Skeleton Clustering: Dimension-Free Density-Aided Clustering

Zeyu Wei
Department of Statistics, University of Washington
and
Yen-Chi Chen
Department of Statistics, University of Washington

January 20, 2023

Abstract

We introduce a density-aided clustering method called Skeleton Clustering that can detect clusters in multivariate and even high-dimensional data with irregular shapes. To bypass the curse of dimensionality, we propose surrogate density measures that are less dependent on the dimension but have intuitive geometric interpretations. The clustering framework constructs a concise representation of the given data as an intermediate step and can be thought of as a combination of prototype methods, density-based clustering, and hierarchical clustering. We show by theoretical analysis and empiri-cal studies that the skeleton clustering leads to reliable clusters in multivariate and high-dimensional scenarios.

Keywords: high-dimensional clustering, density estimation, density-based clustering, k-means clustering

1 Introduction

Density-based clustering (Azzalini and Torelli, 2007; Menardi and Azzalini, 2014; Chacon, 2015) is a popular framework for grouping observations into clusters dened based on the underlying probability density function (PDF). In practice, when the PDF is usually unknown, it is estimated via the random sample and the estimated PDF is then used to obtain the resulting clusters. Many clustering methods have been proposed within the framework of density-based clustering. The mode clustering (Li et al., 2007; Chacon and Duong, 2013; Chen et al., 2016) nds clusters via the local modes of the underlying PDF. When the kernel density estimator (KDE) is used for density estimation, the mode clustering can be done easily via the mean-shift algorithm (Fukunaga and Hostetler, 1975; Cheng, 1995; Carreira-Perpina, 2015). Another famous density-based clustering approach is the level-set clustering (Cuevas et al., 2000, 2001; Mason et al., 2009; Rinaldo et al., 2012), which creates clusters as the connected components of high-density regions. The well-known DBSCAN (Density-Based Spatial Clustering of Applications with Noise) method (Ester et al., 1996) is also a special case of level-set clustering. Moreover, the cluster tree (Stuetzle and Nugent, 2010; Chaudhuri and Dasgupta, 2010; Chaudhuri et al., 2014; Eldridge et al., 2015; Kim et al., 2016) is a density-based clustering approach combining information from both modes and level sets. This method creates a tree structure with each leaf representing a mode and the tree describes the evolution of level-set clusters at dierent density levels.

Compared to the classical k-means clustering (Lloyd, 1982; Hartigan and Wong, 1979; Pollard, 1982) and the model-based clustering methods (Fraley and Raftery, 2002), a density-based clustering approach is capable of nding clusters with irregular shapes and gives an

intuitive interpretation based on the underlying PDF. Furthermore, dening clusters based on the density function makes it possible to view the clustering problem as an estimation problem: the clusters from the true PDF are the parameters of interest and the estimated clusters are sample quantities utilized for approximation.

Although density-based clustering enjoys many advantages, it has a fundamental limitation: the curse of dimensionality. Because a density-based clustering method often involves a density estimation step, it does not scale well with the dimension. Specically, the convergence rate of a density estimator is $O_P(n^{\frac{2}{4+d}})$ under usual smoothness conditions (Scott, 2015; Wasserman, 2006), which is slow when d is large. To overcome the curse of dimensionality and apply density-based clustering to high-dimensional data, we follow the idea of merging a large number of clusters (Peterson et al., 2018; Almodovar-Rivera and Maitra, 2020; Fred and Jain, 2005; Maitra, 2009; Shin et al., 2019), to explicitly construct a graph representation of the data based on the initial protoclusters and propose density-aided similarity measures suitable for high-dimensional settings.

The idea of merging prototypes has also attracted great attention from model-based clustering to overcome the limitations of parametric assumptions. In particular, there are several methods for merging Gaussian-mixture models (Hennig, 2010) such as Dip test approach (Hartigan and Hartigan, 1985), ridgeline elevation (Ray and Lindsay, 2005), misclassication method (Tibshirani and Walther, 2005), multi-layer approach (Li, 2005), entropy-based method (Baudry et al., 2010), level set-based method (Scrucca, 2016), and modal clustering (Chac**ó**n, 2019). The work by Aragam et al. (2020) reconstructs a nonparametric mixture model by tting the data with a large number of general nonparametric mixture components and then partitions them into a small number of nal clusters.

Our idea can be summarized as follows. We rst nd a large set of protoclusters (called knots) by running k-means clustering. Nearby knots are then connected by edges to form a graph that we call the skeleton. The similarities between connected knots are measured by density-aided criteria that are estimable even in high dimensions. Finally, we merge knots according to a linkage criterion to create the nal clusters. Because the construction involves creating a skeleton representation of the data, we call this method Skeleton Clustering.

To illustrate the limitation of the classical approaches and to highlight the eectiveness of skeleton clustering, we conduct a simple simulation in Figure 1. It is a d = 200 dimensional data consisting of ve components with non-spherical shapes. The actual structure is in 2-dimensional space as illustrated in Figure 1. We add Gaussian noises in other dimensions to make it a d = 200 dimensional data (see Section 5 for more details). Traditional k-means and spectral clustering fail to nd the ve components and the mean shift algorithm cannot form clusters due to the high dimensionality of the data. However, our proposed method (bottom-right panel) can successfully recover the underlying ve components.

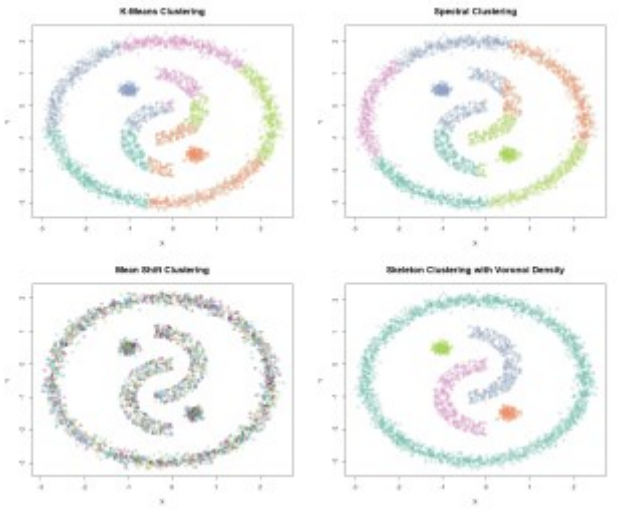


Figure 1: Yinyang Data with dimension 200. On the bottom-right is the clustering result of the skeleton clustering with the proposed Voronoi density similarity measure.

Outline. In section 2, we describe the skeleton clustering framework. In section 3, we introduce similarity measures that can be utilized in the skeleton clustering framework. In section 4, we provide some consistency results of the sample similarity measures and the clustering performance guarantee. In section 5, we present simulation results to demonstrate the eectiveness of skeleton clustering in dealing with dierent data scenarios and to guide some choices in the framework for applications. In section 6, we test the performance of skeleton clustering on real datasets. In section 7, we conclude the paper and point out some directions for future research.

2 Skeleton Clustering Framework

Algorithm 1 Skeleton clustering

Input: Observations $X_1; ; X_n$, nal number of clusters S.

1. Knot construction. Perform k-means clustering with a large number of k; the centers are the knots (Section 2.1).
2. Edge construction. Apply approximate Delaunay triangulation to the knots (Section 2.2).
3. Edge weights construction. Add weights to each edge using either Voronoi density, Face density, or Tube density similarity measure (Section 3).
4. Knots segmentation. Use linkage criterion to segment knots into S groups based on the edge weights (Section 2.4).
5. Assignment of labels. Assign a cluster label to each observation based on which knot group the nearest knot belongs to (Section 2.5).

In this section, we formally introduce the skeleton clustering framework. Let $X = fX_1; : : : ; X_n g$ be a random sample from an unknown distribution with density p supported on a compact set $X \ 2 \ R^d$. The goal of clustering is to partition X into clusters $X_1; : : : X_S$, where S is the nal number of clusters.

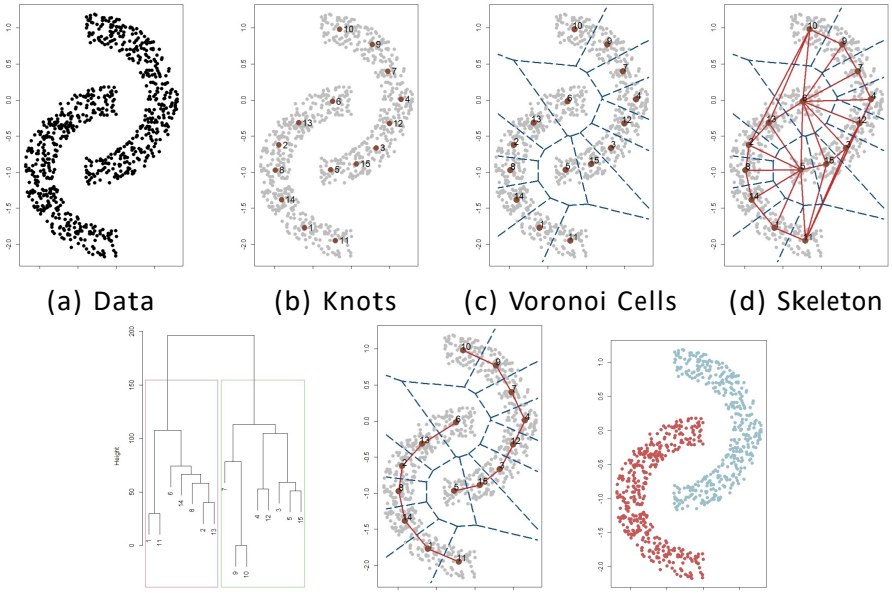
A summary of the skeleton clustering framework is provided in Algorithm 1. Figure 2

illustrates the overall procedure of the skeleton clustering method. Starting with a collection of observations (panel (a)), we rst nd knots, the representative points of the entire data (panel (b)). Then we compute the corresponding Voronoi cells induced by the knots (panel (c)) and the edges connecting the nearby Voronoi cells (panel (d)). For each edge in the graph, we compute a density-aided similarity measure that quanties the closeness of each pair of knots. For the next step, we segment knots into groups based on a linkage criterion (single linkage in this example), leading to the dendrogram in panel (e). Finally, we choose a threshold that cuts the dendrogram into $S = 2$ clusters (panel (f)) and assign a cluster label to each observation according to the knot-cluster that it belongs to (panel (g)).

In summary, the skeleton clustering consists of the following ve steps: (1) Knots construction, (2) Edges construction, (3) Edge weights construction, (4) Knots segmentation, and (5) Assignment of labels. In what follows in this section, we provide a detailed description of each step except Step 3. Step 3 is the key step in our clustering framework where we incorporate the information from the underlying density for clustering in a less dimension-dependent way and we defer the detailed discussion of Step 3 to Section 3 and Section 4. We include a short analysis of the computational complexity of our skeleton clustering framework in Appendix A.

2.1 Knots Construction

The construction of knots is a step aiming at nding representative points in the data that can help measure similarities between regions in the later stage. The knots can be viewed as landmarks inside the data where we can shift our focus from the entire data to these local



(a) Data (b) Knots (c) Voronoi Cells (d) Skeleton

(e) Dendrogram (f) Segmentation (g) Clustering

Figure 2: Skeleton Clustering illustrated by Two Moon Data (d=2).

locations. A simple but reliable approach for constructing knots is the k-means algorithm. We apply the k-means algorithm with a large number k S the desired number of nal clusters, and this procedure behaves like overtting the k-means. Notably, we do not use the k-means procedure to obtain nal clustering, but, instead, we use it as an intermediate step to nd concise representations of the original data.

The number of knots k is a key parameter in the knots construction step. It controls the trade-o between the quality of the data representation and the reliability of each knot. More knots can give a better representation of the data, but, if we have too many knots, the number of observations per knot will be small, so the uncertainty in estimation in the later stage will be large. We nd that a simple reference rule for k to be around $\underline{p}$n works well in our empirical studies (Section F.1). In practice, it is also advisable to prune knots with a small number of corresponding observations because the density-aided weights (in Step 3, Section 3) are estimated locally by the data belonging to each pair of knots.

Knots with a few data points can lead to unstable similarity measurements and unreliable nal clustering. Moreover, to take care of observations in the low-density areas that could cause problems for the k-means clustering, one may rst pre-process or denoise the data by removing observations in the low-density area and then apply the k-means clustering to nd out the knots.

In this work, we use overtting k-means as the default way for knots construction, but there are alternative approaches to nd knots such as subsampling, the coreset construction methods (Bachem et al., 2017; Turner et al., 2020), and the Self-Organizing Maps (SOM) (Heskes, 2001). We show in Appendix F.2 that the SOM can also be used to nd knots but requires more careful treatments such as removing knots with few or even no observations and the performance is slightly worse than that of the overtting k-means. The k-medians algorithm can be another alternative method but it gave an unstable result when the dimension is large. Therefore, we choose to use the overtting k-means algorithm in this work and recommend using it in practice.

Remark 1. Since the k-means algorithm does not always nd the global optimum, we repeat it many times with random initial points (generally 1; 000 times or more) and choose the one with the optimal objective function. This works well for all of our numerical analyses. Moreover, since we are only using k-means as a tool to nd a useful representation, we do not need to nd the actual global optimum. All we need is a set of knots forming a useful representation.

2.2 Edges Construction

With the constructed knots, our next step is to nd the edges connecting them. Let $c_1; \ ; c_k$ be the given knots and we use $C = \ fc_1; \ ; c_k g$ to denote their collection of them.

We add an edge between a pair of knots if they are neighbors, with the neighboring condition being that the corresponding Voronoi cells (Voronoi, 1908) share a common boundary. The Voronoi cell, or Voronoi region, C_j, associated with a knot c_j is the set of all points in X whose distance to c_j is the smallest compared to other knots (See Figure 3). That is,

$$C_j = \{x \in X : d(x, c_j) \leq d(x, c_{j'}) \ \forall j' = j\}; \tag{1}$$

where $d(x, y)$ is the usual Euclidean distance. Therefore, we add an edge between knots (c_i, c_j) if $C_i \cap C_j = \emptyset$. Such resulting graph is the Delaunay triangulation (Delaunay, 1934) of the set of knots C and we denote it as $DT(C)$. In a nutshell, the skeleton graph in our framework is given by the Delaunay triangulation of C.

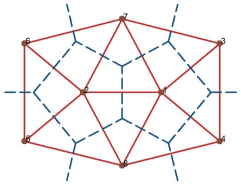


Figure 3: Voronoi Tessellation as blue dashed lines and Delaunay Triangulation by red solid lines.

The Delaunay triangulation graph is conceptually intuitive and appealing and is utilized by some clustering methods to identify connected components (Azzalini and Torelli, 2007; Scrucca, 2016), but empirically the computational complexity of the exact Delaunay triangulation algorithm has an exponential dependence on the ambient dimension d (Amenta et al., 2007; Chazelle, 1993). Given our multivariate and even high-dimensional data setting, exact Delaunay triangulation is empirically unfavorable due to its high computational cost (Polianskii and Pokorny, 2020). Therefore, in practice, we approximate the exact Delaunay Triangulation with $DT(C)$ by examining the 2-nearest knots of the sample data points. The key observation is that, if the Voronoi cells of two knots c_i, c_j share a nontrivial boundary,

there is likely to be a non-empty region of points whose 2-nearest knots are $c_i; c_j$. Consequently, for approximation, we query the two nearest knots for each data point and have an edge between $c_i; c_j$ if there is at least one data point whose two nearest neighbors are $c_i; c_j$. The complexity of the neighbor search depends linearly on the dimension d, which is desirable for high-dimensional setting (Weber et al., 1998), and this sample-based approximation to the Delaunay Triangulation has reliable empirical performance.

2.3 Edge Weight Construction

Given the constructed edges and knots, we assign each edge a weight that represents the similarity between the pair of knots. In this work, we propose some novel density-aided quantities as the edge weights. Since the description of the similarity measures is more involved, we defer the detailed discussion of the similarity measures to Section 3. It is worth noting here that the similarity measures proposed in this work are estimated based on surrogates of the underlying density function (hence density-aided) and the estimation procedure has minimal dependence on the ambient dimension. Therefore, the estimations of the newly proposed similarity measures are reliable even under high-dimensional settings.

2.4 Knots Segmentation

Given the weighted skeleton graph, the next step is to partition the knots into the desired number of nal clusters, and we apply hierarchical clustering by converting the similarity measures into distances. Particularly, for given similarity measures $fs_{ij}g_{i=j}$ where only connected pairs can take nonzero entries and let $s_{max} = max_{i=j} s_{ij}$, we dene the

corresponding distances as $d_{ij} = 0$ if $i = j$ and $d_{ij} = s_{max} \quad s_{ij}$ otherwise.

The choice of linkage criterion for hierarchical clustering may depend on the underlying geometric structure of the data. We analyze several linkage criteria under various simulation scenarios in Appendix E. Generally, single linkage gives reliable clustering results when the components are well-separated, but average linkage works better when there are overlapping clusters of approximately spherical shapes. Therefore, in practice, such a choice of linkage should be made based on some exploratory understanding of the data structure, and experimenting with dierent linkage methods is computationally tractable as only the knots need to be segmented.

The number of nal clusters S is an essential parameter for the hierarchical clustering procedure but can be unknown. The dendrograms given by hierarchical clustering can be a helpful tool in this situation, displaying the clustering structure at dierent resolutions. Consequently, analysts can experiment with dierent numbers of nal clusters and choose a cut that preserves the meaningful structures based on the dendrograms, which takes little extra computation. However, it is worth pointing out that with the presence of noisy data points, the nal number S being larger than the true number of meaningful components may be needed to achieve better clustering results (see Appendix E).

Remark 2. Although the dendrogram for knots given by our method is not exactly the cluster trees, the pruning graph cluster tree procedure proposed in Nugent and Stuetzle (2010) with excess mass can be applied to help decide the nal segmentation. Peterson et al. (2018) also presented similar ideas choosing the nal number of clusters by looking at the lifetime of the clusters in the dendrogram. Additionally, the traditional \elbow" methods can be used to determine the number of clusters. An inferential choice can also be made using the gap statistics (Tibshirani et al., 2001).

2.5 Assignment of Labels

In the previous step, we created S groups of knots and each group has a cluster label. To pass the cluster membership to each observation, we assign a hard clustering label to each observation according to which group its nearest knot belongs. For instance, if an observation X_i is closest to knot c_j and c_j belongs to cluster ‘, we assign cluster membership label ‘ to observation X_i.

Remark 3. There are other methods in clustering literature for assigning labels of observations based on identied structures. Azzalini and Torelli (2007) and Scrucca (2016) assign unlabelled data based on density ratios. DBSCAN and HDBSCAN (Campello et al., 2015; Ester et al., 1996) assign labels (and identify noisy points) based on k-nearest-neighbor considerations. One may use these alternatives to assign the cluster label to each observation.

3 Density-Based Edge Weights Construction

To incorporate the information of density into clustering, we calculate the edge weights in the constructed skeleton based on the underlying density function. However, the conventional notion of PDF is not feasible in multivariate or even high-dimensional data due to the curse of dimensionality. To resolve this issue, we introduce three density-related quantities that are estimable even when the dimension is high.

3.1 Voronoi Density

The Voronoi density (VD) measures the similarity between a pair of knots $(c_j; c_{'})$ based on the number of observations whose 2-nearest knots are c_j and $c_{'}$. We start with dening the

Voronoi density based on the underlying probability measure and then introduce its sample analog. Given a metric d on R^d, the 2-Nearest-Neighbor (2-NN) region of a pair of knots $(c_j; c_{j'})$ is dened as

$$A_{jj'} = \{x \in X : d(x; c_i) > \max\{d(x; c_j); d(x; c_{j'})\}; \forall i = j; j'\}: \quad (2)$$

In this work, we take d(:; :) to be the usual Euclidean distance and use ||:|| to denote the Euclidean norm. An example 2-NN region of a pair of knots is illustrated in Figure 4.

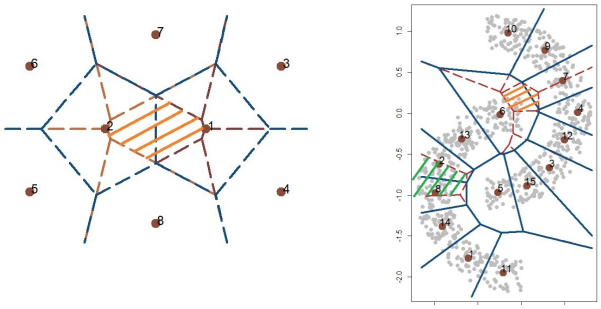


Figure 4: Left: Orange shaded area illustrates the 2-NN region of knots 1; 2. Right: Shaded areas illustrate the 2-NN region of knots 6; 7 and knots 2; 8.

Following the idea of density-based clustering, two knots $c_j; c_{j'}$ belong to the same clusters if they are in a connected high-density region, and we would expect the 2-NN region of $c_j; c_{j'}$ to have a high probability measure. Hence, the probability $P(A_{jj'}) = P(X_1 \in A_{jj'})$ can measure the association between c_j and $c_{j'}$ (see illustration in Figure 4 right). Based on this insight, the Voronoi density measures the edge weight of $(c_j; c_{j'})$ with

$$S_{jj'}^{VD} = \frac{P(A_{jj'})}{\|c_j - c_{j'}\|}: \quad (3)$$

Namely, we divide the probability of the in-between region by the mutual Euclidean distance. The division of the distance adjusts for the fact that 2-NN regions have dierent sizes and provides more weights to edges between knots close in distance. However, such division makes the Voronoi density to be in the unit of $1/\|c_j - c_{j'}\|$ and hence can be scale-dependent.

In practice, we estimate $S_{j'}^{VD}$ by a sample average. Specically, the numerator $P(A_{j'})$ is estimated by $\hat{P}_n(A_{j'}) = \frac{1}{n}\sum_{i=1}^{n} I(X_i \in A_{j'})$ and the nal estimator for the VD is

$$\hat{S}_{j'}^{VD} = \frac{\hat{P}_n(A_{j'})}{\|c_j - c_{j'}\|}. \tag{4}$$

Note that here we are assuming that $c_1, \ldots, c_k$ as given beforehand. In the sample version, we replace them with the sample analog $\hat{c}_1, \ldots, \hat{c}_k$ and replace the region $A_{j'}$ by $\hat{A}_{j'}$.

The Voronoi density can be computed in a fast way. The numerator, which only depends on 2-nearest-neighbors calculation, can be computed eciently by the k-d tree algorithm (Bentley, 1975). For high-dimensional space, space partitioning search approaches like the k-d tree can be inecient but a direct linear search still gives a short run-time (Weber et al., 1998), and with a large number of observations approximate nearest neighbor algorithms can be incorporated. The denominator requires distance calculation and can be burdensome in high-dimensional settings, but note that we only need to calculate the distance for edges present in $\hat{D}_T(C)$, which is far less than $k(k-1)/2$, where k is the number of knots. Hence, the calculation of VD can be carried out in a fast way even for high-dimensional data with a large sample size.

3.2 Face Density

Here we present another density-based quantity to measure the similarity between two knots. Since the Voronoi cell of a knot describes the associated region, a natural way to measure the similarity between two knots is to investigate the shared boundary of the corresponding Voronoi cells. If two knots are highly similar, we would expect the boundary to lie in a high-density region and to be surrounded by many observations. Based on this idea,

we dene the Face Density (FD) as the integrated PDF over the \face" (boundary) region. Note that, although the density is involved in FD, by integrating over the face region the problem reduces to a 1-dimensional density estimation task regardless of the dimension of the ambient space. Formally, let the face region between two knots $c_j; c_{j'}$ be $F_{jj'} = C_j \setminus C_{j'}$. At the population level, the FD is dened as

$$S_{jj'}^{FD} = \int_{F_{jj'}} p(x)\, d\mu_{d-1}(dx) = \int_{F_{jj'}} dP(x); \tag{5}$$

where $\mu_m(dx)$ denotes the m-dimensional volume measure.

To estimate the FD, we utilize the idea of kernel smoothing in combination with data projection. By the construction of the Voronoi diagram, the boundary of two Voronoi cells is orthogonal to the line passing through the two corresponding knots (called the 'central line') and intersects the central line at the middle point regardless of the dimension of the data (see Figure 3 for reference). Therefore, we estimate the FD by rst projecting the observations onto the central line and then using the 1-dimensional kernel density estimator(KDE) to evaluate the density at the midpoint. Specically, x two knots $c_j; c_{j'}$, let $C_j; C_{j'}$ be the corresponding Voronoi cells, and denote $\pi_{jj'}(x)$ as the projection of $x \in X$ onto the central line passing through c_j and $c_{j'}$, we dene the estimator $\hat{S}_{jj'}^{FD}$ to be

$$\hat{S}_{jj'}^{FD} = \frac{1}{nh} \sum_{X_i \in C_j \cup C_{j'}} K\left(\frac{\pi_{jj'}(X_i) - (c_{j'} + c_j)/2}{h}\right) \tag{6}$$

where K is a smooth, symmetric kernel function (e.g. Gaussian kernel) and $h > 0$ is the bandwidth that controls the amount of smoothing. It is noteworthy that, while conventional kernel smoothing suers from the curse of dimensionality (Chen, 2017; Chacón et al., 2011; Wasserman, 2006), the kernel estimator in equation (6) bypasses it.

