The Filtered Mode Tree

David J. Marchette[1] and Edward J. Wegman[2]

The mode tree is a useful tool for visualizing the modal structure of a density. Locations of modes of the density are plotted as a function of the bandwidth used in the kernel estimate of the density. Since the mode tree uses a single bandwidth in the kernel estimator, it exhibits all the drawbacks that a single bandwidth kernel estimator has, particularly for densities with large tails or differences in the scales of the modes. A modification is presented which uses the filtered kernel estimator, a version of the kernel estimator which uses a small number of bandwidths. The two mode trees are compared on some synthetic data, and on a data set from DNA flow cytometry.

Key Words: Bump hunting; Graphical methods; Kernel density estimation; Filtered kernel density estimation; Mode estimation; Multimodality.

1. Introduction

One of the first things one wants to know about a data set is how its distribution. For most data, unless *a priori* information leads us to a particular parametric family, a nonparametric estimate of the density (often a histogram) is constructed. The next question, given a rough idea of the shape of the density, is how much of the structure represents the underlying density and how much is an artifact. For example, one wants to know how many modes there are, and where they are.

This is difficult to assess from any single estimate, since the variability of the data and the uncertainties inherent in nonparametric density estimation will sometimes hide modes due to

[1] Naval Surface Weapons Center, Code B10, Dahlgren Va 22448

[2] Professor, Computational Sciences and Informatics, George Mason University, Fairfax Va 22030

oversmoothing and sometimes exhibit spurious modes due to undersmoothing. Often one makes a series of plots using different assumptions and methods and then decides from the plots what structure is supported by the data. Thus, good visualization tools are essential for density-based data analysis.

Silverman (1981), gave a method for assessing the mode structure using kernel estimators, and Minnotte and Scott (1993) used this approach to design a visualization technique to aid in the assessment of these features. This approach will be discussed and an extension of it will be described which makes the method more easily interpreted in certain situations.

Kernel estimators are well known and used extensively in nonparametric density estimation and regression. Good introductory references are Silverman (1986) and Scott (1992). Given iid data $x_1,...,x_n$, we construct the kernel estimator for the density as:

$$\hat{f}(x) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x-x_i}{h}\right) \tag{1.1}$$

where K is the kernel, usually a density (in fact, in practice K is often the normal density). The bandwidth, h, determines the amount of smoothing of the estimator and, hence, determines the number of modes in the estimate. Silverman (1981) showed that the number of zeros of all derivatives of the estimate (1.1) is monotone decreasing in h for the normal kernel. This fact will be exploited to give a method of visualizing the modal structure of the density.

2. The Mode Tree

The kernel estimator thus gives a method for investigating the number of modes of a density: plot the density for a range of bandwidths h. As h decreases, new modes will appear and old modes will remain (if the kernel is normal). Unfortunately, it is difficult to visualize all these different plots simultaneously and it is with this in mind that the mode tree was invented.

The mode tree of Minnotte and Scott (1993) is a very simple and powerful idea: plot the modes of the kernel estimator as the bandwidth h varies. As h decreases, new modes split off from old modes, and this simple plot encapsulates all this information. More information can be added to the basic mode tree, such as position of the antimodes and the magnitude of the density at the mode as indicated in Minnotte and Scott's paper.

Figures 1 and 2 illustrate the mode tree and point out a problem with the mode tree that our new technique, to be described below, is designed to address. We have drawn 500 data points from the mixture distribution .4 N(-5,.1) + .4 N(.5,.1) + .2 N(0,10) illustrated in Figure 1, with the solid curve indicating the true distribution. Note that the histogram is oversmoothed and does not pick out the two modes of the density. This oversmoothing is inevitable in a density with long tails, such as this one, if one is to avoid "spiking" in the tails, i.e. as long as a single bin width histogram is used.

