A Statistical Approach to Clustering

Lizzy Coda

December 9, 2024

Outline

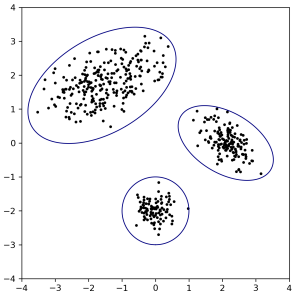
What is Clustering?

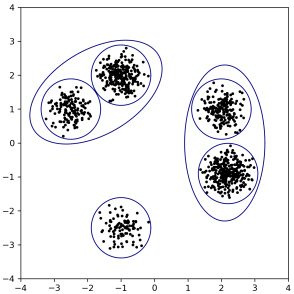
Definition (Clustering)

A clustering of a set $\mathcal{X}$ is a partition of $\mathcal{X}$ into subsets called clusters.

Definition (Hierarchical Clustering)

A hierarchical clustering, or cluster tree, of $\mathcal{X}$ is a collection of subsets of $\mathcal{X}$, referred to as clusters, that has a nested structure in that two clusters are either disjoint or nested.





The Computer Scientist's POV

To a computer scientist, the correct clustering (or hierarchical clustering) of a dataset might be viewed as one that solves an optimization problem (Puzicha et al., 2000; Dasgupta, 2016).

A good algorithm either solves or approximates a solution to this problem, ideally with some guarentees.

For example:

- Partition the data into k clusters so that the within cluster sum of squares is minimized (k-means problem)

$$\min_{\{C_i\}_{i=1}^k} \sum_{i=1}^{k} \sum_{X_j \in C_i} ||X_j - \mu_i||_2^2$$

Note that from this perspective, that dataset is viewed as fixed, and the cluster definition or best clustering is based on the data alone.

The Computer Scientist's POV

More examples:

- Partition the data into k clusters so that the largest diameter of the clusters is minimized (Grosswendt and Roeglin, 2017)

$$\min_{\{C_i\}_{i=1}^k} \max_{1 \le i \le k} diam(C_i)$$

- Partition a graph into k clusters to optimize some notion of a cut (Von Luxburg, 2007)

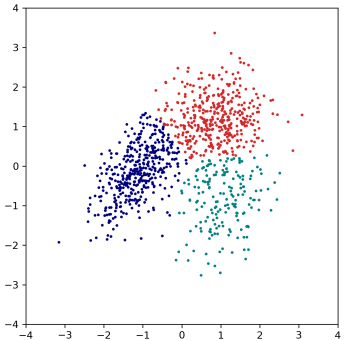
$$\min_{\{C_i\}_{i=1}^k} \frac{1}{2} \sum_{i=1}^{k} W(C_i, \overline{C_i})$$

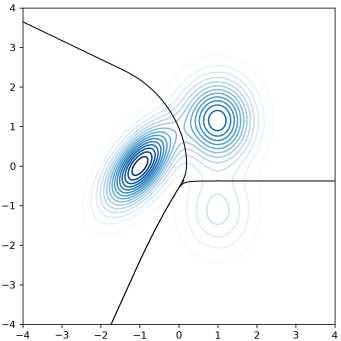
In some applications, the notion of a good clustering can be further loosened. For example, according to Shi and Malik (2000), in task of image segmentation, a good clustering need only "extract the global impression of an image".

The Statistician's POV

In statistics, the dataset is assumed representative of an underlying population and the correct clustering of a dataset is one that is consistent with the population level clustering.

A good clustering algorithm is consistent (exact, in the large-sample limit).



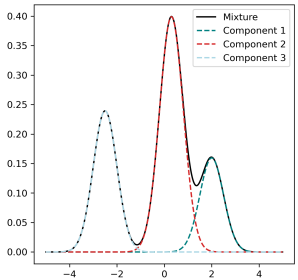


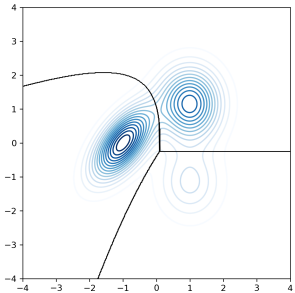
How should the population clustering be defined?

In the mixture-model based definition, points are clustered according to the most likely component in the mixture (Fraley and Raftery, 2002; Bouveyron et al., 2019).

$$f(x) = \sum_{i=1}^{k} \alpha_i f_i(x)$$

$$C_i = \{x : \alpha_i f_i(x) = \max_{1 \le j \le k} \alpha_j f_j(x)\}.$$

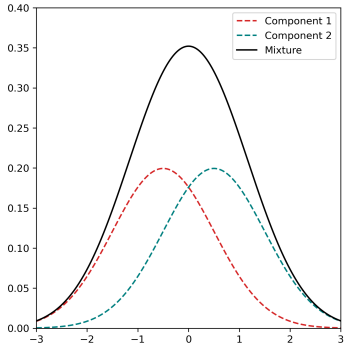


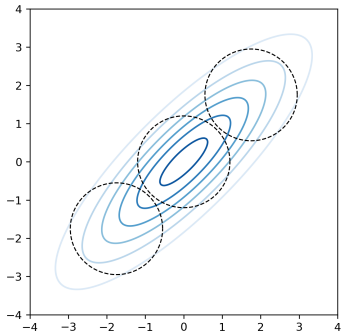


How should the population clustering be defined?

The mixture model based definition is problematic:

- Oversegments unimodal densities
- May require a choice of model (i.e. spherical Gaussian)
- May require a choice in number of components or require a very large number of components to approximate well





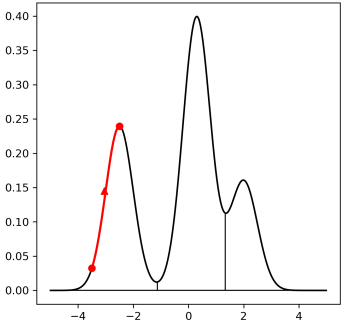
How should the population clustering be defined?

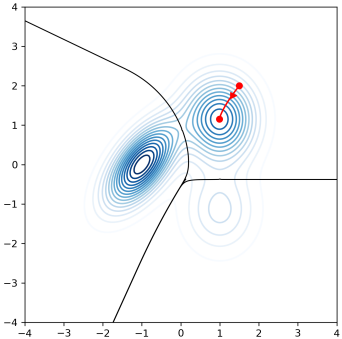
In the gradient flow definition, points are assigned to the nearest mode in the direction of the gradient (Fukunaga and Hostetler, 1975).

$$\gamma_x(0) = x, \qquad \dot{\gamma}_x(t) = \nabla f(\gamma_x(t)), \qquad t \geq 0$$

The clusters correspond to the basins of attraction of the modes.

$$\{x : \gamma_x(\infty) = x^*\}$$



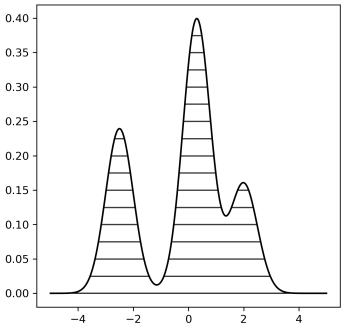


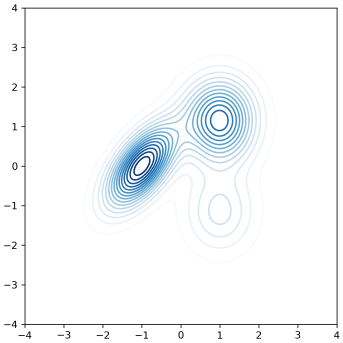
Under mild regularity conditions, these basins of attraction partition $\text{supp}(f)$ up to a set of measure zero.

