Clustering with Confidence: A Low-Dimensional Binning Approach

Rebecca Nugent and Werner Stuetzle

Abstract We present a plug-in method for estimating the cluster tree of a density. The method takes advantage of the ability to exactly compute the level sets of a piecewise constant density estimate. We then introduce *clustering with confidence*, an automatic pruning procedure that assesses significance of splits (and so clusters) in the cluster tree; the only user input required is the desired confidence level.

1 Introduction

The goal of clustering is to identify distinct groups in a data set and assign a group label to each observation. Ideally, we would be able to find the number of groups as well as where each group lies in the feature space with minimal input from the user. To cast clustering as a statistical problem, we regard the data, $\mathbf{X} = \{\mathbf{x_1}, \ldots, \mathbf{x_n}\} \in \mathbf{R^p}$, as a sample from some unknown population density $p(\mathbf{x})$. There are two statistical approaches. While parametric (model-based) clustering assumes the data have been generated by a finite mixture of g underlying parametric probability distributions $p_g(\mathbf{x})$ (often multivariate Gaussian) (Fraley and Raftery 1998; McLachlan and Basford 1988), the nonparametric approach assumes a correspondence between groups in the data and modes of the density $p(\mathbf{x})$. Wishart first advocated searching for modes as manifestations of the presence of groups (Wishart 1969); nonparametric clustering should be able to "resolve distinct data modes, independently of their shape and variance". Hartigan expanded this idea and made it more precise (Hartigan 1975, 1981).

Define a level set $L(\lambda; p)$ of a density p at level λ as the subset of the feature space for which the density exceeds λ: $L(\lambda; p) = \{\mathbf{x} | p(\mathbf{x}) > \lambda\}$. Its connected components are the maximally connected subsets of a level set. For any two connected components A and B, possibly at different levels, either $A \subset B$, $B \subset A$, or

R. Nugent (✉)
Department of Statistics, Carnegie Mellon University, Pittsburgh, PA, USA
e-mail: rnugent@stat.cmu.edu

H. Locarek-Junge and C. Weihs (eds.), *Classification as a Tool for Research*,
Studies in Classification, Data Analysis, and Knowledge Organization,
DOI 10.1007/978-3-642-10745-0_12, © Springer-Verlag Berlin Heidelberg 2010

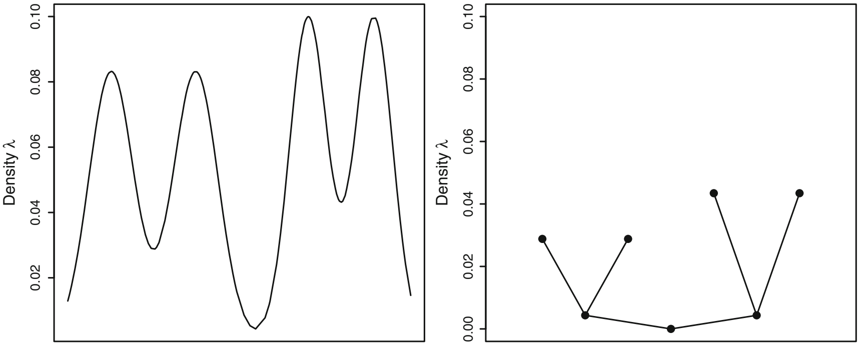

Fig. 1 (a) Density with four modes; (b) cluster tree with three splits/four leaves

$A \cap B = \emptyset$. This hierarchical structure of the level sets is summarized by the *cluster tree* of p.

