Skeleton Clustering: Dimension-Free Density-Aided Clustering

Zeyu Wei and Yen-Chi Chen

Department of Statistics, University of Washington, Seattle, WA

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 empirical studies that the skeleton clustering leads to reliable clusters in multivariate and high-dimensional scenarios. Supplementary materials for this article are available online.

1. Introduction

Density-based clustering (Azzalini and Torelli 2007; Menardi and Azzalini 2014; Chacón 2015) is a popular framework for grouping observations into clusters defined 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, Ray, and Lindsay 2007; Chacón and Duong 2013; Chen, Genovese, and Wasserman 2016) finds 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-Perpinán 2015). Another famous density-based clustering approach is the level-set clustering (Caivas, Febrero, and Fraiman 2000, 2001; Mason and Polonik 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, Belkin, and Wang 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 different 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 finding clusters with irregular shapes and gives an intuitive interpretation based on the underlying PDF. Furthermore, defining 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 used 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. Specifically, the convergence rate of a density estimator is \( O_p \left(n^{-\frac{2}{d+4}}\right) \) under usual smoothness conditions (Wasserman 2006; Scott 2015), 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 (Fred and Jain 2005; Maitra 2009; Peterson, Ghosh, and Maitra 2018; Shin, Rinaldo, and Wasserman 2019; Almodovar-Rivera and Maitra 2020), 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), misclassification 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 (Chacón 2019).

The work by Aragam et al. (2020) reconstructs a nonparametric mixture model by fitting the data with a large number of general nonparametric mixture components and then partitions them into a small number of final clusters.

Our idea can be summarized as follows. We first find 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 final 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 effectiveness of skeleton clustering, we conduct a simple simulation in Figure 1. It is a $d = 200$ dimensional data consisting of five components with nonspherical shapes. The actual structure is in two-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 find the five 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 five components.

Outline. In Section 2, we describe the skeleton clustering framework. In Section 3, we introduce similarity measures that can be used 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 effectiveness of skeleton clustering in dealing with different 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 article and point out some directions for future research.

2. Skeleton Clustering Framework

In this section, we formally introduce the skeleton clustering framework. Let $X = \{X_1, \ldots, X_n\}$ be a random sample from an unknown distribution with density $p$ supported on a compact set $\mathcal{X} \subset \mathbb{R}^d$. The goal of clustering is to partition $\mathcal{X}$ into clusters $X_1, \ldots, X_S$, where $S$ is the final number of clusters.

Algorithm 1 Skeleton clustering

**Input:** Observations $X_1, \ldots, X_n$, final 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).

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 first find 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 quantifies 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 five steps: (i) Knots construction, (ii) Edges construction, (iii) Edge weights construction, (iv) Knots segmentation, and (v) 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 Sections 3 and 4. We include a short analysis of the computational complexity of our skeleton clustering framework in Appendix A, supplementary materials.

### 2.1. Knots Construction

The construction of knots is a step aiming at finding 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 locations. A simple but reliable approach for constructing knots is the $k$-means algorithm. We apply the $k$-means algorithm with a large number $k \gg S$ the desired number of final clusters, and this procedure behaves like overfitting the $k$-means. Notably, we do not use the $k$-means procedure to obtain final clustering, but, instead, we use it as an intermediate step to find concise representations of the original data.

The number of knots $k$ is a key parameter in the knots construction step. It controls the tradeoff 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 find that a simple reference rule for $k$ to be around $\sqrt{n}$ works well in our empirical studies (Appendix E1, supplementary materials). 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 final clustering. Moreover, to take care of observations in the low-density areas that could cause problems for the $k$-means clustering, one may first preprocess or denoise the data by removing observations in the low-density area and then apply the $k$-means clustering to find out the knots.

In this work, we use overfitting $k$-means as the default way for knots construction, but there are alternative approaches to find knots such as subsampling, the coreset construction methods (Bachem, Lucic, and Krause 2017; Turner, Liu, and Rigollet 2020), and the Self-Organizing Maps (SOM) (Heskes 2001). We show in Appendix E2, supplementary materials that the SOM can also be used to find knots but requires more careful treatments such as removing knots with few or even no observations.

---

Figure 2. Skeleton clustering illustrated by Two Moon Data ($d = 2$).
Figure 3. Voronoi Tessellation as blue dashed lines and Delaunay Triangulation by red solid lines.

and the performance is slightly worse than that of the overfitting $k$-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 overfitting $k$-means algorithm in this work and recommend using it in practice.