How should the population (hierarchical) clustering be defined?

In Hartigan's cluster tree the clusters correspond to the connected components of the upper level sets (Hartigan, 1975).

$$\mathcal{C}_f = \bigcup_{\lambda} cc(\{x : f(x) \geq \lambda\})$$





How should the population (hierarchical) clustering be defined?

Hartigan's cluster tree is generally accepted as the definition of hierarchical clustering at the population level and has been used in subsequent works:

- Balakrishnan et al. (2013), Chaudhuri et al. (2014), Eldridge et al. (2015), and Wang et al. (2019) propose estimators of Hartigan's tree and prove they are consistent (meaning exact in the large-sample limit).
- Kim et al. (2016) proposes a method to construct confidence sets for Hartigan's tree.

This hierarchical clustering is fully compatible with the gradient flow definition of clustering (Arias-Castro and Qiao, 2023a,b).

Hartigan provides minimal motivation for this definition beyond observing that each cluster C in his tree "conforms to the informal requirement that C is a high-density region surrounded by a low-density region"(Hartigan, 1975).

Outline

Previous Axiomatic Approaches to Clustering

Several works have explored axiomatic approaches to defining clustering algorithms that take as input a finite number of data points:

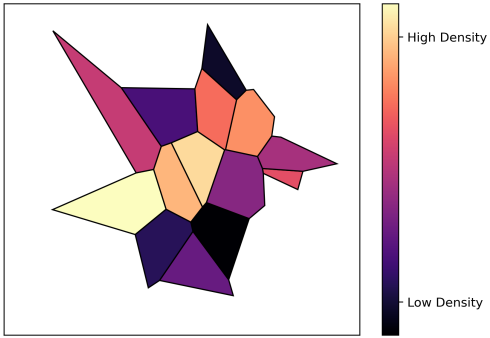
- Kleinberg proposes three axioms (scale-invariance, richness, and consistency) and proves no clustering algorithm can simultaneously satisfy all three (2002).
- Kleinberg (2002), Cohen-Addad et al. (2018), and Zadeh and Ben-David (2012) prove the existence of clustering algorithms under relaxations of Kleinberg's original axioms.
- Puzicha et al. (2000) and Ackerman and Ben-David (2009) propose a set of axiom for a clustering objective function.
- Jardine et al. (1967) and Carlsson and Mémoli (2010) propose a set of axioms for hierarchical clustering algorithms.

Preliminaries

We first consider the class $\mathcal{F}$ of piecewise constant functions with connected, bounded support. A function in $\mathcal{F}$ is of the form

$$f = \sum_{i=1}^{m} \lambda_i \mathbb{I}_{A_i}; \qquad \lambda_i > 0$$

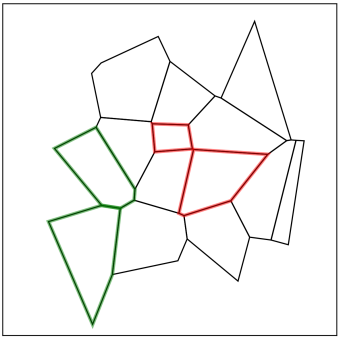
where each A_i is a connected, bounded region with connected interior, and $\text{supp}(f) = \bigcup_{i=1}^{m} \bar{A}_i$ has connected interior. Without loss of generality, also assume $A_i \cap A_j = \emptyset$ for $i \neq j$.



Preliminaries

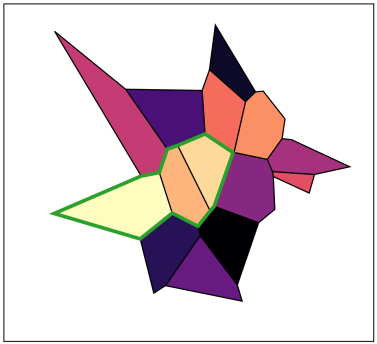
Definition

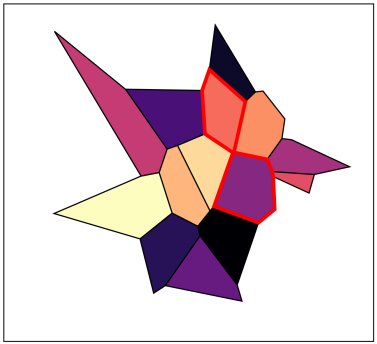
Given a collection of sets $\mathcal{A} = \{A_i\}$, we define the neighborhood of A_i as $\mathcal{N}(A_i) = \bigcup \{A_j : \text{int}(\overline{A_i} \cup \overline{A_j}) \text{ is connected}\}$

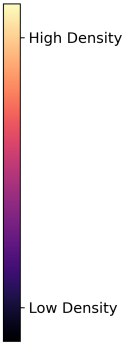


Note that $A_j \subseteq \mathcal{N}(A_i) \iff Ai \subseteq \mathcal{N}(A_j)$, so that we may speak of A_i and A_j as being neighbors, which we will denote by $A_i \sim A_j$

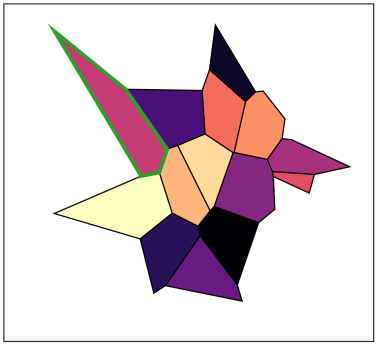
Axiom 1: Clusters have connected interior