The cluster tree is easiest to define recursively (Stuetzle 2003). Each node N of the tree represents a subset $D(N)$ of the support $L(0; p)$ of p and is associated with a density level $\lambda(N)$. The root node represents the entire support of p and is associated with density level $\lambda(N) = 0$. To determine the daughters of a node, we find the lowest level λ_d for which $L(\lambda_d; p) \bigcap D(N)$ has two or more connected components. If no such λ_d exists, then $D(N)$ is a mode of the density, and N is a leaf of the tree. Otherwise, let $C_1, C_2, \ldots, C_n$ be the connected components of $L(\lambda_d; p) \bigcap D(N)$. If $n = 2$, we create two daughter nodes at level λ_d, one for each connected component; we then apply the procedure recursively to each daughter node. If $n > 2$, we create two connected components C_1 and $C_2 \bigcup C_3 ... \bigcup C_n$ and their respective daughter nodes and then recurse. We call the regions $D(N)$ the "high density clusters" of p. This recursive binary tree also can accommodate level sets with more than two connected components with repeated splits at the same height. Figure 1 shows a univariate density with four modes and the corresponding cluster tree with initial split at $\lambda = 0.0044$ and subsequent splits at $\lambda = 0.0288, 0.0434$. Estimating the cluster tree is a fundamental goal of nonparametric cluster analysis.

There are several previously suggested clustering methods based on level sets and other level set estimation procedures. In general, they are heuristic in nature or require subjective decisions from the user (Wishart 1969; Walther 1997; Cuevas et al. 2000, 2001; Stuetzle 2003; Klemelä 2004).

2 Cluster Trees: Piecewise Constant Density Estimates

We can estimate the cluster tree of a density p by the cluster tree of a density estimate $\hat{p}$. However, for most density estimates, computing the cluster tree is a difficult problem; there is no obvious method for computing and representing the level sets.

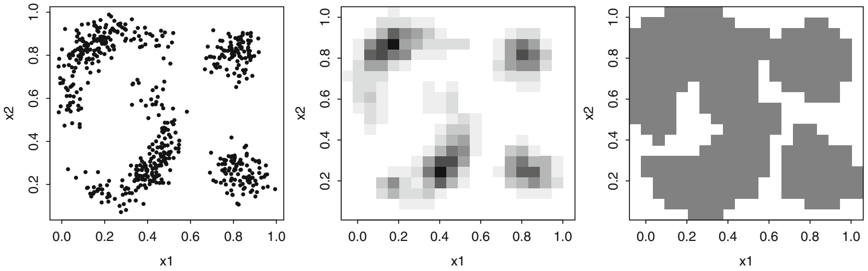

Fig. 2 (a) Four well-separated groups; (b) BKDE, 20×20 grid (c) L(0.00016;BKDE)

Exceptions are density estimates that are piecewise constant over (hyper-)rectangles. Let $B_1, B_2, \ldots, B_N$ be the rectangles, and let $\hat{p}_i$ be the estimated density for B_i. Then $L(\lambda; \hat{p}) = \bigcup_{\hat{p}_i > \lambda} B_i$. If the dimension is low enough, any density estimate can be reasonably binned. Here we use the Binned Kernel Density Estimate (BKDE $= \hat{p}$), a binned approximation to an ordinary kernel density estimate, on a grid we can computationally afford (Wand and Jones 1995; Hall and Wand 1996). We use 10-fold cross validation to estimate the bandwidth h.

Figure 2a has four well-separated groups, two curvilinear and two spherical; a grey-scale heat map of a BKDE on a 20×20 grid ($h = 0.0244$) is in Fig. 2b. Figure 2c shows the level set $L(0.00016; \hat{p})$; since we have a split, we would create two daughter nodes at this height. When a cluster tree node N has been split into daughters N_l, N_r, the high density clusters $D(N_l)$, $D(N_r)$, also referred to as the cluster "cores", do not necessarily form a partition of $D(N)$. We refer to the bins (and their observations) in $D(N) \backslash (D(N_l) \cup D(N_r))$, e.g. the white bins in Fig. 2c, as the "fluff". We assign each fluff bin B to N_r if the Manhattan distance $d_M(B, D(N_r)) = \|B - D(N_r)\|_1$ is less than $d_M(B, D(N_l))$. If $d_M(B, D(N_r)) > d_M(B, D(N_l))$, then B is assigned to N_l. In case of ties, the algorithm arbitrarily chooses an assignment. The cluster cores and fluff represented by the leaves of the cluster tree form a partition of the support of $\hat{p}$ and a corresponding partition of the observations. The same is true for every subtree of the cluster tree.