Remark 1. Since the $k$-means algorithm does not always find the global optimum, we repeat it many times with random initial points (generally 1000 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 find a useful representation, we do not need to find 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 find the edges connecting them. Let $c_1, \ldots, c_k$ be the given knots and we use $C = \{c_1, \ldots, c_k\}$ 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 (Voronoï 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 $\mathcal{X}$ whose distance to $c_j$ is the smallest compared to other knots (see Figure 3). That is,

$$C_j = \{x \in \mathcal{X} : d(x, c_j) \leq d(x, c_\ell) \ \forall \ell \neq j\},$$

where $d(x, y)$ is the usual Euclidean distance. Therefore, we add an edge between knots $(c_i, c_j)$ if $C_i \cap C_j \neq \emptyset$. 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 $C$.

The Delaunay triangulation graph is conceptually intuitive and appealing and is used 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 $d$ (Chazelle 1993; Amenta, Attali, and Devillers 2007). 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 $\hat{DT}(C)$ by examining the two-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 two-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, Schek, and Blott 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 final clusters, and we apply hierarchical clustering by converting the similarity measures into distances. Particularly, for given similarity measures $\{s_{ij}\}_{i\neq j}$ where only connected pairs can take nonzero entries and let $s_{\text{max}} = \max_{i \neq j} s_{ij}$, we define the corresponding distances as $d_{ij} = 0$ if $i = j$ and $d_{ij} = s_{\text{max}} - s_{ij}$ 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, supplementary materials. 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 different linkage methods is computationally tractable as only the knots need to be segmented.

The number of final 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 different resolutions. Consequently, analysts can experiment with different numbers of final 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 final number $S$ being larger than the true number of meaningful components may be needed to achieve better clustering results (see Appendix E, supplementary materials).

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 final segmentation. Peterson, Ghosh, and Maitra (2018) also presented similar ideas choosing the final 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, Walther, and Hastie 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 $\ell$, we assign cluster membership label $\ell$ to observation $X_i$.

Remark 3. There are other methods in clustering literature for assigning labels of observations based on identified structures. Azzalini and Torelli (2007) and Scrucca (2016) assign unlabeled data based on density ratios. DBSCAN and HDBSCAN (Ester et al. 1996; Campello et al. 2015) 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 $(c_j, c_\ell)$ based on the number of observations whose two-nearest knots are $c_j$ and $c_\ell$. We start with defining the Voronoi density based on the underlying probability measure and then introduce its sample analog. Given a metric $d$ on $\mathbb{R}^d$, the 2-Nearest-Neighbor (2-NN) region of a pair of knots $(c_j, c_\ell)$ is defined as

$$A_{j\ell} = \{x \in \mathcal{X} : d(x, c_j) > \max\{d(x, c_i), d(x, c_j)\}, \forall i \neq j, \ell\}. \quad (2)$$

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.

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

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

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 different 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 $1/\|c_j - c_\ell\|$ and hence can be scale-dependent.

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

$$\tilde{S}_{j\ell}^{VD} = \frac{\hat{P}_n(A_{j\ell})}{\|c_j - c_\ell\|}. \quad \text{(4)}$$

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 $A_{j\ell}$ by $\hat{A}_{j\ell}$.

The Voronoi density can be computed in a fast way. The numerator, which only depends on two-nearest-neighbors calculation, can be computed efficiently by the k-d tree algorithm (Bentley 1975). For high-dimensional space, space partitioning search approaches like the k-d tree can be inefficient but a direct linear search still gives a short run-time (Weber, Schek, and Blott 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 $\hat{D}(C)$, 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 define 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 one-dimensional density estimation task regardless of the dimension of the ambient space. Formally, let the face region between two knots \( c_j, c_\ell \) be \( F_{j\ell} = C_j \cap C_\ell \). At the population level, the FD is defined as
\[
S_{j\ell}^{FD} = \int_{F_{j\ell}} p(x) \mu_{d-1}(dx) = \int_{F_{j\ell}} d\mathbb{P}(x),
\]
where \( \mu_m(dx) \) denotes the \( m \)-dimensional volume measure.

To estimate the FD, we use 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 first projecting the observations onto the central line and then using the one-dimensional Kernel Density Estimator (KDE) to evaluate the density at the midpoint. Specifically, fix two knots \( c_j, c_\ell \), let \( C_j, C_\ell \) be the corresponding Voronoi cells, and denote \( \Pi_{j\ell}(x) \) as the projection of \( x \in \mathcal{X} \) onto the central line passing through \( c_j \) and \( c_\ell \), we define the estimator \( \hat{S}_{j\ell}^{FD} \) to be
\[
\hat{S}_{j\ell}^{FD} = \frac{1}{nh} \sum_{X_i \in C_j \cup C_\ell} K \left( \frac{\Pi_{j\ell}(X_i) - (c_\ell + c_j)/2}{h} \right)
\]
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 suffers from the curse of dimensionality (Wasserman 2006; Chacón, Duong, and Wand 2011; Chen 2017), the kernel estimator in equation (6) bypasses it.