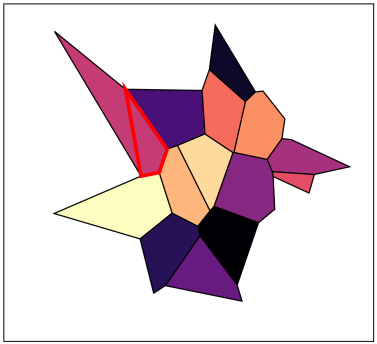


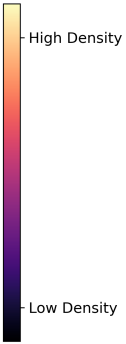




Axiom 2: Clusters do not partition connected regions of constant density

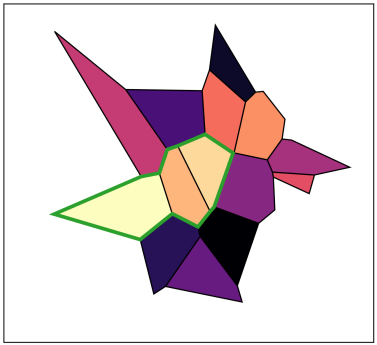


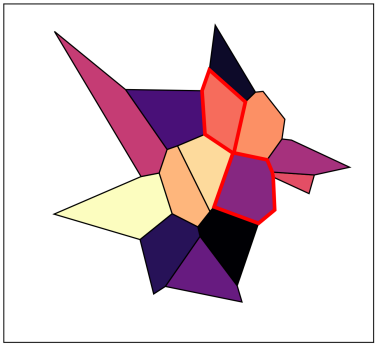


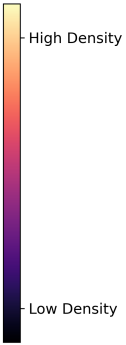


Any $C \in \mathcal{C}$ is of the form $C = \bigcup_{i \in I} A_i$ for some $I \subseteq \{1, 2, ..., m\}$.

Axiom 1 (revisted): Clusters have connected interior

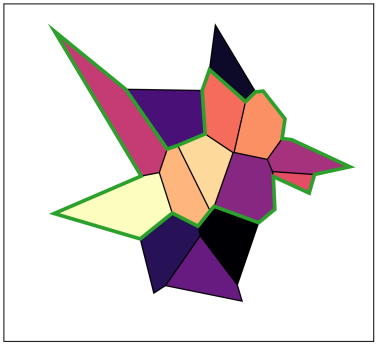


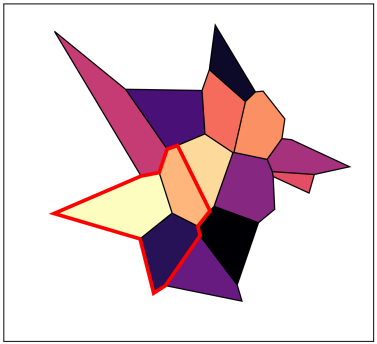


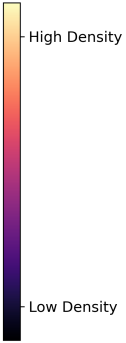


If $C \in \mathcal{C}$ and $A_i, A_j \subseteq C$, then there are $A_{k_1}, \ldots, A_{k_n} \subseteq C$ such that $A_i \sim A_{k_1} \sim \cdots A_{k_n} \sim A_j$.

Axiom 3: Clusters are surrounded by regions of lower density







For any $C \in \mathcal{C}$, it holds that

$$\inf_{x \in C} f(x) > \sup_{x \in \mathcal{N}(C) \setminus C} f(x)$$

Here, if $C = \bigcup_{i \in I} A_i$, then $\mathcal{N}(C) = \bigcup_{i \in I} \mathcal{N}(A_i)$

The finest axiom cluster tree

Definition (Finer cluster tree)

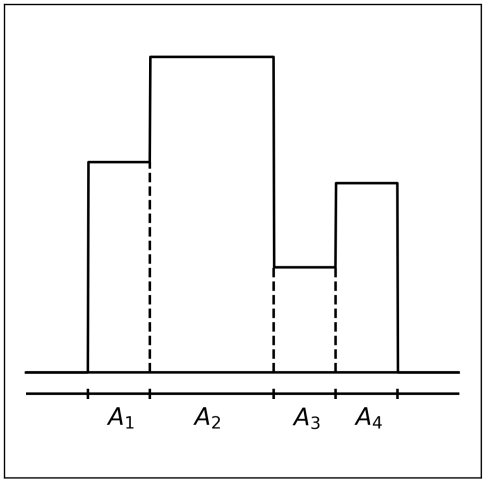
We say that a cluster tree $\mathcal{C}$ is finer than (or a refinement of) another cluster tree $\mathcal{C}'$ if $\mathcal{C}$ includes all the clusters of $\mathcal{C}'$, namely, $C \in \mathcal{C}' \implies C \in \mathcal{C}$.

Proposition

For any $f \in \mathcal{F}$, there exists a unique finest hierarchical clustering of f among those satisfying the axioms.

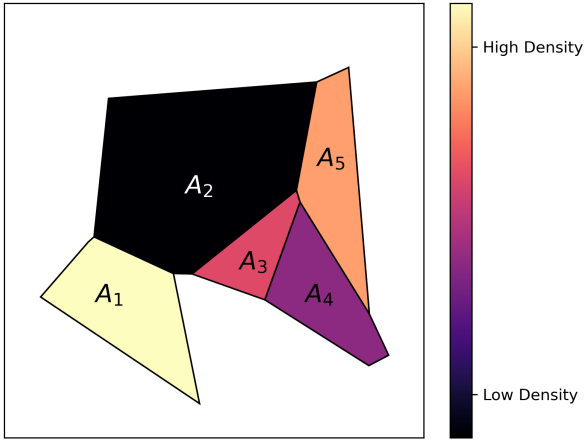
We will call this tree the finest axiom cluster tree, denoted by $\mathcal{C}_f^*$.

Example: The finest axiom cluster tree



$$\mathcal{C}_f^* = \{A_2, A_4, A_1 \cup A_2, A_1 \cup A_2 \cup A_3 \cup A_4\}$$

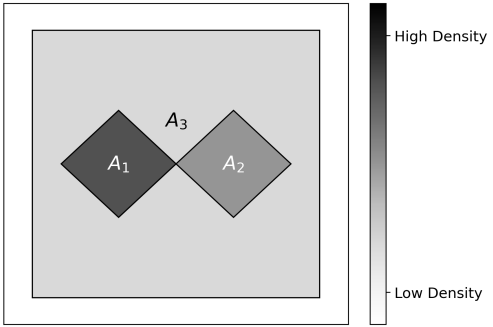
Example: The finest axiom cluster tree



$$\mathcal{C}_f^* = \{A_1, A_5, A_3 \cup A_5, A_3 \cup A_4 \cup A_5, A_1 \cup A_2 \cup A_3 \cup A_4 \cup A_5\}$$

Comparison with Hartigan's Cluster Tree

For $f \in \mathcal{F}$, the clusters in Hartigan's cluster tree $\mathcal{H}_f$ need not satisfy Axiom 1, and thus in general $\mathcal{H}_f \neq \mathcal{C}_f^*$.



In the above, $\mathcal{H}_f = \{A_1, A_1 \cup A_2, A_1 \cup A_2 \cup A_3\}$ but $A_1 \cup A_2 \notin \mathcal{C}_f^*$ and $\mathcal{C}_f^* = \{A_1, A_2, A_1 \cup A_2 \cup A_3\}$.