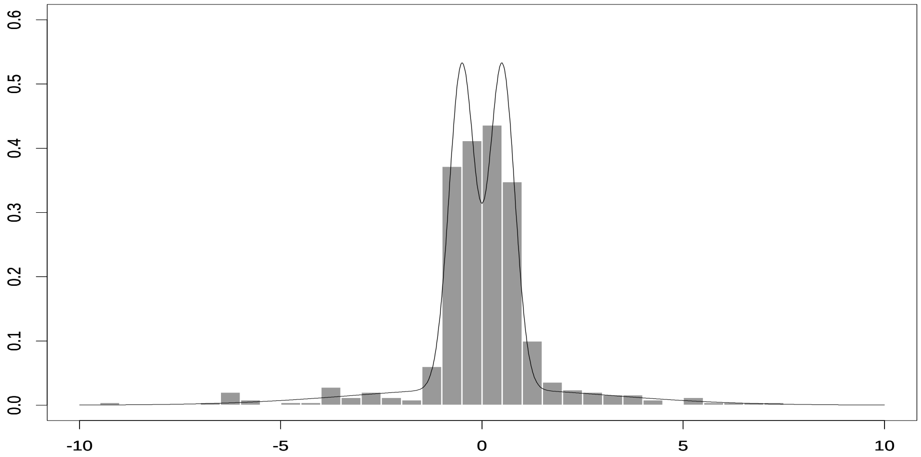

Figure 1: Histogram of 500 data points drawn from .4 N(-0.5,.1) + .4 N(0.5,.1) + .2 N(0,10). True density is shown as a solid curve.

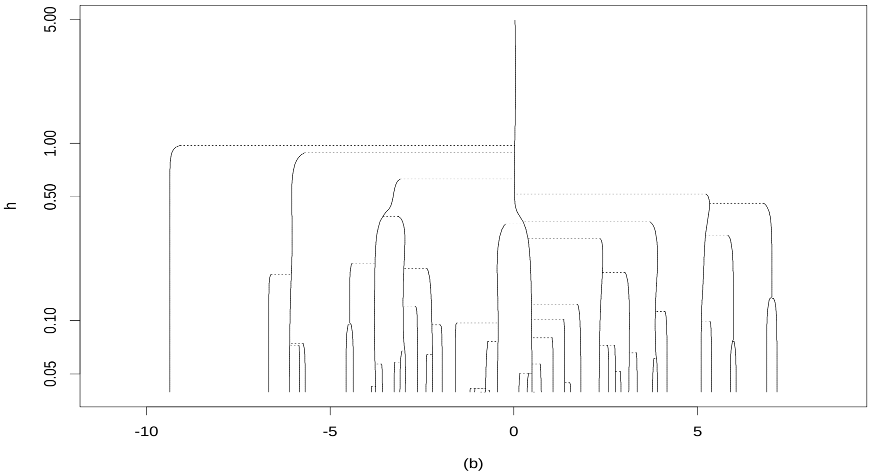

Figure 2: Mode tree for the data in Figure 1.

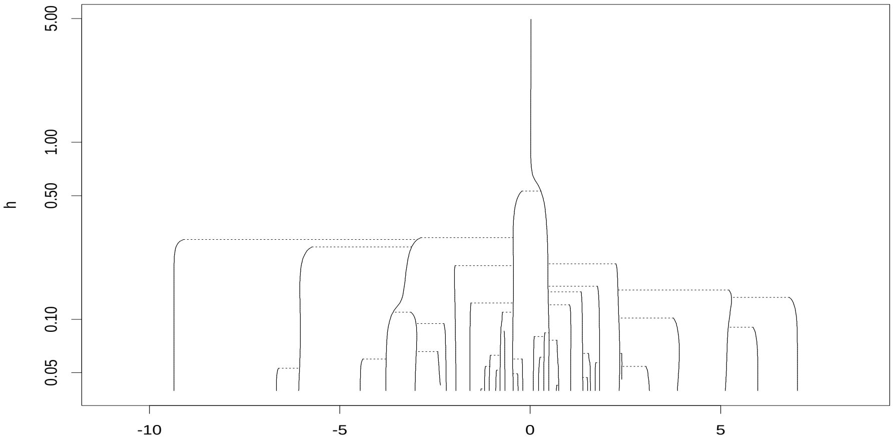

Figure 3: “Rescaled” mode tree for the data in Figure 1.