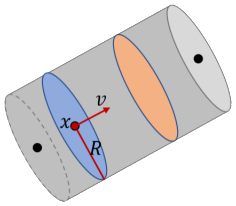


Figure 5: The disk area centered at x with a radius R and a direction .

3.3 Tube Density

While FD is conceptually appealing, the characterization of the face between two Voronoi cells could be challenging since the shapes of the boundaries can be irregular, and the characteristics of the boundaries can aect the estimation of the Face density from a theoretical perspective (Section D.2). Here we propose a measure similar to the Face density measure but has a predened regular shape. For a point x, we dene the Disk Area centered at x with radius R and normal direction (see Figure 5 for an illustration) as

$$\text{Disk}(x; R;) = \{y : \|x \quad y\| \quad R; (x \quad y)^T = 0\} \tag{7}$$

To measure the similarity between knots c_j and $c_{'}$, we examine the integrated density within the disk areas along the central line. In more detail, the central line can be expressed as $\{c_j + t(c_{'} \quad c_j) : t \ 2 \ [0; 1]\}$, and any point on the central line can be written as $c_j + t(c_{'} \quad c_j)$ for some t. For a point $c_j + t(c_{'} \quad c_j)$, we dene the integrated density in the disk region (called Disk Density) as

$$\text{pDisk}_{j';R}(t) = P(\text{Disk}(c_j + t(c_{'} \quad c_j); R; c_{'} \quad c_j)) = \int_{\text{Disk}(c_j+t(c_{'} \quad c_j);R;c_{'} \quad c_j)} p(x)dx: \tag{8}$$

The Tube Density (TD) measures the similarity between c_j and $c_{'}$ as the minimal disk density along the central line, i.e.,

$$S^{TD}_{j,'} = \inf_{t2[0;1]} \text{pDisk}_{j';R}(t) \tag{9}$$

In other words, with given $c_j; c'$, we survey all Disk Density along the central line and retrieve the inmum as the similarity measure between two knots.

In this work, we set R based on the root mean squared distances within each Voronoi cell. Specically, for knot c_j and the corresponding Voronoi cell C_j, we calculate

$$R_j = \sqrt{\frac{1}{jC_j j} \sum_{X' 2 C_j} kX' \quad c_j k^2} \tag{10}$$

where $jC_j j$ denotes the size of set C_j. With the uniform radius paradigm where the radius is the same for all pairs of knots, we set $R = \frac{1}{k} \sum_{j=1}^{k} R_j$. Our empirical studies show that this rule leads to good clustering performances and theoretical analysis also shows that this reference rule for R leads to the consistency of the sample analog of the TD.

Note that the radius may also be chosen adaptively for each pair: we set the disk radius at c_j to be R_j for all knots and set the disk radius along the edge to be the linear interpolation of the radii at the two connected knots. The comparison between the uniform and adaptive R is presented in Appendix F.6, and similar clustering performance is observed for the two approaches. Hence we use uniform R by default for simplicity.

Similar to the FD, we estimate the TD by a projected KDE. Let $_{j'}(x)$ be the projection of a point x on the line through $c_j; c'$. We rst estimate the pDisk via

$$\hat{p}\text{Disk}_{j';R}(t) = \frac{1}{nh} \sum_{i=1}^{n} K \frac{j'(\ _i) \quad c_j \quad t(c' \quad c_j)}{h} I(jjX_i \quad _{j'}(X_i)jj \quad R)$$

and then estimate the TD as

$$\hat{\delta}^{TD}_{j'} = \inf_{t2[0;1]} \hat{p}\text{Disk}_{j';R}(t): \tag{11}$$

where the inmum is approximated by grid search.

Remark 4. The estimations of the FD and the TD involve the use of the projected kernel density estimation, and we discuss the choices of kernel and the bandwidth selections for

kernel density estimations in Appendix F.3. By default, we use the Gaussian kernel with the normal scale bandwidth selector (NS) (Chacon et al., 2011) for the best empirical results.

4 Asymptotic Theory of Edge Weight Estimation

In this section, we focus on the theoretical properties of the similarity measures to theoretically explain the eectiveness of the newly proposed density-aided similarity measures. We assume the set of knots $C = \{c_1, \ldots, c_k\}$ is given and non-random to simplify the analysis because (1) it is hard to quantify k-means uncertainty, and (2) with large k, it is extremely likely for k-means to stuck within the local minimum. Note that this implies the corresponding Voronoi cells $C = \{C_1, \ldots, C_k\}$ and the 2-NN regions $\{A_{j'}\}_{j=1,\ldots,k;j=}$ (Equation 2) of all pairs of knots are xed as well. We allow $k = k_n$ to grow with respect to the sample size n. Theoretical results for Voronoi density are described in this section and theoretical properties for the Face density and Tube density are deferred to Appendix B and C respectively. In summary, the consistency of FD and TD are obtained based on the analysis of KDE with additional geometric considerations, resulting in rates similar to that of the 1-dimensional KDE under some regularity conditions. All proofs are included in Appendix D.

4.1 Voronoi Density Consistency

We start with the convergence rate of the VD. We consider the following condition:

(B1) There exists a constant c_0 such that the minimal knot size $\min_{(j;')2E} P(A_{j'})$ and $\min_{(j;')2E} \|c_j - c_{'}\|$.

where $(j;\ell) \in E$ means that there is an edge between knots $c_j; c_\ell$ in the Delaunay Triangulation. Condition (B1) is a condition requiring that no Voronoi cell $A_{j\ell}$ has a particularly small size and all edges have sucient length. This condition is mild because when the dimension of data d is xed, the total number of edges in the Delaunay triangulation of k points scale at rate O(k) (Berg et al., 2008). Because the volume shrinks at rate $O(k^{-1})$, the distance is expected to shrink at rate $O(k^{-1=d})$.

Remark 5. If we assume there exists constant $c_1; c_2 > 0$ that the density function $f_P(x) > c_1 > 0$ for P a:e: and that $\min_{(j;\ell)} |A_{j\ell}| \ c_2 \frac{|X|}{k}$ where $|X|$ is the volume of the support, then we have $\min_{(j;\ell)\in E} P(A_{j\ell}) \ \frac{c_1 c_2}{k}$. So (B1), as implied by the above two common conditions, is a relatively weak assumption.

Theorem 1 (Voronoi Density Convergence). Assume (B1). Then for any pair $j = \ell$ that shares an edge, the similarity measure based on the Voronoi density satises

$$\frac{\hat{S}^{VD}_{j\ell}}{S^{VD}_{j\ell}} - 1 = O_p\left(\sqrt{\frac{k}{n}}\right); \tag{12}$$

$$\max_{j;\ell} \frac{\hat{S}^{VD}_{j\ell}}{S^{VD}_{j\ell}} - 1 = O_p\left(\sqrt{\frac{k}{n}\log k}\right); \tag{13}$$

when $n \to 1; k \to 1; \frac{n}{k} \to 1$.

Theorem 1 provides the convergence rates of the sample-based Voronoi density to the population version of Voronoi density. This result is reasonable because when the knots C are given, the randomness in the sample-based Voronoi density is just the empirical proportion in each cell, so it is a square-root-rate estimator based on the eective local sample size n=k. Consequentially, Theorem 1 suggests that estimating the Voronoi density is easy in the multivariate case when the knots are given{there is no dependency with respect to the

ambient dimension. The extra log k factor in the uniform bound (Equation 13) comes from the Gaussian concentration bounds.

4.2 Performance Guarantee for Voronoi Density

We provide below a performance guarantee for skeleton clustering with Voronoi density in terms of the adjusted Rand Index (Rand, 1971; Hubert and Arabie, 1985), which measures the agreements between two clusterings after adjusting for permutation chance. To simplify the problem, we dene the true clusters as the connected components of the skeleton graph with edges having true Voronoi density similarities $S_{j,}^{VD}$ over a known threshold > 0. We show below that cutting the skeleton graph based on estimated edge similarities at the same threshold recovers the true clustering with a high probability. Since the knots are xed, the clustering error comes from partitioning knots into the wrong groups, so we will focus on the adjusted Rand Index of clustering the knots. Let the true partition of the knots be $L = fLg_{‘=1;:::;L}$, where $L_‘$ contains all the knot indices belonging to the partition ‘. Let the partition based on estimated edge similarities be $L\hat{}$. We assume that

(P1) The true partition L under the threshold remains the same when the thresholding level is within $((1 \quad "); (1 + "))$ for some $" > 0$.

This is a mild assumption because when we vary the threshold level , only a nite number of values will create a change in the partition. So (P1) holds under almost all values of except for a set of Lebesgue measure 0. Let $ARI(L; L\hat{})$ denotes the adjusted Rand Index of the estimated partition.

Theorem 2 (Adjusted Rand Index Guarantee). Assume (B1) and (P1) and let $p_{min} =$

$\min_{j,j'} P(A_{j,j'})$, then

$$P\left\{ ARI(L, \hat{L}) < 1 \right\} \geq k(k-1) \exp\left(\frac{\frac{1}{2}\varepsilon^2 p_{min} n}{(1 - p_{min}) + \frac{1}{3}\varepsilon} \right) \tag{14}$$

Theorem 2 shows that we have a good chance of recovering the "true" clusters defined by the actual Voronoi density. The above bound is derived from the uniform concentration bound of the Voronoi density.

5 Simulations

To study the effectiveness of skeleton clustering as a clustering method, we conduct several Monte Carlo experiments. In this section, we present some empirical results to illustrate the performance of skeleton clustering in multivariate and high-dimensional settings (with additional data examples in Appendix G). Generally, our framework with the Voronoi density similarity measure is superior among all the compared clustering methods. In Appendix E, we use a systematic set of simulation studies to discuss the choice of linkage criteria within our clustering framework when dealing with different datasets and at the same time to demonstrate the robustness of the proposed framework to noisy data points and overlapping clusters. We include some additional simulations to support some choices within our framework in Appendix F. The R implementation of the skeleton clustering methods along with some simulations can be founded at https://github.com/JerryBubble/skeletonMethods.

5.1 High-dimensional Setting

In this section, we demonstrate the performance of skeleton clustering on simulated datasets: the Yinyang data and the Mickey data. We also include a simulated dataset

consisting of manifold structures of dierent dimensions, called the Manifold Mixture data, in Appendix G.1 and an additional simulation called the Ring data in Appendix G.2. For the simulations within Section 5.1 and Appendix G, when using the skeleton clustering methods, the number of knots is set to be $k = [\sqrt{n}]$ and the knots are chosen by k-means with 1000 random initialization. We select smoothing bandwidth by the normal scale bandwidth selector for the FD and TD, and the radius of TD is set to be the same for all edges with the value chosen as described in Section 3.3. We use single linkage hierarchical clustering when merging knots into nal clusters with the true number of nal clusters S being provided.

To highlight the importance of density-aided similarity measures, we include a similarity measure called the average distance (AD) for comparison. AD measures the similarity between c_j and $c_{j'}$ using the inverse of the average Euclidean distances between all pairs of observations in the two corresponding Voronoi cells. All simulations are repeated 100 times to obtain the distribution of the empirical performances.

5.1.1 Yinyang Data

The Yinyang dataset is an intrinsically 2-dimensional data containing 5 components: a big outer circle with 2000 uniformly distributed data points, two inner semi-circles each with 200 data points generated as 2D Gaussian with standard deviation 0:1, and two clumps each with 200 data points (generated with the shapes.two.moon function with default parameters in the clusterSim library in R (Walesiak and Dudek, 2020)). The total sample size is $n = 3200$ and, according to our reference rule, we choose $k = [\sqrt{3200}] = 57$ knots for the skeleton clustering procedure. To make the data high-dimensional, we include additional variables from a Gaussian distribution with mean 0 and standard deviation 0:1, and we increase the

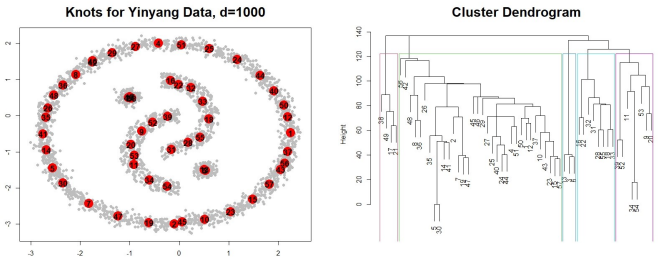


Figure 6: Knots chosen by k-means on Yinyang data and the Dendrogram for single linkage hierarchical clustering with similarity measured by Voronoi density.

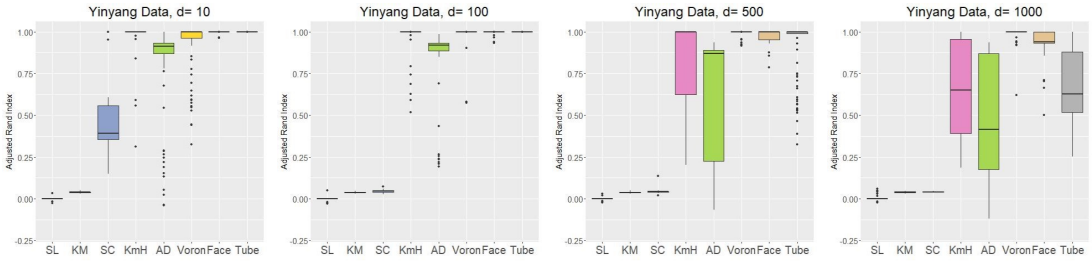


Figure 7: Comparison of the nal clustering performance in terms of adjusted Rand Index with dierent clustering methods on Yinyang Data with dimensions 10, 100, 500, and 1000.

dimension of noise variables so that the total dimensions are $d = 10; 100; 500; 1000$. We present results with larger standard deviations for the noisy variable in Appendix F.7.

We empirically compare with the following clustering approaches: direct single-linkage hierarchical clustering (SL), direct k-means clustering (KM), spectral clustering (SC), and the merging K-means with hierarchical clustering method proposed by Peterson et al. (2018)(KmH). We include the comparison with merging model-based clusters approaches in Appendix F.9.

For skeleton clustering, we present the results with average distance density (AD), Voronoi density (Voron), Face density (Face), and Tube density (Tube). Since this is simulated data, we know that there are exactly 5 clusters and we know which cluster an observation belongs to. The true number of clusters is provided to all the clustering algorithms. We use the adjusted Rand Index to measure the performance of each clustering method.

The results are given in Figure 7. We observe that when dimension increases, traditional methods (SL, KM, SC) fail to give good clustering results while skeleton clustering can generate nearly perfect clustering. The KmH approach has better performance than the skeleton clustering using average distance, but skeleton clustering with other proposed density-aided similarity measures has better clustering results. This illustrates the effectiveness of using the skeleton clustering framework and highlights the advantage of using the proposed density-aided weights in clustering large-dimensional data. Across all the data dimensions, the Voronoi density, the simplest measure among the three proposed similarity measures, gives the best performance in the skeleton clustering framework.

5.1.2 Mickey Data

The simulated Mickey data is an intrinsically 2-dimensional data consisting of one large circular region with 1000 data points and two small circular regions each with 100 data points. As a result, the structures have unbalanced sizes. The total sample size is $n = 1200$ and we choose the number of knots to be $k = [\sqrt{1200}] = 35$. We include additional variables with random Gaussian noises to make it a high dimensional data ($d = 10; 100; 500; 1000$) the same way as in Section 5.1.1. The left panel of Figure 8 shows the scatter plot of the first two dimensions.

We perform the same comparisons as done on the Yinyang data with the true number of components $S = 3$ provided to all the clustering algorithms, and the results are displayed in Figure 9. All methods perform well when d is small but starting at $d = 100$, traditional methods fail to recover the underlying clusters. On the other hand, all methods in the skeleton clustering framework and the KmH approach work well even when $d = 1000$.

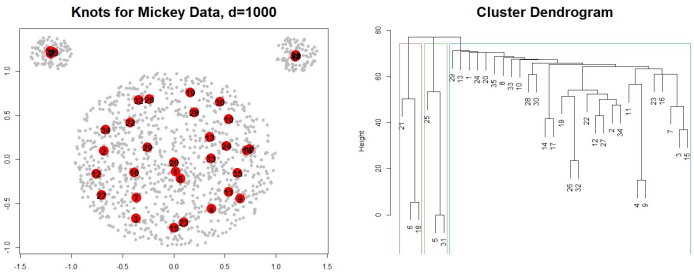


Figure 8: An illustration of the analysis of the Mickey data with dimension 100.

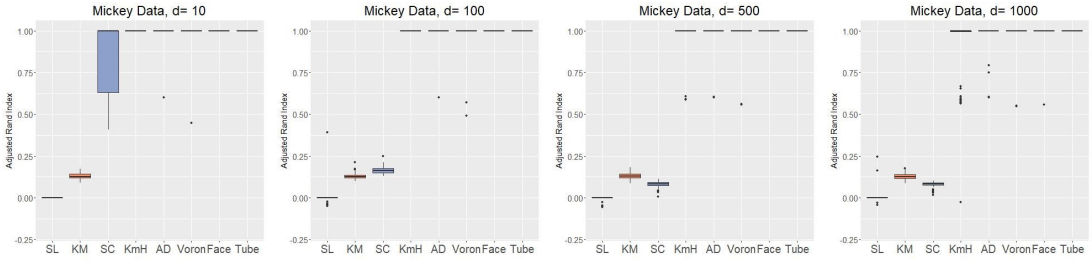


Figure 9: Comparison of adjusted Rand index using dierent similarity measures on Mickey data with dimensions 10, 100, 500, 1000.

6 Real Data

In this section, we apply skeleton clustering to one real data example: the graft-versus-host disease (GvHD) data (Brinkman et al., 2007). Additionally, we analyze the Zipcode data (Stuetzle and Nugent, 2010) in Appendix H.1 and the Olive Oil data (Tsimidou et al., 1987) in Appendix H.2.

GvHD is a signicant problem in the eld of allogeneic blood and marrow transplantation which occurs when allogeneic hematopoietic stem cell transplant recipients when donor-immune cells in the graft attack the tissues of the recipient. The data include samples from a patient with GvHD containing $n_1 = 9083$ observations and samples from a control patient with $n_2 = 6809$ observations. Both samples include four biomarker variables, CD4, CD8, CD3, and CD8. Previous studies (Lo et al., 2008; Baudry et al., 2010) have identied the presence of high values in CD3, CD4, CD8 cell sub-populations as a signicant characteristic

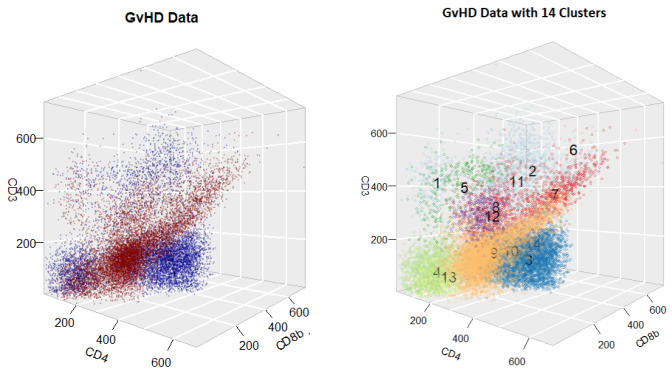


Figure 10: Left: 3D scatterplot of the positive sample (red) and the control sample (blue). Right: Final clustering result of combined GvHD data.

in the GvHD positive sample and a major objective of our analysis is to rediscovery this region with the proposed skeleton clustering methods. In addition, our skeleton clustering procedure shows more information and leads to a novel two-sample test.

The two samples are plotted in the left panel of Figure 10 focusing on the three key variables (CD3, CD4, CD8) with blue points from the control sample and the red points from the GvHD positive sample. We observe that, in addition to the high CD3, CD4, CD8 region, the distribution of the positive sample is dierent from the control sample also in some region with medium to the low CD3, CD4, and CD8. Later we will demonstrate that our clustering framework can identify all such dierences in distributions.

To apply the skeleton clustering for a fair comparison of the two samples, we rst construct knots from each sample separately. Specically, we apply the k-means method to nd $k_1 = [\sqrt{n_1}]$ knots for the positive sample and nd $k_2 = [\sqrt{n_2}]$ knots for the control sample. This ensures that both samples are well-represented by knots. We then combine the two samples into one dataset and combine the two sets of knots into one set with $k_1 + k_2$ knots. We create edges among the combined knots and apply the Voronoi density (VD) to measure

the edge weights. To segment the knots, we use the average linkage criterion because there is no clear gap between major components and the analysis in Appendix E suggests average linkage for this scenario. The skeleton clustering result is displayed in the right panel of Figure 10 with the number of nal clusters chosen to be S = 14. This choice follow Baudry et al. (2010) where the authors suggest to t 9 components on the positive sample and 3 to 5 components on the control sample. We choose S = 9 + 5 = 14 on the combined data to give a reasonable representation of the structures in the data, and, by empirical exploration, this choice leads to a good clustering result.

For further insights, we examined the weighted proportion of positive observations in each cluster. A proportionally smaller weight is assigned to each positive observation to accommodate the fact that there are more positive observations ($n_1 = 9083 > n_2 = 6809$). After such normalization, a weighted proportion of 0:5 means that the positive and control observations are balanced in one region. A summary of the weighted proportion of clusters is presented in Table 1. We note that clusters 7,9,12, and 13 are majorly composed of positive observations (proportion > 0:75), and clusters 3 and 6 are majorly composed of observations from the control sample (proportion < 0:25). We also include the p-value for testing if the proportions equal 0:5. Admittedly, because we use the data to nd clusters and use the same data to do the test, the p-values in Table 1 may tend to be small.

Clusters with majorly positive observations and clusters with majorly control observations are depicted in the two panels in Figure 11. Cluster 7 corresponds to the high CD3, CD4, CD8 region identied by previous works with nearly all data points belonging to the positive patient. Cluster 6 is also scattered in the high CD3, CD4, CD8 region but has all the observations coming from the control sample. However, the small size (only 17 data

Cluster	1	2	3	4	5	6	7
Size	202	948	3881	1859	338	17	812
Prop	.458	.343	.008	.296	.341	.000	.934
p-value	.30	$7\ 10^{20}$	0	310^{63}	410^{8}	$1\ 10^{4}$	610^{103}
Cluster	8	9	10	11	12	13	14
Size	468	6191	251	37	478	402	8
Prop	.690	.888	.673	.669	.794	.841	.310
p-value	210^{13}	0	110^{6}	.09	610^{30}	$3\ 10^{33}$	.52

Table 1: Table of the sizes of the clusters and the weighted proportion of positive observations within each cluster. A proportion of 0:5 indicates that the two samples have equal proportions in the region. The p-value is the simple proportional test to examine if the two samples have equal proportions in that cluster.

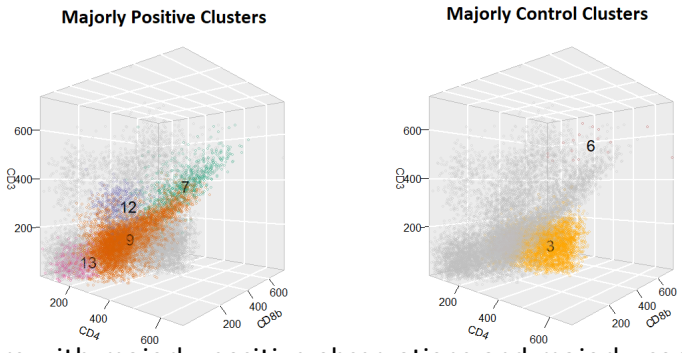


Figure 11: Clusters with majorly positive observations and majorly control observations

points) of Cluster 6 makes it unclear if it is a real structure or due to pure randomness. Overall our method succeeds in identifying the CD3+ CD4+ CD8+ area for the GvHD-positive patient like the previous model-based clustering approaches. Note that the data we are using are two individuals from the original 31 individuals in the GvHD study, which does not account for the inter-individual variability.

Our clustering approach has some additional ndings. Clusters 9, 12, and 13 also have high proportions of positive samples. These clusters are in the mid to low CD3, CD4, CD8 region. For the control case, in addition to the small Cluster 6, Cluster 3 is a large cluster with nearly all the observations from the control sample. It is located in the high CD8 but low CD3 and CD4 region.

