors.

1 INTRODUCTION

Clustering algorithms are unsupervised learning tools that uncover inherent structures in unlabeled datasets. The foundational *partitional* method, k-means (KM) (Lloyd, 1982), requires specifying the number of clusters k a priori and favors convex clusters, motivating alternatives such as Kernel k-means (KKM) and Spectral Clustering (SC), which map data to a space where clusters become linearly (or graph-theoretically) separable (Dhillon et al., 2004; von Luxburg, 2007).

Density-based methods rely on nonparametric density estimates and differ in their design choices: Density-Based Spatial Clustering of Applications with Noise (DBSCAN) (Ester et al., 1996) identifies clusters as contiguous high-density regions separated by low-density regions, while Density Peaks (DP) (Rodriguez & Laio, 2014) and Mean Shift (MS) (Comaniciu & Meer, 2002) assign points to the modes of their underlying density functions. Extensions such as Hierarchical DBSCAN (HDBSCAN) (Campello et al., 2013) and DP Advanced (DPA) (d'Errico et al., 2021) address cluster hierarchies.

We introduce the **Spectral Density Peaks** (SDP) clustering algorithm, a modification of DP that leverages density estimates, together with their associated *uncertainties*, within the SC framework. In particular, clusters are identified as metastable states of a Markov chain constructed from the data and analyzed via Robust Perron Cluster Analysis (PCCA+) (Röblitz & Weber, 2013) in its extension to non-reversible chains with complex eigenvalues, namely complex-valued PCCA+ (cPCCA+) (Frank et al., 2024). The contributions of this work include: modeling the local density structure of a dataset by a *transition matrix* $\mathbf{P}$ that encodes the probability of one point being the nearest higher-density neighbor of another; *connecting* the graph-theoretical and density-based clustering paradigms, with the aim of mitigating their respective limitations; validating the SDP algorithm on synthetic and real-world data against state-of-the-art competitors.

The remainder of Section 1 reviews related work on integrating graph-theoretical and density-based clustering; Section 2 covers SDP preliminaries, Section 3 details the algorithm, Section 4 presents experiments, and Section 5 concludes.

1.1 RELATED WORK

An integration of the density-based MS algorithm within the graph-theoretical SC method is proposed (Ozertem et al., 2008), relying on anisotropic, variable-bandwidth kernel density estimation in order to match local data structures. More recently, a framework (Craig et al., 2022) interpolates

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

between MS and SC through Fokker–Planck equations on graphs, establishing new connections with diffusion maps. An alternative approach (Hess et al., 2019) combining the DBSCAN algorithm with SC aims to overcome their respective limitations, at the cost of introducing additional hyperparameters. A recent work (Beer et al., 2023) introduces a density-connectivity distance to formalize density-based clustering without relying on heuristic design choices, providing a unified framework encompassing both DBSCAN and SC methods.

To the best of our knowledge, the SDP method introduced in the present contribution is the first to approach DP clustering from a graph-theoretical perspective.

2 BACKGROUND

Density-based clustering commonly relies on nonparametric density estimators (Watson & Leadbetter, 1963; Mack & Rosenblatt, 1979); we adopt the Point-Adaptive k-NN estimator (PAk) (Rodriguez et al., 2018), which provides both a logarithmic density estimate and an analytic uncertainty at each point without a fixed cutoff parameter; density estimators lacking a native uncertainty estimate can instead rely on bootstrapping.

In the DP clustering algorithm, local density estimates are combined with pairwise distances between data points via a decision graph (Rodriguez & Laio, 2014), whose interpretation requires visual inspection. The DPA method alleviates this heuristic by separating genuine peaks from finite-sample fluctuations through a free parameter Z (d'Errico et al., 2021), which controls the level of statistical confidence. The Component-wise Peak-Finding (CPF) (Tobin & Zhang, 2024) clustering method addresses the sensitivity of DP to noisy density estimates, at the cost of introducing additional tunable hyperparameters.

Normalized SC (NSC) has emerged as the advocated SC formulation (von Luxburg, 2007; von Luxburg et al., 2008). Each data point $\mathbf{x}^{(i)}$, for $i = 1, \ldots, N$, is associated with a vertex of an *undirected graph* whose edges are weighted by pairwise similarities, defining a symmetric weighted adjacency matrix $\mathbf{W}$. The normalized Laplacian matrix is obtained as $\mathbf{L} = \mathbf{I} - \mathbf{D}^{-1}\mathbf{W}$, where $\mathbf{D}$ is the diagonal degree matrix with nonzero entries $D_{i,i} = \sum_{j=1}^{N} W_{i,j}$ and $\mathbf{I}$ is the N-dimensional identity matrix; a gap after the k-th eigenvalue in the sorted spectrum of $\mathbf{L}$ indicates that the data embedding in $\mathbb{R}^k$, spanned by the first k eigenvectors (i.e., corresponding to the first k sorted eigenvalues) of $\mathbf{L}$, can be partitioned into k clusters (Jianbo & Malik, 2000). Despite its solid theoretical foundation, the performance of NSC is known to degrade on structures of varying size and density (Nadler & Galun, 2006).

The NSC method admits an interpretation as a *random walk* on the similarity graph (Aldous & Fill, 2002) with transition matrix $\mathbf{P} = \mathbf{D}^{-1}\mathbf{W}$ (Meilă & Shi, 2000; 2001). However, the symmetry of $\mathbf{W}$ is equivalent to the detailed balance equation, and therefore NSC is restricted to *reversible* Markov chains. Strongly connected to NSC (Weber et al., 2004), the PCCA+ algorithm (Röblitz & Weber, 2013) leverages the data embeddings spanned by first k eigenvectors of $\mathbf{P}$ lying within a $(k-1)$-dimensional simplex $\sigma_{k-1} \subset \mathbb{R}^k$ to provide robust fuzzy memberships. While PCCA+ does not assume reversibility, it is restricted to transition matrices with real dominant (i.e., largest in absolute value) eigenvalues (Frank et al., 2024).

Extensions of SC have been proposed to address directed graphs underlying non-reversible Markov chains. Extensions to directed graphs typically require a unique stationary distribution, limiting them to ergodic chains, enforceable via teleportation at the cost of altering the original dynamics (Zhou et al., 2005; Palmer & Zheng, 2021; Brin & Page, 1998), or introduce Hermitian Laplacians or directionality-aware vertex measures (Laenen & Sun, 2020; Sevi et al., 2025).

The Generalized PCCA (Fackeldey et al., 2018; Reuter et al., 2019) algorithm extends PCCA+ to coarse-grain non-reversible Markov chains, addressing the potential occurrence of dominant complex eigenvalues, by replacing the eigenvalue problem with a real-valued Schur decomposition of the transition matrix $\mathbf{P}$. A *real* representation of the *subspace* spanned by the first k Schur vectors is obtained by cPCCA+ (Frank et al., 2024) through the replacement of any complex conjugate pair among the first k eigenvectors of $\mathbf{P}$ with their real and imaginary components. Remarkably, the only free parameter is the number of clusters k. Its selection can be guided by a measure of

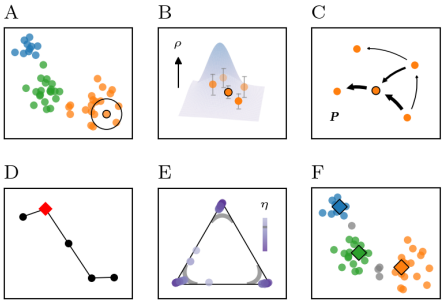


Figure 1: Overview of SDP clustering. (A) Colors by ground truth labels. (B) Density and uncertainty estimates, over the true ρ. (C) Probabilities than an arrow target is the nearest higher-density neighbor of its source; thicker arrows indicate larger $\mathbf{P}$ entries. (D) Optimal k (red diamond), via Timescale Separation Threshold (TST) score maximization. (E) Data simplex colored by normalized membership entropy η; lighter colors indicate higher η. The η contour (gray) marks the halo threshold. (F) Colors by SDP labels, halo points (gray) and density peaks (diamonds). Best viewed in color.

crispness (Röblitz & Weber, 2013), possibly paired with the minChi criterion (Weber et al., 2006) to preliminarily restrict to a range of nearly optimal numbers of clusters $[k_{\min}, k_{\max}]$.

3 Method

An overview of the SDP clustering algorithm is provided in Figure 1, based on $N = 50$ samples from a mixture of three isotropic bivariate Gaussian distributions. In panels A–C of Figure 1, a reference point and its four nearest neighbors are highlighted; in panel D, $[k_{\min}, k_{\max}] = [2, 6]$.

To theoretically ground the method, we assume the data points $\mathbf{x}_i \in \mathbb{R}^d$ are sampled from an unknown continuous probability distribution with density function ρ. The ground-truth population clusters are defined as the domains of attraction under the continuous density-ascent flow $\dot{\phi}_t(\mathbf{x}) = \nabla\rho(\phi_t(\mathbf{x}))$ where each trajectory converges to one of the density modes $m^{(c)}$.The goal of the proposed Spectral Density Peaks (SDP) method is to construct a discrete finite-state probabilistic ascent dynamics, via a transition matrix $\mathbf{P}$, that preserves these continuous domains of attraction. By preserving these basins, the domains of attraction become almost closed sets for the Markov chain, allowing cluster identification via coarse-graining. To this end, we assume that each point $\mathbf{x}_i$ is associated with a logarithmic density estimate $\log\hat{\rho}_i$, and an uncertainty estimate $\hat{\sigma}_i$. As noted in Section 2, this entails no loss of generality; therefore, SDP is *independent* of the density estimator. Let $\mathbf{R}$ be the pairwise distances matrix under any metric suitable for the considered dataset.

3.1 Density-Based Transition Matrix

We derive a *density-dominance* matrix $\mathbf{C}$ encoding the probability that one point has higher density than another. Its diagonal entries are explicitly set to zero, i.e., $C_{i,i} = 0$, for $i = 1, \ldots, N$. Being the logarithm strictly increasing, the event defining the off-diagonal elements of $\mathbf{C}$, i.e., point j having higher density than point i, reads $\log\rho_i - \log\rho_j < 0$, for $i, j = 1, \ldots, N$ and $i \neq j$.