### 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 affect the estimation of the Face density from a theoretical perspective (Appendix D.2, supplementary materials). Here we propose a measure similar to the Face density measure but with a predefined regular shape. For a point \( x \), we define the Disk Area centered at \( x \) with radius \( R \) and normal direction \( v \) (see Figure 5 for an illustration) as
\[
\text{Disk}(x, R, v) = \{ y : \| x - y \| \leq R, (x - y)^T v = 0 \}
\]
To measure the similarity between knots \( c_j \) and \( c_\ell \), we examine the integrated density within the disk areas along the central line.

In more detail, the central line can be expressed as \( \{ c_j + t(c_\ell - c_j) : t \in [0, 1] \} \), and any point on the central line can be written as \( c_j + t(c_\ell - c_j) \) for some \( t \). For a point \( c_j + t(c_\ell - c_j) \), we define the integrated density in the disk region (called Disk Density) as
\[
\text{pDisk}_{j\ell,R}(t) = \mathbb{P} \left( \text{Disk}(c_j + t(c_\ell - c_j), R, c_\ell - c_j) \right) = \int_{\text{Disk}(c_j + t(c_\ell - c_j), R, c_\ell - c_j)} p(x) dx.
\]
The Tube Density (TD) measures the similarity between \( c_j \) and \( c_\ell \) as the minimal disk density along the central line, that is,
\[
\hat{S}_{j\ell}^{TD} = \inf_{t \in [0,1]} \text{pDisk}_{j\ell,R}(t)
\]
In other words, with given \( c_j, c_\ell \), we survey all Disk Density along the central line and retrieve the infimum as the similarity measure between two knots.

In this work, we set \( R \) based on the root mean squared distances within each Voronoi cell. Specifically, for knot \( c_j \) and the corresponding Voronoi cell \( C_j \), we calculate
\[
R_j = \left[ \frac{1}{|C_j| - 1} \sum_{x_i \in C_j} \| x_i - c_j \|^2 \right]^{1/2}
\]
where \( |C_j| \) 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, supplementary materials, 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 \( \Pi_{j\ell}(x) \) be the projection of a point \( x \) on the line through \( c_j, c_\ell \). We first estimate the pDisk via
\[
\text{pDisk}_{j\ell,R}(t) = \frac{1}{nh} \sum_{i=1}^{n} K \left( \frac{\Pi_{j\ell}(X_i) - (c_\ell - c_j)/2}{h} \right) I(||X_i - \Pi_{j\ell}(X_i)|| \leq R)
\]
and then estimate the TD as
\[
\hat{S}_{j\ell}^{TD} = \inf_{t \in [0,1]} \text{pDisk}_{j\ell,R}(t).
\]
where the infimum 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 the kernel and the bandwidth selections for kernel density estimations in Appendix F.3, supplementary materials. By default, we use the Gaussian kernel with the normal scale bandwidth selector (NS) (Chacón, Duong, and Wand 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 effectiveness of the newly proposed density-aided similarity measures. We assume the set of knots $C = \{c_1, \ldots, c_k\}$ is given and nonrandom to simplify the analysis because (i) it is hard to quantify k-means uncertainty, and (ii) with large $k$, it is extremely likely for k-means to stick within the local minimum. Note that this implies the corresponding Voronoi cells $C = \{C_1, \ldots, C_k\}$ and the 2-NN regions $\{A_{ij}\}_{i=1, \ldots, k, j \neq k}$ (2) of all pairs of knots are fixed 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 referred to Appendices 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 one-dimensional KDE under some regularity conditions. All proofs are included in Appendix D, supplementary materials.

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 $\min_{(j, \ell) \in E} P(A_{j\ell}) \geq \frac{c_0}{d}$ and $\min_{(j, \ell) \in E} |c_j - c_\ell| \geq \frac{c_0}{k^{1/2}}$.

