stats

MDPI

Article

Refined Mode-Clustering via the Gradient of Slope

Kunhui Zhang * and Yen-Chi Chen

Department of Statistics, University of Washington, Seattle, WA 98195, USA; yenchic@uw.edu
* Correspondence: zhangkh@uw.edu

Abstract: In this paper, we propose a new clustering method inspired by mode-clustering that not only finds clusters, but also assigns each cluster with an attribute label. Clusters obtained from our method show connectivity of the underlying distribution. We also design a local two-sample test based on the clustering result that has more power than a conventional method. We apply our method to the Astronomy and GvHD data and show that our method finds meaningful clusters. We also derive the statistical and computational theory of our method.

Keywords: clustering; mode-clustering; gradient descent; two-sample test

check for updates

Citation: Zhang, K.; Chen, Y.-C. Refined Mode-Clustering via the Gradient of Slope. *Stats* **2021**, *4*, 486–508. https://doi.org/10.3390/stats4020030

Academic Editors: Marta Nai Ruscone and Daniel Fernández

Received: 9 April 2021
Accepted: 24 May 2021
Published: 1 June 2021

1. Introduction

Mode-clustering is a clustering analysis method that partitions the data into groups by the local modes of the underlying density function [1–4]. A density local mode is often a signature of a cluster, so mode-clustering leads to clusters that are easy to interpret. In practice, we estimate the density function from the data and perform mode-clustering via the density estimator. When we use a kernel density estimator (KDE), there exists a simple and elegant algorithm called the mean-shift algorithm [5–7] that allows us to compute clusters easily. The mean-shift algorithm has made the mode-clustering a numerically friendly problem.

When applied to a scientific problem, we often use a clustering method to gain insight from the data [8,9]. Sometimes, finding clusters is not the ultimate goal. The connectivity among clusters may yield valuable information for scientists. To see this, consider the galaxy sample from the Sloan Digital Sky Survey [10] in Figure 1. While the original data is 3D, here we use a 2D slice of the original data to illustrate the idea. Each black dot indicates the location of a galaxy at a particular location in the sky. Astronomers seek to find clusters of galaxies and their connectivity, since these quantities (clusters and their connections) are associated with the large-scale structures in the universe. Our method finds the underlying connectivity structures without assuming any parametric form of the underlying distribution. In the middle panel, we display the results by the usual mode-clustering method, which only shows clusters, but not how they connect with each other. On the other hand, our proposed method is given in the right panel, which finds a set of dense clusters (purple regions) along with some regions serving as bridges connecting clusters (green areas) and a set of low-density regions (yellow regions). Thus, our clustering method allows us to better identify the structures of galaxies.

We improve the usual mode-clustering method by (1) adding additional clusters that can further partition the entire sample space, and (2) assigning an attribute label to each cluster. The attribute label will indicate if this cluster is a 'robust cluster' (a cluster around a local mode; purple regions in Figure 1), a 'boundary cluster' (a cluster bridging two or more robust clusters; green regions in Figure 1), or an 'outlier cluster' (a cluster representing low-density regions; yellow regions in Figure 1). With this refined clustering result, we gain further insights into the underlying density function and are able to infer the intricate structure behind the data. Furthermore, we can apply our improved clustering method to the two sample tests. In this case, we can identify the local differences between the two populations and provide a more sensitive result. Note that in the usual case of cluster analysis, adding more clusters is not a preferred idea. However, if our goal is to detect the underlying structures (such as finding the connectivity of high-density regions in the galaxy data in Figure 1), using more clusters as an intermediate step to find connectivity could be a plausible approach.

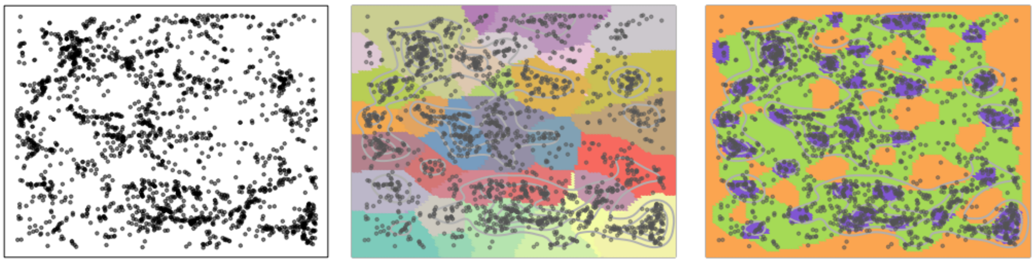


Figure 1. Using two clustering methods to learn the cosmic webs. **Left:** the raw galaxy data from the Sloan Digital Sky Survey. **Middle:** the clustering result using the conventional mode/mean-shift clustering. This conventional mode-clustering method fails to detect the connectivity among clusters. **Right:** the clustering result based on our method, where the color indicates different types of clusters.

To summarize, our main contributions are as follows:

- We propose a new clustering method by the slope function that has an additional attribute label of each cluster (Section 3).
- We propose new two-sample tests using the clustering result (Section 4).
- We introduce a visualization method using the detected clusters (Algorithm 3).
- We derive both statistical and computational guarantees of the proposed method (Section 7).

Related work. The idea of using local modes to cluster observations can be dated back to [5], where the authors used local modes of the KDE to cluster observations and propose the mean-shift algorithm for this purpose [5,11]. mode-clustering has been widely studied in statistics and the machine-learning community [3,4,7,12–14]. However, the KDE is not the only option for mode-clustering—[1,15] proposed a Gaussian mixture model method, and [16] used a fuzzy clustering algorithm, and [17] introduced a nearest-neighbor density method.

Outline. The paper is organized as follows. We start with a brief review on mode-clustering in Section 2 and formally introduce our method in Section 3. In Section 4, we combine the two-sample test and our approach to create a local two-sample test. We use simulations to illustrate our method on simple examples in Section 5. We show the applicability of our approach to three real datasets in Section 6. Finally, we study both statistical and computational theories of our method in Section 7.

2. Review of Mode-Clustering

We start with a review of mode-clustering [2,4,12,18]. The concept of mode-clustering is based on the rationale of associated clusters to the regions around the modes of the density. When the density function is estimated by the kernel density estimator, there is an elegant algorithm called the mean-shift algorithm [5] that can easily perform the clustering.

In more detail, let p be a probability density function with a compact support $\mathbb{K} \subset \mathbb{R}^d$. Starting at any point x, mode-clustering creates a gradient ascent flow $\gamma_x(t)$ such that

$$\gamma_x(0) = x, \quad \gamma_x'(t) = \nabla p(\gamma_x(t)).$$

Namely, the flow $\gamma_x(t)$ starts at point x and moves according to the gradient at the present location. Let $\gamma_x(\infty) = \lim_{t\to\infty} \gamma_x(t)$ be the destination of the flow $\gamma_x(t)$. According to the Morse theory [19,20], when the function is smooth (being a Morse function), such a flow converges to a local maximum of p except for starting points in a set of the Lebesgue measure 0. The mode-clustering partitions the space according to the destination of the gradient flow, that is, for two points x, y, they will be assigned to the same cluster if $\gamma_x(\infty) = \gamma_y(\infty)$. For a local mode η, we define its basin of attraction as $D(\eta) = \{x : \gamma_x(\infty) = \eta\}$. The basin of attraction describes the set of points that belongs to the same cluster.

In practice, we do not know p, so we replace it by a density estimator, $\hat{p}_n$. A common approach to estimate p as the kernel density estimator, in which $\hat{p}_n$ is

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

where K is a smooth function (also known, according to the Morse theory, as the kernel function), such as a Gaussian kernel, and $h > 0$ is the smoothing bandwidth that determines the amount of smoothness. Since we used a nonparametric density estimator, we did not need to assume any parametric assumptions on the shape of the distribution.) With this choice, we the define a sample analogue to the flow $\gamma_x(t)$ as

$$\hat{\gamma}_x(0) = x, \quad \hat{\gamma}_x'(t) = \nabla \hat{p}(\hat{\gamma}_x(t))$$

and partition the space according to the destination of $\hat{\gamma}_x$.

3. Clustering via the Gradient of Slope

3.1. Refining the Clusters by the Gradient of Slope

As is mentioned previously, the mode-clustering has some limitations that the resulting clusters do not provide enough information on the finer structure of the density. To resolve this problem, we introduce a new clustering method by considering gradient descent flows of the 'slope' function. Let $\nabla p(x)$ be the gradient of p. Define the slope function of p as $s(x) = \|\nabla p(x)\|^2$. Namely, the slope function is the squared amplitude of the density gradient.

An interesting property of the slope function is that the minimal points $\{x : s(x) = 0\} = \{x : \nabla p(x) = 0\} = \mathcal{C}$ form the collection of critical points of p, so it contains local modes of p as well as other critical points, such as saddle points and local minima. According to the Morse theory [21,22], there is a saddle point between two nearby local modes when the function is a Morse function. A Morse function is a smooth function f, such that all eigenvalues of Hessian Matrix of f at every critical point are away from 0. This implies that saddle points may be used to bridge connecting regions around two local modes.

With this insight, we propose to create clusters using the gradient 'descent' flow of $s(x)$. Let $\nabla s(x)$ be the gradient of the slope function. Given a starting point $x \in \mathbb{R}^d$, we construct a gradient descent flow as follows:

$$\pi_x(0) = x, \ \ \pi_x'(t) = -\nabla s(\pi_x(t)). \tag{1}$$

That is, π_x is a flow starting from x and moving along the direction of ∇s. Similar to mode-clustering, we use the destination of gradient flows to cluster the entire sample space.

Note that if the slope function s is a Morse function, the corresponding PDF p will also be a Morse function, as described in the following Lemma.

Lemma 1. *If $s(x)$ is a Morse function, then $p(x)$ is a Morse function.*

Throughout this paper, we will assume that the slope function is Morse. Thus, the corresponding PDF will also be a Morse function and all critical points of the PDF will be well-separated.

3.2. Type of Clusters

Recall that $\mathcal{C}$ is the collection of critical points of density p. Let $\mathcal{S}$ be the collection of local minima of the slope function $s(x)$. It is easy to see $\mathcal{C} \subset \mathcal{S}$, since any critical point of p has gradient 0, so it is also a local minimum of s.

Thus, the gradient flow in Equation (1) leads to a partition of the sample space. Specifically, let $\pi_x(\infty)$ be the destination of the gradient flow $\pi_x(t)$. For an element $m \in \mathcal{C}$, let $\mathbb{S}(m) = \{x : \pi_x(\infty) = m\}$ be its basin of attraction.

We use the sign of eigenvalues of $\nabla^2 p(x)$ to assign an additional attribute to each basin, so the set $\{\mathbb{S}(m) : m \in \mathcal{C}\}$ forms a collection of meaningful disjoint regions. In more detail, for a critical point $m \in \mathcal{C}$ such that $p(m) > \delta$ for a small threshold δ, its $\mathbb{S}(m)$ is classified according to

$$\mathbb{S}(m) \text{ is a} \begin{cases} \text{robust cluster} & \text{if } s(m) = 0, \lambda_1(m) < 0; \\ \text{outlier cluster} & \text{if } s(m) = 0, \lambda_d(m) > 0; \\ \text{boundary cluster} & \text{otherwise,} \end{cases} \tag{2}$$

where $\lambda_l(x)$ is the l-th ordered eigenvalue of $\nabla^2 p(x)$ ($\lambda_1(x) \geq \ldots \geq \lambda_d(x)$). In the case of $p(m) \leq \delta$, we always assign it as an outlier cluster. Note that the threshold δ was added to stabilize the numerical calculation. In other words, we refer to a basin of attraction in $\mathbb{S}(m)$ as a robust cluster if $m \in \mathcal{C}$ is a local mode of p. If m is a local minimum of p, then we call its basin of attraction an outlier cluster. The remaining clusters, which are regions connecting robust clusters, are denoted as boundary cluster. Note that the regions outside the support are, by definition, a set of local minima. We assign the same cluster label to those x whose destination $\pi_x(\infty)$ is outside the support, which is an outlier cluster.

