Distribution Free Decomposition of Multivariate Data *

Dorin Comaniciu and Peter Meer

Department of Electrical and Computer Engineering
Rutgers University, Piscataway, NJ 08855, USA

Abstract. A practical approach to nonparametric cluster analysis of large data sets is presented. The number of clusters and the cluster centers are derived by applying the mean shift procedure on a reduced set of points randomly selected from the data. The cluster boundaries are delineated using a k-nearest neighbor technique. The resulting algorithm is stable and efficient, allowing the cluster decomposition of a 10000 point data set in only a few seconds. Complex clustering examples and applications are discussed.

1 Introduction

In image understanding the feature spaces derived from real data most often have a complex structure and *a priori* information to guide the analysis may not be available. The significant features whose recovery is necessary for the solution of a task, correspond to clusters in this space. The number of clusters, their shape and rules of assignment have to be discerned solely from the given data.

The feature space can be regarded as a sample drawn from an unknown probability distribution. Representing this distribution with a parametric model (e.g., Gaussian mixture) will introduce severe artifacts since then the shape of the delineated clusters is predefined. Nonparametric cluster analysis, on the other hand, uses the *modes* of the underlying probability density to define the cluster centers and the *valleys* in the density to define the boundaries separating the clusters.

To estimate the probability density several nonparametric techniques are available: multivariate histogram, the nearest neighbor method, kernel estimation, [5, 12, 14]. For higher dimensional feature spaces, multivariate histograms are less useful due to their exponentially growing number of bins with the space dimension, as well as due to the artifacts introduced by the quantization. The nearest neighbor method is prone to local noise (which makes difficult the accurate detection of the modes), and the obtained estimate is not a probability density since it integrates to infinity [14, p. 96]. For low to medium data sizes kernel estimation is a good practical choice; it is simple, and for kernels obeying

* This research was supported by the NSF under the grant IRI-9530546.

mild conditions the estimate is asymptotically unbiased, consistent in a mean-square sense, and uniformly consistent in probability.

The two nonparametric techniques discussed in this paper belong to the class of kernel estimators. In the first technique the underlying density is estimated and a hierarchical data structure is derived, based on which the data is decomposed. An example is the *graph theoretical* approach [5, p. 539]. In the second technique density gradient estimation [4] is used, the modes being detected with the hill climbing *mean shift* procedure [1].

Note that both the density and the density gradient estimation require the search for the data points falling in the neighborhood delineated by the employed kernel. This task is called *multidimensional range searching* [13, p. 373]. However, unlike the nearest neighbor search which can be performed in logarithmic time [3, 9], the performance of the multidimensional range searching is difficult to predict for a particular data set [13, p. 385]. Therefore, for applications involving large data sets (e.g., multispectral image segmentation [2], image restoration, speech and image coding), both the kernel estimation and density gradient estimation become computationally expensive, their complexity being proportional to the square of the number of data points. The attempt to reduce computations by subsampling the data leads to inaccuracy, most notably in the tails [10].

As a solution to the problem described above, this paper presents a practical algorithm for unsupervised nonparametric clustering of large data sets. The algorithm is based on the mean shift procedure, being simple, efficient, and easy to implement. In Section 2 the principles behind the kernel density and the density gradient estimation are reviewed, and the specific clustering techniques are discussed in Section 3. The proposed algorithm is presented in Section 4, with experimental results shown in Section 5.

2 Density and Density Gradient Estimation

Let $\mathbf{X}_1 \ldots \mathbf{X}_n$ be a set of n points in the d-dimensional Euclidean space R^d. The *multivariate kernel density estimate* obtained with kernel $K(\mathbf{x})$ and window width h, computed in the point $\mathbf{x}$ is defined as [14, p. 76]

$$\hat{f}(\mathbf{x}) = \frac{1}{nh^d} \sum_{i=1}^{n} K\left(\frac{\mathbf{x} - \mathbf{X}_i}{h}\right). \tag{1}$$

