Density Clustering

김지수 (Jisu KIM)

통계이론세미나 - 위상구조의 통계적 추정, 2023 가을학기

The lecture note is largely based on Larry Wasserman's lecture notes.

In a clustering problem we aim to find groups in the data. Unlike classification, the data are not labeled, and so clustering is an example of *unsupervised learning*. The connection between clustering and topology is clearest if we focus on density-based methods for clustering.

Example. Figures 1 and 2 show some synthetic examples where the clusters are meant to be intuitively clear. In Figure 1 there are two blob-like clusters. Identifying clusters like this is easy. Figure 2 shows four clusters: a blob, two rings and a half ring. Identifying clusters with unusual shapes like this is not quite as easy. In fact, finding clusters of this type requires nonparametric methods.

Density-Based Clustering I: Modes

Let $p : \mathbb{R}^d \to \mathbb{R}$ be a density of X. Recall that a point $x \in \mathbb{R}^d$ is a critical point of p if its gradient ∇p is zero at x, i.e., $\nabla p(x) = 0$. The Hessian $H_x(p)$ is the derivative of the gradient ∇p at x.

Definition ([1, Definition 3.1])**.** p is a *Morse function* if its critical points are all non-degenerate, that is, the determinant of the Hessian $H_x(p)$ is nonzero for all critical points x.

Assume that p has modes $m_1, \ldots, m_{k_0}$ and that p is a Morse function. For Morse function, m is a mode of p if and only if $\nabla p(m) = 0$ and all eigenvalues of $H_m(p)$ are negative. We can use the modes to define clusters as follows.

Mode Clustering

Given any point $x \in \mathbb{R}^d$, there is a unique gradient ascent path, or integral curve, passing through x that eventually leads to one of the modes. We define the clusters to be the "basins of attraction" of the modes, the equivalence classes of points whose ascent paths lead to the same mode.

Definition ([1, Definition 3.17])**.** An *integral curve* through x is a path $\pi_x : \mathbb{R} \to \mathbb{R}^d$ such that $\pi_x(0) = x$ and

$$\frac{d}{dt}\pi_x(t) = \nabla p(\pi_x(t)). \tag{1}$$

Integral curves never intersect (except at stationary points) and they partition the space. Equation (1) means that the path π follows the direction of steepest ascent of p through x.

Definition. The destination of the integral curve π through a (non-mode) point x is defined by

$$\mathrm{dest}(x) = \lim_{t\to\infty} \pi_x(t). \tag{2}$$

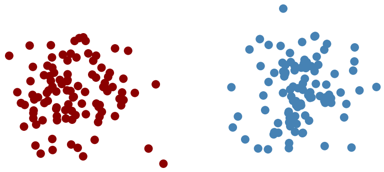


Figure 1: A synthetic example with two "blob-like" clusters.

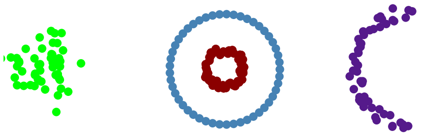


Figure 2: A synthetic example with four clusters with a variety of different shapes.

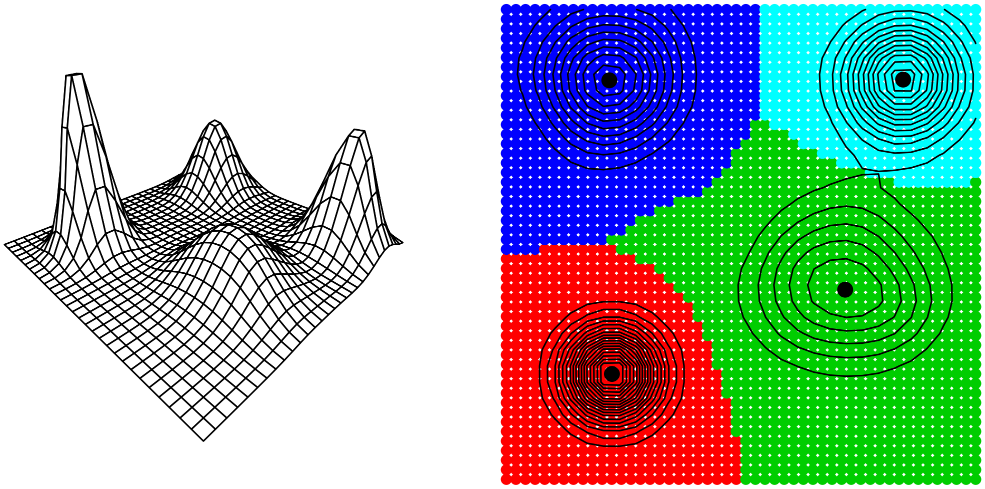


Figure 3: The left plot shows a function with four modes. The right plot shows the ascending manifolds (basins of attraction) corresponding to the four modes.

It can then be shown that [1, Proposition 3.19] for all x, $\text{dest}(x) = m_j$ for some mode m_j. That is: all integral curves lead to modes.

Definition ([1, Definition 4.1])**.** For each mode m_j, define the sets

$$\mathcal{A}_j = \Big\{ x : \ \text{dest}(x) = m_j \Big\}. \tag{3}$$

These sets are known as the *ascending manifolds*, and also known as the cluster associated with m_j, or the basin of attraction of m_j.