This data could be thought of as an outlier contaminated data set, where the two modes in the center are what are of interest and the long tails are due to contamination. We want to know how many modes there are in this data. Figure 2 shows the mode tree of Minnotte and Scott applied to this data. Note that one must mentally ignore the outliers in order to see the central modes. The problem is that the single bandwidth kernel estimator must undersmooth the tails and hence generate a large number of modes in the tails before it can detect the extra structure in the middle. The histogram in Figure 1 shows clearly that the tails are relatively long and that this will be a problem. What we propose is a modification to the kernel estimator to incorporate this “multiscale” information.

Looking at the histogram, one might conservatively hypothesize that the data is a two component mixture (outlier model) and fit this mixture to the data. In all the mixture fits we discuss, the EM algorithm (see for example Titterington et al (1985)) is used. The mixture fit is approximately 0.8 N(0.03,0.4) + 0.2 N(-0.1,11.7). Imagine using the bandwidths appropriate to the separate components of this mixture to rescale the mode tree. Deferring the details of how this is to be accomplished for a moment, consider Figure 3. Now we clearly see the two modes, and the “spurious” modes have all been de-weighted relative to the true modes. This graphic seems to be much more easily interpreted than the one in Figure 2.

Consider the two mode trees together. There appears to be a correspondence between each new mode in each tree, up to a point. In fact, the second tree appears to be a smooth distortion of the first, as if we had pushed the mode up and the tails down. In a sense, this is precisely what has happened. We now describe how this is accomplished.

3. The Filtered Kernel Estimator

We propose a modification of the kernel estimator which allows a small number of bandwidths to be used and gives a mechanism both for choosing these bandwidths and limiting their scope along the support of the density. This allows us to weight the different modes according to their estimated variance as illustrated in Figure 1.

Of course, many different methods have been proposed for constructing kernel estimators with separate bandwidths for each kernel (see Silverman (1985), Scott (1994), and Wand and Jones (1995) for discussions of some of these). One of these methods could be used in place of the kernel estimator in the mode tree. However, for data analysis and exploration, it would be desirable to have just a few bandwidths (for example: one for the tails and one for the mode) that can be adjusted independently to examine the effect this has on the estimator. Thus we seek a method for providing this kind of flexibility.

The technique is described in more detail and generality in Marchette et al (1994). We will consider a specific case of the estimator here. In order to motivate this estimator, consider a normal outlier density: p N(0,1) + (1-p) N(0,σ^2). As σ becomes large, the single bandwidth kernel estimator must either oversmooth the mode or undersmooth the tails. One would like to use a large h in the tails and a smaller one near the mode. In some sense, we would like to choose the one appropriate to the N(0,σ^2) term in the tails, and the one appropriate to the N(0,1) term near the mode. With this in mind, we define the filtered kernel estimator.

Let

$$f(x) = \sum_{j=1}^{m} \pi_j f_j(x) \qquad (3.1)$$

be given (usually a mixture fit to the data), with

$$f_j(x) = N\left(\mu_j, \sigma_j^2\right). \qquad (3.2)$$

With the above definition, we define the filter functions to be the posteriors of the mixture components:

$$\rho_j(x) = \frac{\pi_j f_j(x)}{f(x)}. \tag{3.3}$$

The filtered kernel estimator is then:

$$\hat{\alpha}(x) = \frac{1}{nh}\sum_{i=1}^{n}\sum_{j=1}^{m}\rho_j(x_i)\,N\left(x, x_i, (h\sigma_j)^2\right). \tag{3.4}$$

In regions where one component of the mixture dominates, the inner sum reduces to approximately the single kernel with a bandwidth scaled by the component's variance. This allows us to have different effective bandwidths (one for each mixture components) in different regions of the support, but without the added burden of choosing these extra bandwidths.

Of course, that last sentence is not quite true: we still have to choose the mixture model (3.1). Quite a bit of work has been done in this arena (see for example Titterington et al 1985), but it is by no means a solved problem. Experience has shown that the method is fairly robust to choices of this mixture model and, moreover it is not required that the data actually be selected according to a mixture model of this type. In fact, if it is known that the data is a mixture of a known number of components, and if it is also known that our mixture estimate is a maximum likelihood solution (rather big "ifs"), then the filtered kernel estimator is unnecessary.