The kernel $K(\mathbf{x})$ is a scalar function which must satisfy the following conditions [4]

$$\sup_{\mathbf{x} \in R^d} |K(\mathbf{x})| < \infty, \quad \int_{R^d} |K(\mathbf{x})| d\mathbf{x} < \infty, \quad \lim_{\|\mathbf{x}\| \to \infty} \|\mathbf{x}\| K(\mathbf{x}) = 0, \quad \int_{R^d} K(\mathbf{x}) d\mathbf{x} = 1, \tag{2}$$

where $\|\cdot\|$ is the Euclidean norm. For optimum performance, the window width h has to be a function of the sample size n. Asymptotic unbiasedness, mean-square consistency, and uniform consistency in probability of the density estimate are assured if the following conditions are satisfied, respectively

$$\lim_{n\to\infty} h(n) = 0, \quad \lim_{n\to\infty} nh^d(n) = \infty, \quad \lim_{n\to\infty} nh^{2d}(n) = \infty. \tag{3}$$

The optimum kernel yielding minimum mean integrated square error (MISE) is the Epanechnikov kernel

$$K_e(\mathbf{x}) = \begin{cases} \frac{1}{2}c_d^{-1}(d+2)(1-\mathbf{x}^T\mathbf{x}) & \text{if } \mathbf{x}^T\mathbf{x} < 1 \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

where c_d is the volume of the unit d-dimensional sphere [14, p. 76]. Uniform and Gaussian kernels are also frequently used. Note that a fast computation of (1) requires a fast multidimensional range searching around $\mathbf{x}$.

The use of a differentiable kernel allows to define the estimate of the density gradient as the gradient of the kernel density estimate (1)

$$\hat{\nabla} f(\mathbf{x}) \equiv \nabla \hat{f}(\mathbf{x}) = \frac{1}{nh^d} \sum_{i=1}^{n} \nabla K\left(\frac{\mathbf{x}-\mathbf{X}_i}{h}\right). \tag{5}$$

Conditions on the kernel $K(\mathbf{x})$ and the window width h to guarantee asymptotic unbiasedness, mean-square consistency, and uniform consistency are derived in [4].

For the Epanechnikov kernel (4) the density estimate (5) becomes

$$\hat{\nabla} f(\mathbf{x}) = \frac{1}{n(h^d c_d)} \frac{d+2}{h^2} \sum_{\mathbf{X}_i \in S_h(\mathbf{x})} [\mathbf{X}_i - \mathbf{x}] = \frac{k}{n(h^d c_d)} \frac{d+2}{h^2} \left(\frac{1}{k} \sum_{\mathbf{X}_i \in S_h(\mathbf{x})} [\mathbf{X}_i - \mathbf{x}] \right) \tag{6}$$

where the region $S_h(\mathbf{x})$ is a hypersphere of radius h having the volume $h^d c_d$, centered on $\mathbf{x}$, and containing k data points. Note that k implicitly depends on $\mathbf{x}$. The last term in (6)

$$M_h(\mathbf{x}) \equiv \frac{1}{k} \sum_{\mathbf{X}_i \in S_h(\mathbf{x})} [\mathbf{X}_i - \mathbf{x}] = \frac{1}{k} \sum_{\mathbf{X}_i \in S_h(\mathbf{x})} \mathbf{X}_i - \mathbf{x} \tag{7}$$

is called the *sample mean shift*. Using a kernel different from the Epanechnikov kernel results in a weighted mean computation in (7). Note again that efficient mean shift computation requires efficient range searching.

The quantity $\frac{k}{n(h^d c_d)}$ is the kernel density estimate $\hat{f}(\mathbf{x})$ computed with the hypersphere $S_h(\mathbf{x})$ (the uniform kernel), and thus we can write (6) as

$$\hat{\nabla} f(\mathbf{x}) = \hat{f}(\mathbf{x}) \frac{d+2}{h^2} M_h(\mathbf{x}), \tag{8}$$

which yields

$$M_h(\mathbf{x}) = \frac{h^2}{d+2} \frac{\hat{\nabla} f(\mathbf{x})}{\hat{f}(\mathbf{x})}. \tag{9}$$

The expression (9) was first derived in [4] and shows that an estimate of the normalized gradient can be obtained by computing the sample mean shift in a

uniform kernel centered on $\mathbf{x}$. The mean shift vector has the direction of the gradient density estimate at $\mathbf{x}$ when this estimate is obtained with the Epanechnikov kernel. Therefore, the Epanechnikov kernel is also called the *shadow* of the uniform kernel [1].

