Enhanced Mode Clustering

Yen-Chi Chen  
Department of Statistics  
Carnegie Mellon University  
yenchic@andrew.cmu.edu

Christopher R. Genovese  
Department of Statistics  
Carnegie Mellon University  
genovese@stat.cmu.edu

Larry Wasserman  
Department of Statistics  
Carnegie Mellon University  
larry@stat.cmu.edu

Abstract

Mode clustering is a nonparametric method for clustering that defines clusters as the basins of attraction of a density estimator’s modes. We provide several enhancements to mode clustering: (i) a “soft” cluster assignment, (ii) a measure of connectivity between clusters, and (iii) an approach to visualizing the clusters.

1 Introduction

Mode clustering is a nonparametric clustering method [8, 7, 10, 13, 18, 2] with three steps: (i) estimate the density function, (ii) find the modes of the estimator, and (iii) define clusters to be the basins of attraction of these modes. The partition defined by these basins of attraction is known as the Morse complex. There are several advantages to using mode clustering relative to other commonly-used methods: (1) There is a clear population quantity being estimated, namely, the Morse complex of the true density. (2) Computation is simple: the density can be estimated with a kernel density estimator, and the modes and basins of attraction can be found with the mean-shift algorithm. (3) There is a single tuning parameter to choose, namely, the bandwidth of the density estimator. (4) It is theoretically well understood since it depends only on density estimation and mode estimation [2, 25, 24].

Despite these advantages, there is room for improvement. First, mode clustering results in a hard assignment; there is no measure of uncertainty as to how well-clustered a data point is. Second, each cluster may connect to other clusters and there is no method for measuring the overlap between clusters. Third, it is not clear how to visualize the clusters when the dimension is greater than two. In this paper, we propose solutions to all these issues.

Related Work. Mode clustering is based on the mean-shift algorithm [13, 8, 10] which is a popular technique in image segmentation and clustering. [18, 3] formally introduced mode clustering to the statistics literature. The idea of clustering based on high density regions had already been proposed in [14].

2 Review of Mode Clustering

Let \( p \) be the density function of a random vector \( X \in \mathbb{R}^d \). Assume that \( p \) has \( k \) local maxima \( \mathcal{M} = \{ m_1, \ldots, m_k \} \) and is a Morse function, meaning that the Hessian of \( p \) at each stationary point is non-degenerate. We do not assume that \( k \) is known. Given any \( x \in \mathbb{R}^d \), there is a unique gradient ascent path, or flow, starting at \( x \) that eventually arrives one of the modes. We define the
Figure 1: Mode clustering. (a): The basins of attraction for each mode. (b): The mean shift algorithm to cluster data points. The red crosses are the local modes.

clusters as the ‘basins of attraction’ of the modes, i.e., the sets of points whose ascent paths have the same mode. Now we give more detail.

The flow through $x$ is a path $\pi_x : \mathbb{R} \rightarrow \mathbb{R}^d$ such that $\pi_x(0) = x$ and

$$\pi_x'(t) = \nabla p(\pi_x(t)).$$  \hfill (1)

A standard result in Morse theory is that integral curves never intersect except at the modes, so the curves partition the space. We define the destination for the integral curve starting at $x$ by

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

Then $\text{dest}(x) = m_j$ for some mode $m_j$. For each mode $m_j$ we define the basin of attraction of $m_j$ by

$$C_j = \{ x : \text{dest}(x) = m_j \}, \quad j = 1, \cdots, k.$$ \hfill (3)

The partition $C = \{ C_1, \ldots, C_k \}$ is called the Morse complex of $p$.