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 sufficient length. This condition is mild because when the dimension of data $d$ is fixed, 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 $f_0(x) > c_1 > 0$ for $P - a.e.$ and that $\min_{(j, \ell)} |A_{j\ell}| \geq c_2 |X|$ where $|X|$ is the volume of the support, then we have $\min_{(j, \ell) \in E} P(A_{j\ell}) \geq \frac{c_2}{n}$. 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 $j \neq \ell$ that shares an edge, the similarity measure based on the Voronoi density satisfies

\[
\frac{\hat{S}_{VD}^j - 1}{\hat{S}_{VD}^\ell - 1} = O_p \left( \sqrt{\frac{k}{n}} \right),
\]

\[
\max_{j \neq \ell} \frac{\hat{S}_{VD}^j - 1}{\hat{S}_{VD}^\ell - 1} = O_p \left( \sqrt{\frac{k}{n} \log k} \right),
\]

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

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 effective local sample size $n/k$. Consequently, 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 (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 define the true clusters as the connected components of the skeleton graph with edges having true Voronoi density similarities $S_{VD}^\ell$ over a known threshold $\tau > 0$. We show below that cutting the skeleton graph based on estimated edge similarities at the same threshold $\tau$ recovers the true clustering with a high probability. Since the knots are fixed, 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 $L^* = \{L_{\ell}^*\}_{\ell = 1, \ldots, L}$, where $L_{\ell}^*$ contains all the knot indices belonging to the partition $\ell$. Let the partition based on estimated edge similarities be $\hat{L}$. We assume that

(P1) The true partition $L^*$ under the threshold $\tau$ remains the same when the thresholding level is within $(\tau(1 - \epsilon), \tau(1 + \epsilon))$ for some $\epsilon > 0$.

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

Theorem 2 (Adjusted Rand Index Guarantee). Assume (B1) and (P1) and let $p_{\text{min}} = \min_{j, \ell} P(A_{j\ell})$, then

\[
P\{\text{ARI}(L^*, \hat{L}) < 1\} \leq k(k - 1) \exp \left( -\frac{1}{\tau^2} \frac{p_{\text{min}} n}{(1 - p_{\text{min}}) + \frac{\tau^2}{2}} \right)
\]

(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, supplementary materials). Generally, our framework with the Voronoi density similarity measure is superior
among all the compared clustering methods. In Appendix E, supplementary materials, 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, supplementary materials. The R implementation of the skeleton clustering methods along with some simulations can be found 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 different dimensions, called the Manifold Mixture data, in Appendix G.1, supplementary materials and an additional simulation called the Ring data in Appendix G.2, supplementary materials. For the simulations within Section 5.1 and Appendix G, supplementary materials, when using the skeleton clustering methods, the number of knots is set to be \( k = \lceil \sqrt{n} \rceil \) 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 final clusters with the true number of final 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 \( c_ℓ \) 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 two-dimensional data containing five 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 \( k = \lceil \sqrt{3200} \rceil = 57 \) knots for the skeleton clustering procedure (Figure 6). 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 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 F7, supplementary materials.

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, Ghosh, and Maitra (2018) (KmH). We include the comparison with merging model-based clusters approaches in Appendix F9, supplementary materials.

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 five 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 two-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 \( k = \lceil \sqrt{1200} \rceil = 35 \). 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 scatterplot 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 \( d = 1000 \).

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, supplementary materials and the Olive Oil data (Tsimidou, Macrae, and Wilson 1987) in Appendix H.2, supplementary materials.

GvHD is a significant problem in the field 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, Brinkman, and Gottardo 2008; Baudry et al. 2010)
have identified the presence of high values in CD3, CD4, CD8β cell sub-populations as a significant characteristic 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, CD8β) 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, CD8β region, the distribution of the positive sample is different from the control sample also in some region with medium to the low CD3, CD4, and CD8β. Later we will demonstrate that our clustering framework can identify all such differences in distributions.

To apply the skeleton clustering for a fair comparison of the two samples, we first construct knots from each sample separately. Specifically, we apply the \( k \)-means method to find \( k_1 = \lceil \sqrt{n_1} \rceil \) knots for the positive sample and find \( k_2 = \lceil \sqrt{n_2} \rceil \) 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, supplementary materials suggests average linkage for this scenario. The skeleton clustering result is displayed in the right panel of Figure 10 with the number of final clusters chosen to be \( S = 14 \). This choice follow Baudry et al. (2010) where the authors suggest to fit nine components on the positive sample and 3–5 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 find 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, CD8β region identified by previous works with nearly all data points belonging to the positive patient. Cluster 6 is also scattered in the high CD3, CD4, CD8β region but has all the observations coming from the control sample. However, the small size (only 17 data 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+ CD8β+ 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 findings. Clusters 9, 12, and 13 also have high proportions of positive samples. These clusters are in the mid to low CD3, CD4, CD8β 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 CD8β but low CD3 and CD4 region.

Model-based clustering approaches Lo, Brinkman, and Gottardo (2008) and Baudry et al. (2010) have an advantage for managing this cytometry data as they can parametrically describe the behaviors of data samples in different regions. The overlapping
between different structures and the overall four-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, supplementary materials. 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 in Appendix, supplementary materials).