It should be pointed out that the purpose of the filtering mixture is not really to get a good estimate of the density, but rather to get the basic shape of the density without all the details. It is the purpose of the mode tree to explore the details. Thus, as in Figure 1, we wish to obtain a conservative density estimate which will give a rough estimate of the local smoothness of the density.

As can be seen in (3.4) we do not have complete independence of the bandwidths in all regions. In regions of high overlap of the mixture components, we have a mixture of kernels at each point, the mixture proportions being the posteriors of the components of the mixture. This is probably what one wants, in most situations, but one could just as easily replace the ρ's in (3.4) with characteristic functions for different regions of the support of the density, if one felt this was necessary. We will not pursue these ideas here.

Our experience has been that one can obtain quite good estimators with conservative estimates of the mixture density. Figure 4 is an example of what is meant here. The filtered kernel estimator is drawn as a solid line while the filtering mixture is dotted. The dashed curve represents the true density. This is the same data as in Figure 1. Even though the filtering mixture has under estimated the number of components of the density, we still see a good estimate of the density in the filtered kernel estimator. As with all kernel estimators, this one depends on the bandwidth h, and in this case h was chosen to be "optimal" in the mean integrated squared error (MISE) sense, under the assumption that the filtering mixture was the correct density. Better estimates could be obtained, since the filtering mixture is not correct, however this illustrates how one might use the

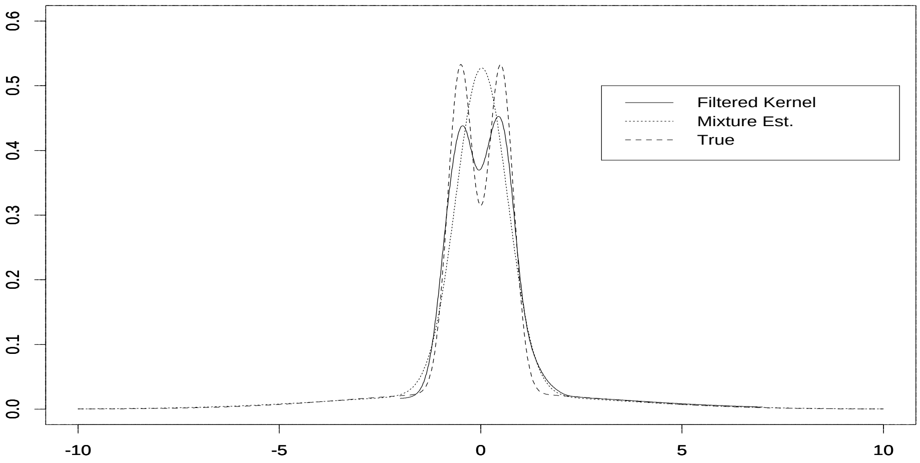

Figure 4: The filtered kernel density estimator compared with the true distribution and the filtering mixture.

algorithm in practice.

The filtered kernel estimator can be made more general, and we briefly consider one possible generalization. Consider having a separate bandwidth for each component that is completely user specified. With this we can hand-craft the estimator near the modes, to bring out the submodal information of interest. The formula for this estimator is:

$$\hat{\alpha}(x) = \frac{1}{nh}\sum_{i=1}^{n}\sum_{j=1}^{m}\rho_j(x_i)\,N\left(x, x_i, (hk_j)^2\right) \tag{3.5}$$

The k_j are now explicitly separated from the variances of the components, and can be adjusted independently until a reasonable fit to the data is achieved, something that is difficult to do with the fully adaptive kernel estimator. In this case, once the k_j's have been fixed, h is varied just as in the standard mode tree. This is of value when the filtering mixture is not thought to be a good estimate of the data in its own right, or when exploratory data analysis is the intent. Other generalizations, including generalizing the filter functions, as mentioned briefly above, are possible, but will not be considered here. We will use the version in (3.4) throughout the remainder of this paper.