Model-based clustering approaches Lo et al. (2008); Baudry et al. (2010) have an advantage for managing this cytometry data as they can parametrically describe the behaviors of data samples in dierent regions. The overlapping between dierent structures and the overall 4-dimensional feature space is also applicable with model-based clustering methods. However, the proposed skeleton clustering approach can result in a graphical representation of each cluster that can be visualized for intuitive understanding. We include the skeleton graphs of the GvHD data clusters from the proposed clustering approach in Appendix F.10. Moreover, model-based approaches can still be limited to some regular shapes of the clusters in the ambient space, while applying the proposed clustering method helps identify clusters with complex structures. Cluster 9, for instance, shows a hammer-like structure based on the skeleton representation (see Figure 41).

Our results suggest a potential procedure for diagnosing GvHD. Biomarkers from a new patient can be divided into clusters with respect to the learned segmentation, and doctors can mainly focus on the sample points that fall into regions 3, 7, 9, 12, and 13. If the patient has many points in Clusters 7, 9, 12, and 13, the patient likely has GvHD. Note that our current result is only based on two individuals and, with a descriptive purpose, is not accounting for the variability between dierent individuals and dierent cases. To use it for practical diagnosis, a more comprehensive analysis based on a larger and more representative sample is required.

7 Conclusion

In this work, we introduce the skeleton clustering framework that can handle multivariate and even high-dimensional clustering problems with complex, manifold-based cluster shapes.

Our method adopts the density-based clustering idea to the high dimensional regime. The key to bypassing the curse of dimensionality is the use of density surrogates such as Voronoi density, Face density, and Tube density that are less sensitive to the dimension. We use both theoretical and empirical analysis to illustrate the eectiveness of the skeleton clustering procedure.

In what follows, we discuss some possible future directions. First, theoretical results accounting for the randomness of knots should be developed. The randomness of knots can aect the clustering performance because the location of knots directly impacts the Voronoi cells, which changes the value of the similarity measures and consequently the cluster label assignments. However, our k-means procedure is unlikely to stop at the global minimum, and it is unclear how to derive a theoretical statement based on local minima properly, so we leave this as future work. Additionally, the proposed skeleton clustering framework can also be potentially used for tasks such as detecting boundary points between clusters, anomaly and noise detection, and community detection in network data (Appendix I). Overall, given the exibility of the skeleton clustering framework, other possibilities by incorporating dierent methods for dierent steps in the framework can be explored.

Acknowledgments

Yen-Chi Chen is supported by NSF DMS-195278, 2112907, 2141808 and NIH U24-AG072122.

Zeyu Wei is supported by NSF NSF DMS - 2112907.

The authors report there are no competing interests to declare.

References

E. Abbe. Community detection and stochastic block models: recent developments. The Journal of Machine Learning Research, 18(1):6446{6531, 2017.

I. A. Almodovar-Rivera and R. Maitra. Kernel-estimated nonparametric overlap-based syncytial clustering. Journal of Machine Learning Research, 21(122):1{54, 2020. URL http://jmlr.org/papers/v21/18-435.html.

N. Amenta, D. Attali, and O. Devillers. Complexity of delaunay triangulation for points on lower-dimensional polyhedra. In Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA '07, pages 1106{1113, USA, 2007. Society for Industrial and Applied Mathematics. ISBN 9780898716245.

B. Aragam, C. Dan, E. P. Xing, and P. Ravikumar. Identiability of nonparametric mixture models and Bayes optimal clustering. The Annals of Statistics, 48(4):2277 { 2302, 2020. doi: 10.1214/19-AOS1887.

A. Azzalini and N. Torelli. Clustering via nonparametric density estimation. Statistics and Computing, 17(1):71{80, 2007.

O. Bachem, M. Lucic, and A. Krause. Practical coreset constructions for machine learning, 2017.

J.-P. Baudry, A. E. Raftery, G. Celeux, K. Lo, and R. Gottardo. Combining mixture components for clustering. Journal of Computational and Graphical Statistics, 19(2):332{353, 2010. doi: 10.1198/jcgs.2010.08111.

J. L. Bentley. Multidimensional binary search trees used for associative searching. Commun. ACM, 18(9):509{517, Sept. 1975.

M. d. Berg, O. Cheong, M. v. Kreveld, and M. Overmars. Computational Geometry: Algorithms and Applications. Springer-Verlag TELOS, Santa Clara, CA, USA, 3rd ed. edition, 2008. ISBN 3540779736.

A. W. Bowman. An alternative method of cross-validation for the smoothing of density estimates. Biometrika, 71(2):353{360, 1984.

R. R. Brinkman, M. Gasparetto, S. J. J. Lee, A. J. Ribickas, J. Perkins, W. Janssen, R. Smiley, and C. Smith. High-Content Flow Cytometry and Temporal Data Analysis for Dening a Cellular Signature of Graft-Versus-Host Disease. Biology of Blood and Marrow Transplantation, 13(6):691{700, jun 2007.

R. J. Campello, D. Moulavi, A. Zimek, and J. Sander. Hierarchical density estimates for data clustering, visualization, and outlier detection. ACM Transactions on Knowledge Discovery from Data, 10(1), jul 2015.

M. A. Carreira-Perpinan. A review of mean-shift algorithms for clustering. arXiv preprint arXiv:1503.00687, 2015.

J. E. Chacon. A Population Background for Nonparametric Density-Based Clustering. Statistical Science, 30(4):518{532, 2015. doi: 10.1214/15-STS526.

J. E. Chac**a**. Mixture model modal clustering. Advances in Data Analysis and Classication, 13:379{404, 6 2019.

J. E. Chacon and T. Duong. Data-driven density derivative estimation, with applications to nonparametric clustering and bump hunting. Electronic Journal of Statistics, 7:499{532, 2013.

J. E. Chacon, T. Duong, and M. P. Wand. Asymptotics for General Multivariate Kernel Density Derivative Estimators. Statistica Sinica, 21(2):807{840, 2011.

K. Chaudhuri and S. Dasgupta. Rates of convergence for the cluster tree. In Proceedings of the 23rd International Conference on Neural Information Processing Systems - Volume 1, NIPS'10, pages 343{351, Red Hook, NY, USA, 2010. Curran Associates Inc.

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

B. Chazelle. An optimal convex hull algorithm in any xed dimension. Discrete & Computational Geometry, (1):377{409, dec 1993. doi: 10.1007/BF02573985.

Y.-C. Chen. A Tutorial on Kernel Density Estimation and Recent Advances. Biostatistics and Epidemiology, 1(1):161{187, apr 2017.

Y. C. Chen, C. R. Genovese, and L. Wasserman. A comprehensive approach to mode clustering. Electronic Journal of Statistics, 10(1):210{241, 2016.

Y. Cheng. Mean shift, mode seeking, and clustering. IEEE transactions on pattern analysis and machine intelligence, 17(8):790{799, 1995.

A. Cuevas, M. Febrero, and R. Fraiman. Estimating the number of clusters. Canadian Journal of Statistics, 28(2):367{382, 2000.

A. Cuevas, M. Febrero, and R. Fraiman. Cluster analysis: A further approach based on density estimation. Computational Statistics and Data Analysis, 36(4):441{459, 2001.

B. Delaunay. Sur la sphere vide. a la memoire de georges voronoi. Bulletin de l'Academie des Sciences de l'URSS. Classe des sciences mathematiques et na, 6:793{800, 1934.

J. Eldridge, M. Belkin, and Y. Wang. Beyond hartigan consistency: Merge distortion metric for hierarchical clustering. volume 40 of Proceedings of Machine Learning Research, pages 588{606, Paris, France, 03{06 Jul 2015. PMLR.

M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, KDD'96, pages 226{231. AAAI Press, 1996.

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

A. L. N. Fred and A. K. Jain. Combining multiple clusterings using evidence accumulation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 27(6):835{850, 2005. doi: 10.1109/TPAMI.2005.113.

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

E. Gine and A. Guillou. Rates of strong uniform consistency for multivariate kernel den-sity estimators. In Annales de l'Institut Henri Poincare (B) Probability and Statistics, volume 38, pages 907{921. Elsevier, 2002.

A. D. Gordon. A Review of Hierarchical Classication. Journal of the Royal Statistical Society. Series A (General), 150(2):119, mar 1987.

S. Graf and H. Luschgy. Foundations of Quantization for Probability Distributions, volume 1730 of Lecture Notes in Mathematics. Springer Berlin Heidelberg, Berlin, Heidelberg, 2000. ISBN 978-3-540-67394-1. doi: 10.1007/BFb0103945.

S. Graf and H. Luschgy. Rates of Convergence for The Empirical Quantization Error. 30(2): 874{897, 2002.

J. A. Hartigan and P. M. Hartigan. The Dip Test of Unimodality. The Annals of Statistics, 13(1):70{84, mar 1985.

J. A. Hartigan and M. A. Wong. Algorithm AS 136: A K-Means Clustering Algorithm. Applied Statistics, 28(1):100, 1979.

C. Hennig. Methods for merging gaussian mixture components. Advances in Data Analysis and Classication 2010 4:1, 4:3{34, 1 2010.

T. Heskes. Self-organizing maps, vector quantization, and mixture modeling. IEEE Trans-actions on Neural Networks, 12(6):1299{1305, 2001.

L. Hubert and P. Arabie. Comparing partitions. Journal of Classication, 2(1):193{218, dec 1985.

M. C. Jones. Simple boundary correction for kernel density estimation. Statistics and computing, 3(3):135{146, 1993.

J. Kim, Y.-C. Chen, S. Balakrishnan, A. Rinaldo, and L. Wasserman. Statistical inference for cluster trees. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, Advances in Neural Information Processing Systems 29, pages 1839{1847. Curran Associates, Inc., 2016.

G. N. Lance and W. T. Williams. A General Theory of Classicatory Sorting Strategies: 1. Hierarchical Systems. The Computer Journal, 9(4):373{380, feb 1967.

J. Li. Clustering based on a multilayer mixture model. Journal of Computational and Graphical Statistics, 14(3):547{568, 2005. doi: 10.1198/106186005X59586.

J. Li, S. Ray, and B. G. Lindsay. A nonparametric statistical approach to clustering via mode identication. Journal of Machine Learning Research, 8(8), 2007.

S. P. Lloyd. Least Squares Quantization in PCM. IEEE Transactions on Information Theory, 28(2):129{137, 1982.

K. Lo, R. R. Brinkman, and R. Gottardo. Automated gating of ow cytometry data via ro-bust model-based clustering. In Cytometry Part A, volume 73, pages 321{332. Cytometry A, apr 2008.

R. Maitra. Initializing partition-optimization algorithms. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 6(1):144{157, 2009. doi: 10.1109/TCBB.2007.70244.

D. M. Mason, W. Polonik, et al. Asymptotic normality of plug-in level set estimates. The Annals of Applied Probability, 19(3):1108{1142, 2009.

G. Menardi and A. Azzalini. An advancement in clustering via nonparametric density estimation. Statistics and Computing, 24(5):753{767, 2014.

F. Murtagh. A Survey of Recent Advances in Hierarchical Clustering Algorithms. The Computer Journal, 26(4):354{359, nov 1983. doi: 10.1093/comjnl/26.4.354.

R. Nugent and W. Stuetzle. Clustering with condence: A low-dimensional binning approach. In Classication as a Tool for Research, pages 117{125. Springer, 2010.

A. D. Peterson, A. P. Ghosh, and R. Maitra. Merging k-means with hierarchical clustering for identifying general-shaped groups. Stat, 7(1):e172, 2018.

V. Polianskii and F. T. Pokorny. Voronoi graph traversal in high dimensions with applications to topological data analysis and piecewise linear interpolation. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '20, page 2154{2164, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450379984. doi: 10.1145/3394486.3403266.

D. Pollard. A Central Limit Theorem for k-Means Clustering. The Annals of Probability, 10 (4):919{926, 1982.

W. M. Rand. Objective Criteria for the Evaluation of Clustering Methods. Journal of the American Statistical Association, 66(336):846, dec 1971.

S. Ray and B. G. Lindsay. The topography of multivariate normal mixtures. The Annals of Statistics, 33(5):2042 { 2065, 2005. doi: 10.1214/009053605000000417.

A. Rinaldo, A. Singh, R. Nugent, and L. Wasserman. Stability of density-based clustering. Journal of Machine Learning Research, 13:905, 2012.

M. Rudemo. Empirical choice of histograms and kernel density estimators. Scandinavian Journal of Statistics, 9:65{78, 1982.

D. W. Scott. Multivariate density estimation: theory, practice, and visualization. John Wiley & Sons, 2015.

L. Scrucca. Identifying connected components in gaussian nite mixture models for clustering. Computational Statistics & Data Analysis, 93:5{17, 2016.

J. Shin, A. Rinaldo, and L. Wasserman. Predictive clustering, 2019.

B. W. Silverman. Density estimation for statistics and data analysis, volume 26. CRC press, 1986.

W. Stuetzle and R. Nugent. A generalized single linkage method for estimating the cluster tree of a density. Journal of Computational and Graphical Statistics, 19(2):397{418, 2010.

R. Tibshirani and G. Walther. Cluster validation by prediction strength. Journal of Computational and Graphical Statistics, 14(3):511{528, 2005. doi: 10.1198/106186005X59243.

R. Tibshirani, G. Walther, and T. Hastie. Estimating the number of clusters in a data set via the gap statistic. Journal of the Royal Statistical Society. Series B: Statistical Methodology, 63(2):411{423, jan 2001.

M. Tsimidou, R. Macrae, and I. Wilson. Authentication of virgin olive oils using principal component analysis of triglyceride and fatty acid proles: Part 1|classication of greek olive oils. Food Chemistry, 25(3):227 { 239, 1987.

P. Turner, J. Liu, and P. Rigollet. A statistical perspective on coreset density estimation, 2020.

G. Voronoi. Recherches sur les paralleloedres primitives. J. reine angew. Math, 134:198{287, 1908.

M. Walesiak and A. Dudek. The choice of variable normalization method in cluster analysis. In K. S. Soliman, editor, Education Excellence and Innovation Management: A 2025 Vision to Sustain Economic Development During Global Challenges, pages 325{340. International Business Information Management Association (IBIMA), 2020. ISBN 978-0-9998551-4-1.

M. P. Wand and M. C. Jones. Multivariate plug-in bandwidth selection. Computational Statistics, 9(2):97{116, 1994.

L. Wasserman. All of Nonparametric Statistics. Springer New York, 2006. doi: 10.1007/0-387-30623-4.

R. Weber, H.-J. Schek, and S. Blott. A quantitative analysis and performance study for similarity-search methods in high-dimensional spaces. In Proceedings of the 24rd International Conference on Very Large Data Bases, VLDB '98, pages 194{205, San Francisco, CA, USA, 1998. Morgan Kaufmann Publishers Inc. ISBN 1558605665.

Y. Zhao. A survey on theoretical advances of community detection in networks. Wiley Interdisciplinary Reviews: Computational Statistics, 9(5):e1403, 2017.

Appendices

A Computational Complexity

Knots construction. The first step of skeleton clustering is choosing knots, and, in this work, we take overfitting k-means as the default method. The k-means algorithm of Hartigan and Wong (Hartigan and Wong, 1979) has time complexity $O(ndkI)$, where n is the number of points, d is the dimension of the data, k is the number of clusters for k-means, and I is the number of iterations needed for convergence. When using overfitting k-means to choose knots, the reference rule is $k = \sqrt{n}$, and hence the complexity is $O(n^{3/2}dI)$. This is a time-consuming step of our clustering framework, and the complexity increases linearly with d. Therefore, preprocessing the data with dimension reduction techniques or using subject knowledge to choose knots can be helpful to speed up this process.

Edges construction. For the edge construction step, we approximate the Delaunay Triangulation with $DT(C)$ by looking at the 2-NN neighborhoods (the Voronoi Density regions in 3.1). Hence the main computational task for our edge construction step is the 2-nearest knot search. We used the k-d tree algorithm for this purpose, which gives the worst-case complexity of $O(ndk^{(1-1/d)})$. Notably, the computation complexity at this step is at the worst linear in d, which is a much better rate than computing the exact Delaunay Triangulation (exponential dependence on d), and our empirical studies have illustrated the effectiveness of such approximation.

Edge weight construction: VD. Next, we consider the computation complexity of the different edge weights measurements. For the VD, its numerator can be computed directly from the 2-NN search when constructing the edges and hence no additional computation is

needed. The denominators are pairwise distances between knots and can be computed with the worst-case complexity of $O(dk^2)$ because the number of nonzero edges is less than $\frac{k(k\ 1)}{2}$. With $k = \sqrt{n}$, we have the total time complexity of computing the VD to be $O(nd)$.

Edge weight construction: FD. For the Face density, we calculate the projected KDE at the middle point for each pair of neighboring Voronoi cells. The projection of one data point onto one central line can be done by matrix multiplication with complexity $O(d)$. Recall that we only use data points in local Voronoi cells for FD calculation, and the local sample size would be at $n_{loc} = O(\sqrt{n})$ under the conditions in Section 4 and the reference rule $k = [\sqrt{n}]$. Together it takes $O(d\sqrt{n})$ to calculate the projected data for one edge. With the projected data, KDE calculation has a time complexity $O(c \log c)$ where $c = max_{j=`}fn_j + n`g$ for any pair of knot indexes $j; `$. Again we have $c = O(n=k) = O(\sqrt{n})$ under the previously mentioned conditions. We need to do KDE for each edge in the skeleton, which gives the overall time complexity of FD weights to $O(k^2d\sqrt{n} + k^2c \log c) = O(n^{3=2}d + n^{3=2} \log n)$.

Edge weight construction: TD. For Tube density, we similarly perform a projected KDE for each edge. Let be the maximum number of points in a tube region $= max_{j;`}\, jfX_i : k_{j`}(X_i)\ \ X_i k\ Rgj$, the data projection again takes $O(d)$ complexity. Suppose the minimum density is obtained by a grid search with m grid points, the KDE step takes a total of $O(m \log)$ for one edge. To compute the whole edge weights matrix with $k = \sqrt{n}$, we have the complexity to be $O(nd + nm \log)$. Under conditions where the tube regions for TD estimations is also of size $= O(n=k) = O(\sqrt{k})$, we have the overall complexity for VD weights calculation to be $O(k^2d\sqrt{n} + k^2c \log c) = O(n^{3=2}d + mn^{3=2} \log n)$,
which is larger than that for FD due to the grid search for minimum density.

Knots segmentation. In this work, we segment the learned weighted skeleton using hierarchical clustering. With links that can be updated by Lance-Williams update (Lance and Williams, 1967) and satisfy the reducibility condition (Gordon, 1987), hierarchical clustering can be carried out with computation complexity $O(N^2)$, where N is the number of points to start the algorithm with (Murtagh, 1983). For our empirical results, we favored single linkage and average linkage, and both satisfy the requirements for an ecient hierarchical clustering algorithm. We perform hierarchical clustering on the $k = \sqrt{n}$ knots, and hence the computation complexity for segmenting the skeleton structure is $O(k^2) = O(n)$.

B Theory for Face Density

Here we derive the convergence rate of the Face Density estimator. Recall that $_d$ is the Lebesgue measure on the d-dimensional Euclidean space and $F_{j'} = C_{'} \setminus C_j$ is the face region between knots $c_j; c_{'}$. Let $@F_{j'}$ be the boundary of $F_{j'}$. We consider the following assumptions:

(D1) (Density conditions) The PDF p has compact support X, is bounded away from zero that $\inf_{x2X} p(x) \ p_{min} > 0$, $\sup_{x2X} p(x) \ p_{max} < 1$, and is Lipschitz continuous.

(B2) (Bounded face region) There exist constants $c_0; c_1$ such that the face area

$$\frac{c_0}{k^{1 \ \frac{1}{d}}} \quad \min_{(j;')2E} \ _{d \ 1}(F_{j'}) \quad \max_{(j;')2E} \ _{d \ 1}(F_{j'}) \quad \frac{c_1}{k^{1 \ \frac{1}{d}}}$$

(B3) (Boundary of face bounded) There exists a constant c_2 such that

$$\max_{(j;')2E} \ _{d \ 2}(@F_{j'}) \quad \frac{c_2}{k^{1 \ \frac{2}{d}}};$$

(B4) (Intersecting angle condition) There is an angle $_0 <$ such that, for every pair of intersecting face regions F_{ij} and $F_{j'}$, the maximal principle angle between the two

subspaces $_{ij;j'}$ satises $_{ij;j'}$ 0

(K1) (Kernel function conditions) The kernel function K is a positive and symmetric function satisfying $\int K^2(x)dx < 1$; $\int |x|K(x)dx < 1$; $\int x^2K(x)dx < 1$:

Assumption (D1) is commonly assumed for the density estimation problem, but usually with higher-order smoothness conditions. Notably, for consistency of the FD estimator, we require only the Lipschitz condition since the bias of the sample estimator will be dominated by a geometric dierence even if we have a higher-order smoothness (see the discussion after Theorem 3 and Appendix D for more detail). Condition (B2) restricts the shared boundary of two Voronoi cells to scale at the rate of $O(k^{-\frac{1}{d}})$. While this condition may seem abstract, it is a mild condition. To illustrate this, suppose we have $k = m^d$ points that are on a uniform grid of $[0;1]^d$ for some integer m. We form the Voronoi cells of these grid points. The (d 1)-dimensional volume of the shared boundary of two neighboring Voronoi cells will scale at rate $O(k^{-\frac{1}{d}})$ as k ! 1. (B3) requires the boundaries of the face regions to scale at most at a rate of $O(k^{-\frac{2}{d}})$, and (B4) requires that we cannot have two nearby faces to be parallel to each other. Assumptions (B3) and (B4) are needed when bounding the geometric dierence between the estimator and the population quantity and are both mild conditions: When the knots form a spherical packing of a smooth region, these conditions hold. Notably, (D1) and (B2) imply (B1) and hence the consistency of FD requires more conditions than the consistency of VD. The condition (K1) is a common assumption on the kernel function (Wasserman, 2006; Scott, 2015) satised by many common kernel functions, including the Gaussian kernel.

Theorem 3 (Face Density). Assume (D1), (K1), and (B2-B4). With h ! 0, k ! 1,

$hk^{1=d} ! \ 0, \frac{nh}{k^{1 \ \frac{1}{d}}} ! \ 1$, then for any pair $j = `$, we have

$$\frac{\hat{S}_{j,`}^{F,D}}{S_{j,`}^{F,D}} 1 = O\left(hk^{1=d} + O_p\left(\sqrt{\frac{k^{1 \ \frac{1}{d}}}{nh}}\right)\right) \tag{15}$$

Theorem 3 shows the convergence rate of estimating the FD. Roughly speaking, the rate is similar to a 1-dimensional density estimation problem. With $d ! \ 1$, we have the rate to be $O\left(h + O_p\left(\sqrt{\frac{k}{nh}}\right)\right) = O\left(h + O_p\left(\sqrt{\frac{1}{n_{loc}h}}\right)\right)$, where $n_{loc} = \Theta\left(\frac{n}{k}\right)$ is the local eective sample size. Therefore, the eect of the ambient dimension is negligible when d is large, and this is because we are estimating a 'projected' density on the central line, which reduces to a 1-dimensional problem.

Noticeably, the bias term in Theorem 3 is of the order O(h). While this rate is optimal under the Lipschitz smoothness (D1) for density estimation problem, it is slower than the conventional rate $O(h^2)$ when we have a bounded second-order derivative of p. One may be wondering if higher-order smoothness of p is assumed, can we improve the convergence rate? Unfortunately, even if p is very smooth, the bias rate will still stay the same at O(h). This is because there are two sources of bias. The rst one is the usual bias from kernel smoothing, which can be improved to higher order if we have high-order derivatives of p. The other source of bias comes from the dierent geometric shapes of the Voronoi cells C_j and $C_`$ (for illustration see Figure 12 in Appendix D). Consider the characterization of central line as $c_j + t(c_` \ c_j)$ for $t 2 [0; 1]$, and the boundary will occur at $t = \frac{1}{2}$. Regions projected onto the central line will be dierent depending on the value of t. Specically, when $t > \frac{1}{2}$, the projected region is from $C_`$ whereas when $t < \frac{1}{2}$, the projected region is from C_j, and those projected regions can have shapes dierent from the face region. This dierence leads

to an additional geometric bias of the order O(h) and cannot be improved by higher-order smoothness of p. In a sense, this bias O(h) is similar to the boundary bias in that the density function is continuous but not dierentiable. However, since the non-dierentiability is caused by the geometric dierence in two nearby Voronoi cells, it is unclear if we can use the conventional boundary-correction kernels (Jones, 1993) to correct this bias.