Since the mean shift vector always points towards the direction of the maximum increase in the density, it can define a path leading to a local density maximum, i.e., to a mode of the density. Note that the normalized gradient in (9) introduces a desirable adaptive behavior: the mean shift step is large for low density regions corresponding to valleys, and decreases as $\mathbf{x}$ approaches a mode.

3 Distribution Free Clustering

Associated with the two estimates (the density and its gradient), there are two basic algorithms of nonparametric clustering. For a given window radius h, both algorithms automatically detect the number of existing clusters and their corresponding boundaries.

Using the density estimate (1) a hierarchical structure of the data can be obtained as follows. For each point $\mathbf{X}_i$ search its neighborhood for a parent $\mathbf{X}_j$, for which the quantity $\left[\hat{f}(\mathbf{X}_j) - \hat{f}(\mathbf{X}_i)\right] \cdot \|\mathbf{X}_j - \mathbf{X}_i\|^{-1}$ is positive and maximum, i.e., $\mathbf{X}_j$ is the steepest uphill from $\mathbf{X}_i$. If the above quantity is negative for all $\mathbf{X}_j$ in the neighborhood, $\mathbf{X}_i$ is declared to be a root node of the tree structure. Root nodes are assumed to be close to a mode of the underlying distribution. Clustering is performed in a natural way by following the branches of the structure. The algorithm, called graph theoretical clustering, is described in detail in [5, p. 538]. The hierarchical structure can also be obtained through iterative thresholding [7] or through splitting [6] of the density estimate.

The second algorithm uses the density gradient estimate to define an iterative, hill climbing technique which detects the modes and the valleys in the underlying distribution. The *mean shift* procedure is an adaptive steepest ascent technique that computes the mean shift vector (7) for each data point, translates the kernel by that amount, and repeats the computations till a mode is reached. Convergence properties of the mean shift procedure, generalizations and applications to clustering are discussed in [1]. A variant of the mean shift, called the maximum entropy clustering is presented in [11]. However, clustering through applying the mean shift procedure to each data point cannot be satisfactory in practical applications since the convergence over the low density regions is poor, while high density regions can present plateaus without a clear local maximum.

When the data set is large (over 10000 points) the most important drawback of the two algorithms discussed above is their computational complexity. They require the density or density gradient estimation at each data point which has a complexity of $O(n^2)$ for a set of n data points. The complexity problem is induced by the lack of efficiency of the multidimensional range searching. The performance of the d-dimensional trees used in range searching is rather difficult to predict for random data [13, p. 385].

In the next section we present a probabilistic mean shift type algorithm which takes in account the difficulties mentioned so far, and whose complexity is $O(mn)$, with $m \ll n$.

4 Clustering Algorithm

The steps of the algorithm are described below.

1. Define a sample set obeying distance and density constraints. To reduce the computational load, a set of m points $\mathbf{X}_1 \ldots \mathbf{X}_m$ called the *sample set* is randomly selected from the data. Two constraints are imposed on the points retained in the sample set. The distance between any two neighbors should not be smaller than h, the radius of the sphere $S_h(\mathbf{x})$, and the sample points should not lie in sparsely populated regions. The latter condition is required to avoid convergence problems for the mean shift procedure. A region is sparsely populated whenever the number of points inside the sphere is below a threshold T_1. Note that the distance and density constraints automatically determine the size m of the sample set. The spheres centered on the sample set cover most of the data points.

2. Apply the mean shift procedure to the sample set. A set containing m *cluster center candidates* is defined by the points of convergence of the m mean shift procedures. Note the decrease in computational complexity which is now $O(mn)$, with $m \ll n$, and that the computation of the mean shift vectors is based on the entire data set. Therefore, the quality of the density gradient estimate is not diminished by the use of sampling.

3. Perturb the cluster center candidates and reapply the mean shift procedure. Since a local plateau can prematurely stop the iterations, each cluster center candidate is perturbed by a random vector of small norm and the mean shift procedure is let to converge again.

4. Derive the cluster centers $\mathbf{Y}_1 \ldots \mathbf{Y}_p$ from the cluster center candidates. Any subset of cluster center candidates which are sufficiently close to each other (for any given point in the subset there is at least another point in the subset such that their distance is less than h), defines a *cluster center*. The cluster center is the mean of the cluster center candidates in the subset. Note that $p \leq m$.