The estimator in (3.5) is one of the reasons we choose not to go the route of the fully adaptive kernel estimator. It seems an advantage to have a small number of “local” tuning factors which can be adjusted to give a better estimate of the data, given the limitations of our pilot estimator (called the filtering mixture in the case of the filtered kernel estimator).

4. The Filtered Mode Tree

The modified mode tree is now simple to define: it is the standard mode tree with the kernel estimator replaced by the filtered kernel estimator. This allows the local rescaling that we desired, with only a “small” cost: fitting the filtering mixture.

The example above (Figure 1) shows the basic idea. As in the outlier model discussed in section 3, we wanted different bandwidths in different regions, but we were not yet willing to make the commitment to more than two components. This also illustrates an important point: the mixture fit need not be aimed at producing a good model of the density. The point of the mode tree is to examine the modal structure of the density. One should be careful not to posit a particular structure in the mixture, and then look for evidence of this structure in the mode tree, unless one is willing to do this for all the different possible structures.

Figure 3 gives another application of the filtered mode tree. We have drawn 500 data points from another mixture distribution (.3 N(-.35,.1) + .3 N(.35,.1) + .2 N(1.8,.8) + .2 N(4,.8)) illustrated in Figure 5, with the solid curve indicating the true distribution. Considering the figure for a moment, one natural intuition about this density is that it has two main groupings (at different scales) a left one and a right one, each of which is “split” into two “minor” modes. While one could have a different view, it is clear that the left most modes are of a different character than the right most ones and this seems a natural grouping.

The mode tree on this data is depicted in Figure 6. Note that it does pick out the two

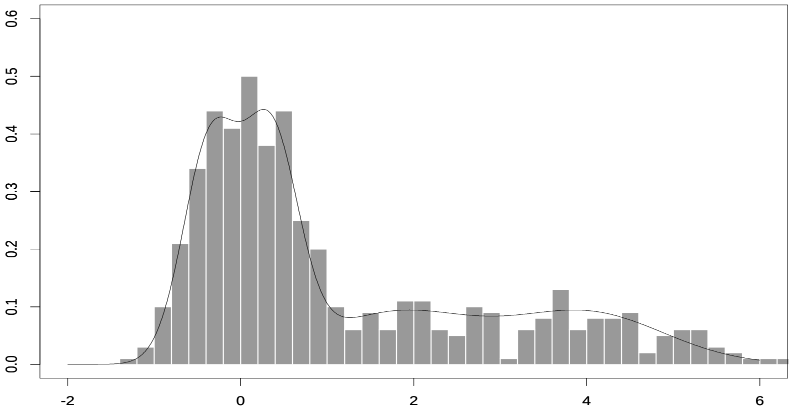

Figure 5: A histogram, overlayed with the true distribution, of data from a 4 component mixture: .3 N(-.35,.1) + .3 N(.35,.1) + .2 N(1.8,.8) + .2 N(4,.8).

modes on the right, but it is difficult to see from this plot whether there is evidence for more than one mode on the left. The reason is that the bandwidth appropriate to detecting the modes on the left greatly undersmooths the right half of the density. The true modes on the left are at the same smoothing level as the spurious modes on the right. Thus, in order to interpret the mode tree for the leftmost modes, one must focus attention on the appropriate area and mentally down-weight the “spurious” modes. Of course, this assumes one has determined which modes are “spurious”. This requires more information than is in this plot.

Note also that four of the modes at the right appear at approximately the same h value, making it difficult, from the picture, to decide what the mode structure should be. Finally, the figure shows the mode at 2 splitting off from the left mode, which is contrary to our (admittedly biased) intuition. The problem once again is that the kernel estimator is using the same bandwidth on the broad right as it is on the narrow left. In some sense, one would always want to use a smaller bandwidth on the left than on the right. Once again a two component mixture is fit to the data. This mixture is approximately 0.66 N(0.1,0.3) + 0.34 N(3.1,2.1). Using this as the filtering mixture, Figure 7 shows the new, rescaled mode tree. Note that the two modes from the left appear at a larger smoothing parameter, h. Note also that we still have “spurious” modes. Thus, no information that was in the original mode tree is lost, it is just reformatted. In a sense we have “leveled the playing field” of the two major modes so that they can be analyzed simultaneously, without extra mental effort.