From Theorem 3, one can see that the optimal bandwidth scales at rate $h \asymp \left(\frac{k^{1 \ 3=d}}{2n}\right)^{1=3}$. Recall that our reference rule sets $k = \sqrt{n}$ so that $n_{loc} = \frac{n}{k} = \sqrt{n}$ is the average number of observations per each knot. When d large, $\frac{3}{d}$ is negligible. Thus, the optimal bandwidth is given by $h \asymp \left(\frac{k}{n}\right)^{1=3} = n_{loc}^{-1=3}$. While our empirical rule $n_{loc}^{-1=5}$ is not optimal in this case, it still gives a consistent estimator and our empirical analysis shows that such choice leads to reliable clustering results; see Appendix F.5.

One may notice that a small k in Theorem 3 leads to a better convergence rate, which suggests to use a small k. While this is true from the perspective of estimation, overall a small k may lead to a poor representation of the data and result in a bad clustering performance. Empirical results show that we need a suciently large number of knots to represent the data in order for the skeleton clustering to perform appropriately. Therefore, our reference rule with $k = \sqrt{n}$ is a suitable balance between the trade-o between representation and estimation. We include an empirical analysis of the eect of k on clustering performance in Appendix F.1.

C Theory for Tube Density

In this section we derive the convergence rate of the Tube Density estimator. We consider the following assumptions, which are slightly stronger than the corresponding ones in the case of the FD:

(D2) (Density conditions) The PDF p has a compact support and is 3-Hölder and $\inf_{x \in \mathcal{X}} p(x) \geq f_{min} > 0$.

(D3) (Disk Density conditions) For any pair $c_j, c_{j'}$, the minimum disk density location $t^* = \operatorname{argmin}_{t \in [0,1]} pDisk_{j',R}(t) \in (0,1)$ is unique and the second derivative of the disk density $pDisk^{(2)}_{j',R}(t^*) \geq c_{min} > 0$.

(K2) (Kernel function conditions) The kernel function K is a positive and symmetric function satisfying $\int x^2 K^{(\alpha)}(x)dx < \infty$; $\int (K^{(\alpha)}(x))^2 dx < \infty$; for all $\alpha = 0, 1, 2$, where $K^{(\alpha)}$ denotes the α-th order derivative of K.

(D2) is a stronger version of (D1) that we require additional smoothness condition of p. We need the 3-Hölder class (slightly weaker than the requirement of third-order derivatives) to obtain the rate of estimating the minimum (Chacon et al., 2011; Chen et al., 2016). Also, a stronger condition (K2) on the kernel function is needed to ensure the gradient estimation is consistent. Fortunately, common kernel functions such as the Gaussian kernel satisfy these conditions.

Theorem 4 (Tube Density Consistency). Assume (D2), (D3), and (K2). Let $h \to 0$, $k \to \infty$, $R \to 0$, $nh^3 \to \infty$, $nhR^{d-1} \to \infty$. Suppose that for every pair $c_j, c_{j'}$, $\inf_{t \in [0,1]} pDisk_{j',R}(t)$ and $\inf_{t \in [0,1]} pDisk_{j',R}(t)$ do not occur at the boundary $t = 0, 1$. Then for any pair $j = j'$ that

shares an edge, we have

$$\widehat{\text{pDisk}}_{j',R}(t) = \text{pDisk}_{j',R}(t) + O(R^{d-1}); \tag{16}$$

$$\frac{\widehat{S}_{j',p}}{S_{j',p}} - 1 = O(h^2) + O_p\left(\sqrt{\frac{1}{nhR^{d-1}}}\right) + O_p\left(\frac{1}{nh^3}\right) \tag{17}$$

Theorem 4 shows that the TD estimator converges to the population TD with a rate consisting of three components. We allow $R \to 0$ as $n \to \infty$ but this result also applies to scenarios where R is fixed. The first component $O(h^2)$ is the usual smoothing bias. The second component $O_p\left(\sqrt{\frac{1}{nhR^{d-1}}}\right)$ is similar to the stochastic variation part from usual KDE but with an additional dependence on R^{d-1}. This is due to the fact that, when $R \to 0$, we are using fewer and fewer observations to perform smoothing, and nR^{d-1} serves as the effective sample size. The third component $O_p\left(\frac{1}{nh^3}\right)$ is due to the error of estimating the location of the minimum. It is a squared term because the density behaves like a quadratic function around its minimum due to (D3).

While the convergence rate of TD requires stronger conditions (D2) and (K2) compared to the conditions (D1) and (K1) when estimating the FD, the TD estimator has a smaller bias than the FD estimator (comparing Theorem 3 and 4). This is because the TD is evaluated on a "regular shape", which leads to a smoother quantity being estimated.

For the stochastic variation part, the second term in Theorem 4 gives $O_p\left(\sqrt{\frac{1}{nhR^{d-1}}}\right)$ while the second term in Theorem 3 gives $O_p\left(\sqrt{\frac{k^{1-\frac{1}{d}}}{nh}}\right)$. Note that empirically we choose R to be the average of the root mean squared distances of each Voronoi cell (Section 3.3), which is of order $O(k^{-1/d})$ with cell sizes to have the same rates. Hence $k^{1-1/d}$ and $\frac{1}{R^{d-1}}$ are at the same rate and the stochastic variation part is comparable for TD and FD estimators.

However, for TD we have another source of variation coming from the uncertainty of the location of minimum, which can cause TD to have a larger variation than the FD estimator.

Based on the above reasoning, our choice of R leads to $R^{d-\frac{1}{1}}$ $k^{1\ 1=d}$, which implies the rate $O(h^2) + O_p\left(\sqrt{\frac{k^{1\ 1=d}}{nh}}\right) + O_p\left(\frac{1}{nh^3}\right)$. Under our reference rule $k = \sqrt{n}$ the optimal bandwidth is $h \asymp n^{(\frac{1}{5}+d)^{\frac{1}{1}}}$. Recall that the local sample size is about $n_{loc} = n=k = \sqrt{n}$ and hence the optimal bandwidth is $h \asymp n_{loc}^{(\frac{1}{5}+d)^{\frac{1}{1}}}$. When $d \to 1$, this leads to $h \asymp n_{loc}^{-1=5}$, which is the same rate on sample size as given by the Silverman's rule of thumb.

Remark 6. Similar uniform bounds of the Face and Tube density can be derived with an extra $\log k$ factor in the rates through the concentration bound for kernel density estimator (Gine and Guillou, 2002). Also, similar concentration bounds on the Adjusted Rand Indexes can be achieved for partition based on the Face and Tube density.

D Proofs

D.1 Proofs for Voronoi Density Results

We restate the assumption:

(B1) There exists a constant c_0 such that the minimal knot size $\min_{(j;\prime)2E} P(A_{j\prime}) \geq \frac{c_0}{k}$ and $\min_{(j;\prime)2E} kc_j - c_\prime k \geq \frac{c_0}{k^{1=d}}$, where $A_{j\prime}$ is the 2-NN region of knots $c_j; c_\prime$ as dened in Equation 2.

Proof of Theorem 1.

For given knots $c_j; c_\prime$, the distance $jjc_j - c_\prime jj$ is also given. We denote the numerator of $S_{j,\prime}^{VD}$ as

$$p_{j\prime} = P(A_{j\prime}) = EI(X_i : d(X_i; c_m) > \max\{d(X_i; c_j); d(X_i; c_\prime); 8m = j; lg)$$

and note that the numerator of $\hat{S}_{j'}^{VD}$ is

$$\hat{P}_n(A_{j'}) = \frac{1}{n}\sum_{i=1}^{n} I\left(X_i : d(X_i; c_m) > \max\{d(X_i; c_j); d(X_i; c'); \forall m = j; l\}\right);$$

which is a sum of binary variables and has variance $\sigma_{j'}^2 = \frac{p_{j'}(1-p_{j'})}{n}$. By Chebyshev's inequality,

$$\hat{P}_n(A_{j'}) - p_{j'} = O_p(\sigma_{j'}^{1/2}) = O_p\left(\frac{p_{j'}(1-p_{j'})^{1/2}}{n}\right)$$

Note that the region $A_{j'}$ is changing with respect to k. The ratio is then

$$\frac{\hat{S}_{j'}^{VD}}{S_{j'}^{VD}} - 1 = \frac{\hat{P}_n(A_{j'})}{P(A_{j'})} - 1 = \frac{1}{p_{j'}} O_p\left(\frac{p_{j'}(1-p_{j'})^{1/2}}{n}\right)$$

$$= O_p\left(\frac{(1-p_{j'})^{1/2}}{np_{j'}}\right) = O_p\left(\frac{(1-c_0/k)^{1/2}}{nc_0/k}\right) = O_p\left(\frac{k^{1/2}}{n}\right)^p$$

by assumption (B1) that $\min_{(j;')\in E} P(A_{j'}) \geq \frac{c_0}{k}$, which completes the proof for Equation 12.

To get the uniform bound, we first start with the concentration bound. Note that $I(X_i \in A_{j'}) - p_{j'}$ has zero mean and $|I(X_i \in A_{j'}) - p_{j'}| \leq 1$. Hence by Bernstein's inequalities, we have

$$P\left(\frac{\hat{P}_n(A_{j'})}{p_{j'}} - 1 > \varepsilon\right) = P\left(\hat{P}_n(A_{j'}) - p_{j'} > \varepsilon p_{j'}\right)$$

$$= P\left(\frac{1}{n}\sum_{i=1}^{n} I(X_i \in A_{j'}) - p_{j'} > \varepsilon p_{j'}\right)$$

$$= 2P\left(\sum_{i=1}^{n} (I(X_i \in A_{j'}) - p_{j'}) > n\varepsilon p_{j'}\right)$$

$$\leq 2\exp\left(-\frac{\frac{1}{2}\varepsilon^2 p_{j'}^2 n^2}{\sum_{i=1}^{n} E(I(X_i \in A_{j'}) - p_{j'})^2 + \frac{1}{3}\varepsilon p_{j'} n}\right)$$

$$= 2\exp\left(-\frac{\frac{1}{2}\varepsilon^2 p_{j'}^2 n^2}{np_{j'}(1-p_{j'}) + \frac{1}{3}\varepsilon p_{j'} n}\right)$$

$$= 2\exp\left(-\frac{\frac{1}{2}\varepsilon^2 p_{j'}^2 n}{p_{j'}(1-p_{j'}) + \frac{1}{3}\varepsilon p_{j'}}\right)$$

Note that plugging in the $p_{j'} = \frac{1}{k}$ rate to above concentration bound we can recover the

$O_p\left(\sqrt{\frac{k}{n}}\right)$ rate in Equation 12. Then by union bound we have

$$
\begin{aligned}
P\left(\max_{(j,j')\in S} |\hat{S}_{j,j'} - S_{j,j'}| > \varepsilon\right) &\le P\left(\max_{j,j'} |\hat{S}_{j,j'} - S_{j,j'}| > \varepsilon\right) \\
&\le \sum_{j,j'} P\left(|\hat{S}_{j,j'} - S_{j,j'}| > \varepsilon\right) \\
&\le \frac{k(k-1)}{2} \max_{j,j'} P\left(\left|\frac{\hat{p}_n(A_{j,j'})}{p_{j,j'}} - 1\right| > \varepsilon\right) \\
&\le k(k-1) \max_{j,j'} \exp\left(-\frac{\frac{1}{2}\varepsilon^2 p_{j,j'}^2 n}{p_{j,j'}(1-p_{j,j'}) + \frac{1}{3}\varepsilon p_{j,j'}}\right) \\
&\le k(k-1) \exp\left(-\frac{\frac{1}{2}\varepsilon^2 p_{\min} n}{(1-p_{\min}) + \frac{1}{3}\varepsilon}\right)
\end{aligned}
$$

where $p_{\min} = \min_{j,j'} p_{j,j'}$. Therefore we can derive the uniform error bound that

$$\max_{j,j'} \frac{\hat{S}_{j,j'}^{VD}}{S_{j,j'}^{VD}} - 1 = O_p\left(\sqrt{\frac{k}{n}\log k}\right);$$

when $n \to \infty$; $k \to \infty$; $\frac{n}{k} \to \infty$.

For comprehensiveness, we provide the denition of the adjusted Rand Index below. For two partitions $X = \{X_1, \ldots, X_r\}$ and $Y = \{Y_1, \ldots, Y_s\}$, let $n_{ij} = |X_i \cap Y_j|$, we have the contingency table

	Y_1	Y_2	$\cdots$	Y_s	Sums
X_1	n_{11}	n_{12}		n_{1s}	a_1
X_2	n_{21}	n_{22}		n_{2s}	a_2
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
X_r	n_{r1}	n_{r2}		n_{rs}	a_r
Sums	b_1	b_2		b_s	

And the Adjusted Rand Index (ARI) adjusting for permutation chance is

$$\mathrm{ARI} = \frac{\sum_{ij}\binom{n_{ij}}{2} - \left[\sum_i \binom{a_i}{2}\sum_j \binom{b_j}{2}\right]\Big/\binom{n}{2}}{\frac{1}{2}\left[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}\right] - \left[\sum_i \binom{a_i}{2}\sum_j \binom{b_j}{2}\right]\Big/\binom{n}{2}}$$

Proof. of Theorem 2 (Performance guarantee for Voronoi density) We note that, assuming (P1),

$$\begin{aligned} P\{\mathrm{ARI}(L, L_0) < 1\} &= P\{\text{there exists at least one wrongly cut edge}\} \\ &= P\Big\{\max_{(j,j')\in S} |\hat{S}_{j,j'} - S_{j,j'}| > \varepsilon\Big\} \\ &\le k(k-1)\exp\left(\frac{\frac{1}{2}\varepsilon^2 p_{\min} n}{(1-p_{\min}) + \frac{1}{3}\varepsilon}\right) \end{aligned}$$

by the uniform bound derived above.

D.2 Proofs for Face Density Consistency

Let $p(x)$ be the density function of the data distribution, let μ_d be the Lebesgue measure on the d-dimensional Euclidean space, let $F_{j,j'} = C_j \cap C_{j'}$ denote the face between knots $c_j, c_{j'}$, and let $\partial F_{j,j'}$ be the boundary of $F_{j,j'}$. We consider the following assumptions: Again, we recall the assumptions:

(D1) (Density conditions) The PDF p has compact support $\mathcal{X}$, is bounded away from zero that $\inf_{x\in\mathcal{X}} p(x) \equiv p_{\min} > 0$, $\sup_{x\in\mathcal{X}} p(x) \equiv p_{\max} < 1$, and is Lipschitz continuous.

(B2) There exist constants c_0, c_1 such that the face area

$$\frac{c_0}{k^{1-\frac{1}{d}}} \le \min_{(j,j')\in E} \mu_{d-1}(F_{j,j'}) \le \max_{(j,j')\in E} \mu_{d-1}(F_{j,j'}) \le \frac{c_1}{k^{1-\frac{1}{d}}}$$

(B3) There exists a constant c_2 such that $\max_{(j,j')\in E} \mu_{d-2}(\partial F_{j,j'}) \le \frac{c_2}{k^{1-\frac{2}{d}}}$

(B4) There is an angle $_0 <$ such that, for every pair of intersecting face regions F_{ij} and $F_{j'}$, the maximal principle angle between the two subspaces $_{ij;j'}$ satises $_{ij;j'}$ $_0$

(K1) (Kernel function conditions) The kernel function K is a positive and symmetric function satisfying $\int K^2(x)dx < 1$; $\int |x|K(x)dx < 1$; $\int x^2K(x)dx < 1$:

Proof of Theorem 3.

Our analysis starts with the usual bias-variance decomposition that

$$\hat{S}_{j'}^{F,D} - S_{j'}^{F,D} = \underbrace{\hat{S}_{j'}^{F,D} - E(\hat{S}_{j'}^{F,D})}_{\text{stochastic variation}} + \underbrace{E(\hat{S}_{j'}^{F,D}) - S_{j'}^{F,D}}_{\text{bias}}:$$

We analyze the two term separately. Before we start our proof, we rst recall some useful notations.

Recall that the face region between two knots $c_j; c_{j'}$ is $F_{j'} = \overline{C_j} \setminus \overline{C_l}$ and $c = c_j + \frac{1}{2}(c_{j'} - c_j) = \frac{1}{2}(c_{j'} + c_j)$ and $L_{j'} = \{c_j + a(c_{j'} - c_j) : a \in [0;1]\}$ is the central line passing through c_j and $c_{j'}$, and for a value $a \in [0;1]$. The face $F_{j'} = \{x \in C_j \cap C_l : _{j'}(x) = c\}$, where $_{j'}$ denotes the projection onto $L_{j'}$. The quantity $_s(dx)$ denotes the integration with respect to s-dimensional volume. We now reparametrize any point in $L_{j'}$ using a unit distance t.

Let $T_{j';t} = \{x \in X : _{j'}(x) = c + t\frac{c_{j'} - c_j}{\|c_{j'} - c_j\|}\}$ be the subspace orthogonal to $L_{j'}$ at the point $c + t\frac{c_{j'} - c_j}{\|c_{j'} - c_j\|}$. t is 1-dimensional distance to c along the line passing through c_j and $c_{j'}$. Let

$$q_{j'}(t) = \int_{(\overline{C_j} \cap \overline{C_{j'}}) \setminus T_{j';t}} p(x)_{d-1}(dx)$$

With these quantities, $S_{j'}^{F,D} = q_{j'}(0)$ and that $q_{j'}(t)$ is a 1-dimensional quantity. Our estimator is

$$\hat{S}_{j'}^{F,D} = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{_{j'}(X_i) - c}{h}\right) I(X_i \in C_j \cap C_{j'}):$$

Bias: We study the bias part rst. A direct computation shows that

$$E[\hat{\beta}^{FD}_{j'}] = E\left[\frac{1}{nh}\sum_{i=1}^{n} K^{j'}(X_i)\,\frac{c}{h}\,I(X_i \in C_j \cup C')\right] \tag{18}$$

$$= \frac{1}{h}\int_{x\in X} K^{j'}(x)\,\frac{c}{h}\,I(x \in \overline{C}_j \cap \overline{C}')p(x)d(dx) \tag{19}$$

$$= \frac{1}{h}\int_{L_{j'}} K\left(\frac{c + t\frac{c' - c_j}{\|c' - c_j\|}}{h}\right)\int_{(\overline{C}_j \cap \overline{C}')\cap T_{j';t}} p(y)d^{-1}(dy)\,d\left(c_j + t\frac{c' - c_j}{\|c' - c_j\|}\right) \tag{20}$$

$$= \frac{1}{h}\int_{R} K\left(\frac{t\|c' - c_j\|}{h}\right) q_{j'}(t)dt \tag{21}$$

$$= \frac{1}{h}\int_{R} K\left(\frac{t}{h}\right) q_{j'}(t)dt \tag{22}$$

$$= \int_{R} K(u)q_{j'}(hu)du; \tag{23}$$

where for the third equality, we split the integration with respect to $c_j + t\frac{c' - c_j}{\|c' - c_j\|} \in L_{j'}$ and the integration with respect to the subspace orthogonal to $L_{j'}$ at $c_j + t\frac{c' - c_j}{\|c' - c_j\|}$. This is possible because all the points in $T_{j';t}$ have the same projection onto $L_{j'}$. For the fourth equality, we used the symmetry of the kernel function. the property of the kernel function that $K(x) = K(\|x\|)$. For the last equality, we used the change of variable that $u = \frac{t}{h}$ and got the simplied form.

The expansion of

$$q_{j'}(t) = \int_{(\overline{C}_j \cap \overline{C}')\cap T_{j';t}} p(y)d^{-1}(dy)$$

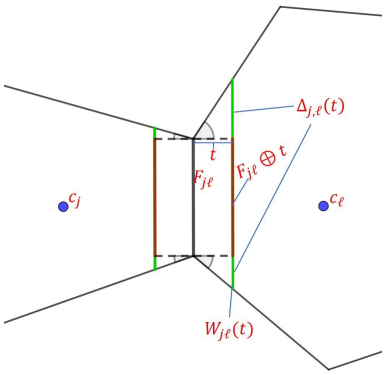


Figure 12: Decomposition of $W_{j'}(t)$. The dark red segment is $F_{j'}$ t, which has the same shape with $F_{j'}$. The green segments consist $_{j;'}(t)$, the part leading to geometric bias.

is more involved when t 0. Let

$$W_{j'}(t) = (\overline{C}_j \ [\ \overline{C}_{'}) \setminus T_{j';t}$$

$$= \begin{cases} \overline{C}_j \setminus T_{j';t}; & t < 0; \\ \overline{C}_{'} \setminus T^{j}_{';t}; & t > 0; \\ (\overline{C}_j \ [\ \overline{C}_{'}) \setminus T_{j';0} = F_{j'}; & t = 0 \end{cases}$$

be the region that leads to $q_{j'}(t)$. For a face $F_{j'}$ and a real number t 2 R, we denote

$$F_{j'} \ t = \left\{ x + t\frac{c_{'} \quad c_j}{\|c_{'} \quad c_j\|} : x \ 2 \ F_{j'} \right\}$$

By the above notation, we can decompose

$$W_{j'}(t) = [F_{j'} \ t] [\ _{j;'}(t);$$

where $_{j;'}(t)$ is the additional region when moving away from t = 0; see Figure 12 for an example.

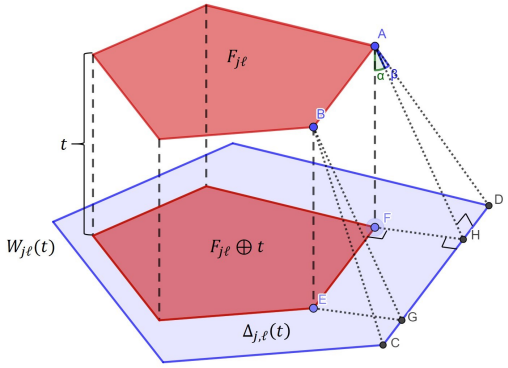


Figure 13: Decomposition of $W_{j'}(t)$. The red regions are $F_{j'}$ and the projected $F_{j'}$ t, while the blue band region denotes $_{j;'}(t)$. All the angles such as $\backslash F A H$ and all the angles such as $\backslash H A D$ are bounded by $_0$ from assumption (B4).

Thus, the dierence

$$q_{j'}(hu) \quad q_{j'}(0) = \int_{W_{j'}(hu)} p(y)\,d_{\ 1}(dy) \quad \int_{W_{j'}(0)} p(y)\,d_{\ 1}(dy)$$

$$= \underbrace{\int_{F_{j'}hu} p(y)\,d_{\ 1}(dy) \quad \int_{F_{j'}} p(y)\,d_{\ 1}(dy)}_{(I)} + \underbrace{\int_{_{j'}(hu)} p(y)\,d_{\ 1}(dy)}_{(II)} :$$

(I) is the usual bias caused by the change of density. Note that the Lipchitz condition in (D1) implies that there is a constant C_g such that $jp(x_1) \quad p(x_2)j \ C_g jx_1 \quad x_2 j$. Since every point can be matched nicely between $F_{j'}$ hu and $F_{j'}$, it can be bounded by

$$j(I)j \ _{d\ 1}(F_{j'})C_g hjuj:$$

(II) is the bias due to the change of volume, so we call it a geometric bias. With an upper bound of the density, (II) can be bounded by $(II) \ _{d\ 1}(_{j;'}(hu)) \ p_{max}$: Thus, we only need to bound the volume $_{d\ 1}(_{j;'}(hu))$.

$_{j;'}(t)$ is illustrated by the blue region in Figure 13. The width of the band region like F H will all be bounded by $t\tan(_0) = O(t)$, and as t ! 0 the surface area (circumference) will be bounded by $O \ _{d\ 2}(@F_{j'})$.

Thus, the volume of the blue region $_{d\ 1}(_{j;'}(t))\ \ O\ _{d\ 2}(@F_{j'})t$, which leads to the bound

$$(II)\ \ O\ hjuj\ _{d\ 2}(@F_{j'})\ p_{max}:$$

Putting altogether, we have

$$q_{j'}(hu)\quad q_{j'}(0)\ _{d\ 1}(F_{j'})C_g hjuj + p_{max}hjuj\ O\ _{d\ 2}(@F_{j'})\tan(_0) \tag{24}$$

This, together with equation (23), implies that

$$jE[S^{F\,D}]\quad q_{j'}(0)j = \int_R K(u)[q_{j'}(hu)\quad q_{j'}(0)]du$$

$$\int K(u)jq_{j'}(hu)\quad q_{j'}(0)jdu$$

$$h\qquad jujK(u)du\quad _{d\ 1}(F_{j'})C_g + p_{max}O\ _{d\ 2}(@F_{j'})$$

$$= O\ \ h\ \ \frac{1}{k^{1\ 1=d}} + O\ \ h\ \ \frac{1}{k^{1\ 2=d}} \tag{B2 3}$$

As a result,

$$jE[\hat{S}_{j'}^{F,D}]\quad S_{j'}^{F,D}j = O\ \ \frac{h}{k^{1\ 1=d}} + O\ \ \frac{h}{k^{1\ 2=d}} \tag{25}$$

Moreover, note that

$$\frac{h}{k^{1\ 1=d}}\quad \frac{k^{1\ 2=d}}{h} = \frac{1}{k^{1=d}} ! \ \ 0 \tag{26}$$

since $k ! \ 1$. Therefore the bias given by the geometric dierence (II) dominates the bias given by the change in density (I). Even if we assume a higher-order derivative, the bias in (II) will still dominate the component in (I).