Our classification of $\mathbb{S}(m)$ is based on the following observations. Regions around local modes of p are where we have strong confidence that these points should belong to the cluster represented by their nearby local modes. Regions around local minima of p are the low-density areas where we should treat them as anomaly points/outliers. Figure 1 provides a concrete example that our clustering method could lead to more scientific insight–the connectivity among robust clusters may reveal intricate structure of the underlying distribution.

Defining different types of clusters allows us to partition the whole space into meaningful subregions. Given a random sample, to assign the cluster label to each of them, we simply examine which basins of attraction these data points fall in and pass the cluster labels from the regions to the data points. After assigning cluster labels to data points, the cluster categories in Equation (2) provide additional information about the characteristics of each data point. Those data points in robust clusters are data points that are highly clustered together; points in the outlier clusters are data points in low-density regions, which could be viewed as anomalies; the rest of points are in the boundary clusters, where these points are not well-clustered and are on the connection regions among different robust clusters.

3.3. Estimators

The above procedure is defined when we have access to the true PDF p. In practice, we do not know p, but we have an IID random sample $X_1, \ldots, X_n$ from p with a compact support $\mathbb{K}$. So we estimate p using $X_1, \ldots, X_n$ and then use the estimated PDF to perform the above clustering task.

While there are many choices of density estimators, we consider the kernel density estimator (KDE) in this paper, since it has a nice form and its derivatives are well-established [14,23–25]. In more detail, the KDE is

$$\hat{p}_n(x) = \frac{1}{nh^d} \sum_{i=1}^{n} K\left(\frac{x - X_i}{h}\right), \qquad \hat{s}_n(x) = \|\nabla \hat{p}_n(x)\|^2,$$

where K is a smooth function (also known as the kernel function) such as a Gaussian kernel, and $h > 0$ is the smoothing bandwidth that determines the amount of smoothness. Note that the bandwidth h in the KDE could be replaced by h_i that depends on each observation. This is called the variable bandwidth KDE in Breiman et al. [26]. However, since the choice of how h_i depends on each observation is a non-trivial problem, so to simplify the problem, we set all bandwidths to be the same.

Based on $\hat{s}_n(x)$, we first construct a corresponding estimated flow using $\nabla \hat{s}_n(x)$:

$$\hat{\pi}_x(0) = x; \quad \hat{\pi}'_x(t) = -\nabla \hat{s}_n(\hat{\pi}_x(t)). \tag{3}$$

An appealing feature is that $\nabla \hat{s}_n(x)$ has an explicit form:

$$\nabla \hat{s}_n(x) = \nabla^2 \hat{p}_n(x) \nabla \hat{p}_n(x), \tag{4}$$

where $\nabla \hat{p}_n(x)$ and $\nabla^2 \hat{p}_n(x)$ are the estimated density gradient and Hessian matrix of p. Thus, to numerically construct the gradient flow $\hat{\pi}_x(t)$, we update x by

$$x \leftarrow x - \gamma \cdot \nabla^2 \hat{p}_n(x) \nabla \hat{p}_n(x), \tag{5}$$

where $\gamma > 0$ is the learning rate parameter. Algorithm 1 summarizes the gradient descent approach.

Algorithm 1: Slope minimization via gradient descent.

```
1. Input: $\hat{p}_n(x)$ and a point $x$.
2. Initialize $x_0 = x$ and iterate the following equation until convergence: ($\gamma$ is a
   step size that could be set to a constant)
       $x_t = x_{t-1} - \gamma \cdot \nabla^2 \hat{p}_n(x_{t-1}) \nabla \hat{p}_n(x_{t-1}).$
3. Output: $x_\infty$ .
```

With an output from Algorithm 1, we can group observations into different clusters, with each cluster labeled by a local minimum of $\hat{s}_n$. We assign an attribute to each cluster via the rule in Equation (2). Note that the smoothing bias could cause some biases around the boundary of clusters. However, when $h \to 0$, this bias will asymptotically be negligible.

4. Enhancements in Two-Sample Tests

Our clustering method can be used as a localized two-sample test. An overview of the idea is as follows. Given two random samples, we first merge them and use clustering method to form partitions of the sample space. Under the null hypothesis, the two samples are from the same distribution, so the proportion of each sample within each cluster should be similar. By comparing the difference in proportion, we obtain a localized two-sample test. Algorithm 2 summarizes the procedure.

In more detail, suppose we want to compare two samples $G_1 = \{X_1, X_2, \ldots, X_N\}$ and $G_2 = \{Y_1, Y_2, \ldots, Y_M\}$. Let $X_1, \ldots X_N \sim P_X$ and $Y_1, \ldots, Y_M \sim P_Y$. The null hypothesis we want to test is $H_0 : P_X = P_Y$ against $H_1 : P_X \neq P_Y$.

Under H_0, the two samples are from the same distribution, so they have the same PDF q. We first pull both samples together to form a joint dataset

$$G_{\mathsf{all}} = \{X_1, \ldots, X_N, Y_1, \ldots, Y_M\}.$$

We then compute the KDE $\hat{p}_n$ using G_{all} and compute the corresponding estimated slope function $\hat{s}_n$ and apply Algorithm 1 to form clusters. Thus, we obtain a partition of G_{all}. Under H_0, the proportion of Sample 1 in each cluster should be roughly the same as the global proportion $\frac{N}{N+M}$. Therefore, we can apply a simple test of the proportion within each cluster to obtain a p-value. In practice, we often only focus on the robust and boundary clusters and ignore the outlier clusters because of sample size consideration. Let $D_1, \ldots, D_J \subset G_{\text{all}}$ be the robust and boundary clusters, and

$$r_0 = N/(N+M); \tag{6}$$

be the global proportion, and

$$r_j = \frac{|D_j \cap G_1|}{|D_j|}. \tag{7}$$

be the observed proportion of cluster D_j. We use the test statistic

$$Z_j = \frac{r_j - r_0}{\sqrt{r_0(1-r_0)/n_j}},$$

where $n_j = |D_j|$ is the total number of the pulled sample within cluster D_j, when H_0 is true and the test statistic Z_j follows from a standard normal distribution asymptotically. Note that since we are conducting multiple tests, we reject the null hypothesis after applying the Bonferroni correction.

Algorithm 2: Local two-sample test.

```
1. Combine two samples ($G_1$ and $G_2$) into one, called $G_{\text{all}}$ and compute $r_0 = \frac{N}{N+M}$
   from Equation (6).
2. Construct a kernel density estimator using $G_{\text{all}}$ and its slope function and apply
   Algorithm 1 to form clusters based on the convergent point.
3. Assign an attribute to each cluster according to Equation (2).
4. Let robust clusters and boundary clusters be $D_1, D_2, \ldots, D_J$, where $D_j \subset G_{\text{all}}$ for
   each $j$.
5. For each cluster $D_j$, compute $r_j$ from Equation (7) and construct $Z$ statistic:

       $Z_j = \frac{r_j - r_0}{\sqrt{r_0(1-r_0)/n_j}}.$

   Find the corresponding $p$-value $p_j$.
6. Reject $H_0$ if $p_j < \alpha/J$ for some $j$ under the significance level $\alpha$.
```

We can apply this idea to other clustering algorithms. However, we need to be very careful when implementing it because we are using data twice–first to form clusters, then again to do two-sample tests. This could inflate the Type 1 error. Our approach is asymptotically valid because the clusters from the estimated slope converge to the clusters of the population slope (see Section 7). Note that our method may not control the Type 1 error in the finite sample situation, but our simulation results in Section 5.2 show that this procedure still controls the Type 1 error. This might be due to the conservative result of the Bonferroni correction.

The advantage of this new two-sample test is that we are using the local information, so if the two distributions only differ in a small region, this method will be more powerful than a conventional two-sample test. In particular, the robust clusters are often the ones with more power because they have a higher sample size, and the bumps in the pulled sample's density could be created by a density bump of one sample but not the other, leading to a region with high testing power. In Section 5, we demonstrate this through some numerical simulations.

4.1. An Approximation Method

The major computational burden of Algorithm 2 comes from Step 2, where we apply Algorithm 1 to 'every observation'. This may be computationally heavy if the sample size is large. Here we propose a quick approximation to the clustering result.

Instead of applying Algorithm 1 to every observation, we randomly subsample the original data (large dimension) or create a grid (low dimension) of points and only apply Algorithm 1 to this smaller set of points. This gives us an approximated set of local minima of the slope function. We then assign a cluster label of each observation according to the 'nearest' local minima.

5. Simulations

In this section, we demonstrate the applicability of our method by applying it to some simulation setups. Note that in practice, we need to choose the smoothing bandwidth h in the KDE. Silverman's rule [27] is one of the most popular methods for bandwidth selection. The idea is to find the optimal bandwidth by minimizing the mean integrated squared error of the estimated density. Silverman [27] proposed to use the normal density to approximate the second derivative of the true density, and use the interquartile range providing a robust estimation of the sample standard deviation. For the univariate case, it is defined as follows:

$$h_s = 1.06 \min\{\frac{\text{IQR}}{1.34}, \hat{\sigma}\} n^{-1/5},$$

where $\hat{\sigma}$ is the sample standard deviation and IQR is the interquartile range. As discussed earlier, we choose $h = C'\left(\frac{\log n}{n}\right)^{\frac{1}{d+8}}$, where C' is a constant. This choice is motivated by theoretical analysis in Section 7 (Theorem 1). In practice, we do not know C', so we applied a modification of Silverman's rule [27]:

$$h = \min\left(\frac{1}{d}\sum_{k=1}^{d}\hat{\sigma}_k, \frac{1}{d}\sum_{k=1}^{d}\frac{\text{IQR}_\text{k}}{1.34}\right) n^{-1/(8+d)}, \tag{8}$$

where $\hat{\sigma}_k$ is the standard deviation of the samples on kth dimension, IQR_k is the interquartile range on kth dimension, and $k = 1, 2, \ldots, d$. Note that our procedure involves estimating both the gradient and Hessian of the PDF. The optimal bandwidth of the two quantities are different, so one may apply two separated bandwidths for gradient and Hessian estimation. However, our empirical studies show that a single bandwidth (optimal for Hessian estimation) still leads to reliable results. Note that this bandwidth selector tends to oversmooth the data in the sense that some density peaks in Figure 6b were not detected (not in purple color).