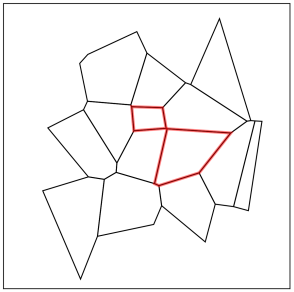
Comparison with Hartigan's Cluster Tree

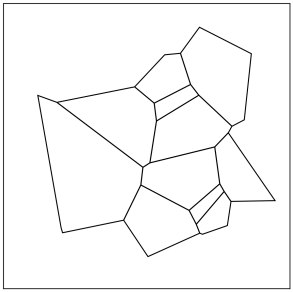
Theorem

For any $f \in \mathcal{F}_{\text{int}}$, it holds that $\mathcal{C}_f^* = \mathcal{H}_f$.

We define $\mathcal{F}_{\text{int}}$ as the class of functions in $\mathcal{F}$ with $\{A_i\}$ having the **interally connected property**

$$\overline{A_i} \cup \overline{A_j} \text{ connected} \implies \text{int}\big(\overline{A_i} \cup \overline{A_j}\big) \text{ connected}$$





Merge distortion pseudometric

Definition

Let $(\mathcal{C}, f)$ be a pair consiting of a cluster tree and underlying density $f : \Omega \to \mathbb{R}$. The merge distortion distance between $(\mathcal{C}, f)$ and $(\mathcal{C}', g)$ is

$$d_M((\mathcal{C}, f), (\mathcal{C}', g)) = \sup_{x,y \in \Omega} |m_{\mathcal{C},f}(x, y) - m_{\mathcal{C}',g}(x, y)|$$

where

$$m_{\mathcal{C},f}(x, y) = \sup_{\substack{C \in \mathcal{C} \\ x,y \in C}} \inf_{z \in C} f(z)$$

Lemma

$$d_M((\mathcal{H}_f, f), (\mathcal{H}_g, g)) \leq \|f - g\|_\infty$$

d_M was introduced by Eldridge et al. (2015) and has gained some popularity in subsequent works that discuss the consistency of hierarchical methods (Kim et al., 2016; Wang et al., 2019).

Extension to continuous functions with connected support

Definition

Given a continuous function f with connected support, we say that $\mathcal{C}$ is an axiom cluster tree for f if there is a sequence $(f_n) \subseteq \mathcal{F}$ such that

$$\lim_{n\to\infty} d_M((\mathcal{C}^*_{f_n}, f_n), (\mathcal{C}, f)) = 0.$$

By the previous results, if there exists a sequence $(f_n) \in \mathcal{F}_{int}$ that uniformly converges to f, then the Hartigan tree $\mathcal{H}_f$ is an axiom cluster tree for f.

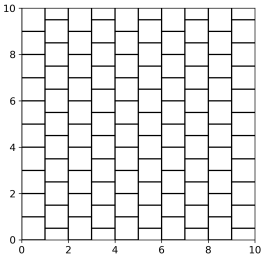
Under what conditions does such sequence exist?

Proposition

Suppose (Ω, d) is a metric space where all closed and bounded sets are compact, and that has the internally connected partition property. Let $f : \Omega \to [0, \infty)$ be continuous with all upper level sets bounded, and such that the upper λ-level set is connected when $\lambda > 0$ is small enough. Then, there is a sequence $(f_n) \in \mathcal{F}_{\text{int}}$ that converges uniformly to f.

Definition

We say that Ω has the internally connected partition property if it is connected, and for any $r > 0$, there exists a locally finite partition $\{A_i\}$ of Ω that has the internally connected property and is such that, for all i, A_i is connected with connected interior and diameter at most r.



Extension to continuous functions with disconnected support

Let f be a function of the form

$$f = \sum_{j=1}^{N} f_j; \qquad \text{supp}(f_j) \cap \text{supp}(f_k) = \emptyset$$

Again, under mild conditions on f, our axiomatic approach recovers Hartigan's cluster tree:

- If each $f_j \in \mathcal{F}_{int}$, extending the axioms gives

$$\mathcal{C}_f^* = \bigcup_{j=1}^{N} \mathcal{C}_{f_j}^* = \bigcup_{j=1}^{N} \mathcal{H}_{f_j} = \mathcal{H}_f$$

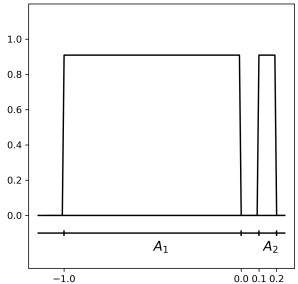
- If each f_j is continuous and satisfies the conditions of the previous theorem, there exists a sequence of piecewise constant functions (f_n) such that

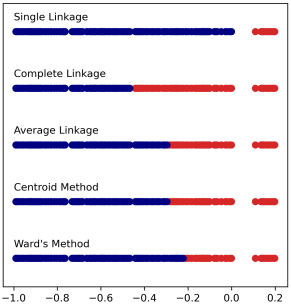
$$\lim_{n \to \infty} d_M((\mathcal{C}_{f_n}^*, f_n), (\mathcal{H}_f, f)) = 0$$

Algorithmic Implications

A good hierarchical clustering algorithm is consistent with the population-level hierarchical clustering.

- Most agglomerative linkage methods (i.e. complete linkage, average linkage) are inconsistent (Hartigan, 1977).
- Despite heavy criticism of it for its "chaining" tendencies, single linkage is consistent in $\mathbb{R}$, and fractionally consistent in $\mathbb{R}^d$ for $d \geq 2$ (Hartigan, 1981; Penrose, 1995).

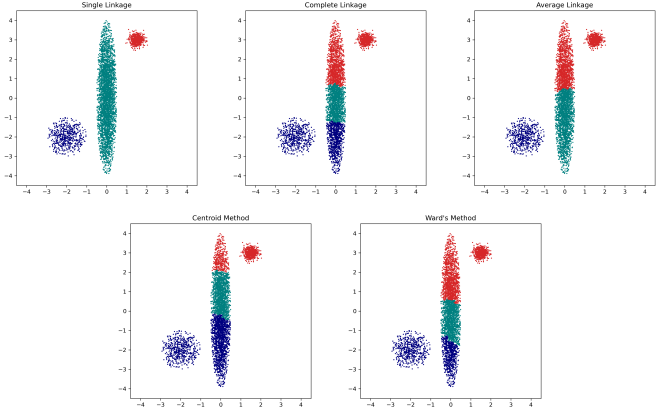




Algorithmic Implications

A good hierarchical clustering algorithm is consistent with the population-level hierarchical clustering.

- Most agglomerative linkage methods (i.e. complete linkage, average linkage) are inconsistent (Hartigan, 1977).
- Despite heavy criticism of it for its "chaining" tendencies, single linkage is consistent in $\mathbb{R}$, and fractionally consistent in $\mathbb{R}^d$ for $d \geq 2$ (Hartigan, 1981; Penrose, 1995).