The $\mathcal{A}_j$'s partition the space. See Figure 3. The collection of ascending manifolds $\{\mathcal{A}_1, \ldots, \mathcal{A}_{k_0}\}$ is called the *Morse complex.*

Given data $X_1, \ldots, X_n$ we construct an estimate $\hat{p}$ of the density. Let $\hat{m}_1, \ldots, \hat{m}_k$ be the estimated modes and let $\hat{\mathcal{A}}_1, \ldots, \hat{\mathcal{A}}_k$ be the corresponding ascending manifolds derived from $\hat{p}$. The sample clusters $C_1, \ldots, C_k$ are defined to be $C_j = \Big\{ X_i : \ X_i \in \hat{\mathcal{A}}_j \Big\}$.

Recall that the kernel density estimator is

$$\hat{p}(x) \equiv \hat{p}_h(x) = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{h^d} K\left(\frac{\|x - X_i\|_2}{h} \right) \tag{4}$$

where K is a smooth, symmetric kernel and $h > 0$ is the bandwidth.[1] The mean of the estimator is

$$p_h(x) = \mathbb{E}[\hat{p}_h(x)] = \int K(t) p(x + th) dt. \tag{5}$$

To locate the modes of $\hat{p}_h$ we use the *mean shift algorithm* [8, 9] which finds modes by approximating the steepest ascent paths. Note that starting from $a^{(0)}$, the gradient ascent algorithm for a function f finds the next point as

$$a^{(n+1)} = a^{(n)} + \nabla f(a^{(n)}).$$

[1] In general, we can use a bandwidth matrix H in the estimator, with $\hat{p}(x) \equiv \hat{p}_H(x) = \frac{1}{n}\sum_{i=1}^{n} K_H(x - X_i)$ where $K_H(x) = |H|^{-\frac{1}{2}} K(H^{-\frac{1}{2}} x)$.

Mean Shift Algorithm

1. Input: $\hat{p}(x)$ and a mesh of points $A = \{a_1, \ldots, a_N\}$ (often taken to be the data points).
2. For each mesh point a_j, set $a_j^{(0)} = a_j$ and iterate the following equation until convergence:

$$a_j^{(s+1)} \longleftarrow \frac{\sum_{i=1}^n X_i K\left(\frac{||a_j^{(s)} - X_i||}{h}\right)}{\sum_{i=1}^n K\left(\frac{||a_j^{(s)} - X_i||}{h}\right)}.$$

   Let $\hat{\mathcal{M}}$ be the unique values of the set $\{a_1^{(\infty)}, \ldots, a_N^{(\infty)}\}$.

3. Output: $\hat{\mathcal{M}}$.

Figure 4: *The Mean Shift Algorithm.*

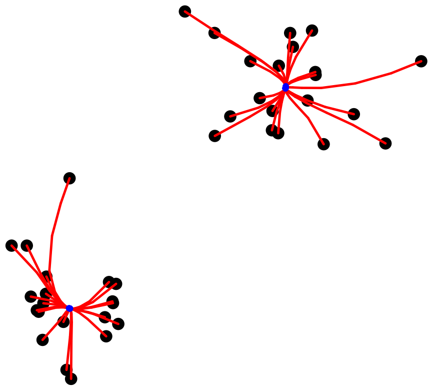


Figure 5: A simple example of the mean shift algorithm.

By applying

$$\begin{aligned}\nabla \hat{p}(a) &= \frac{1}{n}\sum_{i=1}^n \frac{1}{h^d} K'\left(\frac{\|a - X_i\|_2}{h}\right)\frac{a - X_i}{h\,\|a - X_i\|_2} \\ &= \frac{1}{n}\sum_{i=1}^n \frac{1}{h^d} K\left(\frac{\|a - X_i\|_2}{h}\right)\frac{a - X_i}{h^2}.\end{aligned}$$

The algorithm is given in Figure 4. The result of this process is the set of estimated modes $\hat{\mathcal{M}} = \{\hat{m}_1, \ldots, \hat{m}_k\}$. We also get the clustering for free: the mean shift algorithm shows us what mode each point is attracted to. See Figure 5.

A modified version of the algorithm is the blurred mean-shift algorithm [2]. Here, we use the data as the mesh and we replace the data with the mean-shifted data at each step. This converges very quickly but must be stopped before everything converges to a single point; see Figures 6 and 7.

What we are doing is tracing out the *gradient flow*. The flow lines lead to the modes and they define the clusters. In general, a flow is a map $\phi : \mathbb{R}^d \times \mathbb{R} \to \mathbb{R}^d$ such that $\phi(x, 0) = x$ and $\phi(\phi(x, t), s) = \phi(x, s + t)$. The latter is called the semi-group property.

Choosing the Bandwidth

As usual, choosing a good bandwidth is crucial. You might wonder if increasing the bandwidth, decreases the number of modes. [12] showed that the answer is yes if you use a Normal kernel.