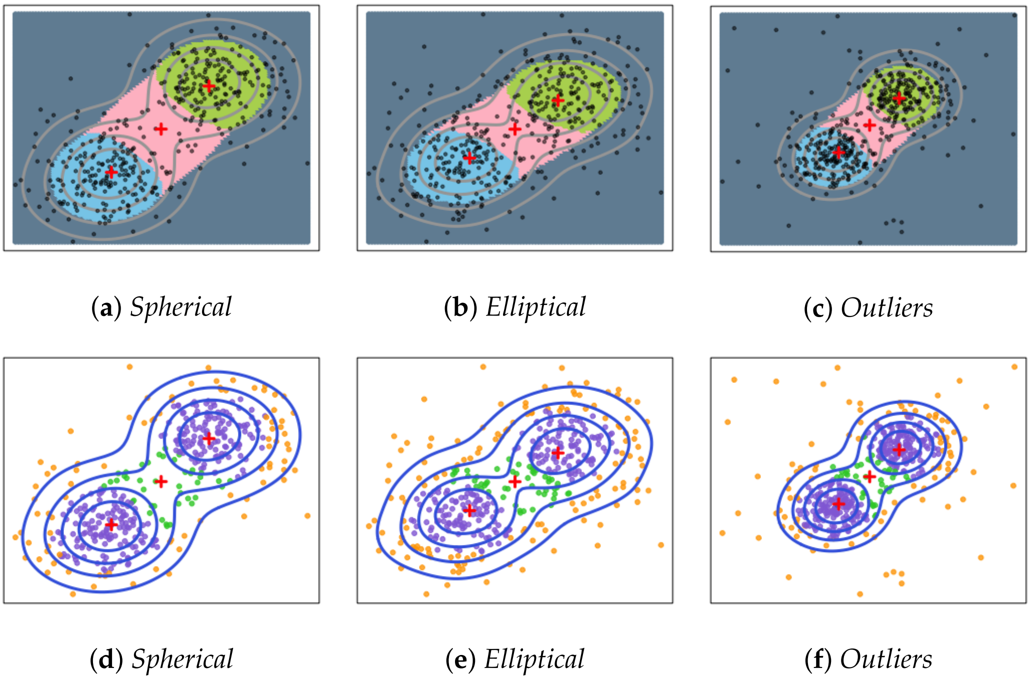
5.1. Clustering

Two-Gaussian mixture. We sample $n = 400$ points from a mixture of two-dimensional normals $N(\boldsymbol{\mu}_1, \boldsymbol{\Sigma})$ and $N(\boldsymbol{\mu}_2, \boldsymbol{\Sigma})$ with equal proportions under the following three scenarios:

- *Spherical*: $\boldsymbol{\mu}_1 = \mathbf{0}$, $\boldsymbol{\mu}_2 = 3e_1 + 3e_2$, and $\Sigma = I_2$.
- *Elliptical*: $\boldsymbol{\mu}_1 = \mathbf{0}$, $\boldsymbol{\mu}_2 = 3e_1 + 3e_2$, and $\Sigma = \text{diag}(1, 3)$. (Note that these clusters are elongated in noise directions.)
- *Outliers*: Same construction as *Spherical*, but with 60 random points (noise) from a uniform distribution over $(-5, 8) \times (-5, 8)$. By design, the outliers differ in such a way that they can only add a little ambiguity.

Note that e_i is the ith standard basis vector, and I_2 is the 2×2 identity matrix. For each scenario, we apply the gradient flow method and draw the contour. If points are outliers, their destinations go to infinity. Thus, we set a threshold to stop them from moving and assign them to outlier clusters.

Figure 2 demonstrates that we identify both two clusters and the boundary of these two clusters. Each colored region is the basin of attraction of a local minimum of $s(x)$ in the picture (a–c). Picture (d–f) provide examples of data points clustering. Given the setting of two equal-sized Gaussian mixture, it is straightforward to verify that the gradient flow algorithm can successfully distinguish points according to their destinations. The purple points represent points that belong to corresponding clusters with strong confidence, while green points represent points in low-density areas that belong to the connection regions among clusters. The yellow points represent points that are not important to any of the clusters. In summary, our proposed method performs well and is not affected by the changes of covariance and outliers.



(**a**) *Spherical* (**b**) *Elliptical* (**c**) *Outliers*

(**d**) *Spherical* (**e**) *Elliptical* (**f**) *Outliers*

Figure 2. Simulations with different data settings. Picture (**a**,**d**), picture (**b**,**e**), and picture (**c**,**f**) display, respectively, the three different simulation scenarios: *Spherical*, *Elliptical*, and *Outliers*. In picture (**a**–**c**), each colored region is the basin of attraction of a local minimum of $s(x)$, while the grey regions are the regions that belong to outlier clusters. Picture (**d**–**f**) provides an example of clustering of data points. Points that labeled purple, green, and orange are assigned to robust, boundary, and outlier clusters, respectively.

Four-Gaussian mixture. To show how boundary clusters can serve as bridges among robust clusters, we consider a four-Gaussian mixture. We sample $n = 800$ from a mixture of four two-dimensional normals $N(0, 0.1I_2)$, $N(0.5e_1, 0.1I_2)$, $N(0.5e_2, 0.1I_2)$ and $N(0.5e_1 + 0.5e_2, 0.1I_2)$ with equal proportion. Then we apply our method and display the result in Figure 3. Each colored region is the basin of attraction of a local minimum of $s(x)$. The red '+'s are the corresponding local minima to each of the basin of attraction. Clearly, we see how robust clusters are connected by the boundary clusters so the additional attributes provide useful information on the connectivity among density modes.

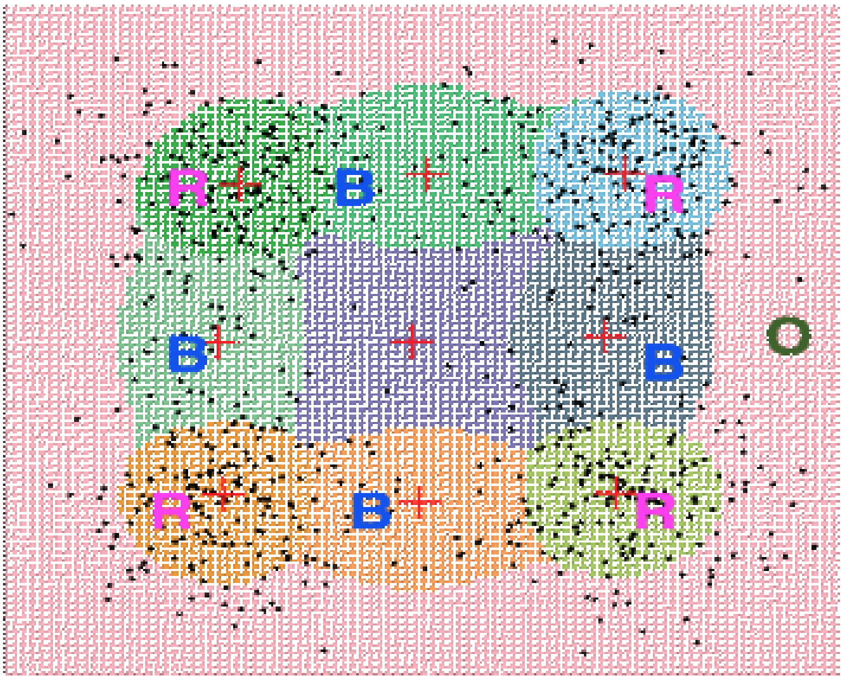
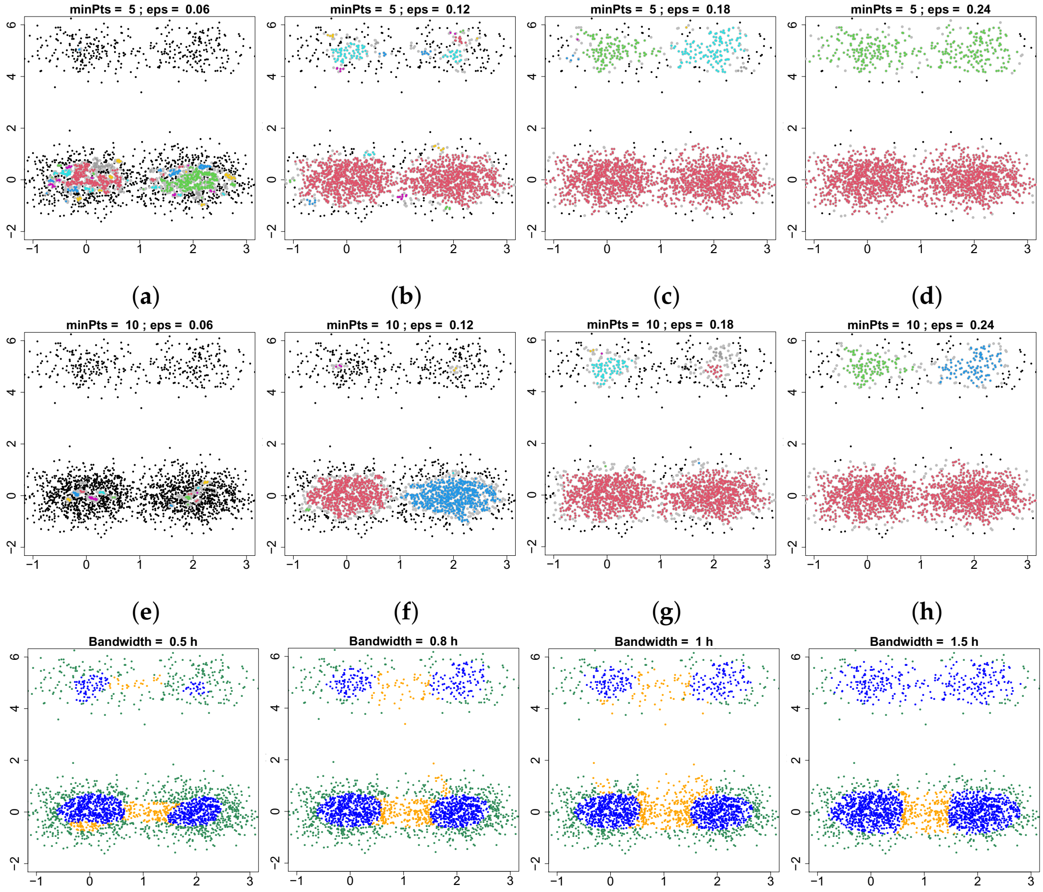


Figure 3. Example of the basins of attraction of a Gaussian mixture. Four groups of data are separated into three types of clusters. We partition the space into 10 parts. 'R' represents the region of the robust cluster, 'B' represents the region of the boundary cluster, and 'O' represents the region of the outlier cluster.

Comparison. To better illustrate the strength of our proposed method, we generate an unbalanced four-Gaussian mixture. We sample $n = 2400$ from a mixture of four two-dimensional normals $N(0, 0.5I_2)$, $N(2e_1, 0.5I_2)$, $N(5e_2, 0.5I_2)$ and $N(2e_1 + 5e_2, 0.5I_2)$ with proportion $\frac{5}{12}, \frac{5}{12}, \frac{1}{12}, \frac{1}{12}$, respectively. Then we apply our method and compare it with the density-based spatial clustering of applications with noise (DBSCAN) [28] in Figure 4. DBSCAN is a classical non-parametric, density-based clustering method that estimates the density around each data point by counting the number of points in a certain neighborhood and applies a threshold minPts to identify core, border and noise points. DBSCAN requires two parameters: the minimum number of nearby points required to form a core point (minPts) and the radius of a neighborhood with respect to a certain point (eps). Two points are connected if they are within the distance of eps. Clusters are the connected components of connected core points. Border points are points connected to a core point, but which do not have enough neighbors to be a core point. Here, we investigate the feasibility of using border points to detect the connectivity of clusters. These two parameters, minPts and eps, are very hard to choose. In the top two rows of Figure 4, we set minPts equal to 5 and 10 and change the value of eps to see if we can find the connectivity of core points using border points (gray points). Our results show that it is not possible to use border points to find the connectivity of the top two clusters and the bottom two clusters at the same time. When we are able to detect the connectivity of bottom two clusters (panel (f)), we are not able to find the top two clusters. On the other hand, when we can find the connectivity of the top two clusters (panel (c,h)), the bottom two clusters have already merged into a single cluster. The limitation of DBSCAN is that it is based on the density level set, so when the structures involve different density values, DBSCAN will not be applicable. In contrast, our method only requires one parameter, bandwidth, and it has good performance in this case. From Figure 4i–l, our method detects four robust clusters and their boundaries correctly. In addition, this result also shows that our method is robust to the bandwidth selection.