In practice, $p(x)$ is unknown and we need to estimate it. Let $X_1, \ldots, X_n$ be a random sample from $p$, and let $K$ be a smooth, symmetric kernel. The kernel density estimator (KDE) with bandwidth $h > 0$ is defined by

$$\hat{p}_n(x) = \frac{1}{nh^d} K\left( \frac{||x - X_i||}{h} \right).$$ \hfill (4)

The modes $\hat{M} = \{ \hat{m}_1, \ldots, \hat{m}_k \}$ of $\hat{p}_n$ and the destinations under $\hat{p}_n$ of any point $x$, $\hat{\text{dest}}(x)$, are both easily found using the mean-shift algorithm [13, 8, 10]. The corresponding basins of attraction and estimated Morse complex are

$$\hat{C}_j = \{ x \in \mathbb{R}^d : \hat{\text{dest}}(x) = \hat{m}_j \}, \quad j = 1, \cdots, \hat{k}$$ \hfill (5)

$$\hat{C} = \{ \hat{C}_1, \ldots, \hat{C}_{\hat{k}} \},$$ \hfill (6)

and the sample clusters are defined by

$$X_j = \{ X_i : X_i \in \hat{C}_j \} = \{ X_i : \hat{\text{dest}}(X_i) = \hat{m}_j \}.$$ \hfill (7)

Remark. We suggest selecting bandwidth targeted at estimating the gradient of $p$ accurately. A slight modification of the suggestion in [6] is

$$h_{NR} = \bar{S}_n \times \left( \frac{4}{d+4} \right)^{\frac{d}{d+4}} \frac{n^{-\frac{1}{d+4}}}{n^{-\frac{1}{d+4}}} \bar{S}_n = \frac{1}{d} \sum_{j=1}^{d} S_{n,j},$$ \hfill (8)

where $S_{n,j}$ is the sample deviation along $j$-th coordinate.
3 Soft Clustering

Mode clustering is a type of hard clustering, where each observation is assigned to one and only one cluster. Soft clustering methods \[20, 19, 22, 23\] attempt to capture the uncertainty in this assignment. This is typically represented by an assignment vector for each point that is a probability distribution over the clusters. For example, whereas a hard-clustering method might assign a point \(x\) to cluster 2, a soft clustering might give \(x\) an assignment vector \(a(x) = (0.01, 0.8, 0.01, 0.08, 0.1)\), reflecting both the high confidence that \(x\) belongs to cluster 2 the nontrivial possibility that it belongs to cluster 5.

Soft clustering can capture two types of cluster uncertainty: population level (intrinsic difficulty) and sample level (variability). The population level uncertainty originates from the fact that even if \(p\) is known, some points are more strongly related to their modes than others. Specifically, for a point \(x\) near the boundaries between two clusters, say \(C_1, C_2\), the associated soft assignment vector \(a(x)\) should have \(a_1(x) \approx a_2(x)\). The sample level uncertainty comes from the fact that \(p\) has been estimated by \(\hat{p}_n\). The soft assignment vector \(a(x)\) is designed to capture both types of uncertainty.

3.1 Soft Mode Clustering

We now present two soft mode clustering methods.

**Method 1: Hitting Probability.** Consider a diffusion (a continuous Markov process) that starts at \(x\) and that tends to move to high density regions. Let \(a^{HP}(x) = (a_1^{HP}(x), \ldots, a_K^{HP}(x))\) where \(a_j^{HP}(x)\) is the conditional probability that mode \(j\) is the first mode reached by the diffusion, given that it reaches one of the modes. In this case \(a(x)\) is a probability vector and so is easy to interpret.

Such diffusions have been used in statistics and machine learning for other purposes \[21, 9\].

In more detail, let \(K_h(x, y) = K \left( \frac{\|x-y\|}{h} \right) \) and define the transition probability

\[
q_h(y|x) = \frac{K_h(x, y)p(y)}{p_h(x)} = \frac{K_h(x, y)p(y)}{\int K_h(x, y)p(y)dy} \tag{9}
\]

This defines a Markov process with \(q(y|x)\) the probability of jumping to \(y\) given that the process is at \(x\). Fortunately, we do not actually have to run the diffusion to estimate \(a^{HP}(x)\).