Therefore, the overall bias can be expressed as reduces to

$$jE[S_{j'}^{D}]\quad S_{j'}^{D}j = O\ \ \frac{}{k^1h^{2=d}} \tag{27}$$

Stochastic variation: For the stochastic variation part, we have

$$\begin{aligned} Var(\hat{S}_{j'}^{F,D}) &= Var\left[\frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{j'(X_i) - c)}{h}\right) I(X_i \in C_j) [C')\right] \\ &\frac{1}{nh^2} E\left[K^2\left(\frac{j'(X_i)}{h}\right) I(X_i \in C_j) [C')\right] \\ &\frac{1}{nh}\int K^2(u)\ q_{j'}(0) + {}_{d-1}(F_{j'})C_g + p_{max}h|u|_{d-2}(@F_{j'})\tan(_0)\ du \\ &\frac{1}{nh}\int K^2(u)\ q_{j'}(0) + O\left(\frac{h}{k^{1-1=d}}\right) + O\left(\frac{h}{k^{1-2=d}}\right) du \end{aligned} \tag{28}$$

by the same decomposition in (23) and the bound in (24) and the assumptions (K1). Note that similar to(26), the second term in (28) is at a slower rate than the third term, so we can simplify it as

$$Var(\hat{S}_{j'}^{F,D}) = O\left(\frac{q_{j'}(0)}{nh}\right) + O_{nk^{1-2=d}:1} \tag{29}$$

Combining (25) and (28), we conclude that for $8j; '$,

$$|\hat{S}_{j'}^{F,D} - S_{j'}^{F,D}| = O_{k^{1-2=d}}\left(\frac{h}{}\right) + O_p\left(\frac{r}{q_{j'}(0)}\right) + O_p\left(\frac{1}{nk^{1-2=d}}\right) \tag{30}$$

Note that the volume of face region $F_{j'}$ decreases when k increases. By assumption (D1) and (B2), we have

$$q_{j'}(0) = S_{j'}^{F,D} \quad p_{min} \min_{(j;')2E} {}_{d-1}(F_{j'}) = p_{min}\, k^{1-\frac{1}{d}} \frac{c_0}{} \tag{31}$$

For the theorem, we again take the ratio between the estimated and the true face density to accommodate the fact that the true face density is decreasing with the number of knots, and we have that This implies that

$$\frac{\hat{S}_{j'}^{F,D}}{S_{j'}^{F,D}} - 1 = O\left(hk^{1=d}\right) + O_p\left(\frac{s}{k^{1-\frac{1}{d}}}\right) + O_p\left(\frac{r}{nh}\right) + O_p\left(\frac{k}{n}\right) \tag{32}$$

When $hk^{1=d} \to 0$,

$$\frac{k^{1-\frac{1}{d}}}{nh} \quad \frac{n}{k} = \frac{1}{hk^{1=d}} \to 1; \tag{33}$$

so the second term dominates the third term in (32) and the rate reduces to

$$\frac{\hat{S}^{FD}_{j,j'}}{S^{FD}_{j,j'}} - 1 = O\left(hk^{1-d}\right) + O_p\left(\sqrt{\frac{k^{1-d}}{nh}}\right); \tag{34}$$

which completes the proof.

D.3 Proofs for Tube Density Consistency

We consider the following assumptions, which are slightly stronger than those in the case of the FD:

(D2) (Density conditions) The PDF p has compact support, is in the 3-Hölder class, and $\inf_{x \in \mathcal{X}} p(x) \geq f_{min} > 0$.

(D3) (Disk Density conditions) For any pair $c_j, c_{j'}$, the minimum disk density location $t^* = \operatorname{argmin}_{t \in [0,1]} pDisk_{j',R}(t) \in (0,1)$ is unique and satisfies $pDisk^{(2)}_{j',R}(t^*) \geq c_{min} > 0$.

(K2) (Kernel function conditions) The kernel function K is a positive and symmetric function satisfying $\int x^2 K^{(\ell)}(x)dx < \infty$; $\int (K^{(\ell)}(x))^2 dx < \infty$; for all $\ell = 0, 1, 2$, where $K^{(\ell)}$ denotes the ℓ-th order derivative of K.

Proof of Theorem 4.

Let $t^* = \operatorname{argmin}_t pDisk_{j',R}(t)$ and $\hat{t}^* = \operatorname{argmin}_t \hat{p}Disk_{j',R}(t)$. Then the tube densities

$$S^{TD}_{j,j'} = \inf_{t \in [0,1]} pDisk_{j',R}(t) = pDisk_{j',R}(t^*);$$

$$\hat{S}^{TD}_{j,j'} = \inf_{t \in [0,1]} \hat{p}Disk_{j',R}(t) = \hat{p}Disk_{j',R}(\hat{t}^*).$$

Since the ratio dierence

$$\frac{\hat{S}^{TD}_{j,j'}}{S^{TD}_{j,j'}} - 1 = \frac{1}{S^{TD}_{j,j'}}\left(\hat{S}^{TD}_{j,j'} - S^{TD}_{j,j'}\right);\ j'$$

we will focus on the dierence $\hat{S}^{DD}_{j',R} - S^{TD}_{j',R}$.

The dierence admits the following decomposition:

$$\begin{aligned}\hat{S}^{DD}_{j',R} - S^{TD}_{j',R} &= \hat{p}Disk_{j',R}(t) - pDisk_{j',R}(t) \\ &= \underbrace{\hat{p}Disk_{j',R}(t) - pDisk_{j',R}(t)}_{(I)} + \underbrace{pDisk_{j',R}(t) - E(\hat{p}Disk_{j',R}(t))}_{(II)} \\ &\quad + \underbrace{E(\hat{p}Disk_{j',R}(t)) - pDisk_{j',R}(t)}_{(III)}.\end{aligned}$$

It is easier to start with term (III) and then term (II) and then term (I).

Recall that

$$q_{v,R}(y) = \int_{Disk(y;R;v)} p(x)dx;$$

and hence $pDisk_{j',R}(t) = q_{c',c_j;R}(c_j - t(c' - c_j))$.

(III): Bias. Note that the kernel weights $w(x) = K\left(\frac{\theta_{j'}(x) - c_j - t(c' - c_j)}{h}\right)$ is the same for all $x \in Disk(c_j - t(c' - c_j); R; c' - c_j)$. Let $L_{j'} = \{c_j - t(c' - c_j) : t \in R\}$ be the line passing through c_j and c'. Then

$$\begin{aligned}E[\hat{p}Disk_{j',R}(t)] &= E\left[\frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{\theta_{j'}(X_i) - c_j - t(c' - c_j)}{h}\right) 1\{\|X_i - \theta_{j'}(X_i)\| \le R\}\right] \\ &= \frac{1}{h}\int_{x\in\mathcal{X}} K\left(\frac{\theta_{j'}(x) - c_j - t(c' - c_j)}{h}\right) 1(\|x - \theta_{j'}(x)\| \le R)p(x)d_{\mathcal{X}}(dx) = \\ &\quad \frac{1}{h}\int_{z\in L_{j'}} K\left(\frac{z - c_j - t(c' - c_j)}{h}\right)\int_{Disk(z;R;c' - c_j)} p(y)d_{\mathcal{X}-1}(dy)dz \\ &= \frac{1}{h}\int_{L_{j'}} K\left(\frac{z - c_j - t(c' - c_j)}{h}\right) q_{c',c_j;R}(z)dz \\ &= \frac{\|c_j - c'\|}{h}\int_{L_{j'}} K\left(\frac{(s - t)\|c_j - c'\|}{h}\right) q_{c',c_j;R}(c_j - s(c' - c_j))ds\end{aligned}$$

where for the third equality we split the integration with respect to $z \in L_{j'}$ and the integration with respect to $y \in Disk(z; R; c' - c_j)$, and for the last equality we set $z = c_j - s(c' - c_j)$ and utilized the symmetry of the kernel function K.

Then by another change of variable that $u = \frac{(s-t)\|c' - c_j\|}{h}$ and Taylor expansion, we have

$$\begin{aligned} E[\widehat{pDisk}_{j';R}(t)] &= \int K(u) q_{c', c_j; R}(c_j - t(c' - c_j) - hu\frac{c' - c_j}{\|c_j - c'\|}) du \\ &= \int K(u) \left[q_{c', c_j; R}(c_j - t(c' - c_j)) + hu\, g_1 + \frac{1}{2}h^2u^2\, g_2 + O(h^2) \right] du \end{aligned}$$

where

$$g_1 = \frac{(c' - c_j)}{\|c_j - c'\|}^T \nabla q_{c', c_j; R}(c_j - t(c' - c_j))$$

$$g_2 = \frac{(c' - c_j)}{\|c_j - c'\|}^T \nabla\nabla q_{c', c_j; R}(c_j - t(c' - c_j)) \frac{(c' - c_j)}{\|c_j - c'\|}$$

When $R \to 0$, assumption (D2) implies that there is a constant C_{d-1} that

$$2p_{min}C_{d-1}R^{d-1} \le \widehat{pDisk}_{j';R}(t) \le 2p_{max}C_{d-1}R^{d-1} = O(R^{d-1}) \tag{35}$$

where $0 < p_{min} \le \inf_{x \in X} p(x), \sup_{x \in X} p(x) \le p_{max} < 1$. Since the disk density is shrinking at rate $O(R^{d-1})$, one can easily verify that the gradient and Hessian of the disk density function are also at rate $O(R^{d-1})$. Namely,

$$g_1 = O(R^{d-1}); \qquad g_2 = O(R^{d-1}):$$

By assumption (D2) we have g_1 and g_2 to be bounded and therefore Thus,

$$\begin{aligned} E[\widehat{pDisk}_{j';R}(t)] &= q_{c', c_j; R}(c_j - t(c' - c_j)) \int K(u)du + h \int uK(u)du \; g_1 \\ &\quad + \frac{1}{2}h^2 \int u^2K(u)du \; g_2 + O(h^2R^{d-1}) \\ &= q_{c', c_j; R}(c_j - t(c' - c_j)) + O(h^2R^{d-1}) \\ &= pDisk_{j';R}(t) + O(h^2R^{d-1}); \end{aligned}$$

where for the second equality we used, by assumption (K)

$$\int K(u)du = 1; \quad \int uK(u)du = 0; \quad \int u^2K(u)du < 1$$

so we conclude that $|E[\widehat{pDisk}_{j';R}(t)] - pDisk_{j';R}(t)| = O(h^2R^{d-1})$

(II): Stochastic variation.

$$\begin{aligned}
Var(\hat{p}Disk_{j',R}(t)) &= Var\left[\frac{1}{nh}\sum_{i=1}^{n} K^{j'}(X_i)\,\frac{c_j - t(c', c_j)}{h}\, I(j|X_i - \pi_{j'}(X_i)| \le R)\right] \\
&\le \frac{1}{nh^2} E\left[K^2_{j'}(X_i)\left(\frac{c_j - t(c', c_j)}{h}\right)^2 I(j|X_i - \pi_{j'}(X_i)| \le R)\right] \\
&= \frac{1}{nh}\int K^2(u)\, q_{c', c_j;R}(c_j - t(c', c_j)) + hu\, g_1 + O(h^2)\, du \\
&= O\left(\frac{1}{nh}\right)
\end{aligned}$$

by the same analysis procedure as for Face Density and the assumptions (D1), (K1).

Now, by assumption (D2), the face density $q_{c', c_j;R}(c_j - t(c', c_j)) = O(R^{d-1})$, which leads to

$$Var(\hat{p}Disk_{j',R}(t)) = O\left(\frac{R^{d-1}}{nh}\right):$$

Therefore,

$$|\hat{p}Disk_{j',R}(t) - E[\hat{p}Disk_{j',R}(t)]| = O_p\left(\sqrt{\frac{R^{d-1}}{nh}}\right)$$

and

$$|\hat{p}Disk_{j',R}(t) - pDisk_{j',R}(t)| = O(h^2 R^{d-1}) + O_p\left(\sqrt{\frac{R^{d-1}}{nh}}\right): \tag{36}$$

(I): Change in position. Finally, we bound the term

$$(I) = |\hat{p}Disk_{j',R}(\hat{t}) - \hat{p}Disk_{j',R}(t)|:$$

Note that the minimizer $\hat{t}$ satises the gradient condition

$$\hat{p}Disk'_{j',R}(\hat{t}) = 0:$$

By a simple Taylor expansion at t_0, we obtain

$$\begin{aligned}(I) &= (\widehat{Disk}_{j',R}(t) - \widehat{Disk}_{j',R}(t_0)) \\ &= (t - t_0)\underbrace{\widehat{Disk}'_{j',R}(t_0)}_{=0} + \frac{1}{2}(t - t_0)^2 \widehat{Disk}''_{j',R}(t_0) + O(|t - t_0|^3) \\ &= O(|t - t_0|^2).\end{aligned}$$

Thus, we only need to derive the rate of $t - t_0$.

Now by the fact that t solves the population gradient condition $\widehat{Disk}'_{j',R}(t) = 0$, we have

$$\begin{aligned}\widehat{Disk}'_{j',R}(t) - Disk'_{j',R}(t) &= \widehat{Disk}'_{j',R}(t) - \widehat{Disk}'_{j',R}(t_0) \\ &= \widehat{Disk}''_{j',R}(t)(t - t_0) + O(|t - t_0|^2).\end{aligned}$$

Because $\widehat{Disk}''_{j',R}(t) \to^P Disk''_{j',R}(t)$ from the analysis of term (II) and (III), we conclude that

$$t_0 - t = O(\widehat{Disk}'_{j',R}(t) - Disk'_{j',R}(t)) = O(h^2 R^{d-1}) + O_P\left(\sqrt{\frac{R^{d-1}}{nh^3}}\right).$$

Note that the above rate analysis follows from the same analysis as term (II) and (III) except that we are using gradient rather than the density.

As a result, we conclude that

$$(I) = O(|t - t_0|^2) = O(h^4 R^{2d-2}) + O_P\left(\frac{1}{nh^3 R^{d-1}}\right).$$

Combining together, we have

$$\begin{aligned}|\widehat{S}^{T,D}_{j',} - S^{T,D}_{j',}| &= (I) + (II) + (III) \\ &= O(h^4 R^{2d-2}) + O_p\left(\frac{R^{d-1}}{nh^3}\right) + O(h^2 R^{d-1}) + O_p\left(\sqrt{\frac{R^{d-1}}{nh}}\right) \\ &= O(h^2 R^{d-1}) + O_p\left(\sqrt{\frac{R^{d-1}}{nh}}\right) + O_p\left(\frac{R^{d-1}}{nh^3}\right).\end{aligned}$$

Using the fact that $S_{j,D}^{T} \geq 2p_{min}C_{d-1}R^{d-1}$ from equation (35), we conclude that

$$\frac{\hat{S}_{jD}}{S_{j,T}^{D}} - 1 = O(h^2) + O_p\left(\frac{r}{nh^{d-1}}\right) + O_p\left(\frac{1}{nh^3}\right);$$

which completes the proof.

E Choice of Linkage

In this section, we use different simulations to investigate the effect of different linkage criteria under our skeleton clustering framework. We start with the same Yinyang data to illustrate how different linkages cope with well-separated clusters in Appendix E.1. Next, we add noisy observations to the Yinyang data and make the comparison again in Appendix E.2. Moreover, we repeat this comparison using different simulation scenarios when there are overlapping clusters; the comparisons in Appendix E.3, E.4, E.5, and E.6.

Except for the linkage criterion, all other procedures are the same with the following settings: we use k-means clustering with $k = \sqrt{n}$ to nd knots and use the Voronoi density as the density-aided similarity measure. We vary the total number of nal clusters from 1 to 40 and compare the adjusted Rand Index (ARI) to the actual cluster label. The entire procedure is repeated 100 times for the comprehensive comparison of various linkage methods from the `hclust` function in R. The medium performances of the resulting clusterings are summarized in Table 2. For datasets without noisy points, we only present the medium ARI at the true number of clusters, while for data with noisy points we show the best medium ARI across different S and record the corresponding S in the bracket. The best linkages for each data scenario are in bold.

	average	centroid	complete	mcquitty	median	minimax	single	Ward
Yinyang,d=10	1.000	0.119	-0.017	1.000	0.111	0.027	1.000	1.000
Yinyang,d=100	1.000	0.098	-0.008	1.000	0.097	0.055	1.000	1.000
Yinyang,d=500	0.560	0.074	-0.028	0.587	0.054	0.062	1.000	0.526
Yinyang,d=10000	0.533	0.107	-0.029	0.555	0.021	0.106	1.000	0.456
MixMickey,d=10	0.731	-0.005	0.017	0.380	0.007	0.010	-0.004	0.194
MixMickey,d=100	0.740	-0.005	0.005	0.341	0.010	0.043	-0.001	0.129
MixMickey,d=500	0.710	-0.003	0.003	0.356	0.013	-0.003	-0.004	0.180
MixMickey,d=10000	0.692	-0.006	-0.014	0.297	0.011	-0.045	-0.006	0.217
MixStar,d=10	0.763	0.0001	0.00532	0.510	0.001	0.0488	0.0001	0.424
MixStar,d=100	0.763	0.0001	0.007	0.540	0.001	0.0503	0.0001	0.415
MixStar,d=500	0.762	0.0001	0.004	0.537	0.001	0.039	0.0001	0.444
MixStar,d=1000	0.721	0.0001	0.005	0.533	0.001	0.050	0.0001	0.418
NoisyYinyang,d=10	0.875(S=4)	0.182(4)	0.102(35)	0.397(3)	0.180(13)	0.132(28)	0.968(16)	0.535(4)
NoisyYinyang,d=100	0.875(S=3)	0.182(6)	0.103(35)	0.798(2)	0.242(20)	0.135(23)	0.999(14)	0.695(4)
NoisyYinyang,d=500	0.875(S=3)	0.121(10)	0.107(28)	0.783(3)	0.209(20)	0.143(21)	0.999(11)	0.539(4)
NoisyYinyang,d=1000	0.875(S=3)	0.176(7)	0.111(27)	0.875(3)	0.193(28)	0.149(19)	0.998(10)	0.372(5)
NoisyMixMickey,d=10	0.686(S=5)	0.119(34)	0.093(29)	0.413(6)	0.077(39)	0.157(15)	0.501(31)	0.235(5)
NoisyMixMickey,d=100	0.700(S=5)	0.141(37)	0.094(29)	0.358(6)	0.095(39)	0.158(16)	0.506(31)	0.221(6)
NoisyMixMickey,d=500	0.697(S=5)	0.095(37)	0.091(30)	0.359(7)	0.098(39)	0.155(17)	0.502(31)	0.232(6)
NoisyMixMickey,d=1000	0.692(S=5)	0.122(36)	0.091(29)	0.386(6)	0.104(39)	0.153(17)	0.497(31)	0.241(5)
NoisyMixStar,d=10	0.783(S=10)	0.109(40)	0.221(30)	0.613(11)	0.140(40)	0.330(17)	0.623(31)	0.476(4)
NoisyMixStar,d=100	0.779(S=9)	0.129(40)	0.220(28)	0.627(10)	0.171(40)	0.334(18)	0.667(30)	0.487(4)
NoisyMixStar,d=500	0.788(S=8)	0.115(40)	0.220(29)	0.604(9)	0.158(40)	0.328(16)	0.651(30)	0.498(4)
NoisyMixStar,d=1000	0.791(S=9)	0.113(40)	0.219(29)	0.599(9)	0.150(40)	0.333(15)	0.621(30)	0.476(4)

Table 2: Comparison of the linkage methods across dierent simulated datasets. All reported values are mediums of 100 random simulations. For datasets without noisy points, the performance at the true number of clusters is reported (S = 5 for Yinyang, S = 3 for Mix Mickey and Mix Star). For datasets with noisy points, we report the best performance across dierent numbers of clusters and include the number of clusters at which the max is achieved in the bracket.

From Table 2, either average linkage or single linkage achieve the best and most reliable performance. Thus, we recommend using one of them as the linkage criterion. We include a more detailed analysis of each dataset in the following subsections and we plot the 5th percentile, medium, and 95th percentile of the adjusted Rand index for single linkage, average linkage, and complete linkage. Plots comparing all the linkages on the dierent datasets are deferred to Appendix E.7.

E.1 Yinyang Data

We begin by comparing the dierent linkage methods on the Yinyang datasets with dierent numbers of noisy dimensions (same data as in Section 5.1.1). The results are shown

in Figure 14. For each dimension (d = 10; 100; 500; 1000), the medium adjusted Rand index of the 100 runs is plotted with the solid line, and the 5 percentile to 95 percentile range is depicted with a lighter color band. The true number of clusters S = 5 is shown as the red dotted vertical line.

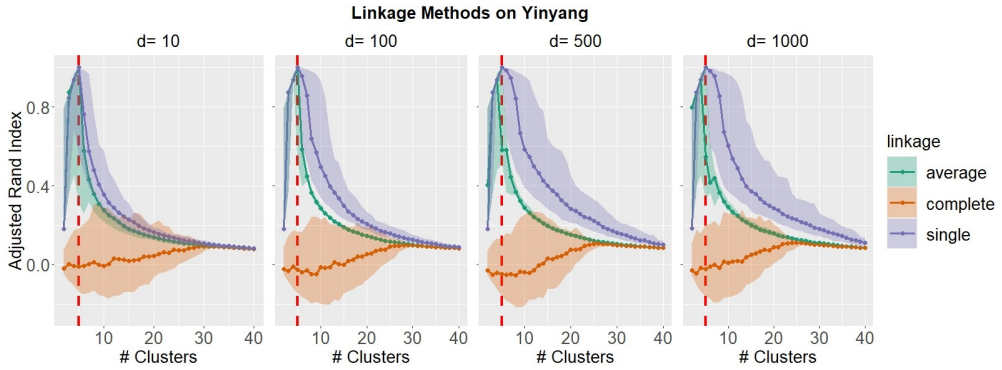


Figure 14: Clustering results with dierent linkage methods across dierent numbers of nal clusters on Yinyang data. The line is for medium and the band is from 5th percentile to 95th percentile. The vertical red dashed line indicates the true number of 5 clusters.

We observe that single linkage and average linkage have similar performance for lower dimensions d = 10 and d = 100, with medium performance achieving nearly perfect clustering at the true number of clusters. However, the clustering results returned by single linkage are more stable, having a narrower band while the band of average linkage is much wider. For cases with higher dimensions d = 500; 1000, we observe single linkage still stably achieves nearly perfect clustering at k = 5, which corroborates our results in Section 5.1.1, but average linkage fails to get such good clustering performance when dimensions get higher. Therefore, single linkage has superior performance on the Yinyang data, arguably because the true manifold of the data has well-separated clusters that single linkage is suitable for separation.

E.2 Noisy Yinyang Data

To create additional noise, we added 640 (20% of the number of signals) noisy points to the Yinyang dataset, sampled uniformly from [3; 3] [3; 3] in the rst two dimensions, with random Gaussian variables in the other dimensions the same way we generated Yinyang data. The adjusted Rand indexes are calculated only for the true signal data points and the results are shown in Figure 15.

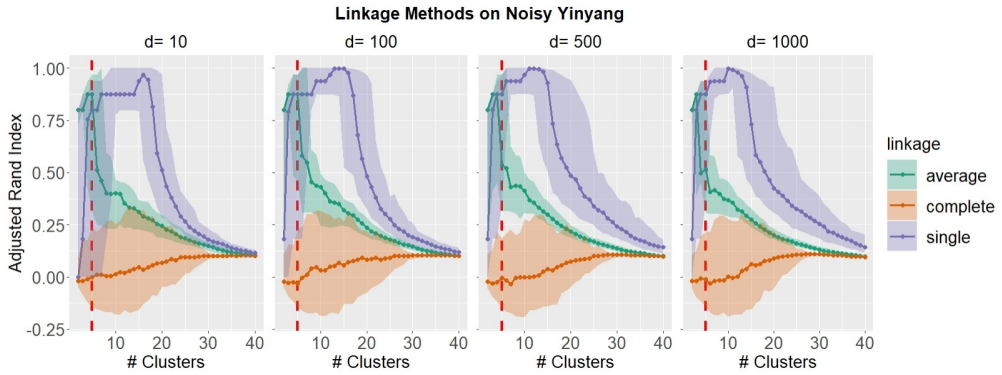


Figure 15: Clustering results with dierent linkage methods across dierent numbers of nal clusters on Yinyang data with noisy points. The vertical red dashed line indicates the true number of 5 clusters.