Algorithmic Implications

Practically, there are robust variants of single linkage that overcome some of the issues with "chaining" and have positive consistency results:

- Robust single linkage (Chaudhuri et al., 2014; Eldridge et al., 2015)
- Hierarchical DBSCAN (Wang et al., 2019).

Clustering in High Dimensions

Wang et al. (2019) shows that the minimax rates for the estimation of the Hartigan cluster tree require the sample size to grow exponentially with the dimension (and in fact match the corresponding minimax rates for density estimation in the L_∞ norm under assumptions of Hölder smoothness on the density).

This is a real limitation of adopting the proposed definition of the cluster tree.

Outline

Main Question

Given a random geometric graph, is it possible to partition the graph in a way which is consistent with the clustering given by the gradient ascent flow of the density?

Definition (Random Geometric Graph)

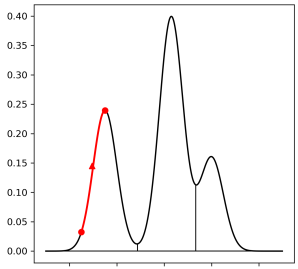
A random geometric graph $\mathcal{G}(\mathcal{Y}, \epsilon)$ has nodes corresponding to points that are generated iid from a density and adjacency matrix $A = (a_{ij})$ where $a_{ij} = 1$ if $||y_i - y_j|| \leq \epsilon$ and $a_{ij} = 0$ otherwise.

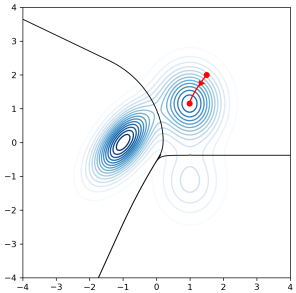
Gradient Flow Clustering

In the gradient flow definition, points are assigned to the nearest mode in the direction of the gradient (Fukunaga and Hostetler, 1975).

$$\gamma_x(0) = x, \qquad \dot{\gamma}_x(t) = \nabla f(\gamma_x(t)), \qquad t \geq 0$$

Under mild regularity conditions, the basins of attraction of the modes partition $\text{supp}(f)$ up to a set of measure zero.





This definition of clustering is fully compatible with the hierarchical clustering defined by Hartigan's cluster tree (Arias-Castro and Qiao, 2023a,b).

Gradient Flow Clustering Algorithms

Given knowledge of the underlying density, a number of mode-seeking algorithms have been proposed to approximate the gradient flow lines:

- Euler Scheme
- Mean Shift (Cheng, 1995)
- Max Slope Shift (Koontz et al., 1976)
- Max Shift (Arias-Castro and Qiao, 2022)

$$x(0) = x; \quad x(k+1) \in \underset{x \in \bar{B}(x(k), r)}{\operatorname{argmax}} f(x), \quad k \geq 1$$

Under mild conditions on the density and Morse regularity assumptions, these algorithms produce a sequence that converges to the correct mode x^*. (Arias-Castro and Qiao, 2022).

Gradient Flow Clustering Methods

In practice the density is not known and must be estimated from data $\mathcal{Y}$. The straightforward strategy is to replace f with a kernel density estimator $\hat{f}_\epsilon$ with bandwidth ϵ.

If only the data points are considered the Max Shift method is:

$$x(0) = x; \quad x(k+1) \in \underset{x \in \mathcal{Y} \cap \bar{B}(x(k), r)}{\operatorname{argmax}} \hat{f}_\epsilon(x), \quad k \geq 1$$

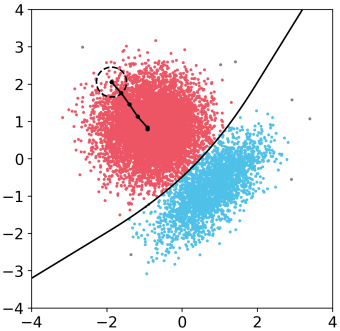
If the kernel is twice differentiable and the bandwidth is chosen so that so that the density estimator is second order consistent, and under mild conditions on the density, the Max Shift method is consistent (Arias-Castro and Qiao, 2022).

Graph Max Shift: Algorithm Description

Graph Max Shift initialized at node i, iteratively moves to neighbor with highest the degree (q_j denotes the degree of node j).

$$i_0 = i; \quad i_t \in \text{argmax}\{q_j : j \sim i_{t-1}\}, \quad t \geq 1.$$

We then cluster together nodes whose paths end at the same mode.



As a final post-processing step, we merge together any two clusters whose modes are within τ hops, where τ is a tuning parameter.

Connection to Max Shift Method

Graph Max Shift applied to $\mathcal{G}(\mathcal{Y}; \epsilon)$ will compute the same hill-climbing paths as the Max Shift method applied to $\mathcal{Y}$ with the flat kernel, and bandwidth and search radius ϵ.

Recall, the Max Shift path is given by

$$x(0) = x; \quad x(k+1) \in \underset{x \in \mathcal{Y} \cap \bar{B}(x(k), r)}{\operatorname{argmax}} \hat{f}_\epsilon(x), \quad k \geq 1$$

where, in this case

$$\hat{f}_\epsilon(x) = \frac{1}{n\epsilon^d} \sum_{i=1}^{n} K\left(\frac{x - y_i}{\epsilon}\right) = \frac{\#\{i : \|x - y_i\| \leq \epsilon\}}{v_d n \epsilon^d} = \frac{q_i}{v_d n \epsilon^d} \propto q_i$$

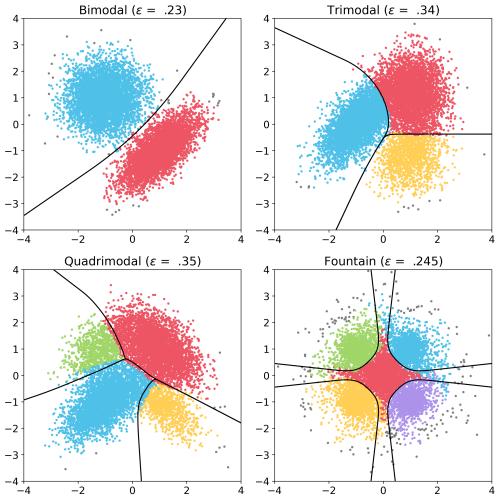
Note, the flat kernel $K(x) = v_d^{-1} \mathbb{I}(\|x\| \leq 1)$ is not differentiable, so the results of Arias-Castro and Qiao (2022) do not apply.