For analytical tractability, a set of operational *assumptions* is introduced. First, we assume that (i) for each data point, the logarithmic density estimation process is affected by additive Gaussian noise with variance given by the corresponding squared uncertainty estimate, i.e., $\log\hat{\rho}_i = \log\rho_i + \varepsilon_i$, with $\varepsilon_i \sim \mathcal{N}(0, \hat{\sigma}_i^2)$. We further assume that (ii) for any pair of distinct data points, $\mathrm{Cov}(\varepsilon_i, \varepsilon_j) = 0$, so to neglect the covariances between estimation errors. This assumption can be relaxed by explicitly estimating $\mathrm{Cov}(\epsilon_i, \epsilon_j)$ via the Jaccard overlap of their k^*-neighborhoods, where k^* is the point-specific optimal number of neighbors used to estimate the local probability density.. As detailed in Appendix G, reinstating this correlation yields negligible changes to the resulting macroscopic topology and spectral gaps, supporting our independent error simplification as a robust, computationally scalable default.

For convenience, let us define $s_{i,j}^2 = \hat{\sigma}_i^2 + \hat{\sigma}_j^2$. We introduce, as auxiliary quantity, the rescaled difference

$$z_{i,j} \coloneqq s_{i,j}^{-1}(\log \hat{\rho}_i - \log \hat{\rho}_j). \tag{1}$$

Under the foregoing assumptions, the standardized difference between estimation errors follows a standard Gaussian distribution, i.e., $s_{i,j}^{-1}(\varepsilon_i - \varepsilon_j) \sim \mathcal{N}(0,1)$. Under assumption (i), the density-dominance event can be simplified to $\varepsilon_i - \varepsilon_j < -(\log \hat{\rho}_i - \log \hat{\rho}_j)$. Dividing both sides of the above inequality by $s_{i,j}$ and denoting by $\Phi(\cdot)$ the standard Gaussian cumulative distribution function, we approximate the probability of interest as

$$C_{i,j} \approx \Pr(s_{i,j}^{-1}(\varepsilon_i - \varepsilon_j) < -z_{i,j}) = \Phi(-z_{i,j}). \tag{2}$$

The off-diagonal elements of $\mathbf{C}$ satisfy the complement rule $C_{i,j} = 1 - C_{j,i}$, which gives the probability that point j does *not* have higher density than point i.

We derive the neighbor-dominance matrix $\mathbf{P} \in \mathbb{R}^{N \times N}$, which directly serves as the transition matrix for our discrete-time Markov chain. The off-diagonal entries quantify the probability that a point is the strictly nearest higher-density neighbor of another, while the diagonal entries (self-loops) capture the probability that a point is the highest-density point within its evaluated transitions. We further assume (iii) conditional independence of pairwise density-dominance events for a reference point i, the joint probabilities can be factorized into products of the corresponding marginals. The transition probabilities are thus defined as:

$$P_{i,j} \coloneqq \begin{cases} C_{i,j} \prod_{l=1}^{N} (1 - C_{i,l})^{\mathbb{1}_{\mathrm{R_{i,l}} < \mathrm{R_{i,j}}}}, & \text{if } i \neq j, \\ 1 - \sum_{l:l \neq i} P_{i,l}, & \text{if } i = j. \end{cases} \tag{3}$$

By construction, $\mathbf{P}_{i,j} \in (0,1)$ and $\sum_{l=1}^{N} \mathbf{P}_{i,l} = 1$ for all $i = 1, \ldots, N$, guaranteeing that $\mathbf{P}$ is a proper row-normalized transition matrix without requiring ad-hoc scaling. In this work, the distances between data points are treated as deterministic.

We define the one-step leakage $L(i) = \sum_{j:c(i) \neq c(j)} P_{ij}$, the probability of exiting a point's true cluster. Appendix A shows this leakage decays exponentially in the density of the screening set $S^{(i)}$ (same-cluster points closer to i than any external point), so domains of attraction act as almost-closed classes for $\mathbf{P}$, justifying metastable-state extraction via cPCCA+. Pseudocode is given in Appendix D.

We note that assumption (i) is *asymptotically* satisfied by density estimators derived by maximizing a likelihood function, either locally, e.g. PAk, or globally, e.g., BMTI (Carli et al., 2025). We acknowledge assumptions (ii) as a working simplification (for instance, violated for overlapping neighborhoods, e.g., in PAk), since estimating $\mathrm{Cov}(\varepsilon_i, \varepsilon_j)$ for $i, j = 1, \ldots, N$ would result in a higher computational burden; similar considerations apply to assumption (iii). Limitations are further discussed in Section 5.

3.2 Transition Matrix Coarse-Graining

The mode-seeking gradient ascent approach of MS clustering is generalized by the Quick Shift (QS) algorithm (Vedaldi & Soatto, 2008; Jiang, 2017) to operate directly in sample space via local density peak (rather than mean) shifts. The QS assignment criterion of a point to its nearest higher-density neighbor is analogous to that of DP clustering. The *intuition* behind the proposed SDP method is to further generalize this idea: treating density estimates as random variables induces a dynamics that is captured by the Markov chain we derive (via the transition matrix $\mathbf{P}$). As formally proven in Appendix A.1, for well-sampled clusters, the effective sampling diverges, which drives the probability of leaking outside a density-ascent basin to zero. Consequently, clustering then amounts to identifying these corresponding basins of attraction, which act as almost closed, metastable states for the Markov chain.

The transition matrix $\mathbf{P}$ is processed by the cPCCA+ algorithm, which requires only the number of clusters k as input, as previously observed, to obtain the $N \times k$ *membership* matrix χ (Röblitz & Weber, 2013). The entries $\chi_{i,m}$ represent the membership degree of point i to the cluster indexed by m, for $m = 1, \ldots, k$. These degrees satisfy both the *positivity* property, i.e., $\chi_{i,m} \geq 0$, for

$i = 1, \ldots, N$ and $m = 1, \ldots, k$, and the *partition of unity* property, i.e., $\sum_{m=1}^{k} \chi_{i,m} = 1$, for $i = 1, \ldots, N$.

From the resulting fuzzy (soft) clustering, a relaxation toward a *hard* $N \times k$ decomposition matrix $\tilde{\chi}$ is obtained by assigning each point to the cluster that yields the highest membership degree, namely

$$\tilde{\chi}_{i,c} = \begin{cases} 1, & \text{if } c = \operatorname{argmax}_{m=1,\ldots,k} \chi_{i,m}, \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

Consequently, points sharing the same *indicator* row in $\tilde{\chi}$ are assigned the same label in the final clustering.

While crispness maximization is the criterion typically used to select k for cPCCA+ (Röblitz & Weber, 2013), we find it unreliable on the SDP chain: its multimodal profile frequently peaks at a coarse, over-merged partition rather than the true metastable count. We instead select k via a non-reversible spectral-gap criterion, that we call *Timescale Separation Threshold* (TST) score, constructed from the transition matrix's own timescales, detailed in Appendix C together with a comparison against crispness on all five synthetic benchmarks.

3.3 Extended Clustering Outputs

In the SDP framework, cluster centers can be naturally defined in analogy with DPA (d'Errico et al., 2021). For each cluster $c = 1, \ldots, k$, the center is identified as the (uncertainty-penalized) *density peak*, i.e., the point within the cluster that maximizes the logarithmic density estimate while accounting for estimation uncertainty. Accordingly, the density peak index of cluster c is found as

$$i_c = \operatorname*{argmax}_{i=1,\ldots,N \,:\, \tilde{\chi}_{i,c}=1} (\log \hat{\rho}_i - \hat{\sigma}_i). \tag{5}$$

The fuzzy memberships can be further exploited to identify points assigned to a cluster with a low level of confidence (as formalized in the following), referred to as *halo* points, again in analogy with DPA (d'Errico et al., 2021). Given a point $i = 1, \ldots, N$, the Shannon entropy of its membership vector $\chi_{i,:}$, denoted by $H(\chi_{i,:})$, is bounded between 0 and $H_{\max} = \log_2(k)$. Therefore, for $i = 1, \ldots, N$, we consider the *normalized entropy*

$$\eta_i = \frac{H(\chi_{i,:})}{H_{\max}} \in [0, 1] \tag{6}$$

as a measure of confidence in the cluster assignment. In particular, $\eta_i = 0$ corresponds to a perfectly crisp (i.e., hard) membership vector, and $\eta_i = 1$ to a uniform membership vector, i.e., $\chi_{i,m} = 1/k$, for $m = 1, \ldots, k$.

The notion of low confidence in cluster assignment can be quantified by introducing a normalized entropy threshold $\eta_{\text{thr}} \in (0, 1)$, and labeling as halo all points for which η_i *exceeds* this value, i.e., $\eta_i \geq \eta_{\text{thr}}$, as they correspond to the least crisp membership vectors. By tuning η_{thr}, the user can control the size of a $(k+1)$-th halo group. The connection between this threshold and the simplex structure geometry is discussed in the following.

3.4 Algorithmic Scalability

As demonstrated in Appendix B, the off-diagonal entries of the neighbor-dominance matrix $\mathbf{P}$ exhibit stepwise exponential decay with distance. This mathematical property allows the transition matrix $\mathbf{P}$ to be populated and stored in a sparse format by restricting computations to a statistically grounded number of nearest neighbors. We recommend employing the PAk estimator (d'Errico et al., 2021), which adaptively determines an optimal local neighborhood size n_i for each data point. This inherently provides a principled cutoff for the sparse approximation of $\mathbf{P}$, ensuring high algorithmic scalability without sacrificing topological accuracy. A comprehensive breakdown of the time and memory complexities is provided in Appendix D.

4 Experiments

All experiments are run on Apple Silicon and x86-64 CPUs, without the need for specialized hardware such as GPUs. The SDP algorithm is implemented using the `DADApy` Python package

Table 1: Clustering performance on the synthetic datasets using $N = 15000$ points per resample. NMI is computed on ground-truth signal points only, while retaining all predicted labels assigned to those points. Results are reported as mean $\pm$ standard deviation over 50 resamples. No entropy-based halo post-processing is applied to SDP.