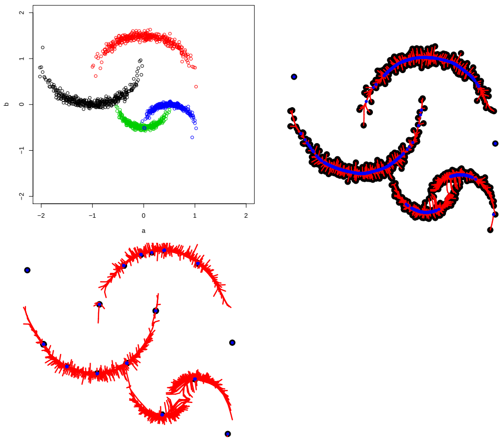


Figure 6: The crescent data example. Top left: data. Top right: a few steps of mean-shift. Bottom left: a few steps of blurred mean-shift.

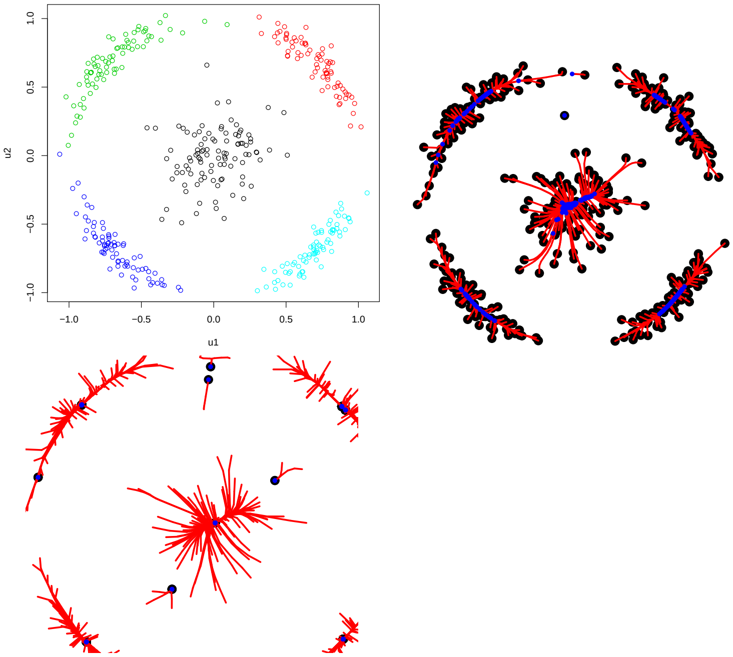


Figure 7: The Broken Ring example. Top left: data. Top right: a few steps of mean-shift. Bottom left: a few steps of blurred mean-shift.

Theorem ([12, Theorem])**.** *Let $\hat{p}_h$ be a kernel density estimator using a Gaussian kernel in one dimension. Then the number of modes of $\hat{p}_h$ is a non-increasing function of h. The Gaussian kernel is the unique kernel with this property.*

We still need a way to pick h. We can use cross-validation as before. One could argue that we should choose h so that we estimate the gradient $g(x) = \nabla p(x)$ well since the clustering is based on the gradient flow.

How can we estimate the loss of the gradient? Consider, first the scalar case. Note that

$$\int (\hat{p}' - p')^2 = \int (\hat{p}')^2 - 2\int \hat{p}p' + \int (p')^2.$$

We can ignore the last term. The first term is known. To estimate the middle term, we use integration by parts to get

$$\int \hat{p}p' = -\int p'' p$$

suggesting the cross-validation estimator

$$\int (\hat{p}'(x))^2 dx + \frac{2}{n}\sum_i \hat{p}_i''(X_i)$$

where $\hat{p}_i''$ is the leave-one-out second derivative. More generally, by repeated integration by parts, we can estimate the loss for the r^{th} derivative by

$$\text{CV}_r(h) = \int (\hat{p}^{(r)}(x))^2 dx - \frac{2}{n}(-1)^r \sum_i \hat{p}_i^{(2r)}(X_i).$$

Let's now discuss estimating derivatives more generally following [3]. Let

$$\hat{p}_H(x) = \frac{1}{n}\sum_{i=1}^n K_H(x - X_i)$$

where $K_H(x) = |H|^{-1/2}K(H^{-1/2}x)$. Let $D = \partial/\partial x = (\partial/\partial x_1, \ldots, \partial/\partial x_d)$ be the gradient operator. Let $H(x)$ be the Hessian of $p(x)$ whose entries are $\partial^2 p/(\partial x_j \partial x_k)$. Let

$$D^{\otimes r}p = (Dp)^{\otimes r} = \partial^r p/\partial x^{\otimes r} \in \mathbb{R}^{d^r}$$

denote the r^{th} derivatives, organized into a vector. Thus

$$D^{\otimes 0}p = p, \qquad D^{\otimes 1}p = Dp, \qquad D^{\otimes 2}p = \text{vec}(H)$$

where vec takes a matrix and stacks the columns into a vector.