During cluster tree construction, $L(\lambda_d; \hat{p}) \bigcap D(N)$ changes structure only when the level λ_d is equal to the next higher value of $\hat{p}(B_i)$ for one or more bins B_i in $D(N)$. We compute the cluster tree of $\hat{p}$ by "stepping through" the bins' sorted unique density estimate values; every increase in level λ_d then corresponds to the removal of one or more bins from the level set $L(\lambda_d; \hat{p})$. We represent $L(\lambda_d; \hat{p}) \cap D(N)$ as an adjacency graph G where the vertices $B_i \in L(\lambda_d; \hat{p}) \cap D(N)$ are connected by an edge if they share a lower-dimensional face. Finding its connected components is a standard graph problem (Robert 2002).

Figure 3a shows the BKDE's cluster tree; the corresponding cluster assignments and partitioned feature space are in Fig. 3b,c. The cluster tree indicates the BKDE has nine modes. The first split at $\lambda = 1.9 \cdot 10^{-16}$ and the mode around (1, 0.5)

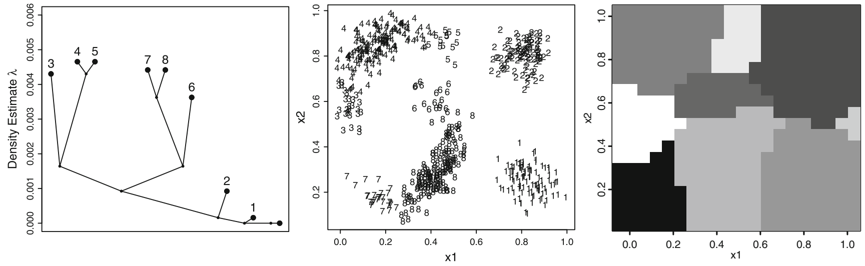

Fig. 3 (a) BKDE cluster tree; (b) cluster assignments; (c) partitioned feature space

in Fig. 3c are artifacts of $\hat{p}$; no observations are assigned to the resulting daughter node. The remaining eight leaves correspond to (a subset of) one of the groups.

We briefly compare these results to common clustering methods (not described here) by comparing the estimated clusters to the true groups using the Adjusted Rand Index (Hubert and Arabie 1985). The ARI is a common measure of agreement between two partitions. Its expected value is zero under random partitioning with a maximum value of one; larger values indicate better agreement. Using the total within-cluster sum of squares criterion, k-means (Mardia et al. 1979) selects five or six clusters with ARI = 0.803, 0.673; for $k = 4$, ARI = 0.924. Model-based clustering (MBC, Fraley and Raftery 1998; McLachlan and Basford 1988) with an unconstrained covariance structure chose ten clusters (ARI = 0.534). The BKDE cluster tree on a 20×20 grid performed comparably ($k = 8$; ARI = 0.781); a 15×15 grid performed slightly better ($k = 6$; ARI = 0.865). The groups are well-separated; however, the two curvilinear groups give an increased number of clusters (k-means, MBC, and the cluster tree). Single, complete, and average hierarchical linkage methods (Mardia et al. 1979) gave perfect agreement given knowing the true number of groups in advance.

For both grid choices, the cluster tree overestimated the number of groups (8,6). Figure 3 illustrates this problem in the approach. While the cluster tree is accurate for the given density estimate, the inherent noise in the density estimate results in spurious modes not corresponding to groups in the underlying population. In our example, the procedure identified the four original groups (post the modal artifact) but erroneously continued splitting the clusters. The corresponding branches of the cluster tree need to be pruned.