Average linkage can achieve slightly better performance than single linkage around the true number of clusters S = 5 for lower dimensions (d = 10; 100), but fails to achieve satisfactory clustering performance when dimensionality gets higher (d = 500; 1000). The performance of single linkage improves with S being slightly larger than the actual number 5 and can yield nearly perfect clusters with S being around 15 to 20. A further investigation reveals that large S will group noisy points into separate clusters and hence improves the clustering performance; see Figure 16. This suggests that our framework may be used for anomaly detection.

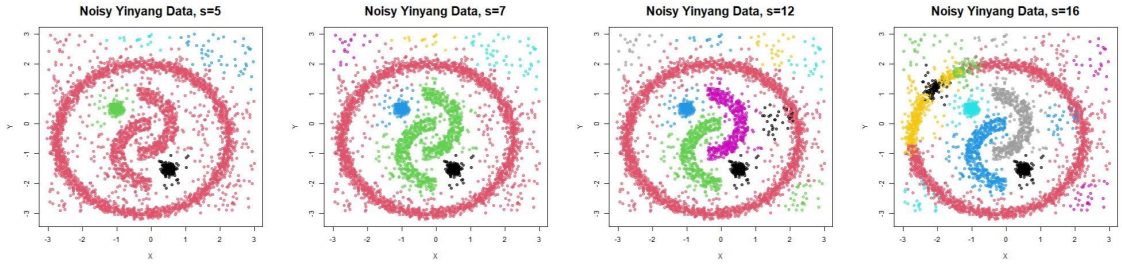


Figure 16: The clustering results with single linkage in skeleton clustering with dierent numbers of nal clusters S for Noisy Yinyang data, d = 1000.

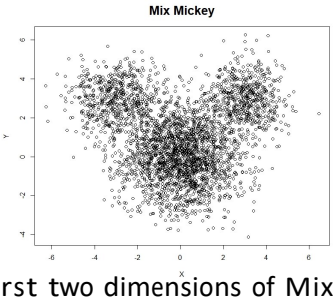


Figure 17: First two dimensions of Mix Mickey data.

E.3 Mix Mickey Data

The well-separated structures in the Yingyang data may provide advantages to the single linkage. To investigate the eect of linkage criteria on the overlapping clusters, we consider a three-Gaussian mixture model in 2D case that we call the Mix Mickey data. The large cluster is centered at $(0; 0)$ with the covariance matrix being a diagonal matrix of 2 and has 2000 points. The two smaller clusters are centered at $(3; 3)$ and $(\ 3; 3)$ respectively, and both have a covariance matrix being a diagonal matrix of 1, and each has 600 points. Random Gaussian variables are added to make the data $d = 10; 100; 500; 1000$ dimensions via the same way we generate the Yinyang data. Figure 17 presents a scatter plot of the rst two dimensions; the three clusters have a substantial amount of overlap so it is dicult for clustering methods to separate them into three distinct clusters. The results under the same linkages analysis pipeline are shown in Figure 18.

Remark 7. GMM can be favored in this data example but is unstable and cannot work with too many noisy dimensions. We present some comparisons including GMM in Appendix F.8.

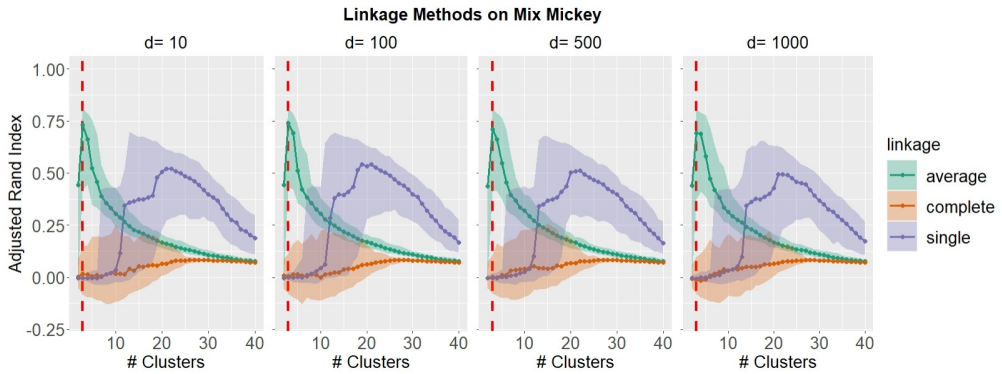


Figure 18: Clustering results with dierent linkage methods across dierent numbers of nal clusters on Mix Mickey data. The vertical red dashed line indicates the true number of 3 clusters.

We observe that average linkage gives good performance at S = 3 (the true number of clusters) and single linkage fails to give a satisfying performance under this scenario, giving non-informative clusters at low S (only extracting small clusters) and too fragmented clusters at high S. The average linkage is a criterion that tends to create spherical clusters with similar sizes and hence is better suited for this simulated data. Thus, our experiment shows that, for data containing overlapping clusters with roughly spherical shapes, the average linkage criterion in the knots segmentation step is preferred.

E.4 Noisy Mix Mickey Data

In this section, we experiment with a scenario with both overlapping clusters and noisy observations. We added 640 (20% of the number of signals) noisy points to the Mix Mickey dataset, sampled uniformly from [6; 6] [5; 6] in the rst two dimensions, with random Gaussian noises in the other dimensions the same way as in Mix Mickey data. The adjusted Rand indices are measured only on the true signal data points with the results shown in

Figure 19.

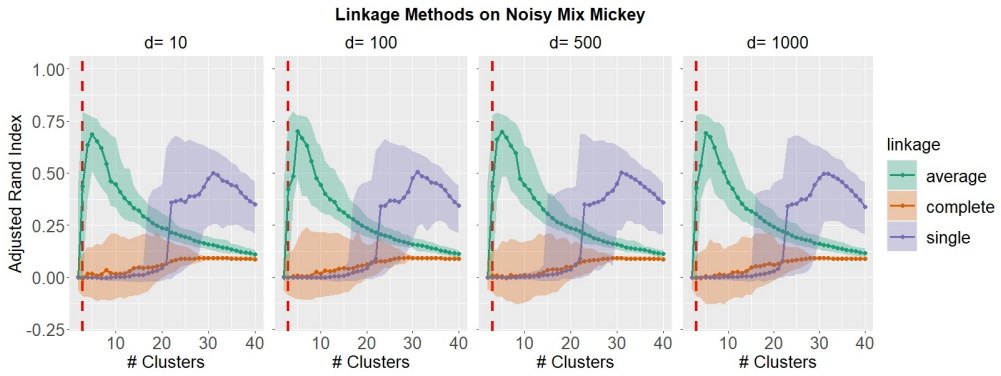


Figure 19: Clustering results with dierent linkage methods across dierent numbers of nal clusters on Mix Mickey data with Noise. The vertical red dashed line indicates the true number of 3 clusters.

Average linkage still gives good performance and is superior to the single linkage, which fails to give reasonable clustering performance under a decent number of clusters. Notably, average linkage achieves the best performance with the S being slightly higher than 3 due to the introduction of noisy data points.

E.5 Mix Star Data

We present here the Mix Star dataset, another 3-GMM data but with a more elongated shape as illustrated in Figure 20. The three clusters are all generated as 2D Gaussian with 5 and 0:3 on the diagonal of the covariance matrix with respective centers at $(4; 0)$, $(\ 4; 0)$, and $(0;\ \ 4)$, and then are rotated to get a star-like shape. Each cluster has 1000 sample points, and random Gaussian variables with standard deviation 0:1 are added to make the data $d = \ 10; 100; 500; 1000$ dimensions. There is still a decent overlap among clusters, but each cluster is more distinct compared to Mix Mickey. We apply the same analysis pipeline as the Yinyang and Mix Mickey data and compare dierent linkage criteria. Figure 21 displays the median clustering performance. Again, we see that average linkage has the best

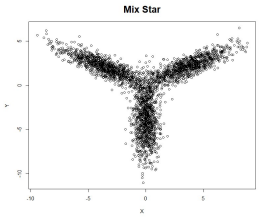


Figure 20: First two dimensions of the Mix Star data.

performance.

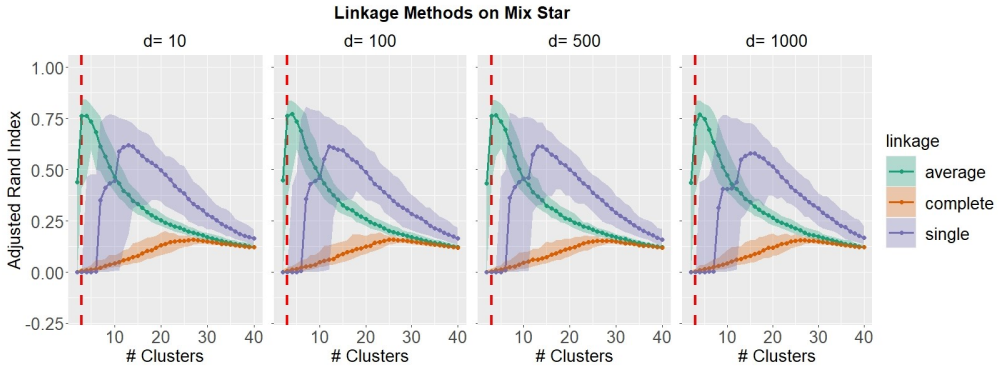


Figure 21: Clustering results with dierent linkage methods across dierent numbers of nal clusters on Mix Star data. The vertical red dashed line indicates the true number of 3 clusters.

E.6 Noisy Mix Star

To investigate the eect of added noises, we make the data similar to the Noisy Mix Mickey by adding 600 (20% of the number of signals) noisy points to the Mix Star dataset, sampled uniformly from [10; 10] [10; 5] in the rst two dimensions, with random Gaussian noises in the other dimensions generated the same way. The results of linkage comparison results are shown in Figure 22. Average linkage still gives the best clustering results in this scenario.

In summary, as illustrated by all the simulations in this section, our skeleton clustering framework is able to handle noisy data points by tuning the number of nal clusters and can cope with overlapping clusters by choosing an appropriate linkage criterion for skeleton

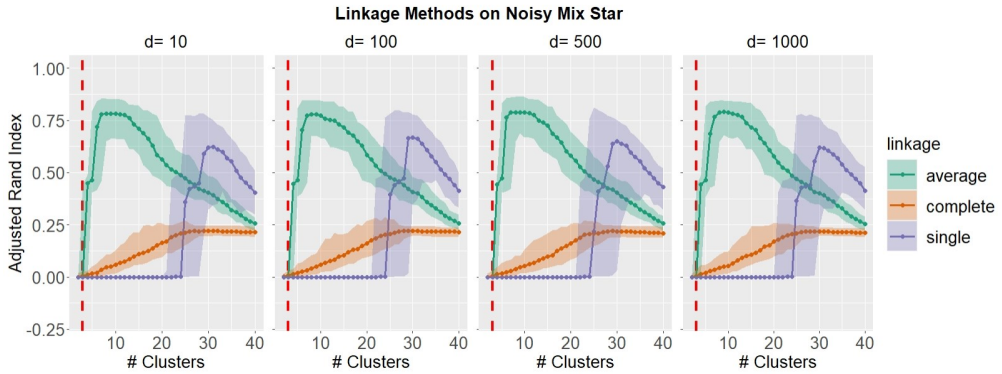


Figure 22: Clustering results with dierent linkage methods across dierent numbers of nal clusters on Mix Star data with Noise.

segmentation. Broadly speaking, the appropriate choice of linkage method depends on the intrinsic geometric structure of the data and may require subject matter knowledge or exploratory analysis. Specically, if the intrinsic clusters are well-separated, single linkage is preferred as it gives clear cuts for disjoint components. But if the clusters are believed to have some degree of overlapping with each cluster approximately spherically shaped, average linkage criterion can lead to better performance.

E.7 All Linkage Comparisons

Figures 23 and 24 display the median clustering performances of all linkage methods under dierent numbers of clusters using Yinyang and noisy Yinyang data. We see that average linkage and single linkage dominate all other methods, while single linkage is superior in those two cases.

Figures 25 and 26 present the median clustering performance under dierent numbers of clusters for the Mix Mickey and noisy Mix Mickey data (same setup in Section E). Similar to the case of Yinyang data, we observe that average linkage and single linkage dominate all other methods, while average linkage is superior among the two.

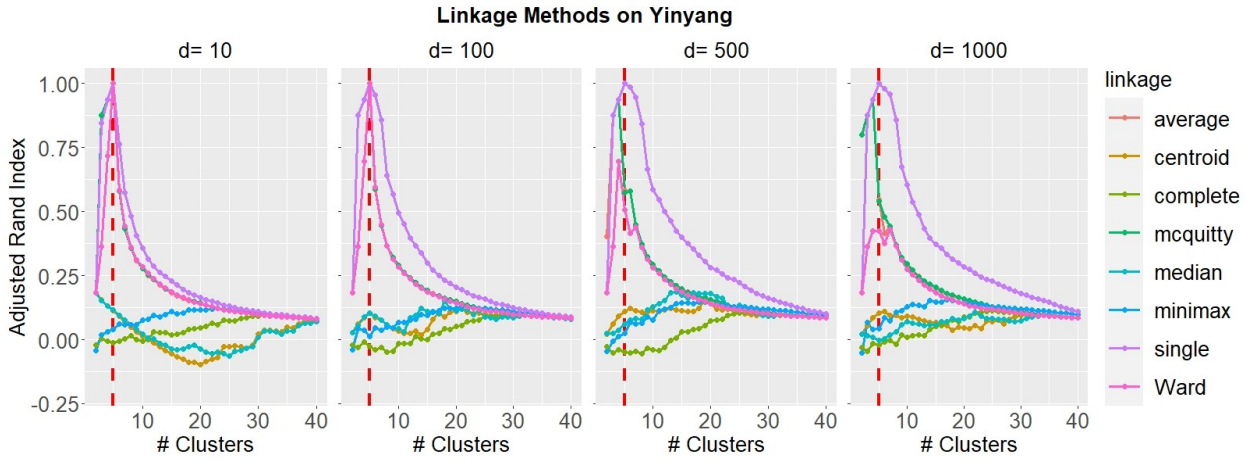


Figure 23: Clustering results with dierent linkage methods across dierent numbers of nal clusters on Yinyang Data.

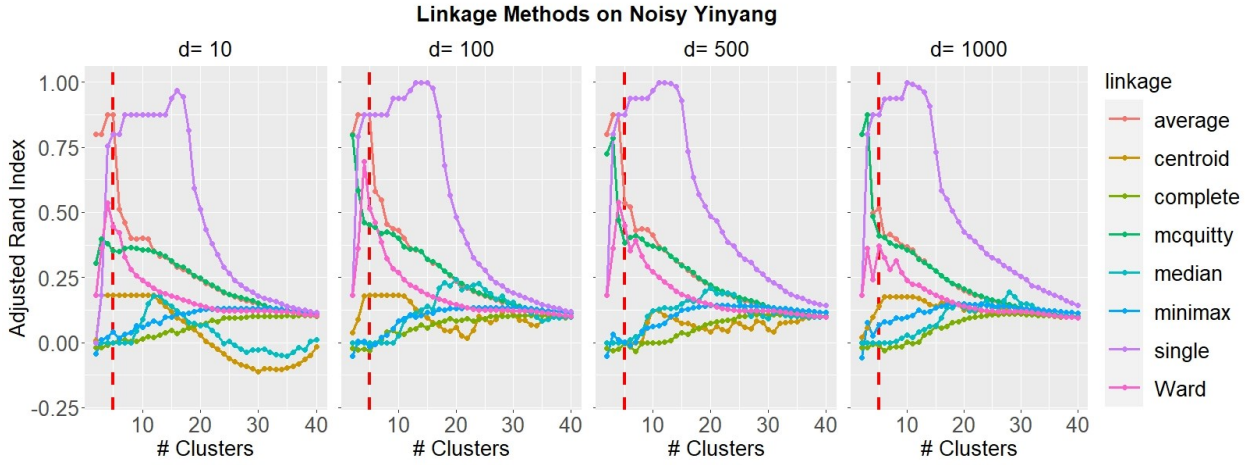


Figure 24: Clustering results with dierent linkage methods across dierent numbers of nal clusters on Noisy Yinyang Data.

To further investigate what the clusters will be like in high dimensions, we present 2D scatterplot of clustering results under S = 3 (nal number of clusters is 3) of the rst two coordinates in Figure 27. We use the data with d = 1000 and color the clusters using red, green, and blue. Clearly, average linkage successfully recovers the actual clusters while other methods fail to recover. Note that single linkage does not perform well because clusters are overlapping with each other.

Figures 28 and 29 present the median clustering performance under dierent numbers

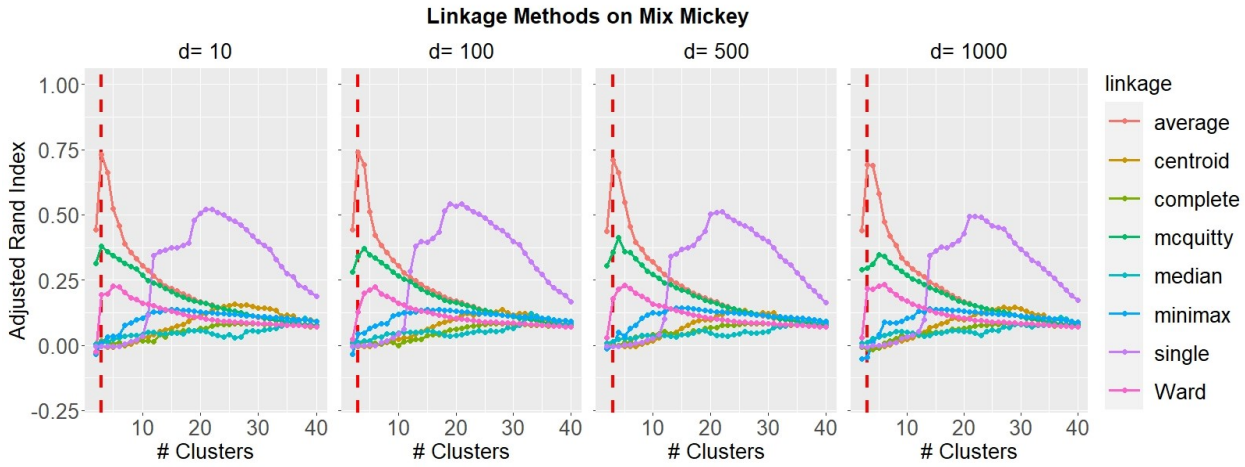


Figure 25: Clustering results with dierent linkage methods across dierent numbers of nal clusters on Mix Mickey data.

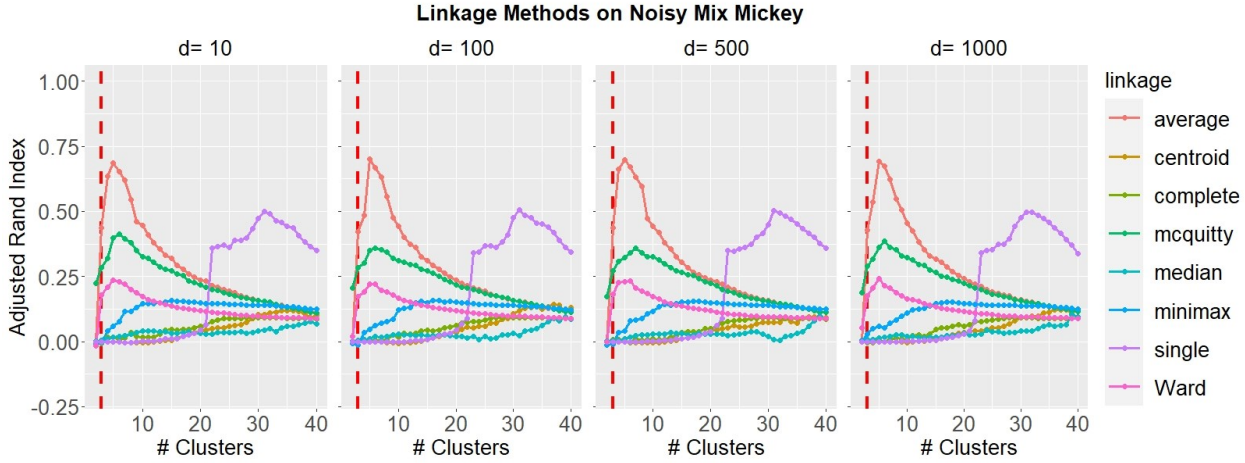


Figure 26: Clustering results with dierent linkage methods across dierent numbers of nal clusters on Mix Mickey data with Noise.

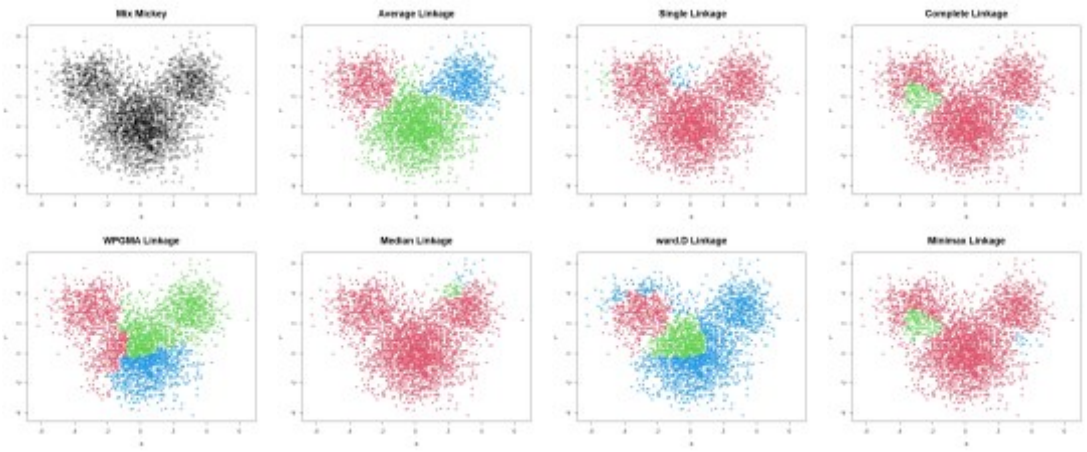


Figure 27: Comparing linkage criteria in segmentation on the Mix Mickey data, d = 1000.

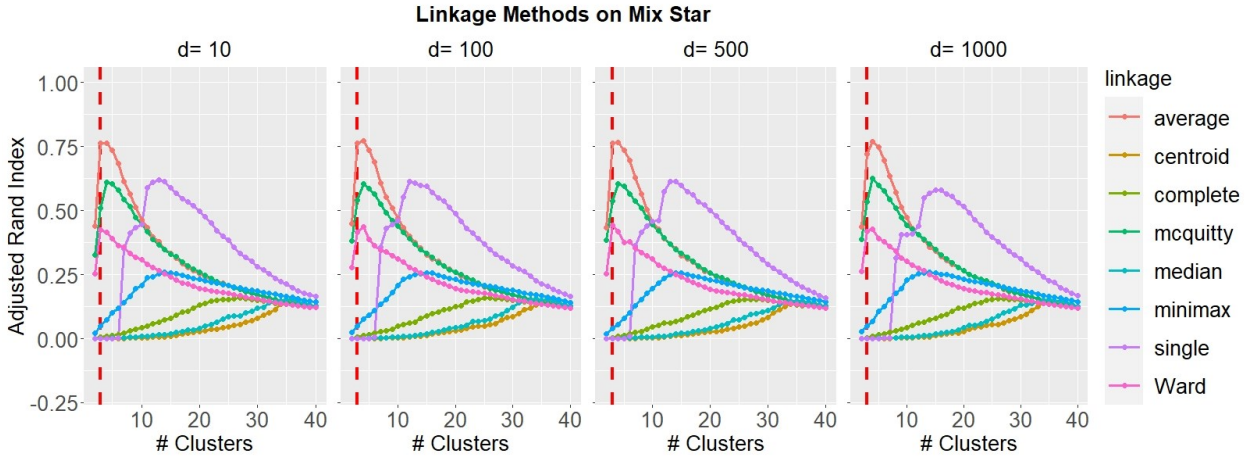


Figure 28: Clustering results with dierent linkage methods across dierent numbers of nal clusters on Mix Star data.

of clusters for the Mix Star and noisy Mix Star data. We observe that average linkage and single linkage dominate all other methods.