The estimate of $D^{\otimes r}p$ is

$$\hat{p}^{(r)}(x) = D^{\otimes r}\hat{p}_H(x) = \frac{1}{n}\sum_{i=1}^n D^{\otimes r}K_H(x - X_i) = \frac{1}{n}\sum_{i=1}^n |H|^{-1/2}(H^{-1/2})^{\otimes r}D^{\otimes r}K(H^{-1/2}(x - X_i).$$

The integrated squared error is

$$L = \int ||D^{\otimes r}\hat{p}_H(x) - D^{\otimes r}p(x)||^2 dx.$$

[4] shows that $\mathbb{E}[L]$ is minimized by choosing H so that each entry has order $n^{-2/(d+2r+4)}$ leading to a risk of order $O(n^{-4/(d+2r+4)})$. In fact, it may be shown that

$$\begin{aligned}\mathbb{E}[L] = &\frac{1}{n}|H|^{-1/2}\text{tr}((H^{-1})^{\otimes r}R(D^{\otimes r}K)) - \frac{1}{n}\text{tr}R^*(K_H \star K_H, D^{\otimes r}p)\\ &+ \text{tr}R^*(K_H \star K_H, D^{\otimes r}p) - 2\text{tr}R^*(K_H, D^{\otimes r}p) + \text{tr}R(D^{\otimes r}p)\end{aligned}$$

where

$$R(g) = \int g(x)g^T(x)dx$$

$$R^*(a, g) = \int (a \star g)(x)g^T(x)dx$$

and $(a \star g)$ is componentwise convolution.

To estimate the loss, we expand L as

$$L = \int ||D^{\otimes r} \hat{p}_H(x)||^2 dx - 2\int \langle D^{\otimes r} \hat{p}_H(x), D^{\otimes r} p(x)\rangle dx + \text{constant}.$$

Using some high-voltage calculations, Chacon and Duong (2013) derived the following leave-one-out approximation to the first two terms:

$$\text{CV}_r(H) = (-1)^r |H|^{-1/2} (\text{vec}(H^{-1})^{\otimes r})^T B(H)$$

where

$$B(H) = \frac{1}{n^2}\sum_{i,j} D^{\otimes 2r}\overline{K}(H^{-1/2}(X_i - X_j)) - \frac{2}{n(n-1)}\sum_{i \neq j} D^{\otimes 2r} K(H^{-1/2}(X_i - X_j))$$

and $\overline{K} = K \star K$ In practice, the minimization is easy if we restrict to matrices of the form $H = h^2 I$.

A better idea is to used fixed (non-decreasing h). We don't need h to go to 0 to find the clusters. More on this when we discuss persistence.

Theoretical Analysis

How well can we estimate the modes?

Theorem. *[7, Theorem 1] Assume that p is Morse with finitely many modes $m_1, \ldots, m_k$. Then for $h > 0$ and not too large, p_h is Morse with modes $m_{h1}, \ldots, m_{hk}$ and (possibly after relabelling),*

$$\max_j ||m_j - m_{jh}|| = O(h^2).$$

With probability tending to 1, $\hat{p}_h$ has the same number of modes which we denote by $\hat{m}_{h1}, \ldots, \hat{m}_{hk}$. Furthermore,

$$\max_j ||\hat{m}_{jh} - m_{jh}|| = O_P\left(\sqrt{\frac{1}{nh^{d+2}}}\right)$$

and

$$\max_j ||\hat{m}_{jh} - m_j|| = O(h^2) + O_P\left(\sqrt{\frac{1}{nh^{d+2}}}\right).$$

Remark: Setting $h \asymp n^{-1/(d+6)}$ gives the rate $n^{-2/(d+6)}$ which is minimax (Tsyabkov 1990) under smoothness assumptions. See also Romano (1988). However, if we take the fixed h point if view, then we have a $n^{-1/2}$ rate.

Proof Outline. Build a small ball B_j around each m_{jh}. We will skip the first step, which is to show that there is one (and only one) local mode in B_j. Let's focus on showing

$$\max_j ||\hat{m}_{jh} - m_{jh}|| = O_P\left(\sqrt{\frac{1}{nh^{d+2}}}\right).$$

For simplicity, write $m = m_{jh}$ and $x = \hat{m}_{jh}$. Let $g(x)$ and $H(x)$ be the gradient and Hessian of $p_h(x)$ and let $\hat{g}(x)$ and $\hat{H}(x)$ be the gradient Hessian of $\hat{p}_h(x)$. Then

$$(0, \ldots, 0)^T = \hat{g}(x) = \hat{g}(m) + (x - m)^T \int_0^1 \hat{H}(m + u(x - m))du$$

and so

$$(x - m)^T \int_0^1 \hat{H}(m + u(x - m))du = (g(m) - \hat{g}(m))$$

where we used the fact that $\mathbf{0} = g(m)$. Multiplying on the right by $x - m$ we have

$$(x - m)^T \int_0^1 \hat{H}(m + u(x - m))(x - m)du = (\hat{g}(m) - \hat{g}(m))^T (x - m).$$

Let $\lambda = \inf_{0\leq u\leq 1} \lambda_{\min}(H(m+u(x-m)))$. Then $\lambda = \lambda_{\min}(H(m)) + o_P(1)$ and

$$(x-m)^T \int_0^1 \hat{H}(x+u(m-x))(x-m)du \geq \lambda ||x-m||^2.$$

Hence, using Cauchy-Schwartz,

$$\lambda ||x-m||^2 \leq ||\hat{g}(m) - g(m)||\ ||x-m|| \leq ||x-m|| \sup_y ||\hat{g}(y) - \hat{g}(y)|| \leq ||x-m|| O_P\left(\sqrt{\frac{1}{nh^{d+2}}}\right)$$

and so $||x-m|| = O_P\left(\sqrt{\frac{1}{nh^{d+2}}}\right)$. □

Remark: If we treat h as fixed (not decreasing) then the rate is $O_P(\sqrt{1/n})$ independent of dimension.