3 Clustering with Confidence

We propose a bootstrap-based automatic pruning procedure that finds simultaneous upper and lower $(1 - \alpha)$ confidence sets for each level set. During cluster tree construction, only splits indicated as significant by the bootstrap confidence sets are taken to signal multi-modality. Spurious modes are discarded during estimation; the only user decision is the confidence level.

3.1 Bootstrap Confidence Sets for Level Sets

We define upper confidence sets (UCS) to be of the form $L^u(\lambda; \hat{p}) = L(\lambda - \delta^u_\lambda; \hat{p})$ and lower confidence sets (LCS) of form $L^l(\lambda; \hat{p}) = L(\lambda + \delta^l_\lambda; \hat{p})$ with $\delta^u_\lambda, \delta^l_\lambda > 0$. By construction, LCS $= L^l(\lambda; \hat{p}) \subseteq L(\lambda; \hat{p}) \subseteq L^u(\lambda; \hat{p}) =$ UCS.

Let $\hat{p}^*_1, \hat{p}^*_2, \dots, \hat{p}^*_m$ be the density estimates for m bootstrap samples of size n drawn with replacement from the original sample. We call a pair $(L(\lambda - \delta^u_\lambda; \hat{p}), L(\lambda + \delta^l_\lambda; \hat{p}))$ a non-simultaneous $(1 - \alpha)$ confidence set for $L(\lambda; p)$ if for $100 \cdot (1 - \alpha)\%$ of the bootstrap density estimates $\hat{p}^*_i$, the upper confidence set $L(\lambda - \delta^u_\lambda; \hat{p})$ *contains* $L(\lambda; \hat{p}^*_i)$, and the lower confidence set $L(\lambda + \delta^l_\lambda; \hat{p})$ *is contained in* $L(\lambda; \hat{p}^*_i)$:

$$P_{boot}\{L(\lambda + \delta^l_\lambda; \hat{p}) \subseteq L(\lambda; \hat{p}^*_i) \subseteq L(\lambda - \delta^u_\lambda; \hat{p})\} \geq 1 - \alpha.$$

Here is one method to determine $\delta^u_\lambda, \delta^l_\lambda$ (Buja 2002). For each bootstrap sample $\hat{p}^*_i$ and each of the finitely many levels of $\hat{p}$, find the smallest $\delta^u_\lambda(i)$ such that $L(\lambda; \hat{p}^*_i) \subseteq L(\lambda - \delta^u_\lambda(i); \hat{p})$ and the smallest $\delta^l_\lambda(i)$ such that $L(\lambda + \delta^l_\lambda(i); \hat{p}) \subseteq L(\lambda; \hat{p}^*_i)$. Choose $\delta^u_\lambda = (1 - \frac{\alpha}{2})$ quantile of the $\delta^u_\lambda(i)$ and $\delta^l_\lambda = (1 - \frac{\alpha}{2})$ quantile of the $\delta^l_\lambda(i)$. By construction, the pair $(L(\lambda - \delta^u_\lambda; \hat{p}), L(\lambda + \delta^l_\lambda; \hat{p}))$ is a $(1 - \alpha)$ non-simultaneous confidence set for $L(\lambda; p)$. To get confidence sets for all λ occurring as values of $\hat{p}$ with simultaneous coverage probability $1 - \alpha$, we simply increase the coverage level of the individual sets until the desired level of simultaneous coverage is reached. Note that the actual upper and lower confidence sets for $L(\lambda; p)$ are the level sets $(L(\lambda - \delta^u_\lambda; \hat{p}), L(\lambda + \delta^l_\lambda; \hat{p}))$ respectively for $\hat{p}$. The bootstrap is used only to find $\delta^u_\lambda, \delta^l_\lambda$.