ALGORITHM	SPIR2	SPIR3	HORSE	AGG	CLUS8
SDP	0.964 ± 0.013	0.988 ± 0.053	0.992 ± 0.005	0.994 ± 0.017	0.992 ± 0.015
KM	0.257 ± 0.006	0.222 ± 0.002	0.701 ± 0.003	0.902 ± 0.004	0.927 ± 0.009
SC	0.529 ± 0.024	0.366 ± 0.014	0.717 ± 0.018	0.867 ± 0.009	0.967 ± 0.002
MS	0.320 ± 0.007	0.431 ± 0.007	0.614 ± 0.023	0.894 ± 0.023	0.976 ± 0.006
HDBSCAN	0.958 ± 0.009	0.906 ± 0.017	0.898 ± 0.028	0.946 ± 0.019	0.920 ± 0.010
DPA	0.891 ± 0.035	0.975 ± 0.058	0.989 ± 0.007	0.993 ± 0.008	0.991 ± 0.007

(Glielmo et al., 2022) for computing distances, logarithmic density estimates, and the corresponding uncertainty estimates. For the cPCCA+ coarse-graining algorithm, the original `cPCCA` MATLAB implementation (sroeblitz, 2022) provided by the authors is adopted.

4.1 SYNTHETIC DATASETS WITH NOISE

We consider five synthetic datasets of dimension $d = 2$: SPIR2 ($k = 2$), SPIR3 ($k = 3$), HORSE ($k = 3$), AGG ($k = 7$), and CLUS8 ($k = 8$) (d'Errico et al., 2021). Ground-truth labels are available for all datasets and include a dedicated noise class. SDP is compared with KM, SC, MS, HDBSCAN, and DPA.

For each dataset, we generate 50 independent resamples of $N = 15000$ points using the resampling procedure implemented in our benchmark pipeline. All methods use Euclidean distance. SDP uses PAk density estimation with its intrinsic k^*-neighborhood sparsification; no normalized-entropy threshold or halo post-processing is used in these experiments. For the competing methods, the hyperparameter configurations reported in Appendix E are evaluated and the configuration maximizing NMI on the ground-truth signal points is retained for each resample. This oracle selection provides a best-case comparison for the competitors, whereas the SDP value of k is selected without using the ground-truth labels.

Because the present SDP evaluation does not include entropy-based halo identification, the primary metric is the NMI computed after excluding only the points labeled as noise in the ground truth. Importantly, no points are removed according to the predicted assignment: if a method assigns a ground-truth signal point to a noise label, that prediction remains in the NMI calculation. This definition evaluates recovery of the signal cluster structure without confounding the comparison with method-specific noise or halo post-processing.

Table 1 reports the mean NMI and standard deviation over the 50 resamples. SDP attains the highest mean NMI on all five datasets, with values ranging from 0.964 on SPIR2 to 0.994 on AGG. DPA is very close to SDP on HORSE, AGG, and CLUS8, while HDBSCAN is close on SPIR2. Across resamples, SDP recovers the ground-truth number of clusters in 100% of SPIR2 runs, 96% of SPIR3 runs, 100% of HORSE runs, 98% of AGG runs, and 92% of CLUS8 runs; the complete cluster-count distributions and competitor settings are reported in Appendix E.

We additionally report the computational cost of the benchmark in Table 2. Runtime is measured as the total wall-clock time required for each resample, including the complete hyperparameter search. SDP requires approximately 16–39 seconds per resample across the five datasets. While it is more computationally demanding than KM, HDBSCAN, and DPA, it remains consistently faster than SC and MS under the evaluated search grids.

Unsupervised Application In order to illustrate the performance of the proposed SDP clustering method in a fully *unsupervised* setting, we run the algorithm on a representative resample from the 50 considered in the above analysis, for the AGG dataset.

Table 2: Median wall-clock time (seconds) across the 50 resamples of each synthetic dataset. Times include the complete hyperparameter search.

ALGORITHM	SPIR2	SPIR3	HORSE	AGG	CLUS8
SDP	16.1	16.8	31.1	39.2	21.8
KM	1.5	1.6	1.3	1.2	1.1
SC	39.5	55.8	51.2	55.6	44.3
MS	102.3	64.7	73.7	51.5	33.0
HDBSCAN	5.5	5.4	5.5	5.4	5.5
DPA	3.1	2.9	3.1	3.7	3.0

The optimal number of clusters is selected within $[2, 10]$ via TST score maximization; in practical applications, it suffices to specify a suitable upper bound to expected number of clusters $k_{\max}$, as the lower bound can be assumed a priori to be $k_{\min} = 2$.

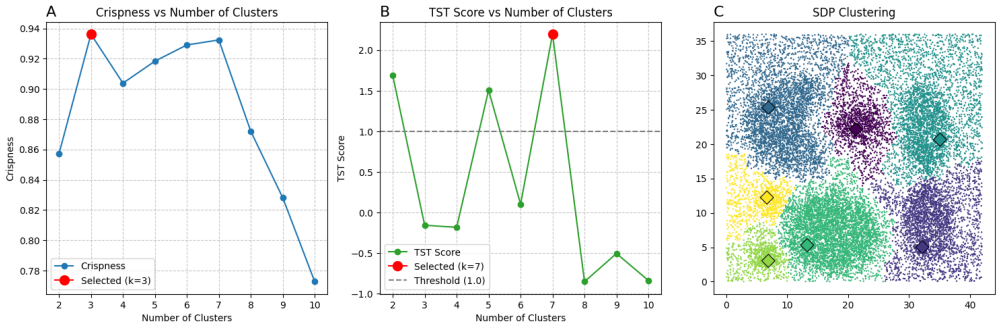


Figure 2: Optimal number of clusters (red diamonds) via crispness and TST score maximization, and the corresponding SDP clustering results and density peaks (diamonds). (A) Crispness profile for AGG. (B) TST profile for AGG. (C) Clustering for AGG using TST result.

The results obtained by SDP are illustrated in Figure 2. On one hand, the crispness maximization criterion (A) shows two peaks making the selection of the number of clusters difficult. On the other hand, TST score maximization (B) is able to correctly recover the true number of clusters. The final clusters are shown in panel (C) of the figure.

4.2 MNIST DATABASE OF HANDWRITTEN DIGITS

The MNIST (training) database (LeCun et al., 1998) is a collection of 60000 grayscale images of handwritten digits, each of size 28×28. Since each digit image is labeled from 0 to 9, MNIST has been widely adopted as a *classification* benchmark; however, a cluster analysis of the MNIST database can provide insights into the *geometric* structure of the data, rather than its semantic organization.

To construct matrix $\mathbf{R}$, image dissimilarities are computed using the one-sided *tangent distance* (Simard et al., 1998), a measure which is not a proper metric, but is designed to increase invariance to small transformations such as translations, rotations, and scaling. We employ the PAk estimator, allowing $\mathbf{P}$ and its auxiliary matrices to be handled in sparse format. The optimal number of clusters in the range $[2, 20]$ is found to be $k = 4$ via TST score maximization. We ultimately select the second peak at $k = 18$, as the $k = 4$ configuration yields a highly imbalanced clustering distribution; we refer the reader to Appendix F for the corresponding TST score plot and a richer explanation of this decision.

Figure 3 visualizes the contingency matrix comparing ground-truth MNIST digit classes to the discovered clusters. The matrix reveals that the clustering algorithm prioritizes structural and morphological features over strict semantic labels. Instead of forcing a one-to-one mapping, the model intelligently oversegments certain digits to isolate distinct handwriting styles, such as separating the

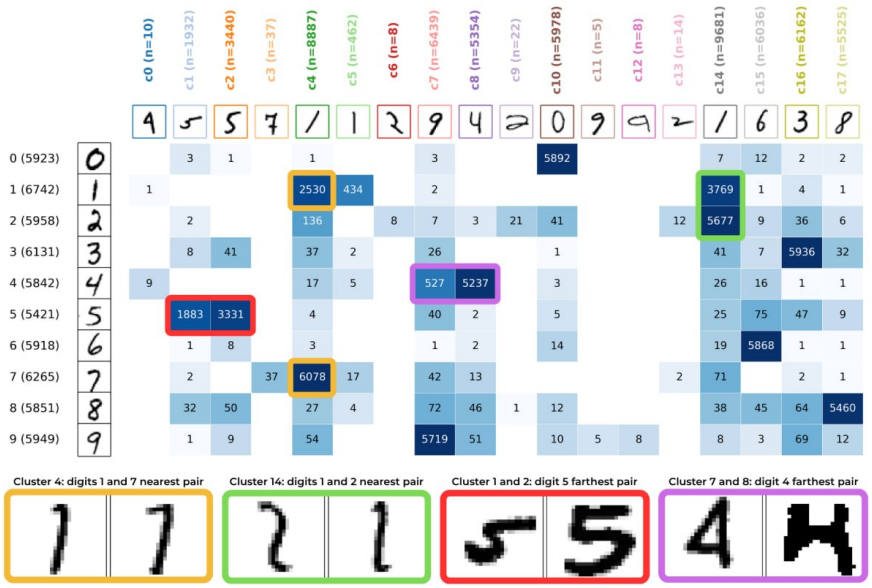


Figure 3: MNIST clustering contingency matrix. Rows denote ground-truth digits and columns represent discovered clusters, topped with cluster representatives. The bottom panels illustrate captured inter-class similarities (e.g., nearest pairs in shared clusters) and intra-class stylistic variances (e.g., farthest pairs split across distinct clusters).

varied stroke patterns of fours and fives. Furthermore, the merging of different digits with highly ambiguous topologies demonstrates that the learned feature space accurately reflects actual visual variance rather than arbitrary predefined classes. This sensitivity to structural nuances confirms the algorithm's robust capacity to map complex data manifolds without forced uniformity. Further detailed explanations of these boundary dynamics are provided in the Appendix F.

4.3 Molecular Dynamics of Peptide CLN025

CLN025 Honda et al. (2008), a stabilized 10-residue variant of the Chignolin peptide Honda et al. (2004), folds into a β-hairpin; above its critical temperature its unfolded state is known to include a distinct denatured collapsed conformation rather than a single unfolded basin (Davis et al., 2012; McKiernan et al., 2017).[1]

In the biomolecular trajectory analysis of CLN025, the density peaks identified by SDP can be interpreted as the conformational states *most frequently* visited over time, expected to correspond to the folded and denatured collapsed ones. We consider a dataset of 41580 frames from a replica-exchange molecular dynamics (REMD) simulation (400 ns, 340 K replica, $\Delta t = 2$ fs) (Carli & Laio, 2021). A high-dimensional representation of the trajectory is given by the set of all pairwise distances between the heavy atoms of CLN025 (93 out of 166), yielding 4278 features; a more compact representation consists of the set of all the 32 dihedral angles of the peptide (Cossio et al., 2011).