(**a**) (**b**) (**c**) (**d**)

(**e**) (**f**) (**g**) (**h**)

(**i**) (**j**) (**k**) (**l**)

Figure 4. Picture (**a**–**f**) displays the simulations using DBSCAN with different parameters settings, where minPts represents the the minimum number of points required to form a dense region and eps represents the radius of a neighborhood with respect to certain point. Picture (**i**–**l**) displays the simulations using our proposed method with different bandwidth, where h represents the bandwidth selected according to Equation (8). In Picture (**a**–**h**), each colored region is the cluster detected by DBSCAN, while the gray and black points are points that are border points and outliers, respectively. In Picture (**i**–**l**), points that are labeled blue, orange, and green are assigned to robust, boundary, and outlier clusters, respectively.

5.2. Two-Sample Test

In this section, we carry out simulation studies to evaluate the performance of the two-sample test in Section 4. We compare our method to three other popular approaches: the energy test [29], the kernel test [30], and KS [31] tests based on each of the two variables.

Our simulation is designed as follows. We draw random samples from a two-Gaussian mixture model in Equation (9):

$$p(x) = a\phi(\mu_1, \Sigma_1) + (1-a)\phi(\mu_2, \Sigma_2), \tag{9}$$

where $\phi(\cdot)$ is a cumulative distribution function of normal distribution. For the first group, we choose the parameters as $a = 0.7$, $\mu_1 = (-1, 0)$, $\mu_2 = (0, 1)$, $\Sigma_1 = \text{diag}(0.3, 0.3)$, and $\Sigma_2 = \text{diag}(0.3, 0.3)$.

In our first experiment (left panel of Figure 5), we generate the second sample from a Gaussian mixture with identical setup, except that the second covariance matrix $\Sigma_2 = \text{diag}(\sigma_2, 0.3)$, and we gradually increase σ_2 from 0.3 (H_0 is correct) to 0.8 to see how the power of the test changes. We generate $n_1 = n_2 = 500$ observations in both samples and repeat the process 500 times to compute the power of the test. This experiment investigates the power as a function of signal strength.

In the second experiment (right panel of Figure 5), we consider a similar setup except that we fix $\Sigma_2 = \text{diag}(0.35, 0.3)$ and vary the sample size from $n_1 = n_2 = 500$ to $n_1 = n_2 = 4000$ and examine how the power changes under different sample size. This experiment examines the power as a function of sample size.

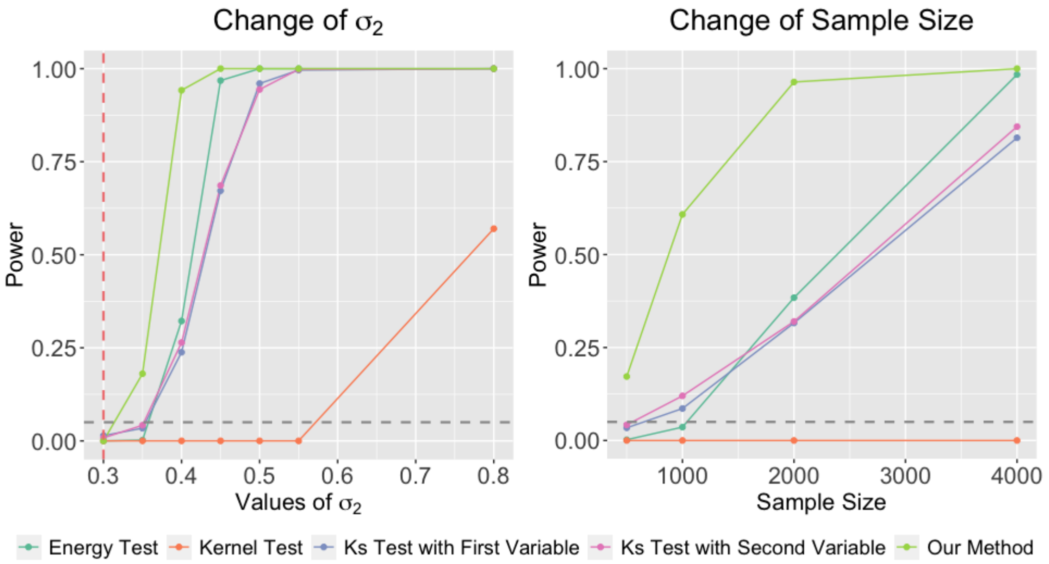


Figure 5. Power analysis of the proposed method. We compare the power of our two-sample test with three other approaches: the energy test, the kernel test, the KS test with only the first variable, and the KS test with only the second variable. In the **left** panel, we vary the variance of the second Gaussian. In the **right** panel, we fix the two distributions and increase the sample size. In both cases, our method has a higher power than the other three naive approaches.

In both experiments, all methods control the Type 1 errors. However, our method has better power in both experiments compared to the other alternatives. Our method is more powerful because we utilize the local information from clustering. In this simulation setup, the difference between the two distributions is the width of second Gaussian component. Our method is capable of capturing this local difference and using it as evidence in the hypothesis test.

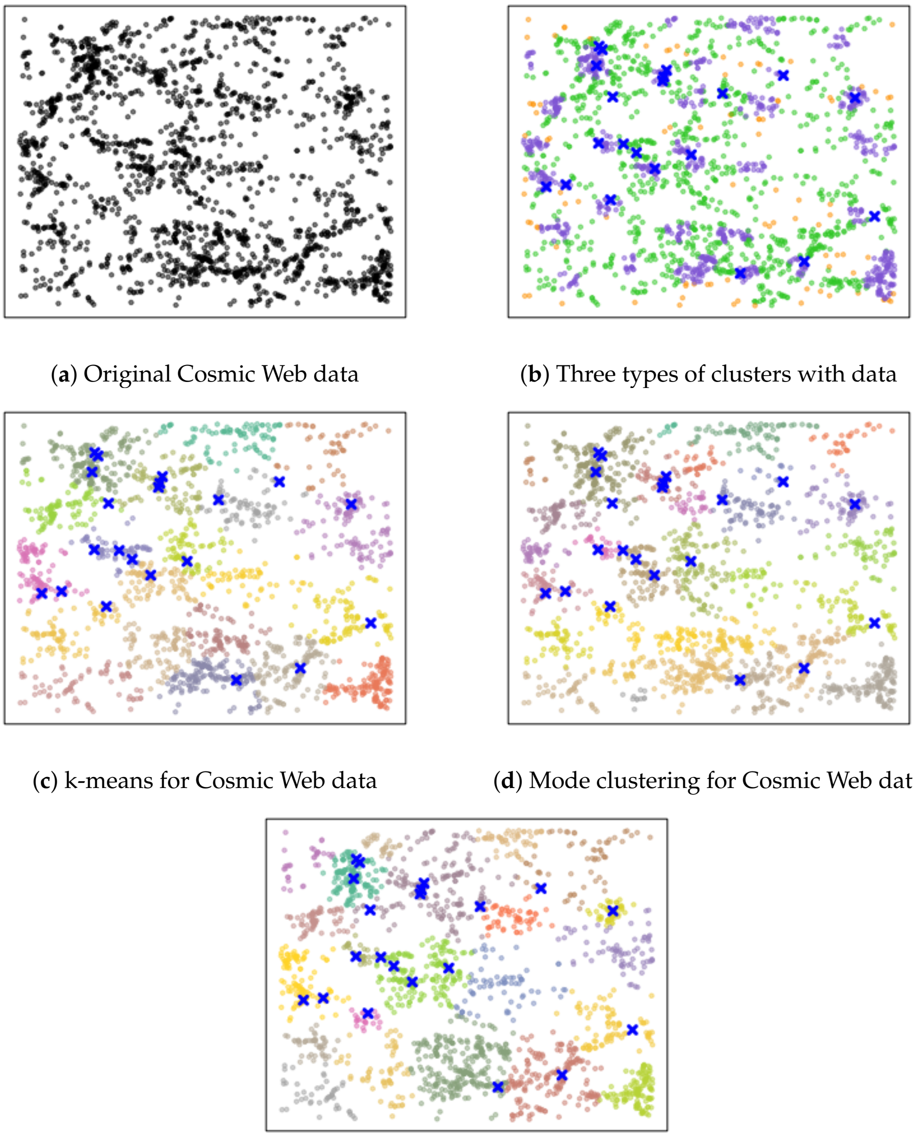
Finally, we would like to emphasize again that two-sample test after clustering has to be used with caution; we are using data twice, so we may not be able to control the Type 1 error. One needs to theoretically justify that the resulting clusters converge to a population limit and apply numerical analysis to investigate the finite-sample coverage.

6. Real Data Application

6.1. Applications to Astronomy

We apply our method to detect the Cosmic Webs [32] from the galaxy sample of the Sloan Digital Sky Survey [10]. It is known that galaxies inside our universe are not uniformly distributed. There are low-dimensional structures where matters are aggregated together. Roughly speaking, there are four types of structures in the Cosmic Webs: galaxy clusters, filaments, sheets, and voids [32]. Galaxy clusters are small regions with lots of matter. Filaments are regions with moderate matter density which connect galaxy clusters. Sheets are weakly dense regions where clusters and filaments are distributed. Voids are vast regions with very low matter density. Because of their properties, galaxy clusters are like zero-dimensional objects (points), filaments are one-dimensional curve-like structures, sheets are two-dimensional surface-like structures, and voids are three-dimensional regions.

Figure 6 displays our result. Note that it is the same data as Section 1. Panel (a) of Figure 6 shows the scatter plot of galaxies in the thin slice of the universe. In Panel (b), we color galaxies according to the types of clusters they belong to; purple, green and orange regions are the robust boundary and outlier clusters, respectively. We mark the locations of known galaxy clusters as blue “×”s [33]. These galaxy clusters are obtained using imaging analysis [34], which is a completely different approach. As can easily be seen, there is a strong agreement between galaxy clusters and the robust regions. Out of the 21 galaxy clusters, 85.71% fall into the robust clusters, and 14.29% fall into the boundary clusters. Moreover, the boundary clusters (green), connecting the robust clusters (purple), behave like the filaments in the Cosmic Webs, and the low-density outlier clusters are similar to the void structures. Figure 6 As for comparison, we display the results from k-means (Figure 6c), traditional mode-clustering (Figure 6d), and Gaussian mixture model (Figure 6e), which are not structurally correlated with the locations of blue “×”s.