Main Theorem

Theorem

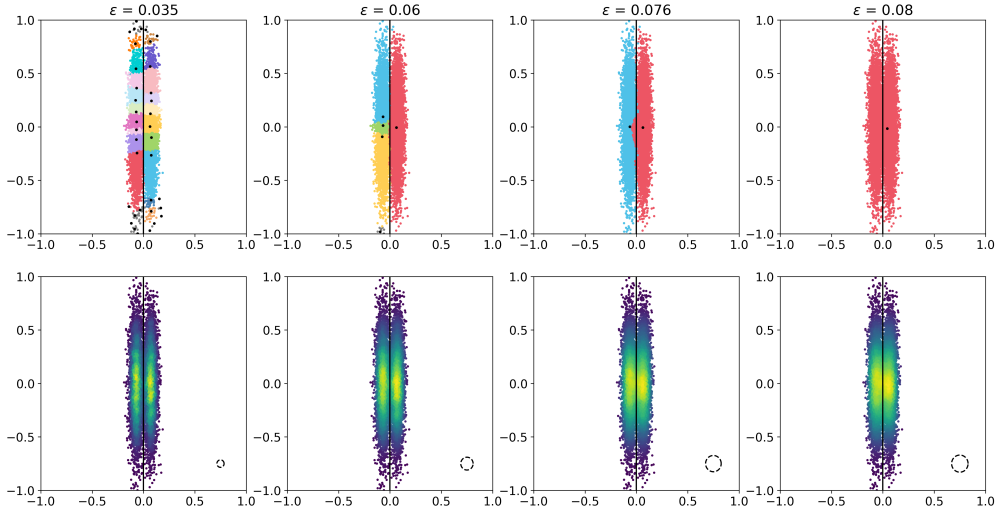
$\mathcal{Y}_n := \{y_1, \ldots, y_n\}$ is generated iid from a density f on $\mathbb{R}^d$ with compact support, twice continuously differentiable, and Morse. Then, with a large enough choice of tuning parameter τ, Graph Max Shift applied to $\mathcal{G}(\mathcal{Y}, \epsilon_n)$ is consistent when $\epsilon_n \to 0$, and $\epsilon_n^{\max\{d+4,2d\}} n / \log n \to \infty$.

Numerical Experiments: Consistency Demonstration



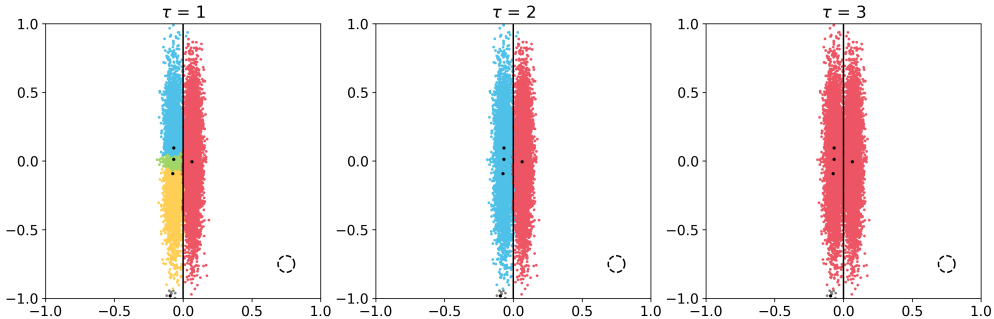
Graph Max Shift on various random geometric graphs with $n = 10^4$, where the underlying density f is a Gaussian mixture with different numbers of components. Mixtures are taken from Chacón (2015).

Numerical Experiments: Effect of ϵ



When ϵ is too small, the density estimator is not very smooth, which results in too many modes. When ϵ is too large, the density estimator is too smooth and points may 'cross' the basins of attraction.

Numerical Experiments: Effect of τ

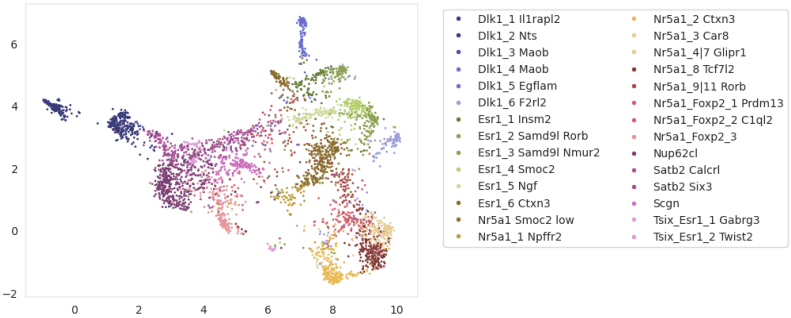


When τ is too small the left cluster is split into multiple clusters, but by increasing $\tau = 2$, the nodes in the left component are merged together. When τ is too large, clusters in different basins of attraction are merged together

Outline

Clustering and Embedding

- t-SNE (Van der Maaten and Hinton, 2008) and UMAP (McInnes et al., 2018) are popular dimensionality reduction methods that group the data points as they are embedded.
- In the biological sciences, it is common practice to cluster the data using some clustering method, embed the data using t-SNE or UMAP, and color the embedding according to the clustering.



UMAP applied to data from Chari and Pachter (2023).

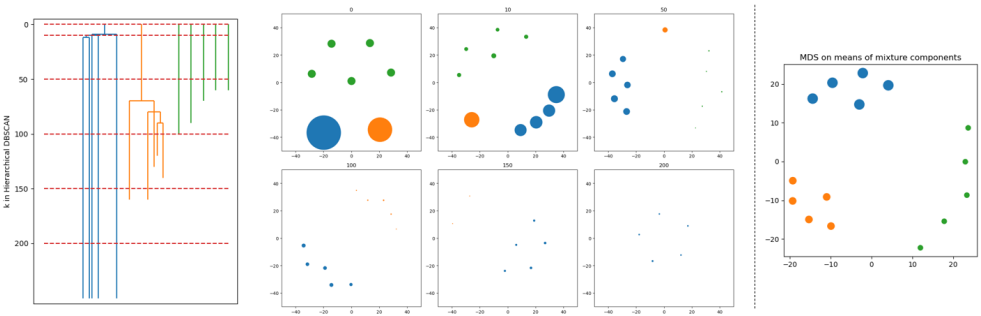
Clustering and Embedding

The embedding methods (UMAP and t-SNE) are somewhat complicated and difficult to understand

- There are minimal theoretical results on these methods, with the exceptions of Arora et al. (2018), Linderman and Steinerberger (2019), and Cai and Ma (2022)
- Recently, there has been some debate in the biological sciences about how UMAP and t-SNE should be used and interpreted (Chari and Pachter, 2023; Lause et al., 2024).

Clustering and Embedding

Can we propose a more theoretically sound, transparent method that clusters data and then embeds it?



The idea of clustering and then embedding was originally proposed by Shepard (1972).

References I

Ackerman, M. and Ben-David, S. (2009). Clusterability: A theoretical study. In *Artificial intelligence and statistics*, pages 1–8. PMLR.

Arias-Castro, E. and Qiao, W. (2022). Clustering by hill-climbing: consistency results. *arXiv preprint arXiv:2202.09023*.

Arias-Castro, E. and Qiao, W. (2023a). Moving up the cluster tree with the gradient flow. *SIAM Journal on Mathematics of Data Science*, 5(2):400–421.

Arias-Castro, E. and Qiao, W. (2023b). A unifying view of modal clustering. *Information and Inference: A Journal of the IMA*, 12(2):897–920.