Density-Based Clustering II: Level Set Clustering

Let p be the density if the data. Let $L_t = \{x :\ p_h(x) > t\}$ denote an upper level set of p. Suppose that L_t can be decomposed into finitely many disjoint sets: $L_t = C_1 \bigcup \cdots \bigcup C_{k_t}$. We call $\mathcal{C}_t = \{C_1, \ldots, C_{k_t}\}$ the level set clusters at level t.

Let $\mathcal{C} = \bigcup_{t\geq 0} \mathcal{C}_t$. The clusters in $\mathcal{C}$ form a tree: if $A, B \in \mathcal{C}$, the either (i) $A \subset B$ or (ii) $B \subset A$ or (iii) $A \cap B = \emptyset$. We call $\mathcal{C}$ the *level set cluster tree.*

The level sets can be estimated in the obvious way: $\hat{L}_t = \{x :\ \hat{p}_h(x) > t\}$. How do we decompose $\hat{L}_t$ into its connected components? This can be done as follows. For each t let

$$\mathcal{X}_t = \{X_i :\ \hat{p}_h(X_i) > t\}.$$

Now construct a graph G_t where each $X_i \in \mathcal{X}_t$ is a vertex and there is an edge between X_i and X_j if and only if $||X_i - X_j|| \leq \epsilon$ where $\epsilon > 0$ is a tuning parameter. Bobrowski et al (2104) show that we can take $\epsilon = h$. G_t is a called a Rips graphs. The clusters at level t are estimated by taking the connected components of the graph G_t. In summary:

- Compute $\hat{p}_h$.
- For each t, let $\mathcal{X}_t = \{X_i :\ \hat{p}_h(X_i) > t\}$.
- Form a graph G_t for the points in $\mathcal{X}_t$ by connecting X_i and X_j if $||X_i - X_j|| \leq h$.
- The clusters at level t are the connected components of G_t.

A Python package, called DeBaCl, written by Brian Kent, can be found at
`http://www.brianpkent.com/projects.html`.

Fabrizio Lecci has written an R implementation, include in his R package: TDA (topological data analysis). You can get it at:
`http://cran.r-project.org/web/packages/TDA/index.html`

Two examples are shown in Figures 8 and 9.

Theory

How well does this work? Define the Hausdorff distance between two sets by

$$H(U,V) = \inf\left\{\epsilon :\ U \subset V \oplus \epsilon \ \text{ and } \ V \subset U \oplus \epsilon\right\}$$

where

$$V \oplus \epsilon = \bigcup_{x\in V} B(x,\epsilon)$$

and $B(x,\epsilon)$ denotes a ball of radius ϵ centered at x. We would like to say that L_t and $\hat{L}_t$ are close. In general this is not true. Sometimes L_t and $L_{t+\delta}$ are drastically different even for small δ. (Think of the case where a mode has height t.) But we can estimate stable level sets. Let us say that L_t is stable if there exists $a > 0$ and $C > 0$ such that, for all $\delta < a$,

$$H(L_{t-\delta}, L_{t+\delta}) \leq C\delta.$$

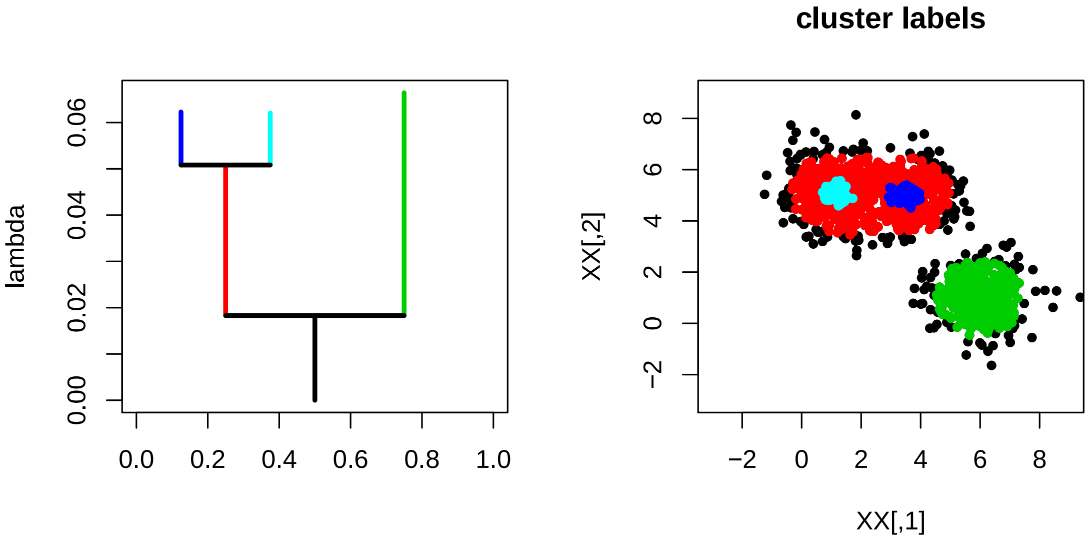


Figure 8: DeBaClR in two dimensions.