F Additional Data Analysis

F.1 Performance with Dierent Number of Knots

We analyze how the number of knots would aect the performance of the skeleton clustering. We empirically test the eect of the number of knots, k, on the nal clustering

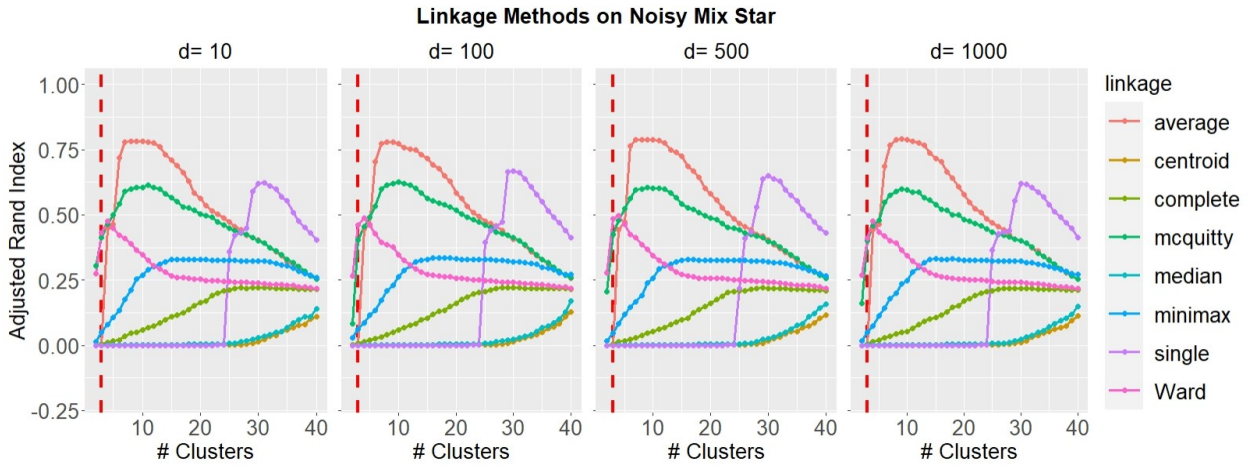


Figure 29: Clustering results with dierent linkage methods across dierent numbers of nal clusters on Mix Star data with Noise.

performance on Yinyang data with dimensions 10; 100; 500 and 1000. For each dimension, we simulated the Yinyang data 100 times, and for each simulated data we carried out the default skeleton clustering procedure with single linkage and dierent k (other steps the same as in Section 5.1.1). Figure 30 displays the median adjusted Rand index given by each method across dierent k, where the reference rule with k = 57 is marked by the vertical dash line. We see that as long as k is suciently large, skeleton clustering works well.

F.2 Self-Organizing Map

The Self-Organizing Map (SOM) is another popular prototype clustering method and can be used as an alternative to k-means clustering in nding knots. Thus, here we conduct a simple experiment to examine the performance of using SOM to nd knots. We examine the performance using Yingyang data with d = 10 to d = 1000. The identical procedure as in Section 5.1.1 is applied except that the knots are now detected by the SOM rather than overtting k-means. The total number of grid points in the SOM is the total number of knots we obtain and, to be comparable to k-means with k = $\sqrt{n}$ knots, we used $dn^{1=4}e$ breaks

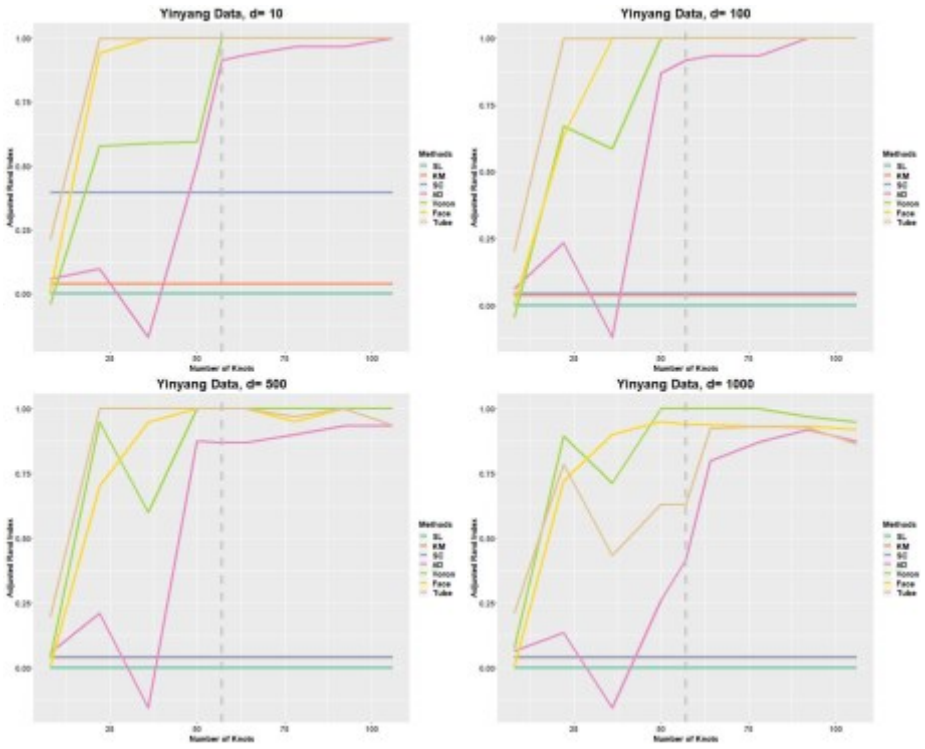


Figure 30: Adjusted Rand indexes of dierent clustering methods against dierent numbers of knots on 100 simulated Yinyang data.

for each dimension of the SOM grid, giving a total of $dn^{1=4}e^2$ initial grid points. However, the SOM may return knots with tiny sample sizes, on which the density-aided similarity measures cannot be calculated. Therefore, we remove knots with less than 3 data points and use the remaining ones for skeleton construction.

Figure 31 summarizes the result. The top left panel shows the knots from the SOM (after post-processing), which are located around the main data structures and are representative of the original data as well. The dendrogram shows the cluster structure of the SOM knots using Voronoi density on one 100-dimensional Yinyang data. In the bottom row, we display the adjusted Rand indices from the clustering methods. Compared to the results of Figure 6, the adjusted Rand indices given by the skeleton clustering with SOM knots are similarly good

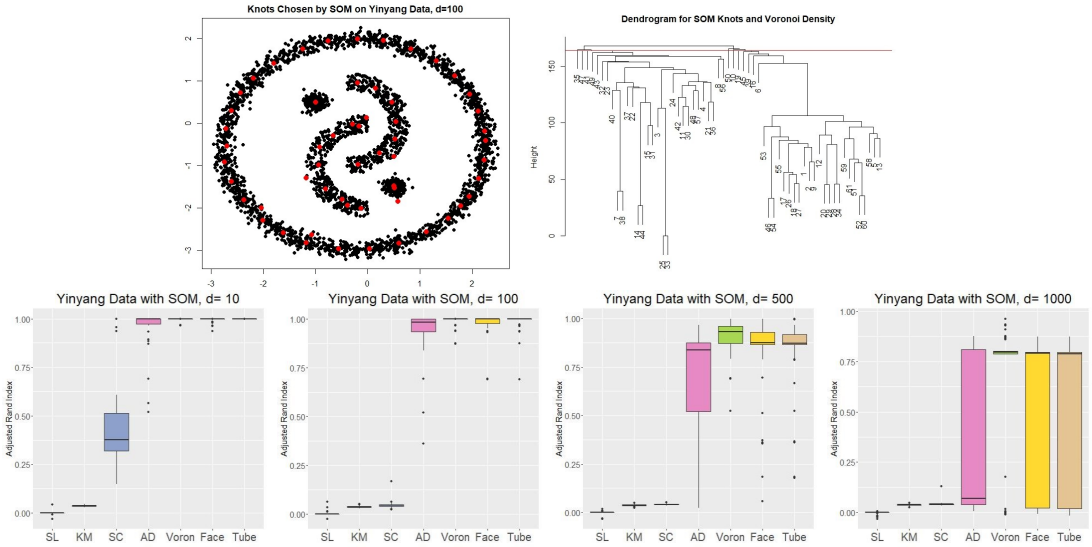


Figure 31: Adjusted Rand indexes using SOM for knots selection on Yinyang data.

when the dimension is not so high (d = 10 and 100). But when the data dimension becomes high (d = 500; 1000), knots constructed by SOM lead to worse clustering results. Therefore, overtting k-means is favored in this work. Another limitation of SOM is that we need to perform some post-processing to remove tiny knots; in the case of k-means, we do not need such a procedure.

F.3 Bandwidth Selection Yinyang Data

The estimations of the FD and the TD involve the use of the projected kernel density estimation, for which the type of kernel and the bandwidth need to be specied. Similar to the usual KDE, the kernel function does not aect the nal performance much, so by default we use the Gaussian kernel in all of our empirical studies. It is worth noting that using the uniform kernel can save some computation since it has compact support, but empirically we nd using the Gaussian kernel leads to better nal clustering results. In what follows, we focus on the bandwidth selection.

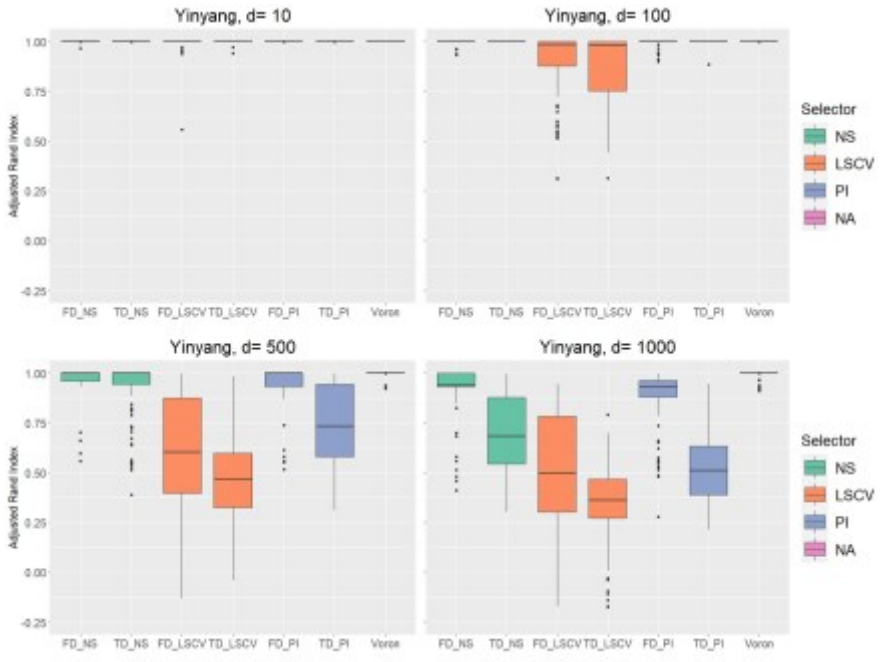


Figure 32: Performance of skeleton clustering on Yinyang data d = 10; 100; 500; 1000 with Face and Tube density by dierent bandwidth selectors. Voronoi density result is included for comparison.

It is known that the bandwidth is a pivotal parameter that can signicantly aect the estimation result of a kernel density estimator. In Figure 32, we conduct a simulation using the Yinyang data with dierent dimensions of noisy Gaussian variables (see Section 5.1.1 for more details) and compare the performance of three common bandwidth selectors: the normal scale bandwidth (NS) (Chacn et al., 2011), the least-squared cross-validation (LSCV) (Bowman, 1984; Rudemo, 1982), and the plug-in approach (PI) (Wand and Jones, 1994). Each edge is allowed to have its own bandwidth. Voronoi density performance results are also included for comparison. We found that the NS performs reliably well while the others may have unstable performance. A similar comparison of the bandwidth selectors on another dataset is presented in Appendix F.4 and the NS also performs relatively better than the other bandwidth selectors.. As a result, we recommend using the NS as the default bandwidth selector. Additionally, since the density estimations are all 1-dimensional, in practice

it is possible to examine the estimated density to assess the degree of oversmoothing or undersmoothing and manually adjust the bandwidth.

In addition to dierent bandwidth selectors, we also study how the bandwidth should depend on the sample size for clustering purpose. In 1-dimensional data, the normal scale bandwidth agrees with Silverman's rule of thumb (Silverman, 1986) giving the bandwidth as $h = \frac{4}{3}^{1=5} b n_{loc}^{1=5}$, where b is the standard deviation of the sample used in the edge weight calculation, and n_{loc} the number of sample points used. Empirically we tested the clustering performance with FD and TD calculated under bandwidth with rates on n_{loc} from 1=3 to 1=10 (see Appendix F.5). We found that the clustering performance with FD and TD generally stays stable with varying bandwidth rates, although a larger bandwidth (slower rate than $O(n_{loc}^{=5})$) may give better clustering results with TD when the dimension of the data is high.

F.4 Bandwidth Selection with Mix Mickey

We present additional results comparing dierent bandwidth selectors on the Mix Mickey dataset generated the same way as in Section E.3. We use average linkage for all the included skeleton clustering approaches. The results are presented in Figure 33. The selectors have similar performances on this Mix Mickey dataset, but NS again seems to perform better with larger dimensions, which corroborates our default choice of using NS for bandwidth.

F.5 Performance under Dierent Bandwidth Rate

In this section we present empirical results on how changing the bandwidth rate aects the performance of clustering. We consider the Yinyang data in Section 5.1.1 with d =

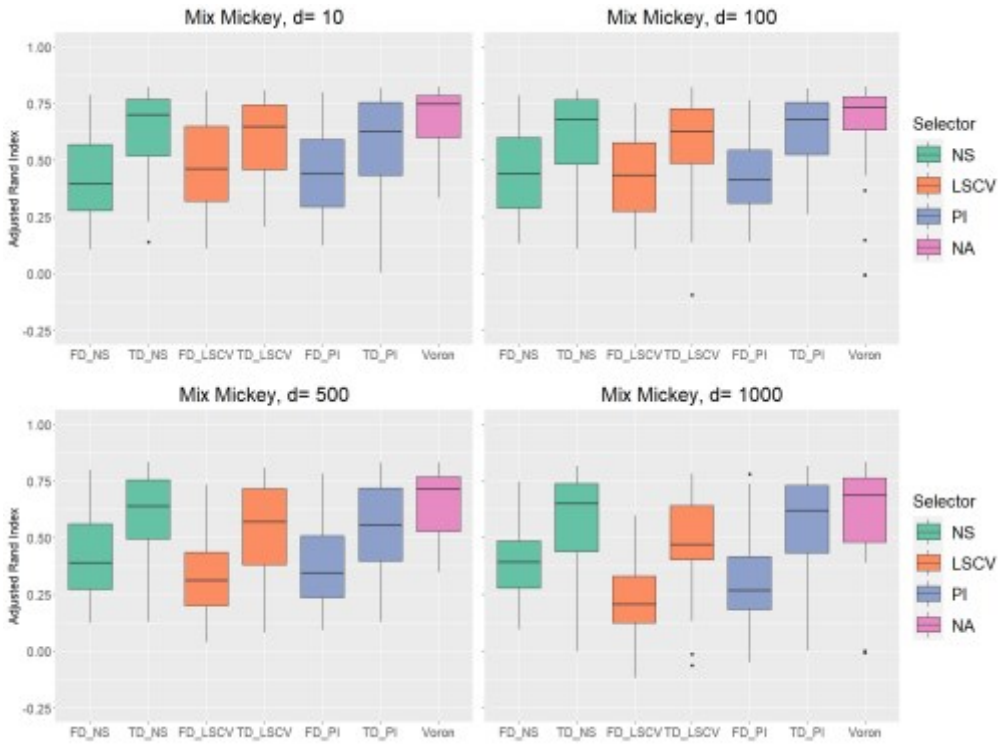


Figure 33: Performance of skeleton clustering on Mix Mickey data d = 10; 100; 500; 1000 with Face and Tube density by dierent bandwidth selectors. Voronoi density result is included for comparison.

10; 100; 500; 1000. We compare the Face and Tube density where the bandwidth is selected by Silverman's rule of thumb with dierent rates, ranging from n_{loc}^{-3} to n_{loc}^{-10}. Note that the original Silverman's rule of thumb will be at rate n_{loc}^{-5}. We repeat the experiment 100 times and record the adjusted Rand index in Figure 34.

When the dimension is low (top panels), all bandwidth within this range works well. When the dimension is large (bottom panels), a slower rate (larger bandwidth) seems to be showing better performance for the TD. Interestingly, the face density yields a robust result across dierent rates of bandwidth. Note that for the TD, the theory (Theorem 4) suggests the choice at rate $h \asymp n_{loc}^{-1}$ is optimal for estimation in large d, the same rate may not lead to the optimal clustering performance. Figure 34 bottom-right panel suggests that the

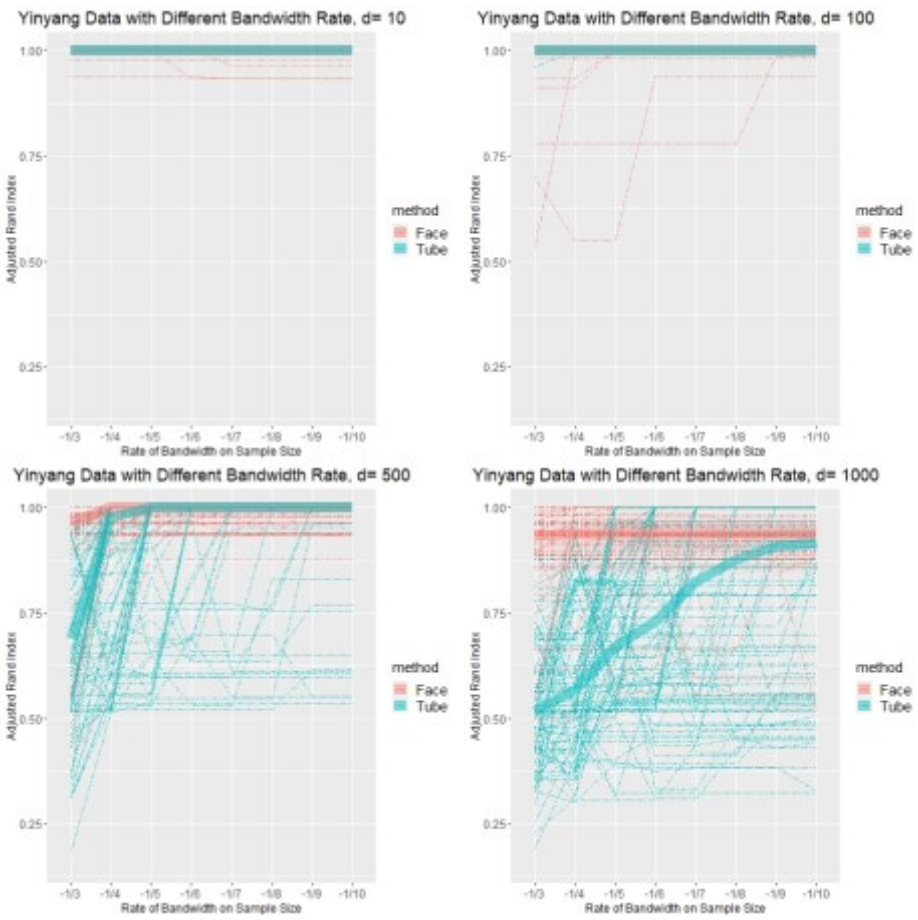


Figure 34: Adjusted Rand indexes of skeleton clustering with Face and Tube density under dierent bandwidth rates on 100 simulated Yinyang datasets. The thick lines indicate the median adjusted Rand index of a given method.

choice h n $^{1=10}_{\text{loc}}$ may have a better clustering performance in this case.

F.6 Adaptive Radius for Tube Density

We compare the clustering performance of Tube density when using xed radius and that when using adaptive radius as described in Section 3.3. The data is the same Yinyang data in Section 5.1.1 and the results are presented in Figure 35. The two approaches (adaptive and xed radius) have a similar performance.

For comprehensiveness, we also compare the clustering performance of Tube density

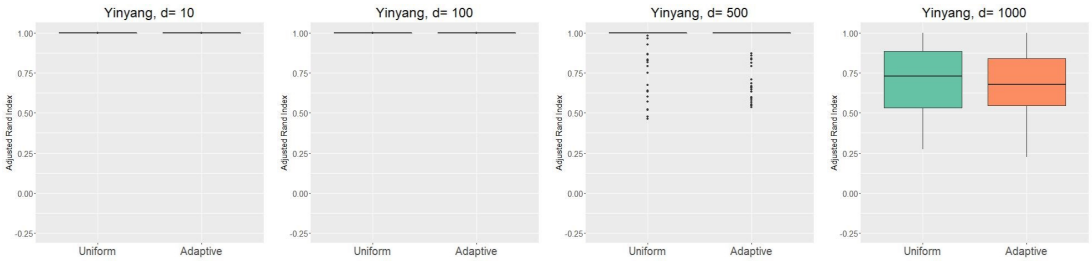


Figure 35: Comparison of radius choices on Yinyang data with dimensions 10, 100, 500, 1000.

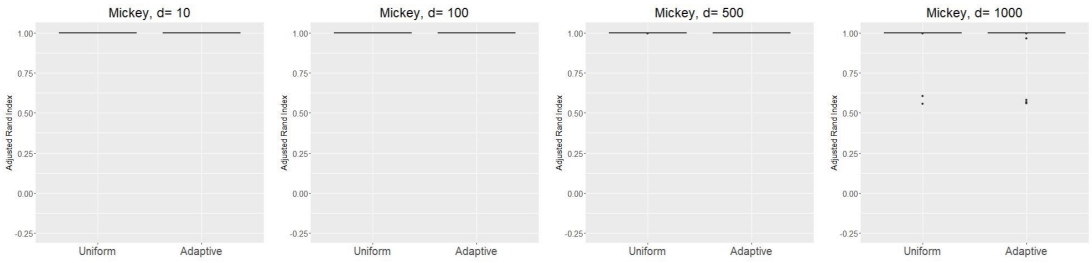


Figure 36: Comparison of radius choices on Mickey data with dimensions 10, 100, 500, 1000.

when using xed radius and that when using adaptive radius on the Mickey data same as in Section 5.1.2, and the results are presented in Figure 36. The two approaches also have a similar performance. In Figure 37, we plot the distribution of the adaptive radius on one Yinyang data and one Mickey data. We note that there are some variations across dierent Voronoi cells, but the variation is not large, and this can be a consequence of constructing the knots using the k-Means method so that the within-cluster variations are minimized.

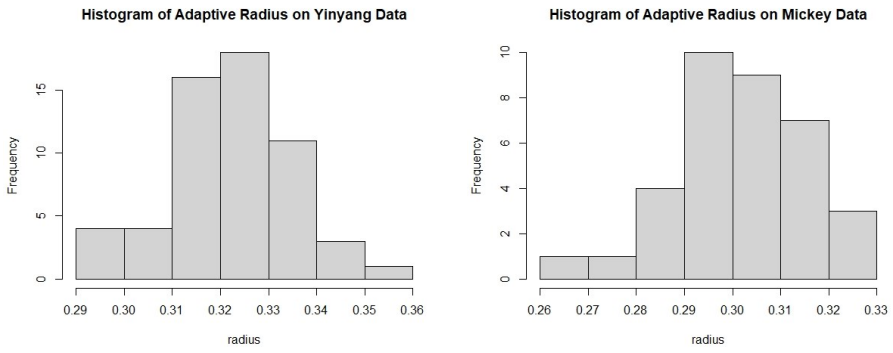


Figure 37: Dispersion of radius on one Yinyang data and one Mickey data.

F.7 Higher Standard Deviations for Noisy Dimensions

We investigate how changing the noise level of the added noisy dimensions of our simulation examples changes the clustering performance. Here we simulate Yinyang data with dierent standard deviations of the added dimensions. We apply the same analysis procedure as in Section 5.1.1 is applied. The adjusted Rand indexes of the clustering methods on 100 simulated datasets with under setting are presented in Figure 38.

We observe that increasing the standard deviation of the noisy dimensions (noise level) has a stronger impact than adding more noisy variables. For example, increasing = 0:1 ! 0:2 scales the standard deviation by a factor of 2 (scales the variance 4 times), but the clustering performance with = 0:2; d = 100 is worse than that with = 0:1; d = 500. However, we still observe that the skeleton clustering with Voronoi density similarity measure can give good clustering performance even under the setting with = 0:4 and d = 100. It can be observed that the VD performance improves under = 0:1 (rst row) v.s. = 0:2 (second row). While this dierence is not large, it is unexpected as in other cases the performance is not better compared to smaller noises. We think that this might be due to Monte Carlo errors.