(**a**) Original Cosmic Web data

(**b**) Three types of clusters with data

(**c**) k-means for Cosmic Web data

(**d**) Mode clustering for Cosmic Web data

(**e**) Gaussian mixture model for Cosmic Web data

Figure 6. We show that the gradient flow method is better in detecting the ‘Cosmic Web’ [32] in our universe. For comparison, we perform the k-means clustering method with 20 centers and traditional mode-clustering to show that our proposed method is better to detect the ‘Cosmic Web’ in our universe. The blue “×”s are the points from image analysis. The results do not structurally correlate with the locations of blue “×”s.

Thus, this analysis reveals the potential of our approach as a good method for detecting the Cosmic Webs with less information. Note that, since our dataset is two-dimensional, we cannot define the cosmic sheet structures.

6.2. Application to GvHD Data

We also apply our method to the GvHD (Graft-versus-Host Disease) data from [35]. The GvHD is a famous example for two-sample test problem. It contains a positive/disease sample and a control/normal sample. There are 9083 observations in the positive sample and 6809 observations in the control sample. Each observation consists of four biomarkers: CD4, CD8b, CD3, and CD8. Our goal is to test whether the positive sample and control sample are from the same distribution or not.

Since the sample size is non-trivial and the dimension is 4, naively applying Algorithm 2 will be computationally heavy, so we apply the approximation method in Section 4.1. We first random select 5% of the whole dataset, including both positive and control samples, as initial points in Algorithm 2. Then, the algorithm to find the local minima and add the attribute label is based on Equation (2). Finally, we assign a cluster label and attribute it to each observation according to an observation's nearest detected local minima of the slope.

Having identified clusters, we perform the two-sample test, and the result is summarized in Table 1. According to Table 1, all groups are significantly different. Thus, we can conclude that the positive sample is from a different distribution than the control sample.

Table 1. Summary of estimated proportion in each group. Note that "Proportion" in the table is referred to as the proportion of the positive group.

Cluster	Proportion	5% CI	95% CI	Z Score	Cluster Type
1	0.910	0.900	0.920	46.980	Robust Cluster
2	0.010	0.010	0.020	−69.620	Robust Cluster
3	0.680	0.650	0.720	5.550	Robust Cluster
4	0.370	0.350	0.390	−17.570	Boundary Cluster
5	0.800	0.770	0.830	11.470	Boundary Cluster
6	0.410	0.380	0.440	−9.920	Boundary Cluster
7	0.920	0.900	0.940	19.170	Robust Cluster
8	0.420	0.370	0.470	−5.930	Boundary Cluster
Overall Proportion	0.570				

The clustering result can be used to visualize the data, since the robust and boundary clusters characterize regions with non-trivial probability mass and each cluster is represented by a minimum of the slope function. The slope minimum within each cluster is the center of that cluster. Algorithm 3 provides a summary of the visualization algorithm. In more detail, we first compute the minimal distance of two different clusters to decide whether two clusters (robust or boundary) are connected. If the value is less than $4 \times \sqrt{h^2 \times d}$, two clusters are connected (neighboring to each other), where d is the number of dimensions. Then we apply multi-dimensional scaling to the centers of robust and boundary clusters to reduce the dimension to 2. Each of these points represents a particular cluster. If two clusters are connected, we add an edge to them on the graph. Finally, we add a pie chart at each cluster's center with a radius corresponding to the total number of observations in that cluster, and partition the pie chart according to the composition from the two samples. Figure 7 shows the 2D visualization of the GvHD data, along with the composition of the two samples in each cluster.

Algorithm 3: Visualization based on slope function.

```
1–4. The same steps as Algorithm 2.
5. Let robust clusters be $\{R_1, R_2, \ldots, R_{J_1}\}$ and boundary clusters be $\{B_1, B_2, \ldots, B_{J_2}\}$.
6. For each pair of $R_{j_1}$ and $B_{j_2}$, compute their Hausdorff distance (minimal distance of all pairs):
       $\mathrm{edge}_{\mathrm{j_1,j_2}} = \mathrm{Haus}(\mathrm{R_{j_1}}, \mathrm{B_{j_2}}).$
7. Apply multidimensional scaling to local minima corresponding to robust and boundary clusters. Let their 2 dimensional representation point be $s_1^*, \cdots s_{J_1+J_2}^*$.
8. For each cluster $D_j$ in $\{R_1, R_2, \ldots, R_{J_1}, B_1, B_2, \ldots, B_{J_2}\}$, plot a pie chart centered at corresponding $s_j^*$ with radius proportional to $\sqrt{|D_j|}$. The pie chart contains two groups, each with ratio $\left(\frac{|D_j \cap G_1|}{|D_j|}, \frac{|D_j \cap G_2|}{|D_j|}\right)$.
9. Label the robust clusters and boundary clusters, and add an edge between a pair of robust cluster $R_{j_1}$ and boundary cluster $B_{j_2}$ if $\mathrm{edge}_{\mathrm{j_1,j_2}} \leq 4 \times \sqrt{h^2 \times d}$, where $d$ is the number of dimensions.
```

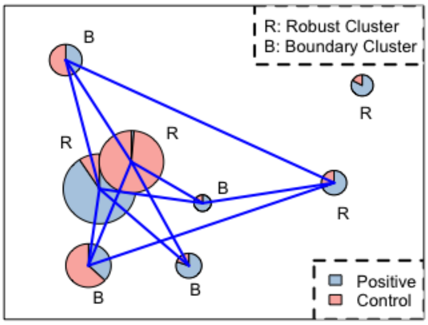


Figure 7. Visualization of GvHD dataset. We apply Algorithm 3 for visulization. Blue lines represent the connections among clusters. Each pie chart describes the total amount of corresponding clusters that is divided between the positive group and the control group.

7. Theory

In this section, we study both statistical and algorithmic convergence of our method. We start with the convergence of estimated minima $\hat{\mathcal{S}}$ to the population minima $\mathcal{S}$ along with the convergence of the gradient flow. Then we discuss the algorithmic convergence of Algorithm 1.

For a set $\mathcal{D}$, we denote its cardinality by $|\mathcal{D}|$. For a function f, we define $\|f\|_\infty = \sup_x |f(x)|$ to be the $\mathcal{L}_\infty$-norm. Let ∇f and $\nabla^2 f$ be the gradient and Hessian matrix of f, respectively. We define $\|f\|_{l,\infty}$ as the element-wise $\mathcal{L}_\infty$-norm for l-th order derivatives of f. Specifically, $\|f\|_{0,\max} = \|f\|_\infty$,

$$\|f\|_{1,\max} = \max_k \|[\nabla f(x)]_k\|_\infty \,,\ \|f\|_{2,\max} = \max_{kk'} \|[\nabla^2 f(x)]_{kk'}\|_\infty,$$

for $k = 1, 2, \ldots, d$ and $k' = 1, 2, \ldots, d$. A twice-differentiable function f is called Morse [19–21] if all eigenvalues of the Hessian matrix of f at critical points are away from 0.

Recall that our data are random sample $X_1, \ldots, X_n$ from a PDF $p(x)$ and $s(x) = \|\nabla p(x)\|_2^2$. Additionally, $\hat{p}_n$, $\nabla \hat{p}_n$ and $\nabla^2 \hat{p}_n$ are the estimated PDF, gradient, and Hessian matrix, respectively. In our analysis, we consider the following assumptions.

Assumptions.

(P) The density function $p(x)$ is four-times bounded and continuously differentiable.

(L) $s(x)$ is a Morse function.

(K) The kernel K is four-times bounded and continuously differentiable. Moreover, the collection of kernel functions and their partial derivatives up to the third order satisfy the VC-type conditions in Giné and Guillou [36]. See Appendix A for more details.

Assumption (P) is slightly stronger than the conventional assumptions for density estimation that we need to be four-times differentiable. This is because we are working with gradient of 'slope', which already involves second derivatives. To control the bias, we need additionally two derivatives, leading to a requirement on the fourth-order derivatives. Assumption (L) is slightly stronger than the conventional Morse function assumption on $p(x)$. We need the slope function to be Morse so that the gradient system is well-behaved. In fact, Assumption (L) implies that $p(x)$ is Morse function due to Lemma 1. Assumption (K) is a common assumption to ensure uniform convergence of a kernel-type estimator; see, for example [37,38].

7.1. Estimation Consistency

With the above assumption, we can show that the local minima of $\hat{s}_n$ converge to the local minima of s.

Theorem 1 (Consistency of local minima of s)**.** *Assume (K), (P) and (L). Let c_1 be the bound for the partial derivatives of s up to the third order and denote the l-th largest eigenvalues of $\nabla^2 s(x)$ by $\lambda_{(s,l)}(x)$ ($l = 1, 2, \ldots, d$, where d is the dimension). Assume:*

(A1) There exists $\eta_1 > 0$ such that for any point x with $\|\nabla s(x)\| \le \eta_1$ and $0 > -\lambda_0'/2 \ge \lambda_{(s,d)}(x)$, we have $\min_{m\in\mathcal{S}} \|m - x\| \le \frac{\lambda_0'}{2dc_1}$, where $0 < \lambda_0' \le |\lambda_{(s,l)}(m)|$ for $l = 1, 2, \ldots, d$ and $m \in \mathcal{S}$.

When $\|\hat{p}_n - p\|_{4,\max}$ is sufficiently small, we have

- *$|\mathcal{S}| = |\hat{\mathcal{S}}|$, and*
- *for every point $m \in \mathcal{S}$, there exists a unique element $\hat{m} \in \hat{\mathcal{S}}$ such that*

$$\|\hat{m} - m\| = O(h^2) + O_P\left(\sqrt{\frac{1}{nh^{d+4}}}\right)$$

Theorem 1 shows two results. First, asymptotically, there will be a one–one corresponding relationship between a population's local minimum and an estimated local minimum. The second result shows the rate of convergence, which is the rate of estimating second derivatives. This is reasonable, since the local minima of s is defined through the gradient of $s(x) = \|\nabla p(x)\|^2$, which requires second derivatives of p.

Note that the fourth-order derivative assumption (P) can be relaxed to a smoothed third-order derivative conditions. We use this stronger condition to simplify the derivation, since the global minima of s are the critical points of p, the consistency of estimating a global minimum only requires a third-order derivative (or a smooth second-order derivative) assumption; see, for example [39,40].

Theorem 1 also implies the rate of the set estimator $\hat{\mathcal{S}}$ in terms of the Hausdorff distance. For given two sets A, B, their Hausdorff distance is

$$\text{Haus}(A, B) = \max\left\{\sup_{x\in A} d(x, B), \sup_{x\in B} d(x, A)\right\},$$

where $d(x, A) = \inf_{y\in A} \|x - y\|$ is the projection distance from point x to the set A.

Corollary 1. *Assume (K),(P), (L), and (A1). When $\|\hat{p}_n - p\|_{4,\max}$ is sufficiently small,*

$$Haus(\hat{\mathcal{S}}, \mathcal{S}) = O(h^2) + O_P\left(\sqrt{\frac{1}{nh^{d+4}}}\right).$$

The above results describe the statistical consistency of the convergent points (local minima) of a gradient flow system. In what follows, we show that the gradient flows will also converge under the same set of assumptions.