Arora, S., Hu, W., and Kothari, P. K. (2018). An analysis of the t-sne algorithm for data visualization. In *Conference on learning theory*, pages 1455–1462. PMLR.

Balakrishnan, S., Narayanan, S., Rinaldo, A., Singh, A., and Wasserman, L. (2013). Cluster trees on manifolds. *Advances in Neural Information Processing Systems*, 26.

Blondel, V. D., Guillaume, J.-L., Lambiotte, R., and Lefebvre, E. (2008). Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008.

References II

Bouveyron, C., Celeux, G., Murphy, T. B., and Raftery, A. E. (2019). *Model-Based Clustering and Classification for Data Science: With Applications in R*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press.

Cai, T. T. and Ma, R. (2022). Theoretical foundations of t-sne for visualizing high-dimensional clustered data. *Journal of Machine Learning Research*, 23(301):1–54.

Carlsson, G. and Mémoli, F. (2010). Characterization, stability and convergence of hierarchical clustering methods. *Journal of Machine Learning Research*, 11(47):1425–1470.

Chacón, J. E. (2015). A population background for nonparametric density-based clustering. *Statistical Science*, 30(4):518 – 532.

Chari, T. and Pachter, L. (2023). The specious art of single-cell genomics. *PLOS Computational Biology*, 19(8):1–20.

Chaudhuri, K., Dasgupta, S., Kpotufe, S., and von Luxburg, U. (2014). Consistent procedures for cluster tree estimation and pruning. *IEEE Transactions on Information Theory*, 60(12):7900–7912.

References III

Cheng, Y. (1995). Mean shift, mode seeking, and clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(8):790–799.

Cohen-Addad, V., Kanade, V., and Mallmann-Trenn, F. (2018). Clustering redemption–beyond the impossibility of kleinberg's axioms. In *Advances in Neural Information Processing Systems*, volume 31.

Dasgupta, S. (2016). A cost function for similarity-based hierarchical clustering. In *Proceedings of the Forty-Eighth Annual ACM Symposium on Theory of Computing*, pages 118–127.

Eldridge, J., Belkin, M., and Wang, Y. (2015). Beyond hartigan consistency: Merge distortion metric for hierarchical clustering. In *Proceedings of The 28th Conference on Learning Theory*, volume 40, pages 588–606. PMLR.

Fraley, C. and Raftery, A. E. (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association*, 97(458):611–631.

Fukunaga, K. and Hostetler, L. (1975). The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Transactions on Information Theory*, 21(1):32–40.

References IV

Grosswendt, A. and Roeglin, H. (2017). Improved analysis of complete-linkage clustering. *Algorithmica*, 78:1131–1150.

Hartigan, J. (1975). *Clustering Algorithms*. Wiley.

Hartigan, J. (1977). Distribution problems in clustering. In *Classification and Clustering*, pages 45–71. Academic Press.

Hartigan, J. A. (1981). Consistency of single linkage for high-density clusters. *Journal of the American Statistical Association*, 76(374):388–394.

Jardine, C., Jardine, N., and Sibson, R. (1967). The structure and construction of taxonomic hierarchies. *Mathematical Biosciences*, 1(2):173–179.

Kim, J., Chen, Y.-C., Balakrishnan, S., Rinaldo, A., and Wasserman, L. (2016). Statistical inference for cluster trees. In *Advances in Neural Information Processing Systems*, volume 29.

Kleinberg, J. (2002). An impossibility theorem for clustering. *Advances in Neural Information Processing Systems*, 15.

Koontz, Narendra, and Fukunaga (1976). A graph-theoretic approach to nonparametric cluster analysis. *IEEE Transactions on Computers*, 100(9):936–944.

References V

Lause, J., Berens, P., and Kobak, D. (2024). The art of seeing the elephant in the room: 2d embeddings of single-cell data do make sense. *PLOS Computational Biology*, 20(10):1–5.

Linderman, G. C. and Steinerberger, S. (2019). Clustering with t-sne, provably. *SIAM journal on mathematics of data science*, 1(2):313–332.

McInnes, L., Healy, J., and Melville, J. (2018). Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*.

Penrose, M. (1995). Single linkage clustering and continuum percolation. *Journal of Multivariate Analysis*, 53(1):94–109.

Puzicha, J., Hofmann, T., and Buhmann, J. M. (2000). A theory of proximity based clustering: Structure detection by optimization. *Pattern Recognition*, 33(4):617–634.

Shepard, R. (1972). Psychological representation of speech sounds. *Human Communication: A unified view/McGraw-Hill*.

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

References VI

Tasic, B., Yao, Z., Graybuck, L. T., Smith, K. A., Nguyen, T. N., Bertagnolli, D., Goldy, J., Garren, E., Economo, M. N., Viswanathan, S., et al. (2018). Shared and distinct transcriptomic cell types across neocortical areas. *Nature*, 563(7729):72–78.

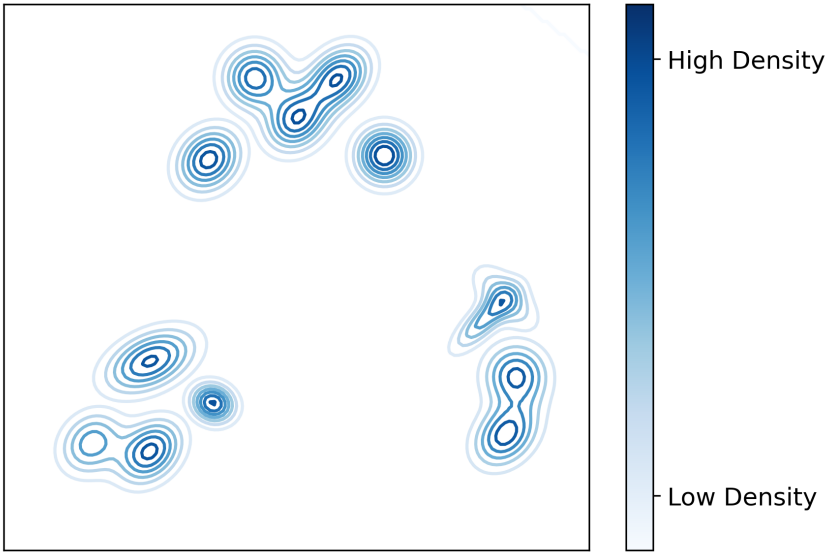
Traag, V. A., Waltman, L., and Van Eck, N. J. (2019). From louvain to leiden: guaranteeing well-connected communities. *Scientific reports*, 9(1):1–12.

Van der Maaten, L. and Hinton, G. (2008). Visualizing data using t-sne. *Journal of machine learning research*, 9(11).

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

Wang, D., Lu, X., and Rinaldo, A. (2019). Dbscan: Optimal rates for density-based cluster estimation. *Journal of Machine Learning Research*, 20(170):1–50.