A nice feature of this example is that the mode at 2 has been linked as splitting off the right-most mode in Figure 7 rather than the left-most mode as in Figure 6, which seems more appropriate for this density.

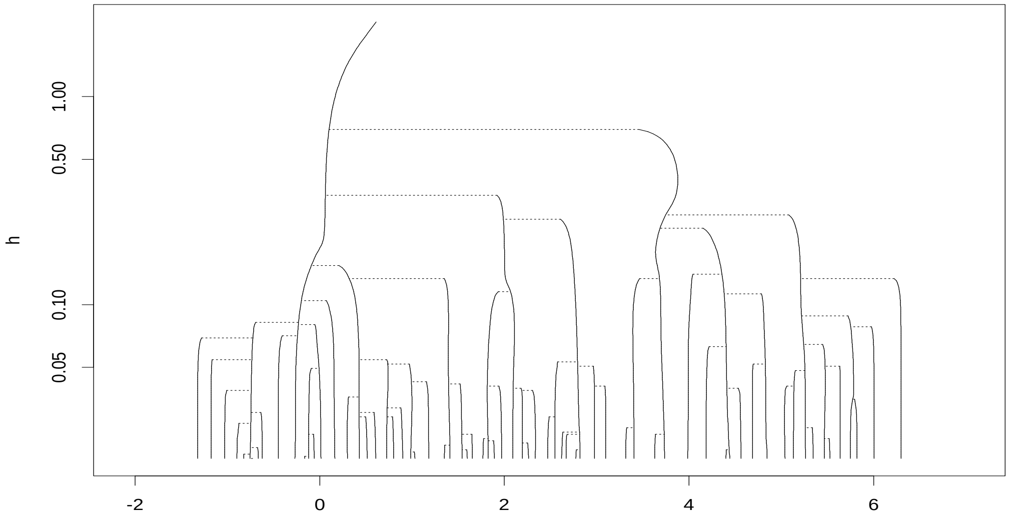

Figure 6: The mode tree for the data in Figure 5.

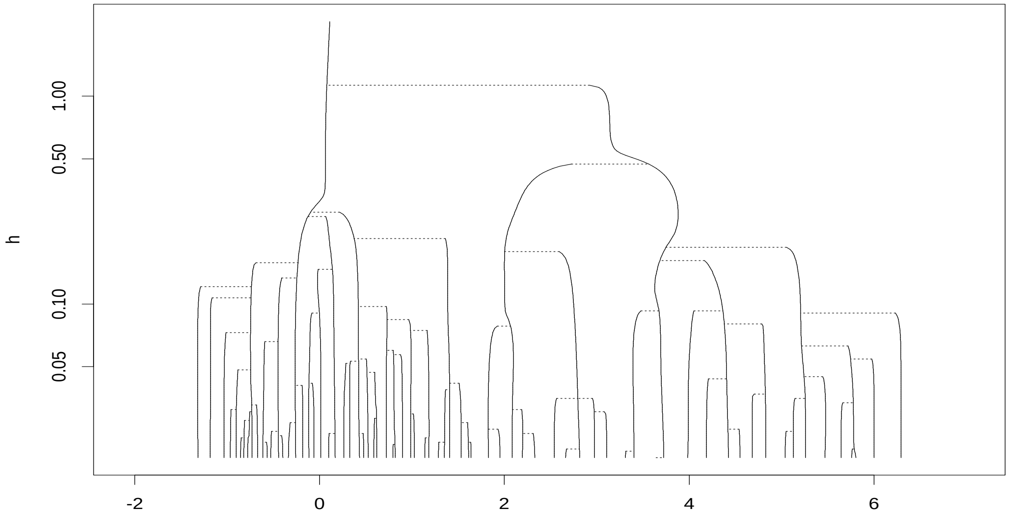

Figure 7: The filtered mode tree for the data in Figure 5.