3.2 Constructing the Cluster Tree

After finding $\delta^u_\lambda, \delta^l_\lambda$ for all λ, we incorporate the bootstrap confidence sets into the cluster tree construction by only allowing splits at heights λ for which the corresponding bootstrap confidence set $(L^l(\lambda; \hat{p}), L^u(\lambda; \hat{p}))$ gives strong evidence of a split. We use a similar recursive procedure to that in Sect. 2. The root node represents the entire support of $\hat{p}$ and is associated with density level $\lambda(N) = 0$. To determine the daughters of a node, we find the lowest level λ_d for which a) $L^l(\lambda_d; \hat{p}) \bigcap D(N)$ has two or more connected components that b) are disconnected in $L^u(\lambda_d; \hat{p}) \bigcap D(N)$. Condition (a) indicates that the underlying density p has two peaks above height λ; condition (b) indicates that the two peaks are separated by a valley dipping below height λ. Satisfying both conditions indicates a split at height λ. If no such λ_d exists, N is a leaf of the tree. Otherwise, let C^l_1, C^l_2 be two connected components of $L^l(\lambda_d; \hat{p}) \bigcap D(N)$ that are disconnected in $L^u(\lambda_d; \hat{p}) \bigcap D(N)$. Let C^u_1 and C^u_2 be the connected components of $L^u(\lambda_d; \hat{p}) \bigcap D(N)$ from the possible $C^u_1, C^u_2, \dots, C^u_n$ that contain C^l_1 and C^l_2

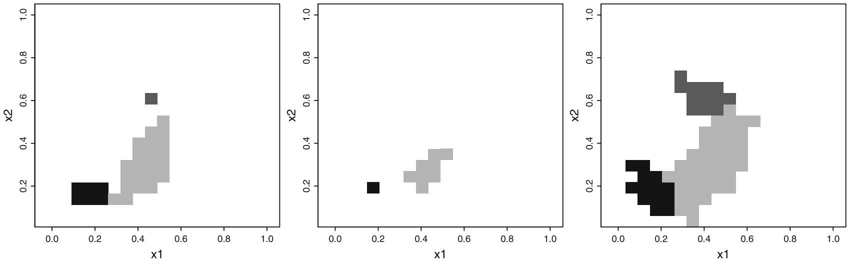

Fig. 4 (a)$L(0.0036; \hat{p})$; (b) LCS, $\delta^l_\lambda = 0.0063$; (c) UCS, $\delta^u_\lambda = 0.0028$

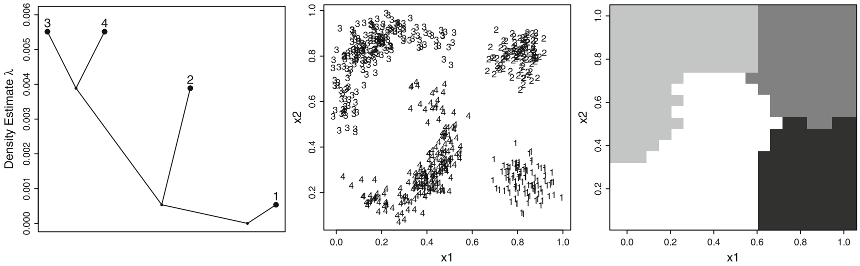

Fig. 5 (a) 95% confidence cluster tree (b) clusters; (c) partitioned feature space

respectively. If $n = 2$, we create two daughter nodes at level λ_d for C^u_1 and C^u_2 and, to each, apply the procedure recursively. If $n > 2$, we create two connected components C^u_1 and $C^u_2 \bigcup C^u_3 ... \bigcup C^u_n$ and respective daughter nodes and recurse.