An approximation to the above diffusion process restricted to \(x, y\) in \(\{X_1, \ldots, X_n, \hat{m}_1, \ldots, \hat{m}_k\}\) is as follows. We define a Markov chain that has \(n + \hat{k}\) states. The first \(\hat{k}\) states are the estimated local modes \(\hat{m}_1, \ldots, \hat{m}_k\) and are each absorbing states. That is, the Markov process stops when it hits any of the first \(\hat{k}\) state. The other \(n\) states correspond to the data points \(X_1, \ldots, X_n\). The transition probability from each \(X_i\) is given by

\[
\begin{align*}
P(X_i \rightarrow \hat{m}_l) &= \frac{K_h(X_i, \hat{m}_l)}{\sum_{j=1}^n K_h(X_i, \hat{m}_j) + \sum_{l=1}^{\hat{k}} K_h(X_i, \hat{m}_l)} \\
P(X_i \rightarrow X_j) &= \frac{K_h(X_i, X_j)}{\sum_{j=1}^n K_h(X_i, X_j) + \sum_{l=1}^{\hat{k}} K_h(X_i, \hat{m}_l)}
\end{align*} \tag{10}
\]

for \(i, j = 1, \ldots, n\) and \(l = 1, \ldots, \hat{k}\). Thus, the transition matrix \(P\) is

\[
P = \begin{bmatrix} I & 0 \\ S & T \end{bmatrix}, \tag{11}
\]

where \(I\) is the identity matrix and \(S\) is an \(n \times \hat{k}\) matrix with element \(S_{ij} = P(X_i \rightarrow \hat{m}_j)\) and \(T\) is an \(n \times n\) matrix with element \(T_{ij} = P(X_i \rightarrow X_j)\). Then by Markov chain theory, the absorbing probability from \(X_i\) onto \(\hat{m}_j\) is given by \(\hat{A}_{ij}\) where \(\hat{A}_{ij}\) is the \((i, j)\)-th element of the matrix

\[
\hat{A} = S(I - T)^{-1}. \tag{12}
\]

We define the soft assignment vector by \(\hat{a}_j^{HP}(X_i) = \hat{A}_{ij}\).
Method 2: Density Distance. We can use the density to define a distance \([5, 26, 11, 4, 16, 1]\) which can then be used to construct a soft assignment vector. Given a density function \(p\), the density distance between points \((x, z)\) is

\[
d_q(x, z) = \inf_{\gamma} \int_0^1 p(\gamma(t)) \frac{2\|\gamma(t)-x\|}{d} dt,
\]

where the infimum is over all smooth paths \(\gamma : [0, 1] \mapsto \mathbb{R}^d\) and \(\gamma(0) = x, \gamma(1) = z\) connecting \(x\) and \(z\). When \(q = 1\), the density distance is the same as the Euclidean distance. When \(q > 1\), the density distance is a shortest path that follows high density regions to connect two points \(x, z\).

Let \(m_1, \ldots, m_k\) be local modes. For each \(x \in \mathbb{R}^d\), we define the soft assignment vector \(a^{DD}(x) = (a_1^{DD}(x), \ldots, a_k^{DD}(x))\) by

\[
a_i^{DD}(x) = \frac{\exp(-\beta_0 d_q(x, m_i))}{\sum_{j=1}^k \exp(-\beta_0 d_q(x, m_j))}.
\]

A consistent estimate of \(d_q(x, z)\) due to \([16]\) is the following. Let \(\mathbb{V}_n = \mathbb{X}_n \cup \{x, z\}\) and \(\pi\) be a path consisting of points

\[
\pi = \{(x_0, x_1, \ldots, x_k) : x_i \in \mathbb{V}_n, x_0 = x, x_k = z\}.
\]

The length of \(\pi\), denoted as \(L_q(\pi)\), is defined to be

\[
L_q(\pi) = \sum_{i=1}^k \|x_i - x_{i-1}\|_2^q,
\]

and a consistent estimator of \(d_q(x, z)\) is

\[
\hat{d}_{q, n}(x, z) = \inf_{\pi} L_q(\pi).
\]

Plugging (16) into (14) gives the soft assignment vector. There are two tuning parameters: \(q\) and \(\beta_0\). In practice, we pick \(q = 2\) and

\[
\beta_0 = (\bar{S}_n)^{-q},
\]

where \(S_n\) is defined in (8). This gives a scale-invariant soft assignment vector.