Theorem 2 (Consistency of gradient flows)**.** *Assume (K), (P) and (L). Then for a fixed point x, when $\frac{nh^{d+8}}{\log n} \to \infty$, $h \to 0$,*

$$\sup_{t \geq 0} \|\hat{\pi}_x(t) - \pi_x(t)\| = \left\{ O(h^{2\alpha}) + O_P\left(\left(\frac{\log n}{nh^{d+4}} \right)^{\frac{\alpha}{2}} \right) \right\} \wedge \left\{ O(h) + O_P\left(\sqrt[4]{\frac{\log n}{nh^d}} \right) \right\},$$

where $\mu_{\min}(x)$ and $\mu_{\max}(x)$ are the minimal and maximal eigenvalues of the Hessian matrix of s evaluated at the destination $\pi_x(\infty)$, and $\alpha = \mu_{\min}(x)/(\mu_{\min}(x) + \mu_{\max}(x))$.

Theorem 2 is mainly inspired by Theorem 2 in Arias-Castro et al. [3]. It shows that starting at a given point x, the estimated gradient flow $\hat{\pi}_x(t)$ is a consistent estimator to the population gradient flow $\pi_x(t)$. One may notice that this result shows that the convergence rate is slowed down by the factor α, which comes from the curvature of s around the local minimum. This is due to the fact that when a flow is close to its convergent point (a local minimum), the speed of flow is decreasing until 0 (when it arrives at a minimum), so the eigenvalues determine the rate of how fast the speed of a flow decreases along a particular direction. When the eigengap (difference between $\mu_{\min}(x)$ and $\mu_{\max}(x)$) is large, even a small perturbation could change the orientation of the flow drastically, leading to a slower convergence rate.

Remark. It is possible to obtain the clustering consistency in the sense that the clustering based on s and $\hat{s}_n$ are asymptotically the same [41]. In [41], the authors placed conditions on the density function and showed that the mode-clustering of $\hat{p}$ leads to a consistent partition of the data compared to the mode-clustering of p. If we generalize their conditions to the slope s, we will obtain a similar clustering consistency result.

7.2. Algorithmic Consistency

In this section, we study the algorithmic convergence of Algorithm 1. For simplicity, we consider the case where the gradient descent algorithm is applied to s. The convergence analysis of gradient descent has been well studied in the literature [42,43] under convex/concave setups. Our algorithm is a gradient descent algorithm but is applied to a non-convex scenario. Fortunately, if we consider a small ball around each local minimum, the function s will still be a convex function, so the conventional techniques apply.

Specifically, we need an additional assumption that is slightly stronger than (L).

(A2) There are positive numbers $R_0, \eta_1, \lambda_0 > 0$ such that for all $x \in B(m, R_0)$, where $m \in \mathcal{S}$, and $B(m, R_0)$ is a ball with center m and radius R_0, all eigenvalues of Hessian matrix $\nabla^2 s(x)$ are above λ_0 and $\|\nabla s(x)\| \leq \eta_1$.

The assumption (A2) is a local strongly convex condition.

Theorem 3 (Convergence of Algorithm 1)**.** *Assume conditions (P), (K), (A1) and (A2). Let the step size in Algorithm 1 be γ. Recall that x_t is the point at iteration time t and x_0 is the initial point. Assume that the step size $\gamma < 1/L$, where $L = \sup_x \|\nabla s(x)\|$. For any initial point x_0 within the ball $B(m, R_0)$, there exists a constant $C_0 < 1$ such that:*

$$\|x_t - m\| \leq (1 - \gamma L)^t \|x_0 - m\|,$$
$$\|s(x_t) - s(m)\| \leq C_0^t \|s(x_0) - s(m)\|.$$

Note that λ_0 is the constant in assumption (A2) and satisfies $\lambda_0 \leq L$; see the proof of this theorem.

Theorem 3 shows that when the initial point is sufficiently close to a local minimum, the algorithm converges linearly [42,43] to the local minimum. Additionally, this implies that the ball $B(m, R_0)$ is always in the basin of attraction of m. However, note that the actual basin could be much larger than $B(m, R_0)$.

8. Conclusions

In this paper, we introduced a novel clustering approach based on the gradient of the slope function. The resulting clusters are associated with an attribute label, which provides additional information on each cluster. With this new clustering method, we propose a two-sample test using local information within each cluster, which improves the testing power. Finally, we developed an informative visualization tool that gives the structure of multi-dimensional data.

We studied our improved method's performance empirically and theoretically. Simulation studies show that our refined clustering method is capable of capturing fine structures within the data. Furthermore, as a two-sample test procedure, our clustering method has better power than conventional approaches. The analysis on Astronomy and GvHD data shows that our method finds meaningful clusters. Finally, we studied both statistical and computational theory of our proposed method. Our proposed method demonstrated good empirical performance and statistical and numerical properties. Finally, we would like to note that while our method works well for the GvHD data (d = 4), it may not be applicable for any higher dimensional data, since our method is a nonparametric procedure involving derivative estimation. The curse of dimensionality prevents us from applying it to data with more dimensions.

Author Contributions: Conceptualization, Y.-C.C.; methodology, K.Z.; software and validation, K.Z.; original draft preparation, K.Z.; review and editing, Y.-C.C. and K.Z.; visualization, K.Z.; supervision, Y.-C.C. All authors have read and agreed to the published version of the manuscript.

Funding: Chen is supported by National Science Foundation DMS-1810960 and DMS-1952781 and National Institutes of Health U01-AG0169761.

Institutional Review Board Statement: Ethical review and approval were waived for this study. There is no human-related subject in this project so no need for IRB review.

Informed Consent Statement: Not applicable.

Data Availability Statement: Codes are available in https://github.com/khzhang/Clustering.git, accessed on 5 May 2021.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Proofs

To explicitly describe the kernel assumption (K), we need to define a few notations first. A vector $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_d)$ of non-negative integers is called a multi-index with $|\alpha| = \alpha_1 + \alpha_2 + \cdots + \alpha_d$ and the corresponding derivative operator is

$$D^\alpha = \frac{\partial^{\alpha_1}}{\partial x_1^{\alpha_1}} \cdots \frac{\partial^{\alpha_d}}{\partial x_d^{\alpha_d}},$$

where $D^\alpha f$ is often written as $f^{(\alpha)}$. The assumption (K) requires the followings. Let

$$\mathcal{K} = \left\{ y \mapsto K^{(\alpha)}\left(\frac{x-y}{h}\right) : x \in \mathbb{R}^d, |\alpha| = l \right\},$$

where $K^{(\alpha)}$ is the partial derivative along $\alpha = (\alpha_1, \cdots, \alpha_d)$ direction and let $\mathcal{K}_r^* = \cup_{l=0}^r \mathcal{K}_l$. $\mathcal{K}_r^*$ is the partial derivatives of the kernel function up to fourth-order. We assume that $\mathcal{K}_4^*$ is a VC-type class. that is, there exists constants A, v, and constant envelope b_0 such that

$$\sup_Q N\left(\mathcal{K}_4^*, \mathcal{L}^2(Q), b_0\epsilon\right) \leq \left(\frac{A}{\epsilon}\right)^v,$$

where $N(T', d_T, \epsilon)$ is the ϵ-covering number for a semi-metric set T' with metric d_T and $\mathcal{L}^2(Q)$ is the $\mathcal{L}_2$ norm with respect to the probability measure Q. While this condition looks complicated, the Gaussian kernel and any smooth compactly supported kernel satisfy this condition; see [36].

For simplicity, we describe some notations which will be used across all proofs. We denote $g_s(x) = \nabla s(x)$ be the gradient of $s(x)$ and $H_s(x) = \nabla^2 s(x)$ be the Hessian matrix. Denote $\hat{g}_s(x) = \nabla \hat{s}_n(x)$ and $\hat{H}_s(x) = \nabla^2 \hat{s}_n(x)$, where $\hat{s}_n$ is the estimator of function s. Let $g(x) = \nabla p(x)$ be the gradient of $p(x)$ and $H(x) = \nabla^2 p(x)$ be the Hessian matrix. Denote $\hat{g}_n(x) = \nabla \hat{p}_n(x)$ and $\hat{H}_n(x) = \nabla^2 \hat{p}_n(x)$, where $\hat{p}_n$ is the estimator of function p. For a smooth function f, recall that we define $\|f\|_{l,\infty}$ be the $\mathcal{L}_\infty$-norm of l-th order derivative. For instance,

$$\|f\|_{0,\infty} = \sup_x \|f(x)\|, \; \|f\|_{1,\infty} = \sup_x \|\nabla f(x)\|_{\max}, \; \|f\|_{2,\infty} = \sup_x \|\nabla^2 f(x)\|_{\max}.$$

Proof of Lemma 1: Recall that $s(x) = \|g(x)\|^2$ and $\nabla s(x) = H(x)g(x)$. Thus, $\mathcal{C} \subset \mathcal{S}$, where $\mathcal{S}$ is the collection of critical points of $s(x)$. In addition, the Hessian matrix of $s(x)$ is

$$\nabla^2 s(x) = T(x),$$

where $T_{kk'}(x) = \left[H^2(x)\right]_{kk'} + \sum_{l=1}^d \frac{\partial H(x)}{\partial x_l} g_l(x)$ and $g_l(x)$ is the l-th component of $g(x)$.

For any $m \in \mathcal{C}$, since $\mathcal{C}$ is the collection of critical points of the density p, we have $g(m) = 0$ and the Hessian of slope function $T(m) = H^2(m)$, since we assume s is a Morse function, the eigenvalues of $T(m)$ is non-zero, which implies the eigenvalues of $H(m)$ is non-zero, thus completes the proof. □

Proof of Theorem 1: We will prove the convergence rate and the one-one correspondence. The first assertion (estimated number of local minima equals the population number of local minima) follows from the one-one correspondence.

Our proof consists of two steps. First, we show that there is a one to one mapping between an estimated local minimum and the corresponding true local minimum. Then we can obtain the rate for the distance by using derivative estimation under assumption (K).

The one to one mapping assertion for local minima can be satisfied by modifying the result of Theorem 1 in [4]. Recall that m is a local minimum of s, let $\hat{m}_n$ be a local minimum of $\hat{s}_n$. From the first two steps of the proof of Theorem 1 in [4], we can get:

$$\min_{m \in \mathcal{S}} \|\hat{m}_n - m\| \leq \frac{\lambda_0'}{2dc_1}$$

when $\|\hat{p}_n - p\|_{4,\max}$ is sufficiently small. Such a local minimum $\hat{m}_n$ of $\hat{s}_n$ is unique, which means there cannot be another critical point for that given local minimum of s. In other words, each m only corresponds to one $\hat{m}_n$ and vice versa. This completes the proof of one to one mapping assertion for local minima.