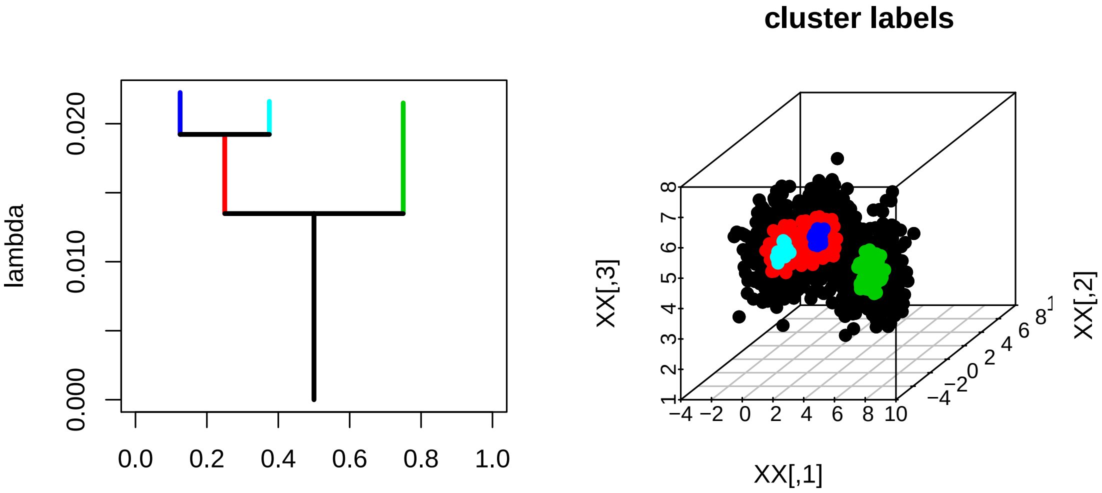


Figure 9: DeBaClR in three dimensions.

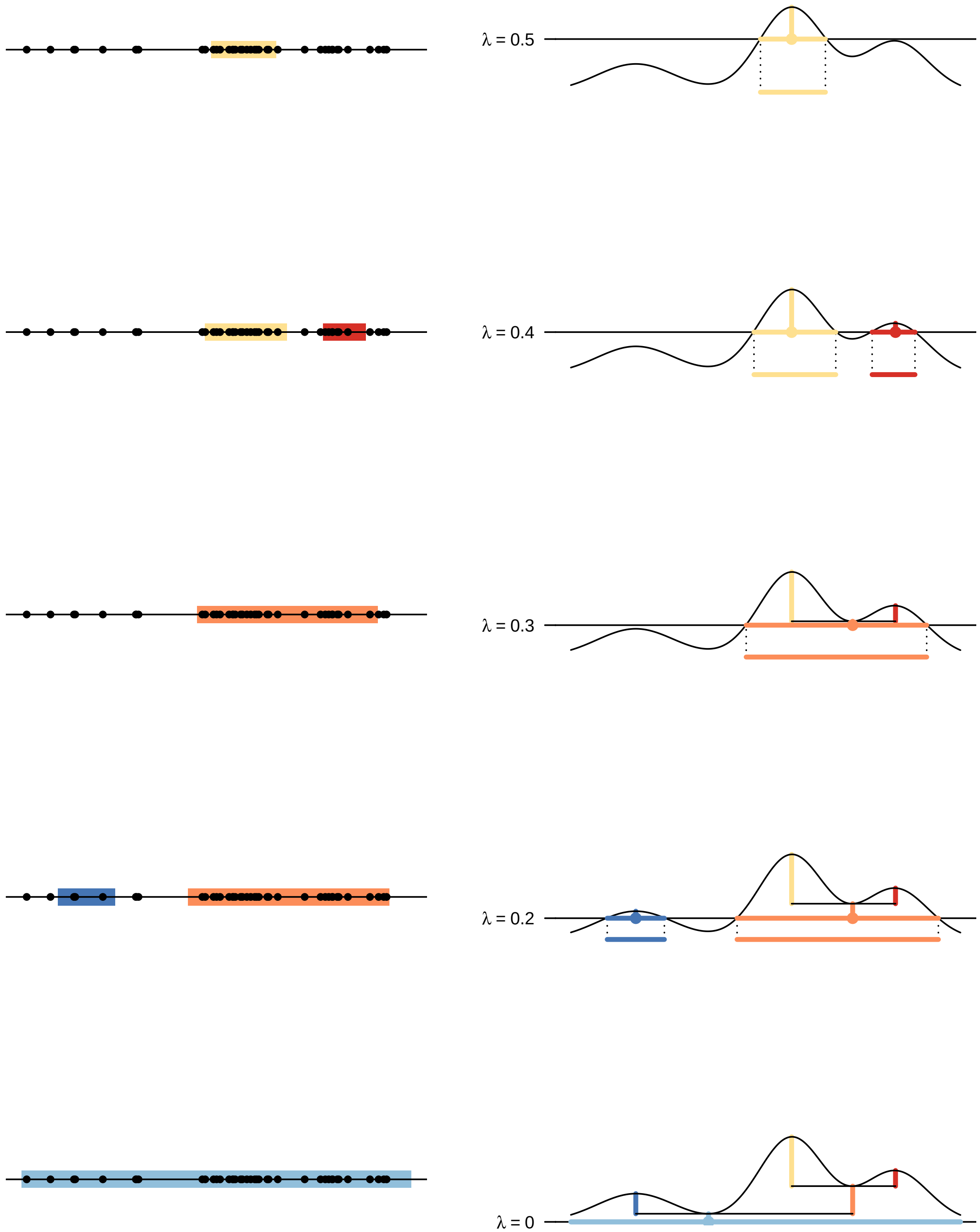
Theorem. *Suppose that L_t is stable. Then $H(\hat{L}_t, L_t) = O_P(\sqrt{\log n/(nh^d)})$.*