4 Further Enhancements

4.1 Measuring the Connectivity of Clusters

Here we define a measure of connectivity of clusters that can be used to provide geometric information and can also be used to help decide when clusters should be merged. We define the connectivity measure by

\[
\Omega_{ij} = \frac{1}{2\lambda} \left( \frac{1}{N_i} \sum_{i=1}^n \tilde{a}_j(X_i)1(X_i \in \tilde{C}_i) + \frac{1}{N_j} \sum_{i=1}^n \tilde{a}_i(X_i)1(X_i \in \tilde{C}_j) \right), \quad i, j = 1, \ldots, \hat{k},
\]

where \(N_i = \sum_{i=1}^n 1(X_i \in \tilde{C}_i)\) is the number of sample in cluster \(\tilde{C}_i\). The matrix \(\Omega\) is useful as a dimension-free, summary-statistic to describe the degree of overlap/interaction among the clusters. If we think of the (hard) cluster assignments as class labels, then \(\Omega\) is analogous to the mis-classification rate between class \(i\) and class \(j\). The connectivity will be large when two clusters are close. We call \(\Omega\) the matrix of connectivity measure or the connectivity matrix.

An application of \(\Omega\) is to merge highly connected clusters. For instance, Figure 2 gives an example of a dataset with a ring-shaped cluster and a small spherical cluster at the center. We find the mode clusters and then we merge clusters with high connectivity. This gives clusters: the ring and the center cluster. Currently, merging is done based on a user-chosen threshold for \(\Omega\), in this case 0.16. Finding a data-driven method to decide when to merge clusters based on \(\Omega\) is left to future work.

Another application of \(\Omega\) is to connect, but not merge, clusters when visualizing the clustering result. This is useful for visualization as discussed in the next section.
4.2 Visualization

Here we present a method for visualizing the clusters that combines multidimensional scaling (MDS) with our connectivity measure for clusters. Given points $X_1, \ldots, X_n \in \mathbb{R}^d$, we use MDS to project the data into $\mathbb{R}^2$. MDS finds $Z_1, \ldots, Z_n \in \mathbb{R}^2$ to minimize $\sum_{i<j} (d(Z_i, Z_j) - d(X_i, X_j))^2$ where $d$ is the Euclidean metric; see [17, 27, 15].

Our approach consists of two stages. At the first stage, we apply MDS on the modes and plot the result in $\mathbb{R}^2$. At the second stage, we apply MDS to points within each cluster along with the associated mode. Then we place the points around the projected modes. We scale the MDS result at the first stage by a factor $\rho_0$ so that each cluster is separated from each other. Figure 3 gives an example.
Recall that \( \hat{\mathcal{M}} = \{\hat{m}_1, \ldots, \hat{m}_K\} \) is the set of estimated local modes and \( X_j \) is the set of data points belonging to mode \( \hat{m}_j \). At the first stage, we perform MDS on \( \hat{\mathcal{M}} \) so that
\[
\{\hat{m}_1, \ldots, \hat{m}_k\} \overset{\text{MDS}}{\longrightarrow} \{\hat{m}_1^\dagger, \ldots, \hat{m}_k^\dagger\},
\]
where \( \hat{m}_j^\dagger \in \mathbb{R}^2 \) for \( j = 1, \ldots, \hat{k} \). We plot \( \{\hat{m}_1^\dagger, \ldots, \hat{m}_k^\dagger\} \).

At the second stage, we consider each cluster individually. Assume we are working on the \( j \)-th cluster and \( m_j, X_j \) are the corresponding local mode and cluster points. We denote \( X_j = \{X_{j1}, \ldots, X_{jN_j}\} \), where \( N_j \) is the sample size for cluster \( j \). Then we apply MDS to the collection of points \( \{m_j, X_{j1}, X_{j2}, \ldots, X_{jN_j}\} \):
\[
\{m_j, X_{j1}, X_{j2}, \ldots, X_{jN_j}\} \overset{\text{MDS}}{\longrightarrow} \{m_j^*, X_{j1}^*, X_{j2}^*, \ldots, X_{jN_j}^*\},
\]
where \( m_j^*, X_{j1}^*, X_{j2}^*, \ldots, X_{jN_j}^* \in \mathbb{R}^2 \). Then we center the points at \( \hat{m}_j^\dagger \) and place \( X_{j1}^*, X_{j2}^*, \ldots, X_{jN_j}^* \) around \( \hat{m}_j^\dagger \). That is, we make a translation to the set \( \{m_j^*, X_{j1}^*, X_{j2}^*, \ldots, X_{jN_j}^*\} \) so that \( m_j^* \) matches the location of \( \hat{m}_j^\dagger \). Then we plot the translated points \( X_{j1}^*, X_{j2}^*, \ldots, X_{jN_j}^* \). We repeat the above process for each cluster to visualize the high dimensional clustering.