5. Validate the cluster centers. Between any two cluster centers $\mathbf{Y}_i$ and $\mathbf{Y}_j$ a significant valley should occur in the underlying density. The existence of the valley is tested for each pair $(\mathbf{Y}_i, \mathbf{Y}_j)$. The sphere $S_h(\mathbf{x})$ is moved with step h along the line defined by $(\mathbf{Y}_i, \mathbf{Y}_j)$ and the number the data points lying in the sphere is counted at each position, i.e., the density is estimated with kernel $S_h(\mathbf{x})$ along the line. Whenever the ratio between $min\left[\hat{f}(\mathbf{X}_i), \hat{f}(\mathbf{X}_j)\right]$ and the minimum density along the line is larger than a threshold T_2, a valley is assumed between $\mathbf{Y}_i$ and $\mathbf{Y}_j$. If no valley was found between $\mathbf{Y}_i$ and $\mathbf{Y}_j$, the cluster center of lower density ($\mathbf{Y}_i$ or $\mathbf{Y}_j$) is removed from the set of cluster centers.

6. Delineate the clusters. At this stage each sample point is associated with a cluster center. To allocate the data points a k-nearest neighbor technique is

employed, i.e., each data point belongs to the cluster defined by the majority of its k-nearest sample points.

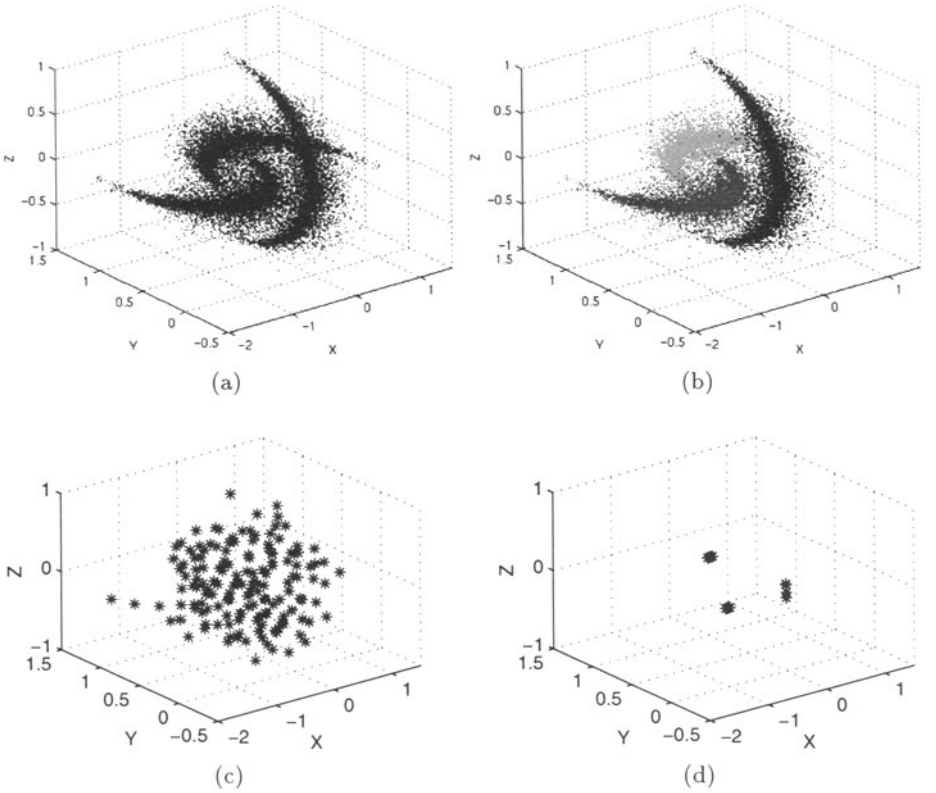


Fig. 1. First experiment. (a) Original data set (32640 points). (b) Cluster delineation (3 clusters). (c) Sample set (167 points). (d) Cluster center candidates.

5 Experimental Results

The clustering algorithm makes use of three parameters: the searching sphere radius h, the threshold T_1 which imposes the density constraint, and the threshold T_2 corresponding to the minimum acceptable peak-valley ratio. All the experimental results described here were obtained with $T_1 = 50$ and $T_2 = 1.2$. Since the experimental data sets had different scales, the sphere radius h has been changed accordingly.