The conformational dynamics time series is first decimated (one frame every 10) to mitigate serial correlations, and the initial 1000 configurations are discarded as burn-in to exclude the initial equilibration phase. As a result, we are left with a dataset of $N = 3158$ frames. We adopt the dihedral angle representation of the trajectory, and consequently employ a 2π-periodic Euclidean distance to construct matrix $\mathbf{R}$. We then apply the PAk estimator, together with the relative sparse representation of $\mathbf{P}$.

[1] At the 343 K melting temperature the folded and unfolded populations are equal; see (McKiernan et al., 2017; Rodriguez et al., 2011) for the full folding-mechanism discussion.

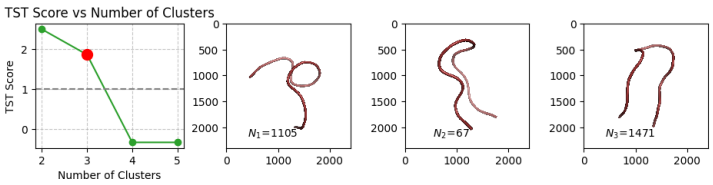


Figure 4: Cluster analysis of an REMD simulation of the peptide CLN025 via SDP. (A) Optimal number of clusters (red diamond) via TST score maximization. (B) - (D) Density Peaks configuration with cluster population.

The corresponding results are reported in Figure 4. From panel A we observe that TST score identifies $k = 3$ as the optimum within $[2, 5]$. Panels B–D report the CLN025 configurations of density peaks, together with the relative cluster population. As in Glielmo et al. (2022), we observe a highly consistent cluster structure defined by the principal macrostates of the molecule. We recognize the first density peak configuration as corresponding to the twisted loop structure; the third cluster corresponds to the folded β-hairpin structure of CLN025; lastly, the second, minimally populated cluster ($N_2 = 67$) captures unstructured or intermediate configurations. This observation is consistent with turn formation being the *rate-limiting* step in the peptide folding process (McKiernan et al., 2017).

A direct comparison with competitor clustering algorithms is not straightforward, as there is no well established ground truth. An analogous analysis of CLN025, performed using the DPA algorithm (Glielmo et al., 2022), yields results consistent with SDP. The present experiment provides evidence of the broad applicability of SDP, supporting its adoption in the analysis of biomolecular trajectories.

5 Conclusion

In this work, we introduced the SDP clustering algorithm, a novel approach to density peaks clustering that identifies clusters by coarse-graining a local density-based Markov chain. Its versatility and minimal requirements for user intervention, together with consistent state-of-the-art performance, highlight its potential for broad applicability across a wide range of clustering tasks. We conclude by discussing the method caveats, outlining possible remedies, and suggesting directions for future research.

Limitations The assumptions underlying SDP are presented in Section 3. Assumption (i), i.e., the Gaussianity of the density estimates, is required to compute the density-dominance probabilities $C_{i,j}$ as in Equation 2, but they could also be evaluated empirically via bootstrap resampling, re-estimating the logarithmic densities for each resample, at a higher computational cost.

Assumptions (ii) and (iii) are strong approximations, since density estimates based on Nearest Neighbors (like PAk) present high correlations. However, in Appendix G, we show that, at least for the toy datasets, including these correlations does not significantly alter the dominance matrix nor the results of the clustering partition, but it does increase the computational cost.

Future Directions While the SDP method explicitly accounts for the uncertainty in density estimates, it assumes *exact* knowledge of the pairwise distances between data points, i.e., of matrix $\mathbf{R}$. A natural extension of the present work is to apply a *softmax* transformation to the distances, in order to obtain probabilistic (soft) neighbor assignments, and, in turn, adjusting Equation 3 accordingly.

AI USE STATEMENT

In this work, we used generative AI tools for language editing purposes, specifically for grammar, syntax corrections during the writing of the manuscript and minimal code generation. We have not used generative AI tools for research ideation, experimental design, data analysis, generation of results, or any other aspect of the scientific content of this work. We have reviewed all AI-assisted work. We manually verified all AI-suggested text modifications to ensure scientific meaning was preserved, and systematically tested all generated code snippets for correctness. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We have included essential implementation snippets in the supplementary materials. Upon acceptance, the complete source code, data, and configuration files will be released in a public GitHub repository.

REFERENCES

D. Aldous and J. A. Fill. Reversible Markov chains and random walks on graphs, 2002. Unfinished monograph, recompiled 2014, available at `https://www.stat.berkeley.edu/users/aldous/RWG/Book_Ralph/book.html`.

A. Beer, A. Draganov, E. Hohma, P. Jahn, C. M. M. Frey, and I. Assent. Connecting the dots – density-connectivity distance unifies DBSCAN, k-center and spectral clustering. In A. K. Singh, Y. Sun, L. Akoglu, D. Gunopulos, X. Yan, R. Kumar, F. Ozcan, and J. Ye (eds.), *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD 2023)*, pp. 80–92, Long Beach, CA, USA, 2023. ACM.

S. Brin and L. Page. The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems*, 30(1):107–117, 1998.

Ricardo JGB Campello, Davoud Moulavi, and Jörg Sander. Density-based clustering based on hierarchical density estimates. In *Pacific-Asia conference on knowledge discovery and data mining*, pp. 160–172. Springer, 2013.

M. Carli and A. Laio. Statistically unbiased free energy estimates from biased simulations. *Molecular Physics*, 119(19–20):e1899323, 2021.

M. Carli, A. Rodriguez, A. Laio, and A. Glielmo. Density estimation via binless multidimensional integration. *Machine Learning: Science and Technology*, 6(2):025052, 2025.

D. Comaniciu and P. Meer. Mean shift: a robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619, 2002.

P. Cossio, A. Laio, and F. Pietrucci. Which similarity measure is better for analyzing protein structures in a molecular dynamics trajectory? *Physical Chemistry Chemical Physics*, 13(22):10421–10425, 2011.

K. Craig, N. Garcia Trillos, and D. Slepčev. Clustering dynamics on graphs: From spectral clustering to mean shift through Fokker–Planck interpolation. In N. Bellomo, J. A. Carrillo, and E. Tadmor (eds.), *Active Particles, Volume 3: Advances in Theory, Models, and Applications*, pp. 105–151. Springer International Publishing, 2022.

J. N. Darroch and E. Seneta. On quasi-stationary distributions in absorbing discrete-time finite Markov chains. *Journal of Applied Probability*, 2:88–100, 1965.

C. M. Davis, S. Xiao, D. P Raleigh, and R. B. Dyer. Raising the speed limit for β-hairpin formation. *Journal of the American Chemical Society*, 134(35):14476–14482, 2012.

I. S. Dhillon, Y. Guan, and B. Kulis. Kernel k-means: spectral clustering and normalized cuts. In W. Kim, R. Kohavi, J. Gehrke, and W. DuMouchel (eds.), *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2004)*, pp. 551–556, Seattle, WA, USA, 2004. ACM.

M. d'Errico, E. Facco, A. Laio, and A. Rodriguez. Automatic topography of high-dimensional data sets by non-parametric density peak clustering. *Information Sciences*, 560:476–492, 2021.

M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In E. Simoudis, J. Han, and U. M. Fayyad (eds.), *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD 1996)*, pp. 226–231, Portland, OR, USA, 1996. AAAI Press.

K. Fackeldey, A. Sikorski, and M. Weber. Spectral clustering for non-reversible Markov chains. *Computational and Applied Mathematics*, 37(5):6376–6391, 2018.

A.-S. Frank, A. Sikorski, and S. Röblitz. Spectral clustering of Markov chain transition matrices with complex eigenvalues. *Journal of Computational and Applied Mathematics*, 444(2):115791, 2024.

A. Glielmo, I. Macocco, D. Doimo, M. Carli, C. Zeni, R. Wild, M d'Errico, A. Rodriguez, and A. Laio. DADApy: Distance-based analysis of data-manifolds in Python. *Patterns*, 3(10):100589, 2022.

S. Hess, W. Duivesteijn, P. Honysz, and K. Morik. The spectACl of nonconvex clustering: A spectral approach to density-based clustering. In P. Van Hentenryck and Z.-H. Zhou (eds.), *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence (AAAI 2019)*, pp. 3788–3795, Honolulu, HI, USA, 2019. AAAI Press.

S. Honda, K. Yamasaki, Y. Sawada, and H. Morii. 10 residue folded peptide designed by segment statistics. *Structure*, 12(8):1507–1518, 2004.

S. Honda, T. Akiba, Y. S. Kato, Y. Sawada, M. Sekijima, M. Ishimura, A. Ooishi, H. Watanabe, T. Odahara, and K. Harata. Crystal structure of a ten-amino acid protein. *Journal of the American Chemical Society*, 11(46):15327–15331, 2008.

Roger A Horn and Charles R Johnson. *Matrix analysis*. Cambridge university press, 2012.

S. Jianbo and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

H. Jiang. On the consistency of quick shift. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Proceedings of the 31st International Conference on Advances in Neural Information Processing Systems (NIPS 2017)*, Long Beach, CA, USA, 2017. Curran Associates, Inc.

S. Laenen and H. Sun. Higher-order spectral clustering of directed graphs. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (eds.), *Proceedings of the 33rd International Conference on Advances in Neural Information Processing Systems (NeurIPS 2020)*, pp. 941–951, Virtual-only, 2020. Curran Associates, Inc.

Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

S. Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2): 129–137, 1982.

Y. P. Mack and M. Rosenblatt. Multivariate k-nearest neighbor density estimates. *Journal of Multivariate Analysis*, 9(1):1–15, 1979.

K. A. McKiernan, B. E. Husic, and V. S. Pande. Modeling the mechanism of CLN025 beta-hairpin formation. *The Journal of Chemical Physics*, 147(10):104107, 2017.

M. Meilă and J. Shi. Learning segmentation by random walks. In T. Leen, T. Dietterich, and V. Tresp (eds.), *Proceedings of the 13th International Conference on Advances in Neural Information Processing Systems (NIPS 2000)*, Denver, CO, USA, 2000. MIT Press.

M. Meilă and J. Shi. A random walks view of spectral segmentation. In T. S. Richardson and T. S. Jaakkola (eds.), *Proceedings of the Eighth International Workshop on Artificial Intelligence and Statistics (AISTATS 2001)*, pp. 203–208, Key West, FL, USA, 2001. PMLR.