Looking at the histogram in Figure 5, it could be argued that there are three components, one on the left and two on the right. So we now fit a 3 component mixture to the data. The filtered mode tree is shown in Figure 8. This seems almost identical to the Figure 6, the standard mode

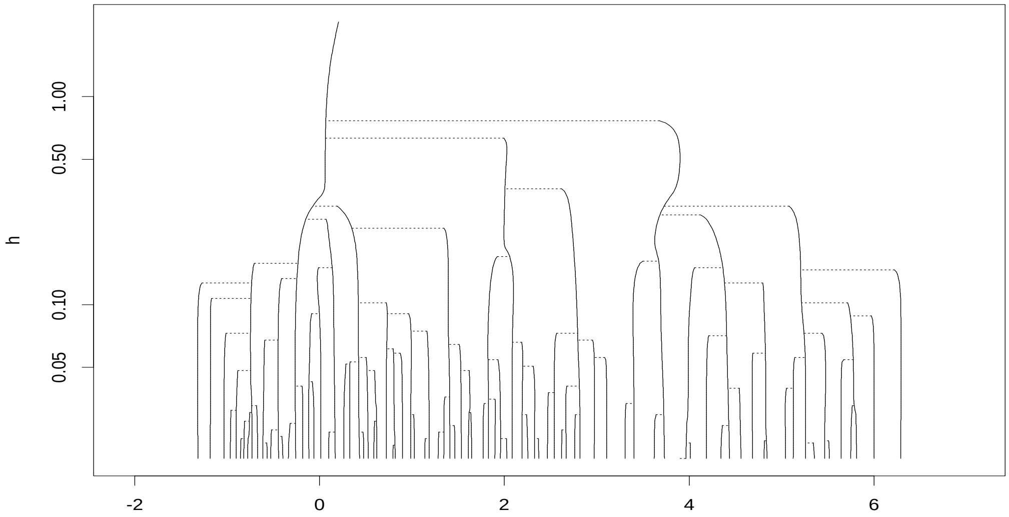

Figure 8: The mode tree for the data in Figure 5 using a three component mixture for the filter.

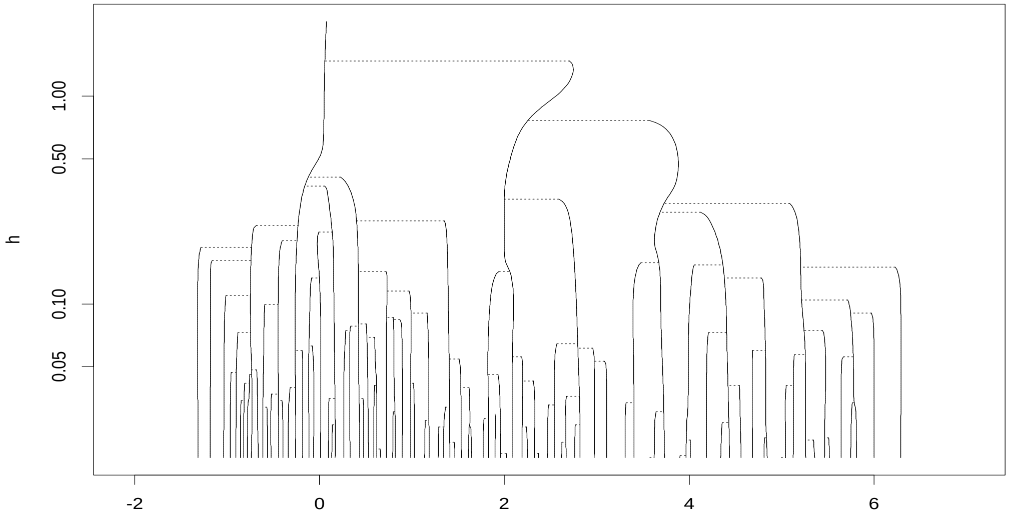

Figure 9: The mode tree for the data in Figure 5 using a four component mixture for the filter.

tree. What has happened? Recall that the filtered mode tree acts by scaling the kernel estimator differently in different regions, as suggested by the filtering mixture. If the components of the filtering mixture have roughly equal variance, the filtered kernel estimator reduces to the standard kernel estimator, and the mode trees are the same (up to a possible rescaling of the y-axis). In this case the fitted mixture is approximately .67 N(0,.3) + .14 N(2,.4) + .19 N(4,.8). Thus, if the filtering mixture has similar variances for all the components, no relative scaling is done and the filtered mode tree reduces to the standard mode tree, since the filtered kernel estimator reduces to the standard kernel estimator in this case.