The first example is shown in Figure 1. The data set contained 32640 points with dimension $d = 3$, grouped into 3 non-linearly separable clusters (Figure 1a). Using a radius $h = 0.2$, the obtained sample set had 167 points (Figure 1c) and converged to 3 cluster centers. In Figure 1b the 3 extracted clusters are shown,

having 11050, 10874, and 10716 points, respectively. The algorithm running time was 20 seconds on a standard workstation.

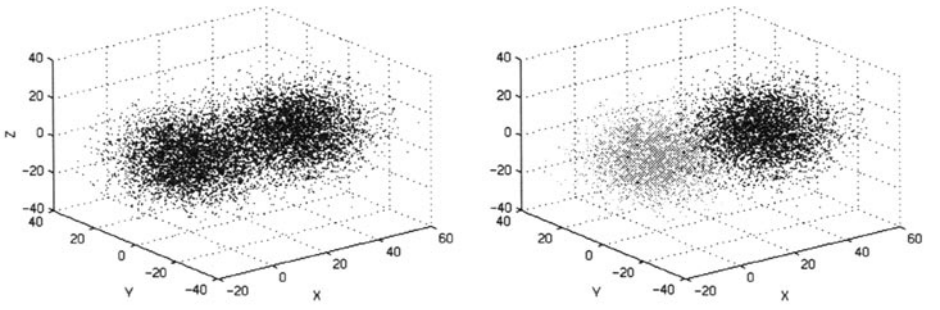


Fig. 2. Second experiment. (a) Original data set (10000 points). (b) Cluster delineation (2 clusters).

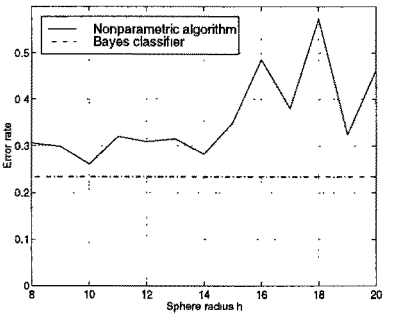


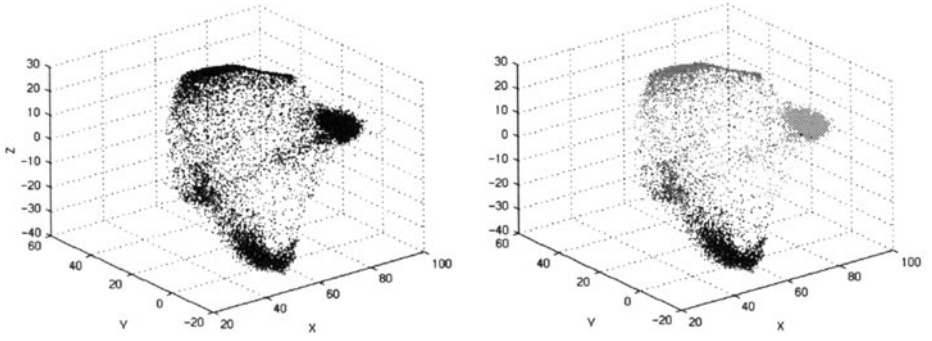
Fig. 3. The error rate of the proposed algorithm for different values of the sphere radius. The dash-dotted line represents the error rate of the Bayes classifier.

A simpler clustering example is shown in Figure 2. The purpose of this experiment was to compare the performance of the nonparametric algorithm with the performance of the classical Bayes classifier. The data set contained 10000 points with dimension $d = 3$ coming from two normal distributions with covariance $10^2\mathbf{I}$ and mean vector $(0,0,0)^T$ and $(40,0,0)^T$, respectively. Figure 2b shows the delineated clusters corresponding to a radius $h = 10$. Using the Bayes classifier the theoretical error rate is 0.234%, due to 234 points that overlap. Figure 3 shows the error rate resulted from our algorithm for sphere radii between 8 and 20. The error rate increases with radius h due to the increase in the boundary delineation error. The straight line in the graph represents the Bayes error rate. The performance of the nonparametric algorithm is very close to that of the Bayes classifier, in spite of no a priori knowledge being used in the nonparametric case.