B. Nadler and M. Galun. Fundamental limitations of spectral clustering. In B. Schölkopf, J. Platt, and T. Hoffman (eds.), *Proceedings of the 19th International Conference on Advances in Neural Information Processing Systems (NIPS 2006)*, Vancouver, BC, Canada, 2006. MIT Press.

U. Ozertem, D. Erdogmus, and R. Jenssen. Mean shift spectral clustering. *Pattern Recognition*, 41 (6):1924–1938, 2008.

W. R. Palmer and T. Zheng. Spectral clustering for directed networks. In R. M. Benito, C. Cherifi, H. Cherifi, E. Moro, L. M. Rocha, and M. Sales-Pardo (eds.), *Complex Networks & Their Applications IX*, pp. 87–99, Cham, Switzerland, 2021. Springer International Publishing.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

B. Reuter, K. Fackeldey, and M. Weber. Generalized Markov modeling of nonreversible molecular kinetics. *The Journal of Chemical Physics*, 150(17):174103, 2019.

S. Röblitz and M. Weber. Fuzzy spectral clustering by PCCA+: application to Markov state models and data classification. *Advances in Data Analysis and Classification*, 7(2):147–179, 2013.

A. Rodriguez and A. Laio. Clustering by fast search and find of density peaks. *Science*, 344(6191): 1492–1496, 2014.

A. Rodriguez, M. d'Errico, E. Facco, and A. Laio. Computing the free energy without collective variables. *Journal of Chemical Theory and Computation*, 14(3):1206–1215, 2018.

Alex Rodriguez, Pol Mokoema, Francesc Corcho, Khrisna Bisetty, and Juan J. Perez. Computational study of the free energy landscape of the miniprotein cln025 in explicit and implicit solvent. *The Journal of Physical Chemistry B*, 115(6):1440–1449, 2011. doi: 10.1021/jp106475c. URL https://doi.org/10.1021/jp106475c. PMID: 21254763.

H. Sevi, G. Debaussart-Joniec, M. Hacini, M. Jonckheere, and A. Kalogeratos. Generalized Dirichlet energy and graph Laplacians for clustering directed and undirected graphs, 2025. URL https://arxiv.org/abs/2203.03221. arXiv preprint.

P. Y. Simard, Y. LeCun, J. S. Denker, and B. Victorri. Transformation invariance in pattern recognition — Tangent distance and tangent propagation. In G. B. Orr and K.-R. Müller (eds.), *Neural Networks: Tricks of the Trade*, pp. 239–274. Springer Berlin Heidelberg, 1998.

sroeblitz. sroeblitz/cPCCA: v1.0.0, 2022. URL https://doi.org/10.5281/zenodo.6782413. Zenodo repository.

J. Tobin and M. Zhang. A theoretical analysis of density peaks clustering and the component-wise peak-finding algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(2): 1109–1120, 2024.

A. Vedaldi and S. Soatto. Quick shift and kernel methods for mode seeking. In D. Forsyth, P. Torr, and A. Zisserman (eds.), *Computer Vision – ECCV 2008*, pp. 705–718, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.

U. von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007.

U. von Luxburg, M. Belkin, and O. Bousquet. Consistency of spectral clustering. *The Annals of Statistics*, 36(2):555–586, 2008.

G. S. Watson and M. R. Leadbetter. On the estimation of the probability density, [i]. *The Annals of Mathematical Statistics*, 34(2):480–491, 1963.

M. Weber, W. Rungsarityotin, and A. Schliep. Perron cluster analysis and its connection to graph partitioning for noisy data. Technical Report 04-39, ZIB, Takustr. 7, 14195 Berlin, 2004.

M. Weber, W. Rungsarityotin, and A. Schliep. An indicator for the number of clusters: Using a linear map to simplex structure. In M. Spiliopoulou, R. Kruse, C. Borgelt, A. Nürnberger, and W. Gaul (eds.), *From Data and Information Analysis to Knowledge Engineering*, pp. 103–110, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

D. Zhou, J. Huang, and B. Schölkopf. Learning from labeled and unlabeled data on a directed graph. In L. De Raedt and S. Wrobel (eds.), *Proceedings of the 22nd International Conference on Machine Learning (ICML 2005)*, pp. 1036–1043, Bonn, Germany, 2005. ACM.

A Leakage Bounds and Basin Preservation

Let $c : \{1, \dots, N\} \to \{1, \dots, k\}$ be the function returning the sample-cluster label of a data point. We formalize the capacity of the transition matrix P to preserve these clusters

Definition A.1 (Leakage Probability). The one-step leakage probability of a data point is defined as the probability that a single transition from that point leaves its cluster

$$L^{(i)} := \sum_{j:c(j)\neq c(i)} P_{i,j} \tag{7}$$

Definition A.2 (Screening Set). For $i = 1, \dots, N$, define the screening set $S^{(i)}$ as the set of indices of same-cluster points closer to i than any outside-cluster point

$$S^{(i)} := \left\{ l \in \{1, \dots, N\} : l \neq i, c(l) = c(i), R_{i,l} < \min_{j:c(j)\neq c(i)} R_{i,j} \right\} \tag{8}$$

Proposition A.1 (Leakage Bound). For $i = 1, \dots, N$, the one-step leakage probability is bounded as

$$L^{(i)} \leq \exp\left(- \sum_{l \in S^{(i)}} C_{i,l} \right) \tag{9}$$

Proof. For $i, j = 1, \dots, N$ with $c(j) \neq c(i)$, let $\bar{S}_j^{(i)}$ be the set of indices, excluding $S^{(i)}$, of points closer to i than j, defined as:

$$\bar{S}_j^{(i)} := \left\{ h \in \{1, \dots, N\} : h \neq i, h \notin S^{(i)}, R_{i,h} < R_{i,j} \right\} \tag{10}$$

Then, we rewrite the off-diagonal transition probabilities isolating the contribution of the screening set as:

$$P_{i,j} = \prod_{l \in S^{(i)}} (1 - C_{i,l}) \left(C_{i,j} \prod_{h \in \bar{S}_j^{(i)}} (1 - C_{i,h}) \right) \tag{11}$$

By bounding the rightmost term by 1, summing over all external points j, and applying the inequality $1 - x \leq \exp(-x)$ for $x \in (0, 1)$, the proposition directly follows.

A.1 Asymptotic Guarantees for Well-Sampled Clusters

To establish when the leakage probability vanishes, we introduce the concept of effective sampling, which quantifies the number of same-cluster points that statistically shield a point from leaving its cluster.

Definition A.3 (Effective Sampling). For a point i, the effective sampling is defined as

$$\mathcal{S}^{(i)} := \sum_{l \in S^{(i)}} \Phi\left(\frac{\log \hat{\rho}^{(l)} - \log \hat{\rho}^{(i)}}{\hat{s}^{(i,l)}} \right) \tag{12}$$

Definition A.4 (Well-Sampled Cluster). A cluster is well-sampled around i if there exists a constant $B < +\infty$ such that

$$\left| \left\{ l \in S^{(i)} : \frac{\log \hat{\rho}^{(i)} - \log \hat{\rho}^{(l)}}{\hat{s}^{(i,l)}} \leq B \right\} \right| \to +\infty \tag{13}$$

Equivalently, before reaching any point outside the cluster, point i sees an increasing number of same-cluster points whose estimated log-density is not arbitrarily lower than its own, when measured in units of the uncertainty $\hat{s}^{(i,l)}$

Theorem A.1 (Effective Sampling Divergence). If the cluster of i is well-sampled around i, then $\mathcal{S}^{(i)} \to +\infty$

Proof. Let $A_i := \left\{ l \in S^{(i)} : \log \hat{\rho}^{(i)} - \log \hat{\rho}^{(l)} \leq B\hat{s}^{(i,l)} \right\}$. By Definition A.4, $|A_i| \to +\infty$

$$\frac{\log \hat{\rho}^{(l)} - \log \hat{\rho}^{(i)}}{\hat{s}^{(i,l)}} \geq -B \tag{14}$$

Since the standard Gaussian cumulative distribution function Φ is strictly increasing, $\Phi\left(\frac{\log \hat{\rho}^{(l)} - \log \hat{\rho}^{(i)}}{\hat{s}^{(i,l)}}\right) \geq \Phi(-B) > 0$

$$\mathcal{S}^{(i)} \geq \sum_{l \in A_i} \Phi\left(\frac{\log \hat{\rho}^{(l)} - \log \hat{\rho}^{(i)}}{\hat{s}^{(i,l)}}\right) \geq |A_i|\Phi(-B) \tag{15}$$

Since $|A_i| \to +\infty$ and $\Phi(-B) > 0$, it follows that $\mathcal{S}^{(i)} \to +\infty$.

Corollary A.1 (Basin Preservation). By recognizing that $\sum_{l \in S^{(i)}} C_{i,l} = \mathcal{S}^{(i)}$, we substitute the effective sampling into the exponential bound from Proposition A.1. As $\mathcal{S}^{(i)} \to +\infty$, the one-step leakage probability $L^{(i)} \to 0$.

B Exponential Decay of the Transition Matrix Entries

We recall the *convexity* property of the exponential function, namely $1 - x \leq e^{-x}$ for all $x \in \mathbb{R}$, with equality holding at $x = 0$. Therefore, each $(1 - C_{i,l})$ factor in the product of Equation 3 is *upper bounded* by $\exp(-C_{i,l})$, with the equality attained for $l = i$, as $C_{i,i} = 0$ for $i = 1, \ldots, N$. By construction, the off-diagonal entries of the transition matrix $\mathbf{P}$ describe transitions from point i to distinct points $j \neq i$, whereas the diagonal entry is defined as

$$P_{ii} = 1 - \sum_{j \neq i} P_{ij},$$

and represents the self-transition probability. Therefore, the following analysis concerns only the off-diagonal entries P_{ij}, with $i \neq j$; for the off-diagonal entries we obtain that

$$P_{i,j} = C_{i,j} \prod_{l=1}^{N} (1 - C_{i,l})^{\mathbb{1}_{\mathrm{R_{i,l}} < \mathrm{R_{i,j}}}} \leq C_{i,j} \exp\left(- \sum_{l=1}^{N} C_{i,l} \mathbb{1}_{\mathrm{R_{i,l}} < \mathrm{R_{i,j}}} \right),$$

where the equality holds if the set $\mathbb{L}(R_{i,j}) = \{l = 1, \ldots, N \colon R_{i,l} < R_{i,j}\}$ contains no indices other than $l = i$, i.e., when j is the nearest neighbor of i. Otherwise, there exists at least one $l \neq i$ such that $R_{i,l} < R_{i,j}$, and since $C_{i,l} > 0$ for $l \neq i$, the inequality holds strict. The exponential argument

$$- \sum_{l=1}^{N} C_{i,l} \mathbb{1}_{\mathrm{R_{i,l}} < \mathrm{R_{i,j}}} = - \sum_{l \in \mathbb{L}(R_{i,j})} C_{i,l}$$

sums the contributions of all points strictly closer to point i than point j. As $R_{i,j}$ increases, the set of indices $\mathbb{L}(R_{i,j})$ enlarges, including additional $C_{i,l} > 0$ terms. Consequently, the exponential function is a monotonically decreasing, stepwise function of $R_{i,j}$, and the off-diagonal $P_{i,j}$ entries decay stepwise exponentially as $R_{i,j}$ increases.