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 different individuals and different 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 effectiveness 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 affect 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, supplementary materials). Overall, given the flexibility of the skeleton clustering framework, other possibilities by incorporating different methods for different steps in the framework can be explored.

### Supplementary Materials

The supplementary materials contain additional theoretical results, the proofs, and additional empirical results. R code implementation is also included.

### Disclosure Statement

The authors report there are no competing interests to declare.

### Funding

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

### ORCID

Zeyu Wei [https://orcid.org/0000-0003-1614-4458](https://orcid.org/0000-0003-1614-4458)

Yen-Chi Chen [https://orcid.org/0000-0002-4485-306X](https://orcid.org/0000-0002-4485-306X)

### References

- Almodovar-Rivera, I. A., and Maitra, R. (2020), “Kernel-Estimated Nonparametric Overlap-based Syncytial Clustering,” *Journal of Machine Learning Research*, 21, 1–54. [1124]
- Amenta, N., Attali, D., and Devillers, O. (2007), “Complexity of Delaunay Triangulation for Points on Lower-Dimensional Polyhedra,” in *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA ’07, pp. 1106–1113, USA. Society for Industrial and Applied Mathematics. [1127]
- Aragam, B., Dan, C., Xing, E. P., and Ravikumar, P. (2020), “Identifiability of Nonparametric Mixture Models and Bayes Optimal Clustering,” *The Annals of Statistics*, 48, 2277–2302. DOI:10.1214/19-AOS1887. [1124]
- Azzalini, A., and Torelli, N. (2007), “Clustering via Nonparametric Density Estimation,” *Statistics and Computing*, 17, 71–80. [1124,1127,1128]
- Bachem, O., Lucic, M., and Krause, A. (2017), “Practical Coreset Constructions for Machine Learning,” arXiv preprint. DOI:10.48550/arXiv.1703.06476 [1126]
- Baudry, J.-P., Raftery, A. E., Celeux, G., Lo, K., and Gottardo, R. (2010), “Combining Mixture Components for Clustering,” *Journal of Computational and Graphical Statistics*, 19, 332–353. DOI:10.1198/jcgs.2010.08111. [1124,1132,1133]
- Bentley, J. L. (1975), “Multidimensional Binary Search Trees used for Associative Searching,” *Communications of the ACM*, 18, 509–517. [1128]
- Berg, M. d., Cheong, O., Kreveld, M. v., and Overmars, M. (2008), *Computational Geometry. Algorithms and Applications* (3rd ed.), Springer-Verlag TELOS, Santa Clara, CA, USA. [1130]
- Brinkman, R. R., Gasparetto, M., Lee, S. J. J., Ribickas, A. J., Perkins, J., Janssen, W., Smiley, R., and Smith, C. (2007), “High-Content Flow Cytometry and Temporal Data Analysis for Defining a Cellular Signature of Graft-Versus-Host Disease,” *Biology of Blood and Marrow Transplantation*, 13, 691–700. [1132]
- Campello, R. J., Moulavi, D., Zimek, A., and Sander, J. (2015), “Hierarchical Density Estimates for Data Clustering, Visualization, and Outlier Detection,” *ACM Transactions on Knowledge Discovery from Data*, 10, 1–51. [1128]
- Carreira-Perpinán, M. A. (2015), “A Review of Mean-Shift Algorithms for Clustering,” arXiv preprint arXiv:1503.00687. [1124]
- Chacón, J. E. (2015), “A Population Background for Nonparametric Density-Based Clustering,” *Statistical Science*, 30, 518–532. DOI:10.1214/15-STSS526. [1124]
- Chacón, J. E. (2019), “Mixture Model Modal Clustering,” *Advances in Data Analysis and Classification*, 13, 379–404. [1124]
- Chacón, J. E., and Duong, T. (2013), “Data-Driven Density Derivative Estimation, with Applications to Nonparametric Clustering and Bump Hunting,” *Electronic Journal of Statistics*, 7, 499–532. [1124]
- Chacón, J. E., Duong, T., and Wand, M. P. (2011), “Asymptotics for General Multivariate Kernel Density Derivative Estimators,” *Statistica Sinica*, 21, 807–840. [1129]
- Chaudhuri, K., and Dasgupta, S. (2010), “Rates of Convergence for the Cluster Tree,” in *Proceedings of the 23rd International Conference on Neural Information Processing Systems - Volume 1*, NIPS’10, pp. 343–351, Red Hook, NY, USA. Curran Associates Inc. [1124]