Note that in practice, the above process may cause unwanted overlap among clusters. Thus, one can scale \( \{\hat{m}_1^\dagger, \ldots, \hat{m}_k^\dagger\} \) by a factor \( \rho_0 > 1 \) to remove the overlap.

## 5 Experiments

We ran two experiments in this section.

**Example 1: Two-Dimensional Example.** We consider \( d = 2 \) and a three Gaussian mixture with three centers \((0,0)\), \((0.5,0)\), \((0.25,0.25)\) and covariance matrix \(0.08^2 I \). The sample size is 600. We use the Gaussian kernel with smoothing bandwidth \( h = 0.085 \) from (8). We display the soft clustering for each data point by three colors (red, green and blue). The intensity for each color shows the component of the soft assignment vector. We also provide the ternary plot that contains all the information on soft assignment vector for each data point. Each point on the ternary plot corresponds to one observation and the location of that point on the ternary plot is the soft assignment vector for that observation. Note that for each point, the sum of all components of the soft assignment vector is 1 so that every vector can be represented by a unique point on the ternary plot. Figure 4 shows the result for applying the density distance (with \( q = 2 \) and \( \beta_0 = 28.87 \) by (17)). Figure 5 gives the analysis based on the hitting probability.

**Example 2: Olive Oil data.** We apply our method to the Olive oil data introduced in [12]. This data set consists of 8 chemical measurements (features) for each observation and the total sample size is \( n = 572 \). Each observation is an olive oil produced in one of 3 regions in Italy and these regions are further divided into 9 areas.

Since these measurements are in different units, we normalize and standardize each measurement. We apply the normal reference rule (8) to select the smoothing bandwidth \( h = 0.587 \). We also merged tiny clusters (with sample size less than 5) into nearby larger clusters.

There are 7 remaining clusters. We visualize these clusters and the measure the connectivity (using the hitting probability). Instead of further merging cluster according to the connectivity measures, we keep these 7 clusters and connect pair of clusters if the connectivity measure is larger than 0.07. Finally, we color each point according to the region. (The information about the region was not used in the clustering.) The result is given in figure 6.

As can be seen, most clusters contain one dominating production area. Even in cases where one cluster contains multiple types of olive oil, the connectivity matrix captures this phenomena. For instance, cluster 2 and 3 both contain Calabria, Sicily and South-Apulia. We do observe a connection between cluster 2 and 3 in Figure 6 and a higher connectivity measure in the matrix for connectivity measures. When a group of oil produced in the same area is separated into two clusters, we observe an edge between these two clusters. This shows that the connectivity measure conveys more information on the hidden interaction between clusters.
Figure 4: The density distance method on the three Gaussian mixtures. We use color for visualizing the confidence to assign one point to each cluster. The ternary plot gives the component of soft assignment vector to each cluster for each data point.

Figure 5: The hitting probability method on the three Gaussian mixtures. We use color for visualizing the confidence to assign one point to each cluster. The ternary plot gives the component of soft assignment vector to each cluster for each data point.
Figure 6: Clustering result for the Olive oil data (d=8). Color denotes produce area. Note that we add edges to those pairs of clusters with connectivity measure > 0.07 (colored by red in the matrix). The width of edge reflects the degree of connection.
6 Conclusion

In this paper, we presented enhancements to mode clustering methods, including several approaches to soft mode clustering, a measure of cluster connectivity, and new visualization methods for high-dimensional data. The cluster connectivity and visualization methods apply to other clustering methods as well. We considered several examples; a complete theoretical analysis of the proposed methods will be treated in a future paper.
References