C Appendix: choosing the number of clusters by a per–cluster spectral score: Timescale Separation Threshold

The clustering above returns, for any requested number of clusters m, a partition $\{I_a\}_{a=1}^{m}$; we propose a criterion to select m. We score each candidate partition by a metastability gap read off the SDP transition matrix, without diagonalising it. Because the SDP chain is a density *ascent* it is strongly non–reversible, so its eigenvalues are complex in general and the usual "count the eigenvalues near 1" spectral gap is not directly well posed; we sidestep this by attaching two *real* timescales to each cluster.

Per–cluster timescales Fix a small diffusion mixing $\lambda \in (0,1)$ (in this work we used always the default value $\lambda = 0.01$) and let $P^{\text{eff}} = (1-\lambda)P + \lambda D$, the SDP chain P regularised by the $k^\star$–NN diffusion D (both row–stochastic). For a cluster I_a let $Q_a = P^{\text{eff}}[I_a, I_a]$ be the block obtained by *killing* the walk on exit. By the Markov property and induction, $\big[Q_a^t\big]_{ij} = \mathbb{P}_i(X_t = j,\ \tau > t)$ with τ the first exit time, so $Q_a^t\mathbf{1}$ is the vector of survival probabilities inside I_a; and $(Q_a\mathbf{1})_i = 1 - \sum_{j\notin I_a} P^{\text{eff}}_{ij} \le 1$, i.e. Q_a is sub–stochastic, strictly so on cells with an edge leaving I_a. For a connected cluster Q_a is nonnegative and irreducible, and its strictly positive diagonal ($P^{\text{eff}}_{ii} \ge (1-\lambda)d_i > 0$, the retained deficit) makes it *primitive*; the Perron–Frobenius theorem for primitive matrices (Horn & Johnson, 2012, Thm. 8.5.1) then gives a spectral radius $\lambda_1(Q_a)$ that is real, simple, and strictly dominant, with positive right/left eigenvectors $w, \hat\nu > 0$. Normalising $\hat\nu^\top\mathbf{1} = 1$,

$$\lambda_1 = \lambda_1\,\hat\nu^\top\mathbf{1} = \hat\nu^\top(Q_a\mathbf{1}) = \sum_i \hat\nu_i (Q_a\mathbf{1})_i < 1, \tag{16}$$

strictly, because $\hat\nu > 0$ and some row of Q_a leaks. The left eigenvector $\hat\nu$ is the quasi–stationary distribution Darroch & Seneta (1965): starting from it, survival is exactly geometric, $\mathbb{P}_{\hat\nu}(\tau > t) = \hat\nu^\top Q_a^t\mathbf{1} = \lambda_1^t$, so the per–step escape probability is $\kappa_a = 1-\lambda_1(Q_a)$ and the mean exit time is $1/\kappa_a$. Conditioned on survival, an arbitrary interior law relaxes to $\hat\nu$ with error envelope $\big(|\lambda_2(Q_a)|/\lambda_1\big)^t$, where λ_2 is the sub–dominant eigenvalue (possibly complex); the relaxation rate is thus set by the modulus $|\lambda_2|$. We define the escape and internal *log–scales*

$$\tau_{\text{out}}(a) := -\log\big(1-\lambda_1(Q_a)\big)\ \ (=\log\ \text{mean exit time}), \qquad \tau_{\text{int}}(a) := -\log\big(1-|\lambda_2(Q_a)|\big), \tag{17}$$

both real by construction: a rotational (complex) internal mode enters τ_{int} only through $|\lambda_2|$ and never inflates the cluster count. In practice the two eigenvalues nearest 1 come from one shift–invert Arnoldi solve on the small block Q_a.

Score and selection rule. A cluster is well posed when it is left slowly and mixes internally quickly, $\tau_{\text{out}} \gg \tau_{\text{int}}$. The two failure modes are read directly from (17). If a cluster is a *merged pair* of genuine basins, its killed block has a second long–lived mode, so $|\lambda_2| \to \lambda_1$ and $\tau_{\text{int}} \to \tau_{\text{out}}$: the gap $\tau_{\text{out}} - \tau_{\text{int}}$ collapses to ≈ 0. If instead it is a *fragment* cut through a dense region, the walk escapes almost immediately, λ_1 drops well below 1, τ_{out} is small, and the gap can go negative. A partition into the correct number of clusters is therefore the finest one in which *every* cluster still has a clear positive gap. Scoring a partition by its worst cluster,

$$S(m) := \min_a \tau_{\text{out}}(a) - \max_a \tau_{\text{int}}(a), \qquad \boxed{\hat m = \max\{\, m :\ S(m) > \theta \,\}} \quad (\theta = 1 \text{ by default}), \tag{18}$$

i.e. we take the finest partition whose least–metastable cluster still shows at least a one–e–fold ($\theta = 1$ nat) separation between escape and internal mixing. Larger θ is more conservative; $\theta = 1$ is used throughout.

Note that $S(m)$ can be understood as a computable surrogate for the classical separation between the m low inter–cluster modes and the internal cluster modes. However, theoretical analysis shows that this identity is not strict and can be violated in some conditions. We therefore use (18) as an *empirical* selection criterion, validated by the experiments rather than by a spectral–gap identity.

C.1 Comparison with crispness on synthetic benchmarks

Empirical evaluations across multiple benchmarks demonstrate the distinct superiority of this localized spectral gap metric over traditional crispness maximization. As highlighted in Table 3, in datasets such as AGG and CLUS8, crispness maximization decisively fails by collapsing the data into a coarse three-cluster partition, whereas the spectral gap criterion successfully and reliably recovers the ground-truth cluster counts. Although the spectral gap functions as a rigorous, worst-cluster metric that can exhibit sensitivity to noise along exceptionally faint density boundaries, it consistently resolves the systemic coarse-peak failures of the crispness methodology without degrading clustering performance on simpler datasets.

		Agreement /10		mean NMI	
Dataset	k	Crisp	TST	Crisp	TST
25% of the original dataset retained					
AGG	7	2	5	0.835	0.980
CLUS8	8	0	8	0.639	0.984
HORSE	3	2	0	0.858	0.796
SPIR2	2	10	10	0.942	0.942
SPIR3	3	3	8	0.805	0.757
50% of the original dataset retained					
AGG	7	0	8	0.866	0.991
CLUS8	8	0	7	0.662	0.988
HORSE	3	5	0	0.907	0.783
SPIR2	2	10	9	0.971	0.955
SPIR3	3	5	9	0.910	0.970
75% of the original dataset retained					
AGG	7	0	10	0.803	0.996
CLUS8	8	0	8	0.716	0.988
HORSE	3	3	0	0.875	0.799
SPIR2	2	10	10	0.968	0.968
SPIR3	3	7	9	0.910	0.971
$N = 10\,000$, *10% noise (1000 noise / 9000 signal)*					
AGG	7	1	9	0.764	0.957
CLUS8	8	0	10	0.785	0.999
HORSE	3	0	6	0.824	0.914
SPIR2	2	10	10	0.939	0.939
SPIR3	3	9	10	0.985	0.998
$N = 10\,000$, *20% noise (2000 noise / 8000 signal)*					
AGG	7	1	9	0.780	0.983
CLUS8	8	0	10	0.597	0.997
HORSE	3	3	3	0.875	0.838
SPIR2	2	10	10	0.950	0.950
SPIR3	3	3	10	0.872	0.943
$N = 10\,000$, *25% noise (2500 noise / 7500 signal)*					
AGG	7	1	10	0.920	0.998
CLUS8	8	0	6	0.615	0.987
HORSE	3	1	2	0.841	0.843
SPIR2	2	10	9	0.952	0.920
SPIR3	3	3	8	0.850	0.820

Table 3: Crispness maximisation (Crisp) versus the spectral (TST) score, the latter selecting the largest m with $S(m) > 1.0$ at $\lambda = 0.01$. **Each row is computed from ten independent resampling experiments.** "Agreement /10" is the number of runs whose selected number of clusters equals the ground–truth k; "mean NMI" is the clustering NMI (without considering the noise) against the ground truth averaged over the ten runs.

D Algorithmic Implementation and Complexity

This section provides the complete step-by-step algorithmic implementation for constructing the density-based transition matrix $\mathbf{P}$ utilized in the Spectral Density Peaks (SDP) clustering algorithm. Additionally, we provide a comprehensive breakdown of the time and memory complexities.

D.1 Transition Matrix Construction

Algorithm 1 details the routine for computing the transition matrix P. In practical applications, this process is optimized by restricting the inner loops to the n nearest neighbors of each point, yielding a sparse matrix representation.