Zadeh, R. B. and Ben-David, S. (2012). A uniqueness theorem for clustering. *arXiv preprint arXiv:1205.2600*.


High Density
Low Density

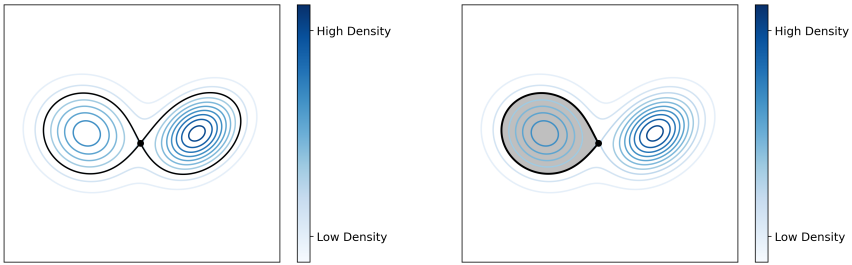
Merge distortion pseudometric

Example

Consider $f = \mathbb{I}_A$ where A has unit measure. Then any collection of subsets of A with a nested structure is a cluster tree for f, and the merge distortion distance between any pair of such cluster trees is zero.

Example

A cluster can be added at a 'split level' so that $\mathcal{C} = \mathcal{H}_f \cup \{R\}$ and $d_M(\mathcal{C}, \mathcal{H}_f) = 0$.



Merge distortion pseudometric

How different can a cluster tree $\mathcal{C}$ such that $d_M(\mathcal{C}, \mathcal{H}_f) = 0$ be from $\mathcal{H}_f$?

- $\mathcal{C}$ can be constructed by removing clusters from $\mathcal{H}_f$ or adding cluster trees on level sets.

$$\mathcal{C} = (\mathcal{H}_f \setminus \{C_i : i \in I\}) \cup \{\mathcal{S}_j : j \in J\}$$

  where $C_i \in \text{cc}(U_{\lambda_i})$ for some $\lambda_i > 0$ such that $\{\lambda_i : i \in I\}$ has empty interior and $\mathcal{S}_j$ is a cluster tree of $\mathcal{L}_{\lambda_j}$ for some $\lambda_j > 0$ such that $\{\lambda_j : j \in J\}$ are all distinct.
- If f is continuous and $\mathcal{C}$ is a closed cluster tree, then $\mathcal{C}$ contains $\mathcal{H}_f$
- If f is continuous with bounded upper level sets and locally finitely many modes and $\mathcal{C}$ is a closed cluster tree, then for every $C \in \mathcal{C}$, $\{f > \inf_{z \in C} f(z)\} \cap C$ is some union of connected components of $\{f > \inf_{z \in C} f(z)\}$.

Hartigan Consistency

Definition (Hartigan Consistency)

Let $\mathcal{H}_f$ be the population cluster tree and let $\hat{\mathcal{C}}_n$ be an estimator of $\mathcal{H}_f$. For $A \in \mathcal{C}_f$, let A_n be the smallest cluster in $\hat{\mathcal{C}}_n$ containing all data points in A. The estimator is Hartigan consistent if for all disjoint clusters $A, B \in \mathcal{H}_f$,

$$\lim_{n \to \infty} \mathbb{P}(A_n \cap B_n = \emptyset) = 1$$

This is a weak notion of consistency.

Clustering and Embedding

While the Louvain and Leiden algorithms (Blondel et al., 2008; Traag et al., 2019) are popular choices for clustering, the exact clustering methods can vary and can seem somewhat arbitrary or unclear:

"In brief, all quality control qualified cells were grouped into very broad categories using known markers, then clustered using high variance gene selection, dimensionality reduction, dimension filtering, and Jaccard–Louvain or hierarchical (Ward) clustering. This process was repeated within each resulting cluster until no more child clusters met differential gene expression or cluster size termination criteria. The entire clustering procedure was repeated 100 times using 80% of all cells sampled at random, and the frequency with which cells co-cluster was used to generate a final set of clusters, again subject to differential gene expression or cluster size termination criteria." (Tasic et al., 2018)

Hierarchical DBSCAN

Algorithm 1 The DBSCAN algorithm.

INPUT: i.i.d sample $\{X_i\}_{i=1}^n$, and $h > 0$.

1. For each $k \in \mathbb{N}$, construct a graph $\mathbb{G}_{h,k}$ with nodes $\{X_i : |B(X_i, h) \cap \{X_j\}_{j=1}^n| \geq k\}$ and edges (X_i, X_j) if $\|X_i - X_j\| < 2h$.
2. Compute $\mathbb{C}(h, k)$, the graphical connected components of $\mathbb{G}_{h,k}$.

OUTPUT: $\{\mathbb{C}(h, k), k \in \mathbb{N}\}$.

Algorithm 2 The modified DBSCAN

INPUT: i.i.d sample $\{X_i\}_{i=1}^n$, a α-valid kernel K and $h > 0$.

1. Compute $\{\widehat{p}_h(X_i), i = 1, \ldots, n\}$.
2. For each $\lambda \geq 0$, construct a graph $\mathbb{G}_{h,\lambda}$ with node set

$$\widehat{D}(\lambda) = \{X_i : \widehat{p}_h(X_i) \geq \lambda\}$$

and edge set $\{(X_i, X_j) : X_i, X_j \in \widehat{D}(\lambda) \text{ and } \|X_i - X_j\| < 2h\}$.

3. Compute $\mathbb{C}(h, \lambda)$, the graphical connected components of $\mathbb{G}_{h,\lambda}$.

OUTPUT: $\widehat{T}_n = \{\mathbb{C}(h, \lambda), \lambda \geq 0\}$.

Algorithms from Wang et al. (2019).

Robust Single Linkage and kNN Estimator

Algorithm 1

1. For each x_i set $r_k(x_i) = \min\{r : B(x_i, r) \text{ contains } k \text{ data points}\}$.
2. As r grows from 0 to ∞:
   (a) Construct a graph G_r with nodes $\{x_i : r_k(x_i) \leq r\}$.
   Include edge (x_i, x_j) if $\|x_i - x_j\| \leq \alpha r$.
   (b) Let $\mathbb{C}_n(r)$ be the connected components of G_r.

Algorithm 2

1. For each x_i set $r_k(x_i) = \min\{r : B(x_i, r) \text{ contains } k \text{ data points}\}$.
2. As r grows from 0 to ∞:
   (a) Construct a graph G_r^{NN} with nodes $\{x_i : r_k(x_i) \leq r\}$.
   Include edge (x_i, x_j) if:

$$\begin{array}{ll} \|x_i - x_j\| \leq \alpha \max(r_k(x_i), r_k(x_j)) & k\text{-NN graph} \\ \|x_i - x_j\| \leq \alpha \min(r_k(x_i), r_k(x_j)) & \text{mutual } k\text{-NN graph} \end{array}$$

   (b) Let $\mathbb{C}_n(r)$ be the connected components of G_r^{NN}.

Algorithms from Chaudhuri et al. (2014).