Proof. Let $r_n = \sqrt{\log n/(nh^d)})$. We need to show two things: (i) for every $x \in L_t$ there exists $y \in \hat{L}_t$ such that $||x - y|| = O_P(r_n)$ and (ii) for every $x \in \hat{L}_t$ there exists $y \in L_t$ such that $||x - y|| = O_P(r_n)$. First, we note that, by earlier results, $||\hat{p}_h - p_h||_\infty = O_P(r_n)$. To show (i), suppose that $x \in L_t$. By the stability assumption, there exists $y \in L_{t+r_n}$ such that $||x - y|| \leq Cr_n$. Then $p_h(y) > t + r_n$ which implies that $\hat{p}_h(y) > t$ and so $y \in \hat{L}_t$. To show (ii), let $x \in \hat{L}_t$ so that $\hat{p}_h(x) > t$. Thus $p_h(x) > t - r_n$. By stability, there is a $y \in L_t$ such that $||x - y|| \leq Cr_n$. □

Cluster Tree with multi-scale

Another way to measure the consistency of the cluster tree is through multi-scale approach, that is, we look at the connected components $\mathcal{C}_t = \{C_1, \ldots, C_{k_t}\}$ of a level set L_t for different values $t \in [0, \infty)$ simultaneously.

Definition. [11, Definition 1] For a density function p, its cluster tree $T_p : \mathbb{R} \to \mathcal{P}(\mathcal{X})$ is a function where $T_p(\lambda)$ is the set of connected components of the upper level set $\{x \in \mathcal{X} : p(x) \geq \lambda\}$.



For measuring a distance between trees that reflects multi-scale structure, we use l_∞ metric.

Definition. [11] The l_∞ metric between trees are defined as $d_\infty(T_p, T_q) = \|p - q\|_\infty$.

With this metric, we make a confidence set for the cluster tree. Recall that an asymptotic $1-\alpha$ confidence set $\hat{C}_\alpha$ is a collection of trees with the property that

$$P(T_p \in \hat{C}_\alpha) = 1 - \alpha + o(1).$$

We let $T_{\hat{p}_h}$ be the cluster tree from the kernel density estimator $\hat{p}_h$, where

$$\hat{p}_h(x) = \frac{1}{nh^d} \sum_{i=1}^{n} K\left(\frac{x - X_i}{h}\right),$$

and the confidence set as the ball centered at $T_{\hat{p}_h}$ and radius ϵ_α, i.e.

$$\hat{C}_\alpha = \left\{T : \, d_\infty(T, T_{\hat{p}_h}) \le \epsilon_\alpha\right\}, \tag{6}$$

where ϵ_α is the bootstrap quantile defined by

$$\epsilon_\alpha = \inf\left\{z : \, \frac{1}{B}\sum_{b=1}^{B} I\Big(||\hat{p}_h^{*b} - \hat{p}_h||_\infty > z\Big) \le \alpha\right\}. \tag{7}$$

Here, $\hat{p}_h^{*b}$ is the density estimator based on the b^{th} bootstrap sample.

Theorem. *[11, Theorem 3] Under minor conditions on the kernel, above confidence set $\hat{C}_\alpha$ in (6) satisfies*

$$P\left(T_h \in \hat{C}_\alpha\right) = 1 - \alpha + O\left(\left(\frac{\log^7 n}{nh^d}\right)^{1/6}\right).$$

Persistence

Consider a smooth density p with $M = \sup_x p(x) < \infty$. The t-level set clusters are the connected components of the set $L_t = \{x : \; p(x) \ge t\}$. Suppose we find the upper level sets $L_t = \{x : \; p(x) \ge t\}$ as we vary t from M to 0. *Persistent homology* measures how the topology of L_t varies as we decrease t. In our case, we are only interested in the modes, which correspond to the zeroth order homology. (Higher order homology refers to holes, tunnels etc.) The idea of using persistence to study clustering was introduced by [6].

Imagine setting $t = M$ and then gradually decreasing t. Whenever we hit a mode, a new level set cluster is born. As we decrease t further, some clusters may merge and we say that one of the clusters (the one born most recently) has died. See Figure 10.

In summary, each mode m_j has a death time and a birth time denoted by (d_j, b_j). (Note that the birth time is larger than the death time because we start at high density and move to lower density.) The modes can be summarized with a persistence diagram where we plot the points $(d_1, b_1), \ldots, (d_k, b_k)$ in the plane. See Figure 10. Points near the diagonal correspond to modes with short lifetimes. We might kill modes with lifetimes smaller than ϵ_α in (7). This corresponds to killing a mode if it is in a $2\epsilon_\alpha$ band around the diagonal. See [10]. Note that the starting and ending points of the vertical bars on the level set tree are precisely the coordinates of the persistence diagram. (A more precise bootstrap approach was introduced in [5].)