To derive the rate for the distance $\|\hat{m}_n - m\|$, note that $\hat{g}_s(\hat{m}_n) = g_s(m) = 0$. By Taylor's theorem,

$$\hat{g}_s(m) - g_s(m) = \hat{g}_s(m) - \hat{g}_s(\hat{m}_n) = \hat{H}_s(m)(m - \hat{m}_n) + O(\|\hat{m}_n - m\|^2).$$

After rearrangement, we obtain:

$$\hat{m}_n - m = -\hat{H}_s^{-1}(m)(\hat{g}_s(m) - g_s(m)) + O(\|\hat{m}_n - m\|^2) = -\hat{H}_s^{-1}(m)\hat{g}_s(m) + R_n,$$

where $R_n = O\big(\|H_s^{-1}(m) - \hat{H}_s^{-1}(m)\| \cdot \|\hat{g}_s(m)\| + \|\hat{m}_n - m\|^2\big)$, which is a second order term, since $H_s(m)$ is a positive definite matrix due to Lemma 1 and assumption (L), the rate of $\hat{m}_n - m$ is determined by the rate of $\hat{g}_s(m)$. By the definition of $\hat{s}$, $\hat{g}_s(x) = \nabla \hat{s}(x) = \hat{H}_n(x)\hat{g}_n(x)$. $\hat{g}_n(x) = \nabla \hat{p}_n(x)$ and $\hat{H}_n(x) = \nabla^2 \hat{p}_n(x)$ are the gradient and Hessian matrix of kernel density estimator $\hat{p}(x)$, and $g(x) = \nabla p(x)$ and $H(x) = \nabla^2 p(x)$ are the gradient and Hessian matrix of true density function $p(x)$. Thus,

$$\begin{aligned}
\hat{g}_s(m) = \hat{g}_s(m) - g_s(m) &= \hat{H}_n(m)\hat{g}_n(m) - H_n(m)g_n(m) \\
&= \hat{H}_n(m)\hat{g}_n(m) - H_n(m)\hat{g}_n(m) + H_n(m)\hat{g}_n(m) - H_n(m)g_n(m) \\
&= \big(\hat{H}_n(m) - H_n(m)\big)\hat{g}_n(m) + H_n(m)(\hat{g}_n(m) - g_n(m)) \\
&= \big(\hat{H}_n(m) - H_n(m)\big)(\hat{g}_n(m) - g_n(m)) + H_n(m)(\hat{g}_n(m) - g_n(m))
\end{aligned}$$

Let $[\beta] = (\beta_1, \beta_2, \dots, \beta_d)$ be a multi-index (each $\beta_l \in [\beta]$ is a non-negative integer and $|[\beta]| = \sum_{l=1}^d \beta_l$). Define $D^{[\beta]} = \frac{\nabla^{\beta_1}}{\nabla x_1^{\beta_1}} \cdots \frac{\nabla^{\beta_d}}{\nabla x_d^{\beta_d}}$ to be the $[\beta]$-th order partial derivative operator [14].

Under smoothness condition [24],

$$D^{[\beta]}\hat{p}_n(x) - D^{[\beta]}p_n(x) = O(h^2) + O_P\left(\sqrt{\frac{1}{nh^{d+2|[\beta]|}}}\right).$$

Thus, under assumption (K), for a fixed point x,

$$\hat{H}_n(x) - H(x) = O(h^2) + O_P\left(\sqrt{\frac{1}{nh^{d+4}}}\right)$$

$$\hat{g}_n(x) - g(x) = O(h^2) + O_P\left(\sqrt{\frac{1}{nh^{d+2}}}\right)$$

So $\hat{g}_s(m) = O(h^2) + O_P\left(\sqrt{\frac{1}{nh^{d+4}}}\right)$, which leads to

$$\hat{m}_n - m = O(h^2) + O_P\left(\sqrt{\frac{1}{nh^{d+4}}}\right).$$

□

Before we discuss the proof of Theorem 2, we first recall a useful result:

Theorem A1 (Rate of convergence of KDE; page 17 of [38])**.** *Assume (P) and (K). Let $\hat{p}_n(x)$ be the kernel density estimator. For each $l = 0, 1, 2, 3, 4$, when $h \to 0$ and $\frac{nh^{d+2l}}{\log n} \to \infty$,*

$$\|\hat{p}_n - p\|_{l,\infty} = O(h^2) + O_P\left(\sqrt{\frac{\log n}{nh^{d+2l}}}\right)$$

Theorem A2 ((Modified) Theorem 2 in Arias-Castro et al. [3])**.** *Suppose f and $\tilde{f}$ are two smooth functions that are three times differentiable. Given a point x_0, let $(x(t)\colon t > 0)$ be the gradient flow of f starting from x_0, and $(\tilde{x}(t)\colon t > 0)$ be the gradient flow of $\tilde{f}$ starting from the same point x_0. Assume that $x(t)$ ends at the local mode x^* and the eigenvalues of $\nabla^2 f(x^*)$ are in* *the interval $[v_1, v_2]$ where $\infty > v_2 \geq v_1 > 0$. Then there exists a constant C depends only on f, x_0, v_1, v_2 such that when max $\{\|f - \tilde{f}\|_{l,\infty} : l = 0, 1, 2, 3\} <$ max $\{C, C^{-1}\}$,*

$$\sup_{t\geq 0} \|\tilde{x}(t) - x(t)\| \leq C max\left\{\sqrt{\|f - \tilde{f}\|_{0,\infty}}, \|f - \tilde{f}\|_{1,\infty}^{\alpha_0}\right\},$$

where $\alpha_0 = \frac{v_1}{v_1+v_2}$.

Proof of Theorem 2: The main idea for this proof is to reverse the direction of the gradient flows described in Theorem 2 in Arias-Castro et al. [3], which establish a stability result for gradient flows of smooth functions f. To apply Theorem A2, the corresponded smooth function $f(x)$ is $s(x)$, and $s(x) = \|\nabla p(x)\|^2$ in our case. Thus, assumption (P) guarantees that $s(x)$ is three times differentiable, since in Theorem A2, it requires $\max\{\|f - \tilde{f}\|_{l,\infty} : l = 0, 1, 2, 3\} < \max\{C, C^{-1}\}$, which means $\max\{\|s(x) - \tilde{s}(x)\|_{l,\infty} : l = 0, 1, 2, 3\}$ is sufficient small. That is $\max\{\|p(x) - \tilde{p}(x)\|_{l,\infty} : l = 0, 1, 2, 3, 4\}$ should be small. By Theorem A1, we can get $\frac{\log n}{nh^{d+8}} \to 0$ if $h \to 0$, which guarantees our assumptions.

Recall that $\mu_{\min}(x)$ and $\mu_{\max}(x)$ are the smallest and largest eigenvalue of $H_s(\pi_x(\infty))$. Thus, all eigenvalues of $H_s(\pi_x(\infty))$ fall into $[\mu_{\min}(x), \mu_{\max}(x)]$, which means $\mu_{\min}(x)$ and $\mu_{\max}(x)$ are corresponding v_1 and v_2 in Theorem A2. Then, we can obtain

$$\sup_{t\geq 0} \|\hat{\pi}_x(t) - \pi_x(t)\| = \left\{O(h^{2\alpha}) + O_P\left(\left(\frac{\log n}{nh^{d+4}}\right)^{\frac{\alpha}{2}}\right)\right\} \wedge \left\{O(h) + O_P\left(\sqrt[4]{\frac{\log n}{nh^d}}\right)\right\},$$

where $\alpha = \frac{\mu_{\min}(x)}{\mu_{\max}(x)+\mu_{\min}(x)}$. □

Finally, the proof of Lemma A1 relies on some useful properties from convex optimization. We first recall a useful lemma.

Lemma A1. *According to Chapter 2 in [42], we have several properties below.*

- *Property 1: When a function $f(x)$ has an L-Lipschitz continuous gradient, then*

$$f(x) - f(y) \leq \langle x - y, \nabla f(y)\rangle + \frac{L}{2}\|x - y\|^2 \quad \textit{for every } x, y \in \mathbb{R}^n. \tag{A1}$$

  In addition, constant L is greater than or equal to the maximum eigenvalue of Hessian matrix of $f(x)$.

- *Property 2:*
  Let $f^* = f(x^*) = \min_x f(x)$, where x^* is the true minimum of the function $f(x)$. The function $f(x)$ is called C_m strongly convex if and only if there exists a constant $C_m > 0$ such that the $f(x) - \frac{C_m}{2}\|x\|^2$ is a convex function. In addition, for each step t, we have:

$$f^* - f(x_t) \geq (x^* - x_t)^T \nabla f(x_t) + \frac{C_m}{2}\|x^* - x_t\|^2, \tag{A2}$$

  which implies

$$(x_t - x^*)^T \nabla f(x_t) \geq f(x_t) - f^* + \frac{C_m}{2}\|x^* - x_t\|^2. \tag{A3}$$

- *Property 3: Let $f^* = f(x^*) = 0$, where x^* is the true minimum of the function $f(x)$. Assume function $f(x)$ has an L-Lipschitz continuous gradient. Then, we have:*

$$f(x) \geq \frac{1}{2L}\|\nabla f(x)\|^2 + f^*. \tag{A4}$$

- *Property 4: By the settings in Property 2 and Property 3, we have:*

$$\|\nabla f(x)\|^2 \geq C_m^2\|x - x^*\|^2 \geq \frac{2(f(x) - f^*)C_m^2}{L} \geq 2f(x)C_m^2/L. \tag{A5}$$

Proof of Lemma A1: Property 1 can be directly obtained by the definition of L-Lipschitz continuity. For property 2, $f(x)$ is strongly convex, so $\|\nabla f(x)\| \geq C_m\|x - x^*\|$, where C_m is smaller than or equal to the minimum eigenvalue of Hessian matrix of $f(x)$. For property 3, $f(x)$ is L-Lipschitz, so $f(x) \leq \frac{L}{2}\|x - x^*\|^2 + f^*$. According to the fact that $f(x) \geq f^* = 0$, then,

$$\begin{aligned} -f(x_t) + f^* &\leq f(x_{t+1}) - f(x_t) \\ &= f(x_t - \gamma\nabla f(x_t)) - f(x_t) \\ &\leq f(x_t - \frac{1}{L}\nabla f(x_t)) - f(x_t) \\ &\leq -\frac{1}{L}\|\nabla f(x_t)\|^2 + \frac{1}{2L}\|\nabla f(x_t)\|^2 \\ &= -\frac{1}{2L}\|\nabla f(x_t)\|^2. \end{aligned} \tag{A6}$$

Thus, the results are as desired. The C_m-strongly convexity implies $\|\nabla f(x)\| \geq C_m\|x - x^*\|$ and the L-Lipscthitz gradient implies $f(x) - f^* \leq \frac{L}{2}\|x - x^*\|$. Thus, the Property 4 holds. □

Proof of Theorem 3: From assumptions (A1) and (A2), there exists a ball with certain radius R_0 around each minimum of s such that all points within that ball have all positive eigenvalues of the Hessian matrix. Let a starting point within a ball to be x_0. Note that within each ball, $s(x)$ is λ_0-strongly convex, since the Hessian matrix has all of its eigenvalues bounded [42]. The constant λ_0 is from assumption (A2).

According to assumption (P) and (L), s is a continuously differentiable function with Lipschitz continuous gradient and Lipschitz constant L. Consider a minimum $m_j \in \mathcal{S}$ and let $s^* = s(m_j) = 0$. According to Property 3 and Property 4 in Lemma A1, we have:

$$s(x_t) \geq \frac{1}{2L}\|\nabla s(x_t)\|^2 \geq \frac{1}{2L}2s(x_t)\lambda_0^2/L. \tag{A7}$$

After rearrangement, we obtain:

$$1 \geq \frac{\lambda_0^2}{L^2}. \tag{A8}$$

For step $t+1$,

$$\begin{aligned} \|x_{t+1} - m_j\|^2 &= \|x_t - m_j - \gamma\nabla s(x_t)\| \\ &= \|x_t - m_j\|^2 - 2\gamma(x_t - m_j)^T\nabla s(x_t) + \gamma^2\|\nabla s(x_t)\|^2 \\ &\leq \|x_t - m_j\|^2 - 2\gamma(s(x_t) - s^* + \frac{\lambda_0}{2}\|m_j - x_t\|^2) + \gamma^2\|\nabla s(x_t)\|^2 \\ &\leq \|x_t - m_j\|^2(1 - \gamma\lambda_0) - 2\gamma s(x_t) + \gamma^2\|\nabla s(x_t)\|^2 \\ &\leq \|x_t - m_j\|^2(1 - \gamma\lambda_0) - 2\gamma s(x_t) + \gamma^2 * 2Ls(x_t) \\ &\leq \|x_t - m_j\|^2(1 - \gamma\lambda_0) \\ &\leq \|x_0 - m_j\|^2(1 - \gamma\lambda_0)^{t+1} \end{aligned} \tag{A9}$$

The first and third inequalities are due to Equations (A3) and (A4). By Equation (A8), $0 < \gamma\lambda_0 \leq \frac{\lambda_0}{L} \leq 1$. This proves the first statement.