We return to the split at $\lambda = 0.0036$, the first split that breaks the lower left curvilinear group into two clusters (Fig. 3a: 6 and 7,8). Figure 4 shows the bootstrap confidence set ($\alpha = 0.05$) for this level set. The original level set $L(0.0036; \hat{p})$ is in Fig. 4a (grey-scale corresponds to Fig. 3 final leaf). The LCS is found to be $\delta^l_\lambda = 0.0063$ higher, i.e. $L(0.0099; \hat{p})$ (Fig. 4b). The UCS is found to be $\delta^u_\lambda = 0.0028$ lower, i.e. $L(0.0008; \hat{p})$ (Fig. 4c). At $\lambda = 0.0008$, the UCS does not have two connected components (no valley). Moreover, even though the LCS does have two connected components, they do not correspond to the two connected components in $L(0.0036; \hat{p})$. We do not have evidence of a significant split and so do not create daughter nodes at this level.

Clustering with Confidence (CWC) with $\alpha = 0.05$ generates the cluster tree and data/feature space partitions in Fig. 5. The cluster tree's significant splits have identified the four original groups as significant clusters (ARI = 1). No other smaller clusters (or modal artifacts) are found. Note that the split heights are higher than the corresponding split heights in the cluster tree in Fig. 3. The CWC procedure required stronger evidence for a split than was available at the lower levels. It performed more favorably than k-means or model-based clustering and provided a measure of confidence for the clusters.

4 Example: "Automatic Gating" in Flow Cytometry

The algorithms presented could be used for any number of dimensions but are more tractable for lower dimensions. For easier visualization of the results, we present a real two-dimensional application from molecular biology. We comment on higher dimensionality in the summary and future work section.

Flow cytometry is a technique for examining and sorting tagged mRNA molecules in a cell population. Each cell's fluorescence level (corresponding to, e.g., gene expression level) is recorded as particles pass in front of a single wavelength laser. We are interested in discovering groups of cells with high fluorescence levels for multiple channels or groups of cells that have different levels across channels. A common identification method is "gating" or subgroup extraction from two-dimensional plots of measurements on two channels. Most commonly, these subgroups are identified by eyeballing the graphs. Clustering techniques would allow for more statistically motivated subgroup identification (Lo et al. 2008).

We have 1,545 flow cytometry measurements on two fluorescence markers (anti-BrdU, binding dye 7-AAD) applied to Rituximab, a therapeutic monoclonal antibody, in a drug-screening project designed to identify agents to enhance its anti-lymphoma activity (Lo et al. 2009). Figure 6a shows the cluster tree (BKDE 15×15; $h = 21.834$); the cluster assignments as well as whether or not the observations are part of a cluster "core" (larger labels) are in Fig. 6b. The cluster tree has 12 leaves (8 clusters, 4 modal artifacts). The core sizes give some evidence as to their eventual significance. For example, cluster 1's core near (500, 1,000) contains one observation; we would not expect cluster 1 to remain in the confidence cluster tree for any reasonable α.

We use CWC to construct a confidence cluster tree for $\alpha = 0.10$; we are at least 90% confident in the generated clusters (Fig. 6c). All modal artifacts have been removed; the smaller clusters are merged into two larger clusters with cores at (200, 200), (700, 300). Note that the right cluster is a combination of the mid to high 7-AAD clusters in Fig. 6b. CWC did not find enough evidence to warrant splitting this larger cluster further into subgroups.

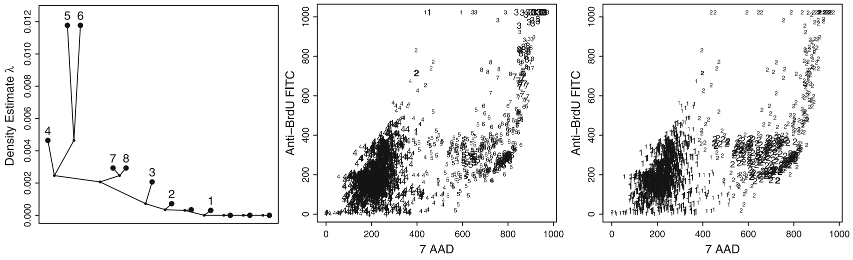

Fig. 6 Flow cytometry measurements on the two fluorescent markers anti-BrdU and 7-AAD; (a) Cluster tree with 12 leaves (8 clusters, 4 artifacts); (b) Cluster assignments; core obs have larger labels (c) 90% confidence cluster assignments