References

[1] Augustin Banyaga and David Hurtubise. *Lectures on Morse homology*, volume 29 of *Kluwer Texts in the Mathematical Sciences*. Kluwer Academic Publishers Group, Dordrecht, 2004.

[2] Miguel Á. Carreira-Perpiñán. Fast nonparametric clustering with gaussian blurring mean-shift. In William W. Cohen and Andrew W. Moore, editors, *Machine Learning, Proceedings of the Twenty-Third International Conference (ICML 2006), Pittsburgh, Pennsylvania, USA, June 25-29, 2006*, volume 148 of *ACM International Conference Proceeding Series*, pages 153–160. ACM, 2006.

[3] José E. Chacón and Tarn Duong. Unconstrained pilot selectors for smoothed cross-validation. *Aust. N. Z. J. Stat.*, 53(3):331–351, 2011.

[4] José E. Chacón, Tarn Duong, and M. P. Wand. Asymptotics for general multivariate kernel density derivative estimators. *Statist. Sinica*, 21(2):807–840, 2011.

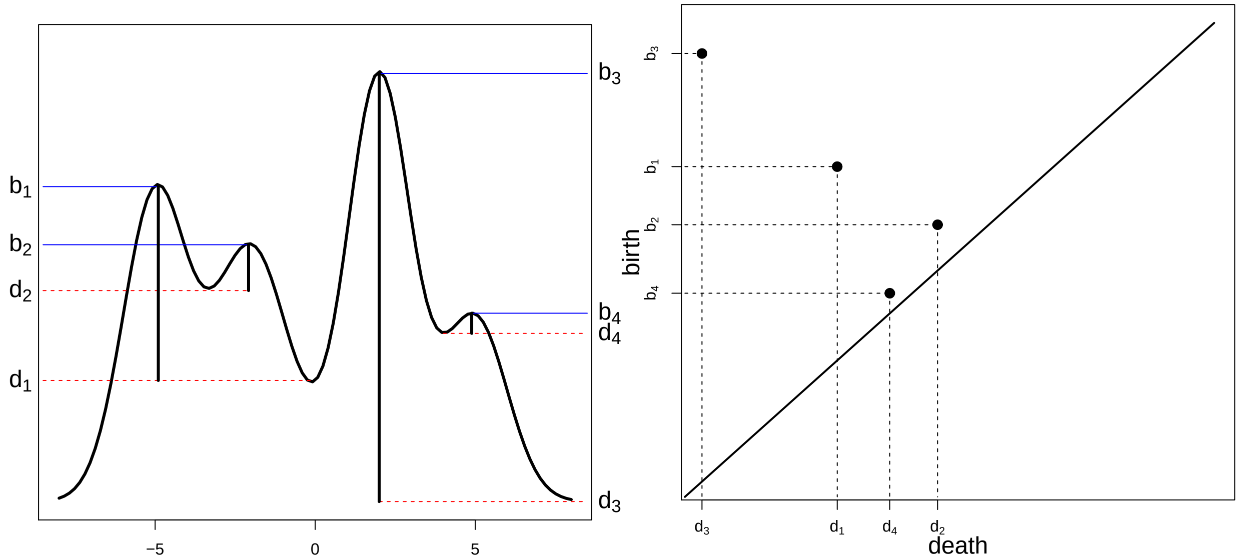


Figure 10: *Starting at the top of the density and moving down, each mode has a birth time b and a death time d. The persistence diagram (right) plots the points $(d_1, b_1), \ldots, (d_4, b_4)$. Modes with a long lifetime are far from the diagonal.*

[5] Frédéric Chazal, Brittany Fasy, Fabrizio Lecci, Bertrand Michel, Alessandro Rinaldo, and Larry Wasserman. Robust topological inference: distance to a measure and kernel distance. *J. Mach. Learn. Res.*, 18:Paper No. 159, 40, 2017.

[6] Frédéric Chazal, Leonidas J. Guibas, Steve Y. Oudot, and Primoz Skraba. Persistence-based clustering in Riemannian manifolds. *J. ACM*, 60(6):Art. 41, 38, 2013.

[7] Yen-Chi Chen, Christopher R. Genovese, and Larry Wasserman. A comprehensive approach to mode clustering. *Electron. J. Stat.*, 10(1):210–241, 2016.

[8] Yizong Cheng. Mean shift, mode seeking, and clustering. *IEEE Trans. Pattern Anal. Mach. Intell.*, 17(8):790–799, 1995.

[9] Dorin Comaniciu and Peter Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(5):603–619, 2002.

[10] Brittany Terese Fasy, Fabrizio Lecci, Alessandro Rinaldo, Larry Wasserman, Sivaraman Balakrishnan, and Aarti Singh. Confidence sets for persistence diagrams. *Ann. Statist.*, 42(6):2301–2339, 2014.

[11] Jisu Kim, Yen-Chi Chen, Sivaraman Balakrishnan, Alessandro Rinaldo, and Larry A. Wasserman. Statistical inference for cluster trees. In Daniel D. Lee, Masashi Sugiyama, Ulrike von Luxburg, Isabelle Guyon, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 1831–1839, 2017.

[12] B. W. Silverman. Using kernel density estimates to investigate multimodality. *J. Roy. Statist. Soc. Ser. B*, 43(1):97–99, 1981.