The third experiment shows the application of our algorithm to the segmentation of the color image in Figure 4a. Clustering is performed in the perceptually uniform $L^*u^*v^*$ color space, each delineated cluster corresponding to homogeneous regions in the image. The color space (Figure 5a) contained 14826 points, and four clusters have been extracted by using a radius of $h = 10$. Note the irregular boundaries of the clusters in Figure 5b. The clustering quality can be assessed by observing the segmented image in Figure 4b, where spatial constraints have been used to remove small regions [2].

(a) (b)

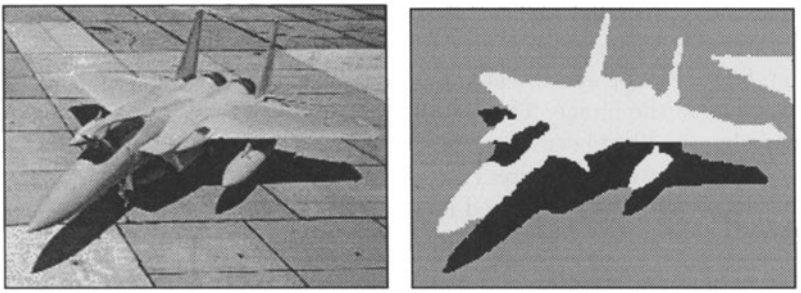
Fig. 4. Third experiment. (a) Original image. (b) Segmented image using nonparametric clustering.



(a) (b)

Fig. 5. Third experiment. (a) Original data set (14826 color points). (b) Cluster delineation (4 clusters).

A second color segmentation example is presented in Figure 4. Using the same radius of $h = 10$, the algorithm extracted three color clusters.



(a) (b)

Fig. 6. Fourth experiment. (a) Original image. (b) Segmented image using nonparametric clustering.

References

1. Y. Cheng, "Mean Shift, Mode Seeking, and Clustering", *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 17, 790–799, 1995.
2. D. Comaniciu, P. Meer, "Robust Analysis of Feature Spaces: Color Image Segmentation", Proc. *IEEE Conf. on Comp. Vis. and Pattern Recognition*, Puerto Rico, 750-755, 1997.
3. J.H. Friedman, J.L. Bentley, R.A. Finkel, "An Algorithm for Finding Best Matches in Logarithmic Expected Time", *ACM Trans. Mathematical Software*, vol. 3, 209-226, 1977.
4. K. Fukunaga, L.D. Hostetler, "The Estimation of the Gradient of a Density Function, with Applications in Pattern Recognition", *IEEE Trans. Info. Theory*, vol. IT-21, 32-40, 1975.
5. K. Fukunaga, *Introduction to Statistical Pattern Recognition*, Boston: Academic Press, 1990.
6. J.A. Garcia, J.F. Valdivia, F.J. Cortijo, and R. Molina, "A Dynamic Approach for Clustering Data", *Signal Processing*, vol. 44, 181-196, 1995.
7. M. Herbin, N. Bonnet, P. Vautrot, "A Clustering Method Based on the Estimation of the Probability Density Function and on the Skeleton by Influence Zones", *Pattern Recognition Letters*, vol. 17, 1141-1150, 1996.
8. A.K. Jain, R.C. Dubes, *Algorithms for Clustering Data*, Englewood Cliff, NJ: Prentice Hall, 1988.
9. S.A. Nene, S.K. Nayar, "A Simple Algorithm for Nearest Neighbor Search in High Dimensions", *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 19, 989-1003, 1997.
10. K. Popat, R.W. Picard, "Cluster-Based Probability Model and Its Application to Image and Texture Processing", *IEEE Trans. Image Process.*, vol. 6, no. 2, 268-284, 1997.
11. K. Rose, E. Gurewitz, G.C. Fox, "Constrained Clustering as an Optimization Method", *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 15, 785-794, 1993.
12. D.W. Scott, *Multivariate Density Estimation*, New York: Wiley, 1992.
13. R. Sedgewick, *Algorithms in C++*, New York: Addison-Wesley, 1992.
14. B.W. Silverman, *Density Estimation for Statistics and Data Analysis*, New York: Chapman and Hall, 1986.
15. G.R. Terrell and D.W. Scott, "Variable Density Estimation", *The Annals of Statistics*, vol. 20, 1236-1265, 1992.