[1] M. Alamgir and U. von Luxburg. Shortest path distance in random k-nearest neighbor graphs. In ICML. icml.cc / Omnipress, 2012.
[2] E. Arias-Castro, D. Mason, and B. Pelletier. On the estimation of the gradient lines of a density and the consistency of the mean-shift algorithm. Technical report, IRMAR, 2013.
[3] A. Azzalini and N. Torelli. Clustering via nonparametric density estimation. Statistics and Computing, 17(1):71–80, 2007. ISSN 0960-3174. doi: 10.1007/s11222-006-9010-y.
[4] A. S. Bijral, N. Ratliff, and N. Srebro. Semi-supervised learning with density based distances, 2011.
[5] O. Bousquet, O. Chapelle, and M. Hein. Measure based regularization. In Advances in Neural Information Processing Systems 16. MIT Press, 2004.
[6] J. Chacón, T. Duong, and M. Wand. Asymptotics for general multivariate kernel density derivative estimators. Statistica Sinica, 2011.
[7] F. Chazal, L. Guibas, S. Oudot, and P. Skraba. Persistence-based clustering in riemannian manifolds. In Proceedings of the 27th annual ACM symposium on Computational geometry, pages 97–106. ACM, 2011.
[8] Y. Cheng. Mean shift, mode seeking, and clustering. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 17(8):790–799, 1995.
[9] R. R. Coifman and S. Lafon. Diffusion maps. Applied and Computational Harmonic Analysis, 2006.
[10] D. Comaniciu and P. Meer. Mean shift: a robust approach toward feature space analysis. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 24(5):603–619, may 2002.
[11] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. Introduction to Algorithms. The MIT Press, 2009.
[12] M. Forina, C. Armanino, S. Lanteri, and E. Tiscornia. Classification of olive oils from their fatty acid composition. Food Research and Data Analysis, 1983.
[13] K. Fukunaga and L. D. Hostetler. The estimation of the gradient of a density function, with applications in pattern recognition. IEEE Transactions on Information Theory, 21:32–40, 1975.
[14] J. Hartigan. Clustering Algorithms. Wiley and Sons, 1975.
[15] T. Hastie, R. Tibshirani, and J. Friedman. The Elements of Statistical Learning. Springer Series in Statistics. Springer New York Inc., New York, NY, USA, 2001.
[16] S. J. Hwang, S. B. Damelin, and A. O. Hero. Shortest path through random points. arXiv: 1202.0045, 2012.
[17] J. Kruskal. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. Psychometrika, 29(1):1–27, 1964. ISSN 0033-3123. doi: 10.1007/BF02289565.
[18] J. Li, S. Ray, and B. Lindsay. A nonparametric statistical approach to clustering via mode identification. Journal of Machine Learning Research, 8(8):1687–1723, 2007.
[19] P. Lingras and C. West. Interval set clustering of web users with rough k-means. Journal of Intelligent Information Systems, 2002.
[20] G. McLachlan and D. Peel. Finite mixture models. John Wiley & Sons, 2004.
[21] B. Nadler, S. Lafon, R. R. Coifman, and I. G. Kevrekidis. Diffusion maps, spectral clustering and eigenfunctions of fokker-planck operators. NIPS, 2005.
[22] R. Nock and F. Nielsen. On weighting clustering. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2006.
[23] G. Peters, F. Crespoc, P. Lingrasd, and R. Weber. Soft clustering - fuzzy and rough approaches and their extensions and derivatives. International Journal of Approximate Reasoning, 2013.
[24] J. P. Romano. Bootstrapping the mode. Annals of the Institute of Statistical Mathematics, 40(3):565–586, 1988.
[25] J. P. Romano et al. On weak convergence and optimality of kernel density estimates of the mode. The Annals of Statistics, 16(2):629–647, 1988.
[26] Sajama and A. Orlitsky. Estimating and computing density based distance metrics. In Proceedings of the 22Nd International Conference on Machine Learning, ICML ’05, 2005.
[27] R. N. Shepard. Multidimensional scaling, tree-fitting, and clustering. Science, 210(4468):390–398, 1980.