5 Summary and Future Work

We have presented a plug-in method for estimating the cluster tree of a density that takes advantage of the ability to exactly compute the level sets of a piecewise constant density estimate. The approach shows flexibility in finding clusters of unequal sizes and shapes. However, the cluster tree is dependent on the (inherently noisy) density estimate. We introduced *clustering with confidence*, an automatic pruning procedure that assesses significance of splits in the cluster tree; the only input needed is the desired confidence level.

These procedures may become computationally intractable as the number of adjacent bins grows with the dimension and are realistically for use in lower dimensions. One high-dimensional approach would be to employ projection or dimension reduction techniques prior to cluster tree estimation. We also have developed a graph-based approach that approximates the cluster tree in high dimensions (Stuetzle and Nugent 2010). CWC then could be applied to the resulting graph to identify significant clusters.

Acknowledgements This work was partially supported by NSF grants DMS-0505824 and DMS-0240019.

References

Buja, A. (2002). Personal communication. Also Buja, A. and Rolke, W. *Calibration for simultaneity: (Re)Sampling methods for simultaneous inference with applications to function estimation and functional data.* In revision.

Cuevas, A., Febrero M., & Fraiman, R. (2000). Estimating the number of clusters. *The Canadian Journal of Statistics, 28*, 367–382.

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

Fraley, C., & Raftery, A. (1998). How many clusters? which clustering method? Answers via model-based cluster analysis. *The Computer Journal, 41*, 578–588.

Hall, P., & Wand, M. P. (1996) On the accuracy of binned kernel density estimators'. *Journal of Multivariate Analysis, 56*, 165–184.

Hartigan, J. A. (1975). *Clustering Algorithms*. London: Wiley.

Hartigan, J. A. (1981). Consistency of single linkage for high-density clusters. *Journal of the American Statistical Association, 76*, 388–394.

Hubert, L., & Arabie, P. (1985). Comparing partitions. *Journal of Classification, 2*, 193–218.

Klemelä, J. (2004). Visualization of multivariate density estimates with level set trees. *Journal of Computational and Graphical Statistics, 13*, 599–620.

Lo, K., Brinkman R., & Gottardo, R. (2008). Automated gating of flow cytometry data via robust model-based clustering. *Cytometry, Part A, 73A*, 321–332.

Lo, K., Hahne, F., Brinkman, R.R., and Gottardo, R. (2009). FlowClust: a Bioconductor package for automated gating of flow cytometry data. *BMC Bioinformatics*, *10*, 145.

Mardia, K., Kent, J. T., & Bibby, J. M. (1979). *Multivariate analysis*. New York: Academic Press.

McLachlan, G. J., & Basford, K. E. (1988). *Mixture models: Inference and applications to clustering*. New York, USA: Marcel Dekker.

Robert, S. (2002). *Algorithms in C, Part 5: Graph Algorithms* (3rd ed.) Reading, MA: Addison-Wesley.

Stuetzle, W. (2003). Estimating the cluster tree of a density by analyzing the minimal spanning tree of a sample. *Journal of Classification, 20*, 25–47.

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

Walther, G. (1997). Granulometric smoothing. *The Annals of Statistics, 25*, 2273–2299.

Wand, M. P., & Jones, M. C. (1995). *Kernel smoothing*. London: Chapman & Hall.

Wishart, D. (1969). Mode analysis: A generalization of nearest neighbor which reduces chaining effect. In A. J. Cole (Ed.), *Numerical Taxonomy* (pp. 282–311). London: Academic Press.