F.8 Mix Mickey with GMM

We compare the performance of Gaussian Mixture Models (GMMs) to our methods using the Mix Mickey data same as in Section E.3. Unfortunately, the GMM method from clusterR package in R cannot work with dimension 500 and 1000 case because of too many noisy dimensions, so we only compare the case of dimension 10 and 100. For the skeleton

clustering, we use average linkage for the segmentation step the same as in Section E.3. Because this data is generated from 3-GMM and we t the GMM with 3 components, the GMM naturally has the best performance. However, our proposed approaches achieve performance comparable to that given by the GMM and are capable of handling high dimensional data (d = 500; 1000).

F.9 Comparison with Combining Mixture Components for Clustering

In this section, we present the empirical comparison with the combining mixture components approach (mComb) from Baudry et al. (2010) with the implementation from the mclust package in R [1]. The mComb algorithm builds upon the Gaussian nite mixture models tted via the EM algorithm and proceeds with a hierarchy of combined clusterings following an entropy-based criterion.

To make a comparison between this model-based merging clusters approach with our proposed skeleton framework, we apply the mComb algorithm to the Yinyang data (Section 5.1.1) and the Mickey data (Section 5.1.2). However, with a large dimension of noisy variables, the initial stage of Gaussian mixture modeling is likely to identify the data points as one large component without distinguishing the dierent components, and the overall algorithm is unstable. Therefore, we only present the empirical results for merging Gaussian mixtures on datasets with dimension d = 10.

We carry out the combining mixture components approach in two ways. In one way, we provide the true number of nal clusters (5 for Yinyang data and 3 for Mickey data) to the algorithm (mCombOracle). In another way, we use the optimal number of clusters by

[1] https://mclust-org.github.io/mclust/

combining mixture components based on the entropy method as proposed in Baudry et al. (2010) (mCombOptim). We repeat the experiments for 100 random instances for the two dierent simulation datasets and the results are presented in Figure 40, where are included the result given by the spectral clustering (SC) and the skeleton clustering with Voronoi density (Voron) for comparison.

On the Yinyang data, the optimal number of clusters chosen by the mComb algorithm varies from 2 to 8, while only 12% of the runs have the optimal number of clusters to be the true number of clusters 5. The performance given by the combining mixture components approach as assessed by the adjusted Rand Index is not satisfactory compared to the classical clustering methods and the skeleton clustering approaches. Merging Gaussian components is not sucient to handle the complex structures of the Yinyang data.

For the Mickey data, 94% of the mCombOptim runs identify 2 to be the optimal number of clusters, while only 6% of the times the optimal number of clusters is determined to be 3, the true number of clusters. Therefore, combining mixture components with an auto-chosen number of clusters does not give good performance. However, the combining mixture components approach with the number of components given does lead to perfect clustering results on the Mickey data where the clusters are spherically shaped.

F.10 Graphical Representation of GvHD Data Clusters

We visualize the skeleton structure of the clusters identied on the GvHD dataset in Section 6. These graph representations are generated by the igraph package in R. Cluster 6 only has 1 knot with 17 corresponding data points and is hence omitted in Figure 41. We observe that most clusters display a hammer-like structure, which is non-spherical and not

favorable for some classical clustering methods. Only Cluster 3 has a spherical shape in this data.

G Additional Simulated Data Examples

G.1 Manifold Mixture Data

In the Yinyang data and the Mix Mickey data experiments, the underlying components are all two-dimensional structures. Here we consider the data composed of structures of dierent intrinsic dimensions called the manifold mixture data. The simulated manifold mixture data, as illustrated in the left panel of Figure 42, consists of a 2-dimensional plane with 2000 data points, a 3-dimensional Gaussian cluster with 400 data points, and an essentially 1-dimensional ring shape with 800 data points. There are a total of 3200 observations and we choose $k = [\sqrt{3200}] = 57$ knots. Similar to the other two simulations, we include Gaussian noise variables to make the data high-dimensional ($d = 10; 100; 500; 1000$) and make comparisons between the same set of clustering methods. The true number of components $S = 3$ is provided to all the clustering algorithms.

Figure 43 summarizes the performance of each method. Traditional methods (SL, KM, and SC) do not perform well when $d > 10$ while all methods of skeleton clustering perform very well when $d \ 500$. Notably, the skeleton clustering with VD still has a perfect performance even when $d = 1000$, whereas skeleton clustering based on other similarity measures gives satisfying results.

G.2 Ring Data

The ring data is constructed by a mixture distribution such that with a probability of $\frac{1}{6}$ we sample from the ring structure and with a probability of $\frac{5}{6}$ we sample from the central part. The ring structure is generated by a uniform distribution over the ring $f(x_1; x_2) : x_1^2 + x_2^2 = 1g$ and is corrupted with an additive Gaussian noise $N(0; 0{:}2^2 I_2)$. The central part is simply a Gaussian $N(0; 0{:}2^2 I_2)$. We generate a total of $n = 1200$ points from the above mixture and add the high dimensional noise with the same procedure as in Section 5.1. The same skeleton clustering approaches are applied as well as the classical approaches, with the nal number of clusters chosen to be 2. The result is displayed in Figure 45. Again, the density-based skeleton clustering methods work well even when the dimension is large.

H Additional Real Data Examples

H.1 Zipcode Data

This dataset consists of $n = 2000$ 16 16 images of handwritten Hindu-Arabic numerals from (Stuetzle and Nugent, 2010). We use the overtting k-means to nd $k = 45$ knots. Similar to the procedure in Section 5.1, we consider four similarity measures to obtain the edge weight: VD, FD, TD, and AD. We use single linkage for the four skeleton clustering approaches and compare them to three traditional methods: direct single linkage hierarchical clustering (SL), direct k-means clustering (KM), and spectral clustering (SC).

The result is shown in the left panel of Figure 46 with the adjusted Rand index plotted against dierent numbers of total clusters S. The gray vertical line indicates $S = 10$, which is the actual number of digits. The skeleton clustering with VD (Voron) gives the

best clustering result in terms of adjusted Rand index at the true 10 clusters and gives good clustering results when the number of clusters is specied to be larger than the truth. However, we note that spectral clustering (SC) and naive k-means clustering (KM) give comparably good results with a small number of clusters.

The right panel of Figure 46 is the \denoised" version of the digits. We estimate the density of each observation by $[\frac{p}{n}]$-nearest-neighbor density estimator and remove the observations with the lowest 10% density. We see that all clustering results are slightly improved, but such improvement may come from the decreased total sample size after denoising. Notably, the skeleton clustering with Tube density (Tube) generates signicantly better clustering results after denoising the data, giving adjusted Rand indexes comparable to skeleton clustering with Voronoi density. This shows skeleton clustering with Tube density can be sensitive to noises in real data but still has the potential to give insightful clustering results.

H.2 Olive Oil Data

We consider another real dataset: the Olive Oil data (Tsimidou et al., 1987), a popular dataset for cluster analysis. This data set represents $d = 8$ chemical measurements on dierent specimens of olive oil produced in 9 dierent regions in Italy (northern Apulia, southern Apulia, Calabria, Sicily, inland Sardinia, and coast Sardinia, eastern and western Liguria, Umbria). There are a total of $n = 572$ observations in the dataset.

The same comparison procedure as in Section H.1 is employed, nding $k = 24$ knots by k-means and using single linkage for the skeleton clustering approaches. The performance of dierent similarity measures is presented in Figure 47. Dierent color denotes dierent similarity measures and the gray vertical line indicates the actual number of clusters 9.

Overall, the skeleton clustering with Voronoi density and Tube density works well; spectral clustering also performs well in this case. The fact that average distance fails to capture clusters in the data highlights the importance of using a density-aided similarity in this case. Note that we also include the clustering performance on the 'denoised' data, in which we remove the 10% observation with the lowest ^{p}n-Nearest-Neighbor density estimate.

I Future Work

We discuss some future directions below:

- Accounting for the randomness of knots. For our current theoretical analysis, we assume that the knots are given and non-random to simplify the problem. But in practice, knots are computed from the sample data with inherent uncertainty. The randomness of knots can aect the clustering performance because the location of knots directly impacts the Voronoi cells, which changes the value of the similarity measures and consequently the cluster label assignments. In particular, observations on the boundary of clusters will be more sensitive to any perturbations in the location of knots. Currently, there are two technical challenges when dealing with random knots. First, the randomness of knots may be correlated with the randomness of estimated edge weight, so the calculation of rates is much more complicated. Second, while there are established theories for k-means algorithm (Graf and Luschgy, 2000, 2002; Hartigan and Wong, 1979), these results only apply to the global minimum of the objective function. In reality, we are unlikely to obtain the global minimum, but instead, our inference is based on a local minimum. It is unclear how to properly derive

a theoretical statement based on local minima, so we leave this as future work.

- Skeleton clustering with similarity matrix. The idea of skeleton clustering may be generalized to data where we only observe the similarity/distance matrices such as network data. Knots can be restricted to indices in the data and we choose them by minimizing some network-based or diusion-related criteria. While Face and Tube density can be dicult to adopt, the Voronoi density is still applicable since we only need the information about pairs of observations. This might provide a new approach for community detection in network data (Zhao, 2017; Abbe, 2017).
- Detecting boundary points between clusters. Our skeleton clustering method can be applied to detect points on the boundary between two clusters. The idea is simple: in the nal cluster assignment, instead of assigning only one label to an observation, we assign h labels to an observation based on the cluster labels of h-nearest knots. The homogeneity of the label assignments can be used as a quantity to detect if a point is on the boundary or in the interior of a cluster and may serve as an uncertainty quantication of clustering. We will pursue this in the future.
- Anomaly and noise detection. As illustrated in Appendix E.2, E.4, and E.6, the single linkage criterion in our Skeleton clustering framework may detect noisy observations in the data. This suggests the possibility of using our approach for noises or anomalies similar to the DBSCAN (Campello et al., 2015; Ester et al., 1996). We will explore this direction in the future.

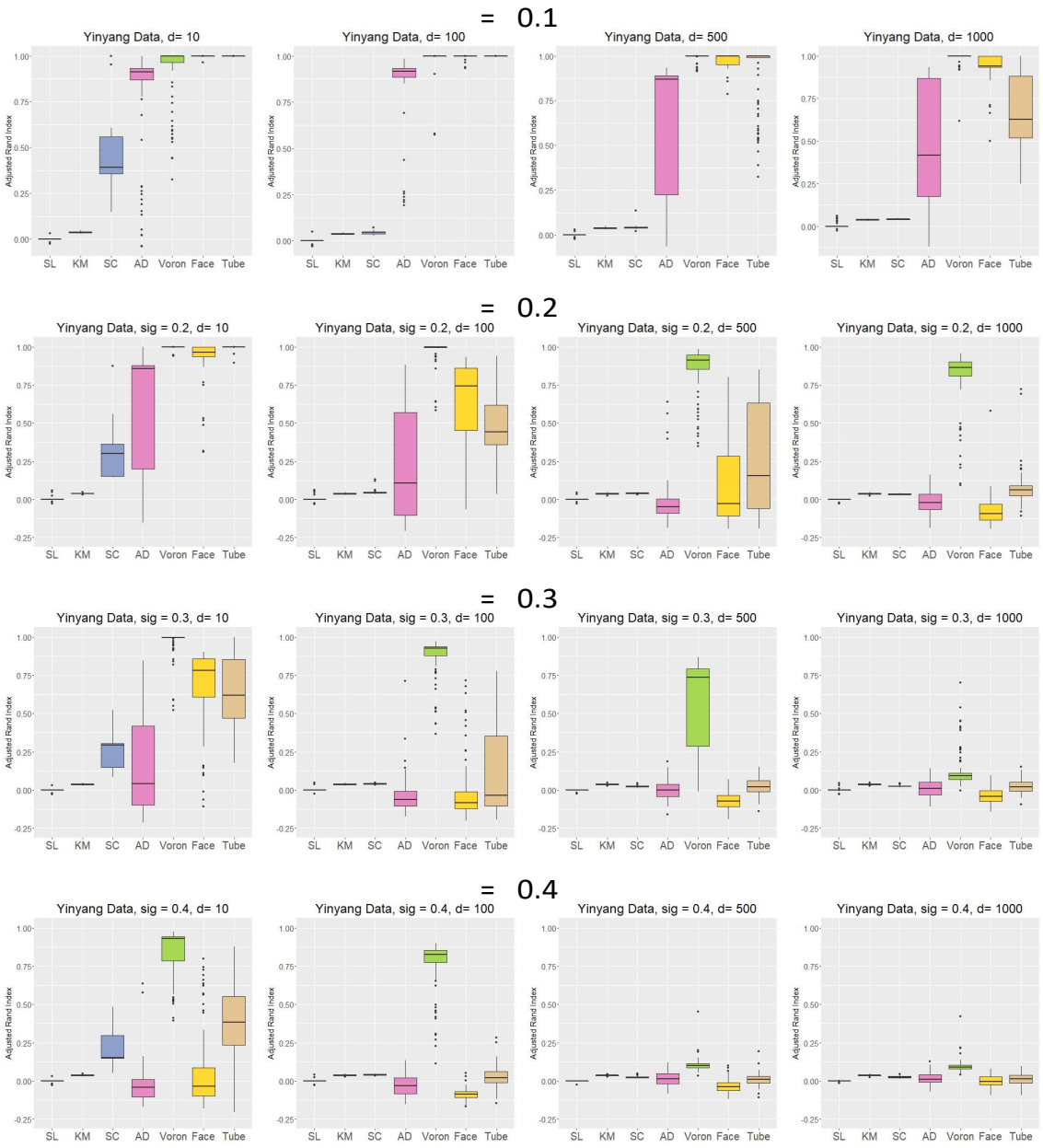


Figure 38: Adjusted Rand index performance of clustering methods on Yinyang data with dierent standard deviation for added dimensions.

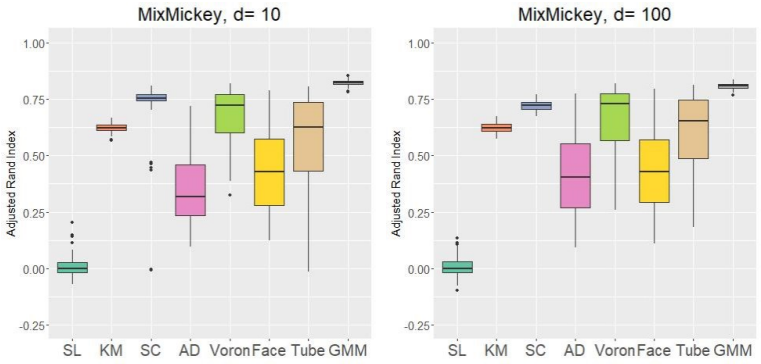


Figure 39: Comparison of clustering methods on Mix Mickey data d = 10; 100 with GMM included.

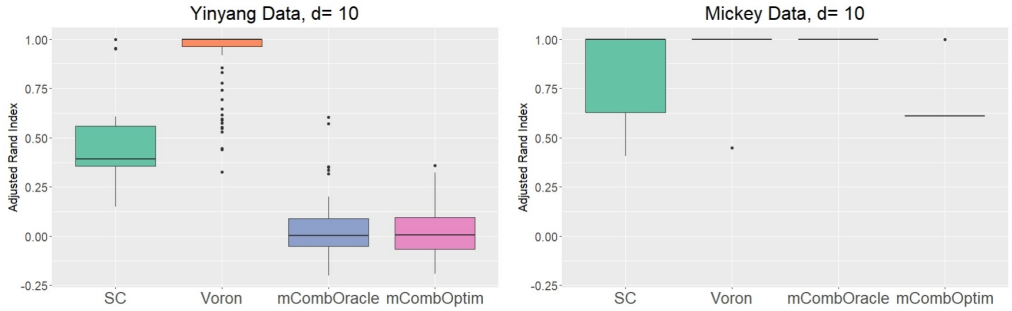
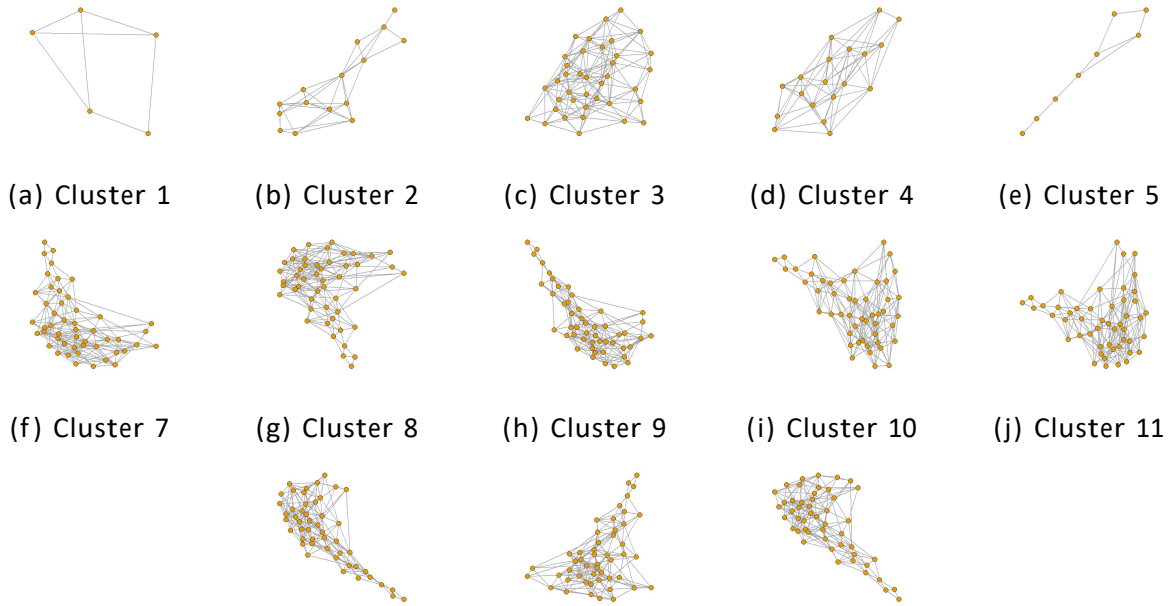


Figure 40: Results from combining mixture components on Yinyang and Mickey data d = 10.



(a) Cluster 1 (b) Cluster 2 (c) Cluster 3 (d) Cluster 4 (e) Cluster 5

(f) Cluster 7 (g) Cluster 8 (h) Cluster 9 (i) Cluster 10 (j) Cluster 11

(k) Cluster 12 (l) Cluster 13 (m) Cluster 14

Figure 41: Skeleton structures of the clusters identied for the GvHD dataset in Section 6

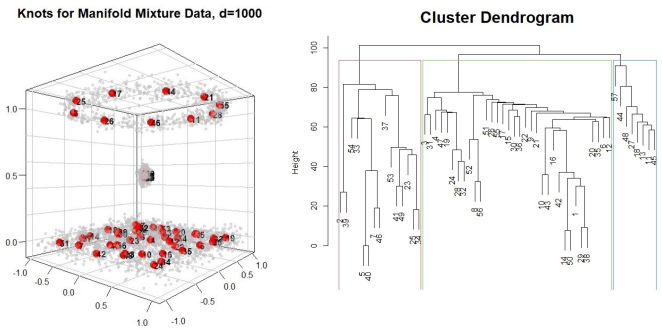


Figure 42: Results on Manifold Mixture data with dimension 100.

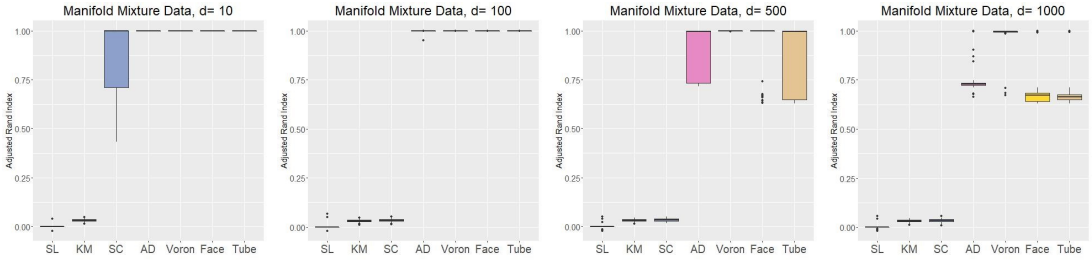


Figure 43: Comparison of adjusted Rand index using dierent similarity measures on Manifold Mixture data with dimensions 10, 100, 500, 1000.

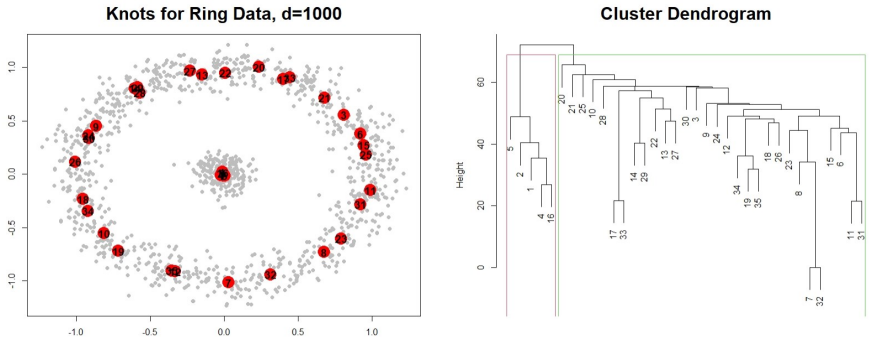


Figure 44: Results on Ring data with dimension 1000.

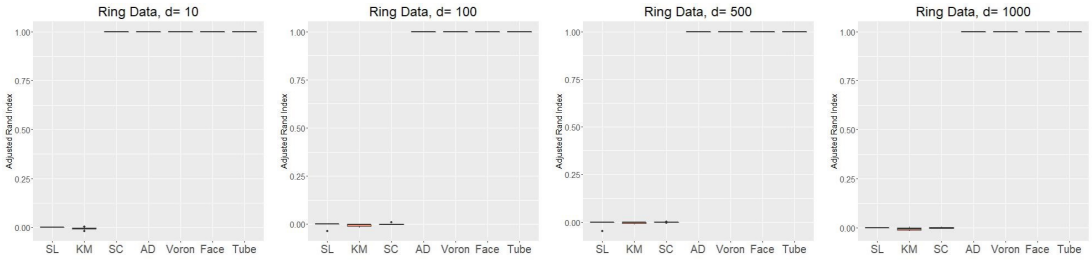


Figure 45: Comparison of the rand index using dierent similarity measures on Ring data with dimensions 10, 100, 500, 1000. Medium of 100 repetitions.

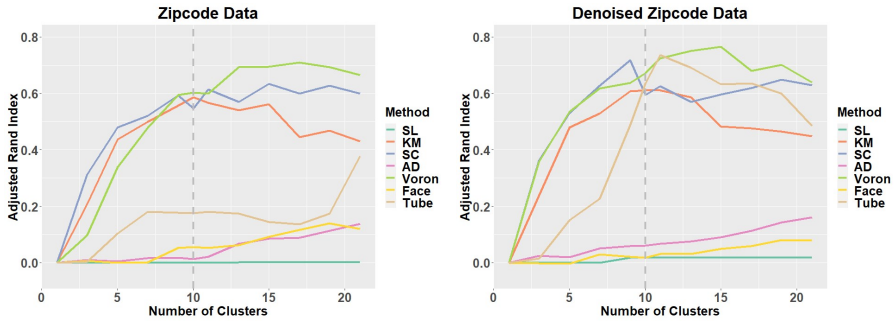


Figure 46: Comparison of dierent similarity measures on all Zipcode Data.

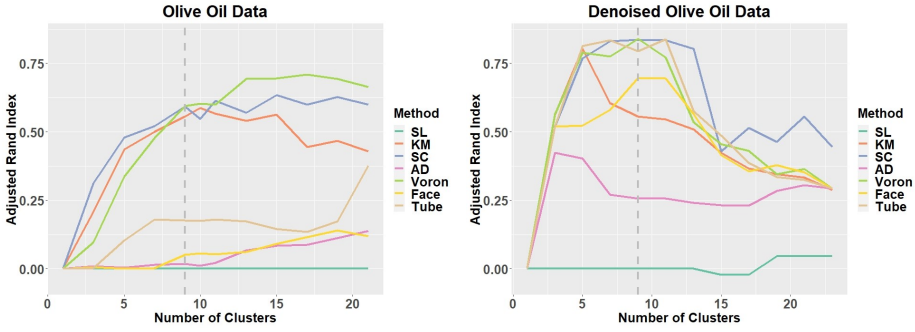


Figure 47: The clustering performance under dierent numbers of nal clusters of the Olive oil data.