Algorithm 1 Transition Matrix Construction

Input: logarithmic density estimates $\log \hat{\rho}_i$, uncertainty estimates $\hat{\sigma}_i$, distances $R_{i,j}$, size N
for $i = 1$ **to** N **do**
  for $j = 1$ **to** N **do**
    if $i = j$ **then**
      $C_{i,j} = 0$
    else
      $s_{i,j}^2 = \hat{\sigma}_i^2 + \hat{\sigma}_j^2$
      $z_{i,j} = s_{i,j}^{-1}(\log \hat{\rho}_i - \log \hat{\rho}_j)$ {Equation 1}
      $C_{i,j} = \Phi(-z_{i,j})$ {Equation 2}
    end if
  end for
end for
for $i = 1$ **to** N **do**
  for $j = 1$ **to** N **do**
    if $i \neq j$ **then**
      $P_{i,j} = C_{i,j} \prod_{l=1}^{N} (1 - C_{i,l})^{\mathbb{1}_{\mathrm{R_{i,l}} < \mathrm{R_{i,j}}}}$ {Equation 3}
    end if
  end for
  $P_{i,i} = 1 - \sum_{j \neq i} P_{i,j}$ {Equation 3}
end for

D.2 Computational Complexity

Let D denote the data dimensionality and n the truncated number of nearest neighbors. When applying the PAk sparsification, the average neighborhood size is $\bar{n} = \sum_{i=1}^{N} n_i / N$. The total execution complexity of the SDP algorithm depends heavily on whether a dense or sparse representation is used:

- **Distances (R):** Limiting the distance matrix R to n nearest neighbors reduces the time complexity from $\mathcal{O}(N^2 D)$ to $\mathcal{O}(N \log N + NnD)$ by utilizing space-partitioning trees for moderate D. The memory complexity is proportionally reduced from $\mathcal{O}(N^2)$ to $\mathcal{O}(Nn)$.
- **Density Estimation:** Given R and the nearest neighbor ordering, the PAk algorithm scales in time as $\mathcal{O}(N\bar{n})$ and requires $\mathcal{O}(N)$ additional memory space.
- **Matrix Construction (P):** Populating the transition matrix P requires $\mathcal{O}(N^3)$ time and $\mathcal{O}(N^2)$ memory in dense format. By leveraging the sparse format restricted to local neighborhoods, both complexities reduce drastically to $\mathcal{O}(Nn)$.
- **Coarse-Graining (cPCCA+):** The cPCCA+ algorithm relies on the eigendecomposition of P. In dense format, this scales as $\mathcal{O}(N^3)$ in time and $\mathcal{O}(N^2)$ in memory. For a sparse P with $\mathrm{nnz}(P)$ nonzero elements (e.g., $\mathrm{nnz}(P) = N\bar{n}$ under PAk truncation), the complexities reduce to $\mathcal{O}(k\ \mathrm{nnz}(P))$ in time and $\mathcal{O}(\mathrm{nnz}(P) + Nk)$ in memory, where k is the specified number of clusters.

E Supplementary Information for Synthetic Datasets with Noise

This section reports the complete experimental settings associated with Section 4.1. We use the `scikit-learn` (Pedregosa et al., 2011) implementations of KM, SC, MS, and HDBSCAN, while DPA is run through `DADApy` (Glielmo et al., 2022). The SC affinity matrix is constructed from a nearest-neighbor graph, and the MS bandwidth is estimated with `sklearn.cluster.estimate_bandwidth`. All experiments comprise 50 resamples of $N = 15000$ points. SDP selects k from $[2, 10]$ and does not use an entropy threshold or halo assignment. For the competing methods, the candidate configurations below are evaluated independently in every resample and the configuration maximizing NMI on ground-truth signal points is retained.

For each method, Table 5 reports the most frequently selected hyperparameter values across the 50 resamples, together with their relative frequencies. The values correspond to the configurations max-

Table 4: Hyperparameter ranges used in the synthetic-data benchmark. Braces denote the discrete values evaluated.

ALGORITHM	HYPERPARAMETER	VALUES
SDP	`n_clusters`	$2, \ldots, 10$ (TST SELECTION)
KM	`n_clusters`	$2, \ldots, 10$
SC	`n_clusters`	$2, \ldots, 10$
	`n_neighbors`	$\{5, 10, 20, 40, 80\}$
MS	`quantile`	$\{0.01, 0.02, 0.04, 0.08, 0.16\}$
HDBSCAN	`min_cluster_size`	$\{50, 100, 200, 400\}$
	`min_samples`	$\{5, 10, 20, 40\}$
DPA	`Z`	$\{0.50, 0.75, \ldots, 3.50\}$

Table 5: Most frequently selected hyperparameter values across the 50 resamples, reported as mode (relative frequency).

ALGORITHM	HYPERPARAMETER	SPIR2	SPIR3	HORSE	AGG	CLUS8
SDP	`n_clusters`	2 (1.00)	3 (0.96)	3 (1.00)	7 (0.98)	8 (0.92)
KM	`n_clusters`	8 (1.00)	10 (1.00)	4 (0.76)	6 (1.00)	10 (1.00)
SC	`n_clusters`	4 (0.80)	9 (0.46)	4 (0.66)	5 (0.84)	7 (1.00)
	`n_neighbors`	20 (0.44)	80 (0.64)	20 (0.34)	80 (0.84)	80 (1.00)
MS	`quantile`	0.01 (0.72)	0.02 (0.92)	0.04 (0.50)	0.08 (0.52)	0.04 (1.00)
HDBSCAN	`min_cluster_size`	100 (0.40)	50 (0.38)	400 (0.46)	100 (0.50)	50 (0.56)
	`min_samples`	5 (0.70)	10/20 (0.34)	40 (0.30)	5 (0.38)	5 (0.90)
DPA	`Z`	3.00 (0.30)	2.75 (0.32)	2.50 (0.42)	1.75 (0.30)	1.00 (0.46)

imizing NMI on the ground-truth signal points. For MS, the bandwidth depends on the resampled data and is therefore summarized through the selected bandwidth quantile rather than the bandwidth itself. In SPIR3, the HDBSCAN values `min_samples`$= 10$ and 20 are tied, each being selected in 34% of resamples.

Table 6 summarizes the number of clusters $\hat{k}$ returned by each method across the 50 resamples. For KM and SC, it coincides with the oracle-selected `n_clusters` hyperparameter. For MS, HDBSCAN, and DPA, the number of clusters is instead determined by the fitted partition obtained under the oracle-selected configuration.

Table 6: Number of clusters $\hat{k}$ returned across the 50 resamples, reported as mode (relative frequency).

ALGORITHM	SPIR2	SPIR3	HORSE	AGG	CLUS8
SDP	2 (1.00)	3 (0.96)	3 (1.00)	7 (0.98)	8 (0.92)
KM	8 (1.00)	10 (1.00)	4 (0.76)	6 (1.00)	10 (1.00)
SC	4 (0.80)	9 (0.46)	4 (0.66)	5 (0.84)	7 (1.00)
MS	263 (0.06)	63 (0.14)	39 (0.12)	5 (0.38)	63 (0.12)
HDBSCAN	2 (0.38)	3 (0.72)	3 (0.94)	10 (0.26)	7 (0.32)
DPA	2 (0.86)	3 (0.88)	3 (1.00)	7 (0.76)	8 (0.44)

For completeness, Table 7 reports NMI computed on all sampled points, including the ground-truth noise class. These values should not be interpreted as a pure cluster-structure comparison because the methods handle noise differently: HDBSCAN explicitly assigns a noise label, whereas the present SDP and DPA runs do not use halo post-processing. The signal-only NMI in Table 1 is therefore the primary benchmark used in the main text.

F SUPPLEMENTARY INFORMATION FOR THE MNIST DATABASE

The contingency matrix in Figure 3 illustrates the tendency of the spectral density-peak clustering algorithm to naturally over-segment the ten ground-truth classes into 18 distinct structural clusters

Table 7: NMI on all sampled points, including the ground-truth noise class, reported as mean ± standard deviation over 50 resamples.

ALGORITHM	SPIR2	SPIR3	HORSE	AGG	CLUS8
SDP	0.683 ± 0.010	0.761 ± 0.044	0.541 ± 0.003	0.601 ± 0.010	0.734 ± 0.009
KM	0.259 ± 0.005	0.208 ± 0.002	0.405 ± 0.007	0.550 ± 0.003	0.686 ± 0.007
SC	0.432 ± 0.020	0.313 ± 0.011	0.416 ± 0.011	0.527 ± 0.006	0.723 ± 0.001
MS	0.353 ± 0.005	0.407 ± 0.004	0.412 ± 0.008	0.548 ± 0.019	0.704 ± 0.004
HDBSCAN	0.926 ± 0.018	0.882 ± 0.013	0.789 ± 0.034	0.769 ± 0.040	0.808 ± 0.008
DPA	0.631 ± 0.024	0.754 ± 0.043	0.542 ± 0.005	0.600 ± 0.007	0.734 ± 0.005

(c0-c17). This behavior is driven by the algorithm's TST criterion, which identify multiple valid density peaks within a single semantic class rather than enforcing a strict ten-bin constraint.

Intra-Class Over-Segmentation The separation of digit 5 primarily across clusters 1 and 2, and digit 4 across clusters 7 and 8, demonstrates the model's sensitivity to stylistic extremes. As shown in the bottom red and purple panels, the algorithm successfully isolates the "farthest pairs" for these digits. For digit 4, this structural divergence corresponds to the separation of closed-loop versus open-top handwriting styles. For digit 5, the separation reflects extreme variations in loop curvature and stroke angle. By relying on local density landscapes and spectral gap metrics, the clustering method avoids merging these morphologically distinct subgroups into single, high-variance domains.

Inter-Class Boundary Ambiguity Conversely, the orange and green panels highlight boundary regions where the spectral gap between different ground-truth digits is minimal, resulting in shared cluster assignments. Cluster 4 contains the nearest pair for digits 1 and 7, capturing variations where the distinguishing top horizontal stroke of the 7 is muted or heavily slanted, making it topologically nearly identical to a 1. Similarly, cluster 14 merges ambiguous variations of digits 1 and 2, mapping a shared density manifold where the distinctive strokes of the 2 are degraded. These nearest-pair overlaps confirm that the clustering assignments faithfully represent continuous transition zones in the underlying pixel-space feature manifold rather than enforcing rigid, artificial semantic separation.

Selection of the k parameter. As illustrated in Figure F, the plot reveals a complex density landscape for determining the optimal k. The algorithm initially highlights a strong peak at $k = 4$, explicitly marked by the red indicator, which rests prominently above the defined significance threshold of 1.0.