Applying L-Lipschitz again and according to the Property 4 from Lemma A1, we have:

$$\begin{aligned} s(x_{t+1}) - s(x_t) &= s(x_t - \gamma \nabla s(x_t)) - s(x_t) \\ &\leq -\gamma \|\nabla s(x_t)\|^2 + \frac{L\gamma^2}{2} \|\nabla s(x_t)\|^2 \\ &= -\gamma(1 - \frac{L\gamma}{2}) \|\nabla s(x_t)\|^2 \\ &\leq -\gamma(1 - \frac{L\gamma}{2}) \frac{2(s(x_t) - s^*)\lambda_0^2}{L}. \end{aligned} \tag{A10}$$

By rearrangements,

$$\begin{aligned} s(x_{t+1}) &\leq s(x_t)\left(1 - 2\gamma\left(1 - \frac{L\gamma}{2}\right)\frac{\lambda_0^2}{L}\right) \\ &= s(x_t)\left(1 - \frac{\lambda_0^2}{L^2} + \lambda_0^2\left(\gamma - \frac{1}{L}\right)^2\right). \end{aligned} \tag{A11}$$

Recall that x_0 is the initial point. By telescoping, we can get:

$$\begin{aligned} s(x_{t+1}) - s(m) &= s(x_{t+1}) - s^* \\ &\leq s(x_0)\left(1 - \frac{\lambda_0^2}{L^2} + \lambda_0^2\left(\gamma - \frac{1}{L}\right)^2\right)^{t+1} \\ &= (s(x_0) - s(m))\left(1 - \frac{\lambda_0^2}{L^2} + \lambda_0^2\left(\gamma - \frac{1}{L}\right)^2\right)^{t+1}. \end{aligned}$$

since $0 < \gamma \leq 1/L$, $-\frac{\lambda_0^2}{L^2} + \lambda_0^2\left(\gamma - \frac{1}{L}\right)^2$ lies in range $(0, \frac{\lambda_0^2}{L^2}]$. By Equation (A8), $\frac{\lambda_0^2}{L^2} \leq 1$, $1 - \frac{\lambda_0^2}{L^2} + \lambda_0^2\left(\gamma - \frac{1}{L}\right)^2 < 1$. This completes the proof. □

References

1. Li, J.; Ray, S.; Lindsay, B.G. A Nonparametric Statistical Approach to Clustering via Mode Identification. *J. Mach. Learn. Res.* **2007**, *8*, 1687–1723.
2. Chacón, J.E. Clusters and water flows: A novel approach to modal clustering through Morse theory. *arXiv* **2012**, arXiv:1212.1384.
3. Arias-Castro, E.; Mason, D.; Pelletier, B. On the Estimation of the Gradient Lines of a Density and the Consistency of the Mean-Shift Algorithm. *J. Mach. Learn. Res.* **2016**, *17*, 1–28.
4. Chen, Y.C.; Genovese, C.R.; Wasserman, L. A comprehensive approach to mode-clustering. *Electron. J. Stat.* **2016**, *10*, 210–241. [CrossRef]
5. Fukunaga, K.; Hostetler, L. The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Trans. Inf. Theory* **1975**, *21*, 32–40. [CrossRef]
6. Cheng, Y. Mean shift, mode seeking, and clustering. *IEEE Trans. Pattern Anal. Mach. Intell.* **1995**, *17*, 790–799. [CrossRef]
7. Carreira-Perpiñán, M.Á. A review of mean-shift algorithms for clustering. *arXiv* **2015**, arXiv:1503.00687.
8. Hastie, T.; Tibshirani, R.; Friedman, J. *The Elements of Statistical Learning*; Springer Series in Statistics; Springer New York Inc.: New York, NY, USA, 2001.
9. Hennig, C.; Meila, M.; Murtagh, F.; Rocci, R. *Handbook of Cluster Analysis*; CRC Press: Boca Raton, FL, USA, 2015.
10. York, D.G.; Adelman, J., Jr.; Anderson, J.J.E.; Bahcall, N.A.; Yasuda, N. The Sloan Digital Sky Survey: Technical Summary. *Astron. J.* **2000**, *120*, 1579–1587. [CrossRef]
11. Comaniciu, D.; Meer, P. Mean shift analysis and applications. In Proceedings of the Seventh IEEE International Conference on Computer Vision, Corfu, Greece, 20–27 September 1999; Volume 2, pp. 1197–1203.
12. Chacón, J.E.; Duong, T. Data-driven density derivative estimation, with applications to nonparametric clustering and bump hunting. *Electron. J. Stat.* **2013**, *7*, 499–532. [CrossRef]
13. Chacón, J.E. A population background for nonparametric density-based clustering. *Stat. Sci.* **2015**, *30*, 518–532. [CrossRef]
14. Chen, Y.C. A tutorial on kernel density estimation and recent advances. *Biostat. Epidemiol.* **2017**, *1*, 161–187. [CrossRef]

15. Scrucca, L. Identifying Connected Components in Gaussian Finite Mixture Models for Clustering. *Comput. Stat. Data Anal.* **2016**, *93*, 5–17. [CrossRef]
16. Bonis, T.; Oudot, S. A fuzzy clustering algorithm for the mode-seeking framework. *Pattern Recognit. Lett.* **2018**, *102*, 37–43. [CrossRef]
17. Jiang, H.; Kpotufe, S. Modal-set estimation with an application to clustering. In *Artificial Intelligence and Statistics*; PMLR: Fort Lauderdale, FL, USA, 2017; pp. 1197–1206.
18. Menardi, G. A Review on Modal Clustering. *Int. Stat. Rev.* **2015**, *84*. [CrossRef]
19. Morse, M. Relations Between the Critical Points of a Real Function of n Independent Variables. *Trans. Am. Math. Soc.* **1925**, *27*, 345–396.
20. Milnor, J.; Spivak, M.; Wells, R. *Morse Theory. (AM-51)*; Annals of Mathematics Studies, Princeton University Press: Princeton, NJ, USA, 1963; Volume 51.
21. Banyaga, A.; Hurtubise, D. *Lectures on Morse Homology*; Texts in the Mathematical Sciences; Springer: Amsterdam, The Netherlands, 2013.
22. Matsumoto, Y. *An introduction to Morse Theory*; American Mathematical Society: Providence, RI, USA, 2002.
23. Wasserman, L. *All of Nonparametric Statistics (Springer Texts in Statistics)*; Springer: Berlin/Heidelberg, Germany, 2006.
24. Chacón, E.J.; Duong, T.; Wand, P.M. Asymptotics for general multivariate kernel density derivative estimators. *Stat. Sin.* **2011**, *21*, 807. [CrossRef]
25. Scott, D. *Multivariate Density Estimation: Theory, Practice, and Visualization*; Wiley Series in Probability and Statistics; Wiley: Hoboken, NJ, USA, 2015.
26. Breiman, L.; Meisel, W.; Purcell, E. Variable Kernel Estimates of Multivariate Densities. *Technometrics* **1977**, *19*, 135–144. [CrossRef]
27. Silverman, B.W. *Density Estimation for Statistics and Data Analysis*; Chapman and Hall: London, UK, 1986.
28. Ester, M.; Kriegel, H.P.; Sander, J.; Xu, X. 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, Portland, OR, USA, 2–4 August 1996; pp. 226–231.
29. Székely, G.J.; Rizzo, M.L. Testing for equal distributions in high dimensions. *InterStat* **2004**, *5*, 1249–1272.
30. Gretton, A.; Borgwardt, K.M.; Rasch, M.J.; Schölkopf, B.; Smola, A. A Kernel Two-sample Test. *J. Mach. Learn. Res.* **2012**, *13*, 723–773.
31. Massey, F.J. The Kolmogorov-Smirnov Test for Goodness of Fit. *J. Am. Stat. Assoc.* **1951**, *46*, 68–78. [CrossRef]
32. Bond, J.R.; Kofman, L.; Pogosyan, D. How filaments of galaxies are woven into the cosmic web. *Nature* **1996**, *380*, 603. [CrossRef]
33. Koester, B.; McKay, T.; Annis, J.; Wechsler, R.H.; Evrard, A.; Bleem, L.; York, D. A MaxBCG catalog of 13,823 galaxy clusters from the sloan digital sky survey. *Astrophys. J.* **2007**, *660*, 239–255. [CrossRef]
34. Koester, B.P.; McKay, T.A.; Annis, J.; Wechsler, R.H.; Evrard, A.E.; Rozo, E.; Bleem, L.; Sheldon, E.S.; Johnston, D. MaxBCG: A Red-Sequence Galaxy Cluster Finder. *Astrophys. J.* **2007**, *660*, 221–238. [CrossRef]
35. Brinkman, R.R.; Gasparetto, M.; Lee, S.J.J.; Ribickas, A.J.; Perkins, J.; Janssen, W.; Smiley, R.; Smith, C. High-Content Flow Cytometry and Temporal Data Analysis for Defining a Cellular Signature of Graft-Versus-Host Disease. *Biol. Blood Marrow Transplant.* **2007**, *13*, 691–700. [CrossRef]
36. Giné, E.; Guillou, A. Rates of strong uniform consistency for multivariate kernel density estimators. In *Annales de l'Institut Henri Poincare (B) Probability and Statistics*; Elsevier: Amsterdam, The Netherlands, 2002; Volume 38, pp. 907–921.
37. Genovese, C.R.; Perone-Pacifico, M.; Verdinelli, I.; Wasserman, L. The geometry of nonparametric filament estimation. *J. Am. Stat. Assoc.* **2012**, *107*, 788–799. [CrossRef]
38. Genovese, C.R.; Perone-Pacifico, M.; Verdinelli, I.; Wasserman, L. Nonparametric ridge estimation. *Ann. Stat.* **2014**, *42*, 1511–1545. [CrossRef]
39. Vieu, P. A note on density mode estimation. *Stat. Probab. Lett.* **1996**, *26*, 297–307. [CrossRef]
40. Chazal, F.; Fasy, B.; Lecci, F.; Michel, B.; Rinaldo, A.; Rinaldo, A.; Wasserman, L. Robust topological inference: Distance to a measure and kernel distance. *J. Mach. Learn. Res.* **2017**, *18*, 5845–5884.
41. Chen, Y.C.; Genovese, C.R.; Wasserman, L. Statistical inference using the Morse-Smale complex. *Electron. J. Stat.* **2017**, *11*, 1390–1433. [CrossRef]
42. Nesterov, Y. *Introductory Lectures on Convex Optimization: A Basic Course*, 1st ed.; Springer Publishing Company: New York, NY, USA, 2014.
43. Ruder, S. An overview of gradient descent optimization algorithms. *arXiv* **2016**, arXiv:1609.04747.