While $k = 4$ represents a mathematically highly scored configuration according to the TST metric, empirical inspection of the resulting assignments reveals a severe structural failure in the form of extreme class imbalance. Let N_c denote the population of cluster c. For $k = 4$, the dataset is almost entirely absorbed by a single dominant macro-cluster, yielding populations of $N_0 = 47976$, $N_1 = 10$, $N_2 = 5969$, and $N_3 = 6045$. This highly skewed partitioning demonstrates that the $k = 4$ setting collapses the rich topological diversity of the MNIST dataset into an uninformative monolith, failing to isolate meaningful morphological features.

Following the initial peaks, the TST score experiences a sharp decline, remaining in deeply negative territory between $k = 6$ and $k = 12$. However, as the number of clusters increases, the metric recovers and forms a distinct, stable secondary local maximum at $k = 18$. Although this second peak sits below the strict 1.0 threshold, it represents the most viable structural equilibrium in the dataset's underlying feature manifold.

Selecting the secondary peak at $k = 18$ directly resolves the extreme imbalance observed at $k = 4$. Instead of forcing the data into a singular mass, this configuration allows the spectral density peak algorithm to intelligently segment the data into granular, morphologically coherent sub-groups. As detailed in the contingency matrix analysis, this higher value of k provides the necessary flexibility to capture complex boundary dynamics, successfully mapping both ambiguous inter-class similarities and extreme intra-class stylistic variances across the dataset.

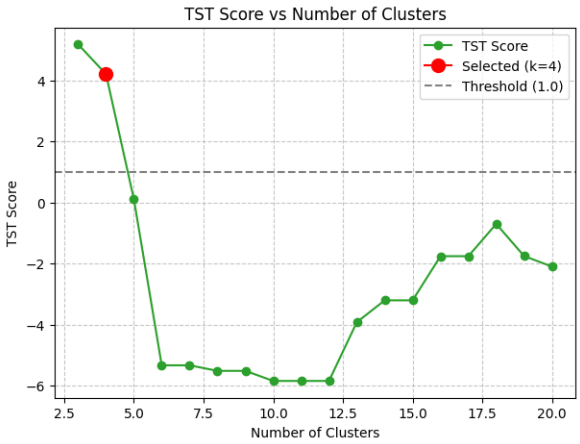


Figure 5: TST score versus number of clusters. The evaluation metric highlights a primary peak at $k = 4$ (red marker) exceeding the 1.0 threshold, and reveals a secondary local maximum at $k = 18$.

G Impact of neighborhood correlation on transition dynamics

As discussed in Section 3, the standard SDP algorithm assumes zero covariance between local density estimation errors for computational efficiency (Assumption (ii)). The ideal approach for capturing manifold topology would incorporate neighborhood correlation; for instance, by utilizing the Jaccard overlap ω_{ij} to correct the pairwise variance.

To account for correlations between neighboring log-density estimates arising from shared k^*-neighborhoods, we model the joint distribution of the log-density differences along each point's neighbor chain as multivariate Gaussian, with the covariance between any two points approximated via the Jaccard overlap of their k^*-neighborhoods. The dominance probability of each candidate neighbor is then obtained as a multivariate-normal lower-orthant probability, conditioned on all previously visited neighbors in the chain, evaluated via numerical integration for low-dimensional cases and Sobol quasi-Monte Carlo sampling otherwise.

The primary objective of this section is to demonstrate that the computationally efficient independent-error approximation successfully preserves the fundamental properties of the underlying manifold geometry. As shown in Table 9, the macroscopic density basins and resulting cluster assignments produced under the independence assumption are virtually identical to those generated with the full correlation correction across all five synthetic benchmark datasets.

This structural consistency is rigorously supported by both information-theoretic and spectral metrics:

- **Cluster Agreement:** The Normalized Mutual Information (NMI) computed between the assignments obtained with and without the correlation correction remains remarkably high, peaking at 0.975 for the CLUS8 dataset.
- **Transition Matrix Perturbation:** The structural difference between the two transition matrices is microscopic, with the Frobenius norm of their difference bounded in the order of 10^{-8}.
- **Spectral Stability:** The L^2 distances between the dominant eigenvalues of the two matrices are effectively zero (ranging from 0 to 10^{-7}). Furthermore, the maximum principal angle between their corresponding invariant subspaces is tightly bounded, ranging from 3.57° (SPIR2) to 12.04° (HORSE).

The third column of Table 9 visualizes this agreement directly at the entry level: each scatter plot shows, for every retained pair (i, j), the transition probability $P_{i,j}$ computed without correlation against the corresponding value computed with the Jaccard-corrected variance. Across all five

datasets, the point cloud concentrates tightly along the identity diagonal, with no systematic curvature or offset in either direction. This indicates that the correlation correction does not introduce a directional bias but rather perturbs individual entries by a small amount consistent with the Frobenius-norm bound reported below. The absence of visible dispersion at this entry-wise level, prior to any aggregation into clusters or spectral quantities, is itself evidence that the independence assumption (ii) is not merely adequate in aggregate but locally accurate throughout the transition matrix.

This entry-level stability makes the apparent deviations in the HORSE dataset highly illustrative. The HORSE dataset attains the weakest agreement on two metrics in Table 9 (NMI 0.867; subspace distance $12.04°$), and this is not a failure of the independence approximation itself but a consequence of a genuine ambiguity in the number of clusters. HORSE ($k = 3$ by ground truth) has two of its three components separated by a comparatively weak density barrier, so the crispness profile places nearly comparable mass on $k = 2$ and $k = 3$; reinstating the neighborhood correlation shifts the crispness values by an amount small in absolute terms but sufficient to occasionally flip the arg max between these two closely competing choices of k. When this occurs, the two clusterings compared in Table 9 disagree on whether the ambiguous pair of components is merged, which is sufficient to depress the NMI and inflate the subspace distance even though the underlying transition matrices remain close in both Frobenius norm (8.21×10^{-8}) and dominant-eigenvalue distance (1.93×10^{-11}).

HORSE therefore illustrates a boundary case of model-selection sensitivity near a genuine k-ambiguity, rather than a breakdown of the independence assumption at fixed k. Consequently, the invariant subspace defining the geometric coordinate system of the clusters is highly stable, justifying the adoption of the independent-error approximation as a robust and scalable default for the SDP algorithm.

Finally, the empirical validation of the full correlation correction was restricted to the five synthetic benchmark datasets due to severe computational bottlenecks. Reinstating the full neighborhood covariance replaces the efficient, rank-one independent error evaluations with dense, per-neighborhood matrix computations. While this overhead is manageable on smaller synthetic manifolds, it becomes computationally prohibitive for larger, high-dimensional data such as the MNIST database. Table 8 quantifies this scaling penalty, contrasting the execution time required to assemble the transition matrix under the default independent-error approximation versus the full Jaccard-corrected model.

Table 8: Comparison of computational time (in seconds) required to assemble the transition matrix across the five synthetic benchmark datasets.

Method	**AGG**	**CLUS8**	**HORSE**	**SPIR2**	**SPIR3**
SDP Default	7.2	4.8	1.7	5.3	1.9
SDP with Correlation	3970.3	5812.5	4655.5	12601.8	1336.3

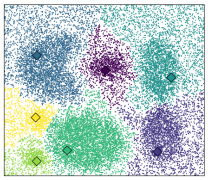
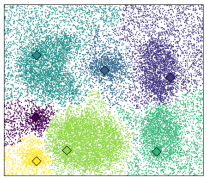
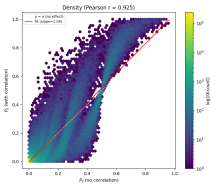
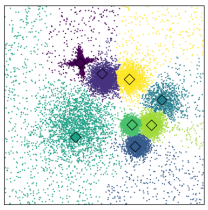
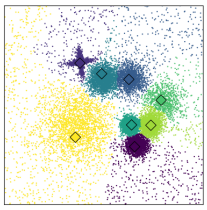
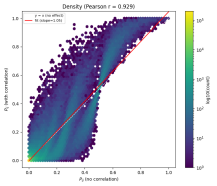
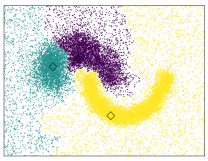
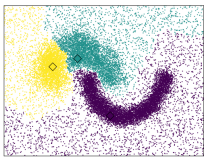
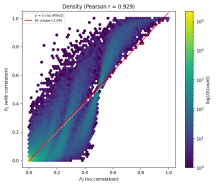
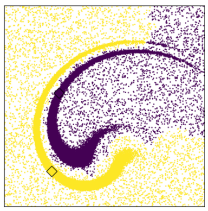
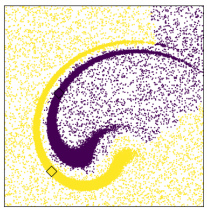
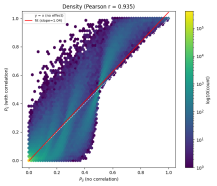
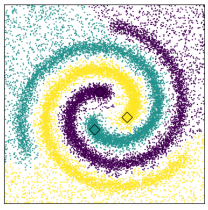
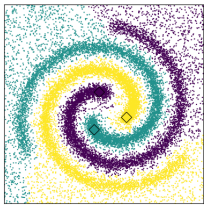
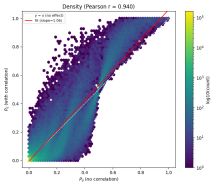
	Without Correlation	With Correlation	$\mathbf{P}_{ij}$	Statistics
AGG				• NMI: 0.964 • Frob. Norm: 8.46×10^{-8} • L^2 eigenvalues: 2.05×10^{-9} • Subspace Distance: 7.65°
CLUS8				• NMI: 0.975 • Frob. Norm: 8.30×10^{-8} • L^2 eigenvalues: 1.95×10^{-7} • Subspace Distance: 8.19°
HORSE				• NMI: 0.867 • Frob. Norm: 8.21×10^{-8} • L^2 eigenvalues: 1.93×10^{-11} • Subspace Distance: 12.04°
SPIR2				• NMI: 0.952 • Frob. Norm: 3.03×10^{-8} • L^2 eigenvalues: 0.0 • Subspace Distance: 3.57°
SPIR3				• NMI: 0.960 • Frob. Norm: 7.91×10^{-8} • L^2 eigenvalues: 7.10×10^{-15} • Subspace Distance: 7.59°

Table 9: Structural comparison of density-peak clustering with and without neighborhood correlation across five benchmark datasets. The correlation correction regularizes local transition probabilities (Col. 3) while rigorously preserving the global manifold topology and macroscopic cluster assignments (Cols. 1, 2, and 4).