Figure 9 shows what happens when we take the next step and fit a 4 component mixture to the data. The estimated mixture is approximately .3 N(-.3,.12) + .3 N(.4,.13) + .2 N(1.9,.63) + .2 N(4.2,.78). We have a slightly larger range in the variances, and the mode tree in Figure 9 looks closer to the one in Figure 7 and closer to our understanding of the data.

This example has pointed out that the gain in the filtered mode tree is fundamentally tied to the difference in variances of the modes, as represented by the filtering mixture. Thus if the filtering mixture does not have different variances in the different components the filtered mode tree gives the same result as the standard mode tree. Note also that if the filtering mixture has variances which are not representative of the underlying structure, the filtered mode tree will give misleading results.

5. Application

We now turn to a real data set. As discussed above, we are interested in data sets for which the filtered mode tree can provide an improvement over the standard mode tree. For this we need data with modes of distinctly different sizes, distributions with long tails, etc.

The data we consider is immunocytometry data. A flow cytometer, for the purposes of this work, counts the amount of DNA in cells. Normal cells in an organism all have essentially the same amount of DNA, except those that are preparing to divide, and those which have undergone meiosis prior to sexual reproduction. Abnormal cells, those which have undergone some kind of genetic change (such as cancer cells) may in many situations have a different amount of DNA from the norm for the organism. Thus by measuring the amount of DNA in a sample, one can in principle determine if there are abnormal cells present.

The data consists of 20,000 counts, depicted in Figure 10. The usual method of analysis is to histogram the data, as is done here, and measure the modes from the histogram. The fundamental questions in flow cytometry, then, are how many modes are there, where are they, and how big are they? The primary regions of interest in this data are the two main clumps of data, around 200 and 400. The overt structure of this data in these regions is apparent. The question is, is there some substructure to these modes?

The standard mode tree is presented in Figure 11. Note that the outliers and noise between the modes has hopelessly complicated the picture. One needs to compare the mode tree picture

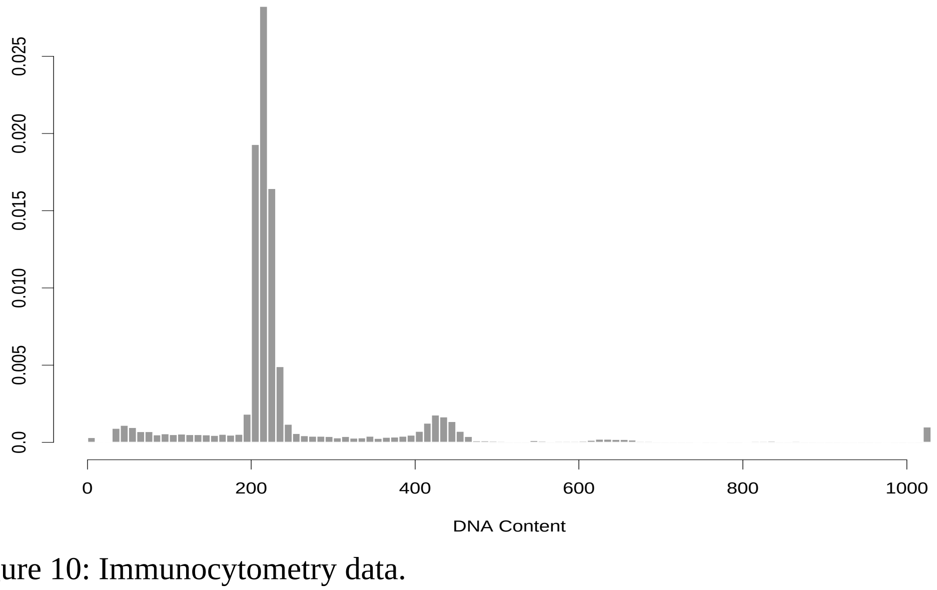

Figure 10: Immunocytometry data.

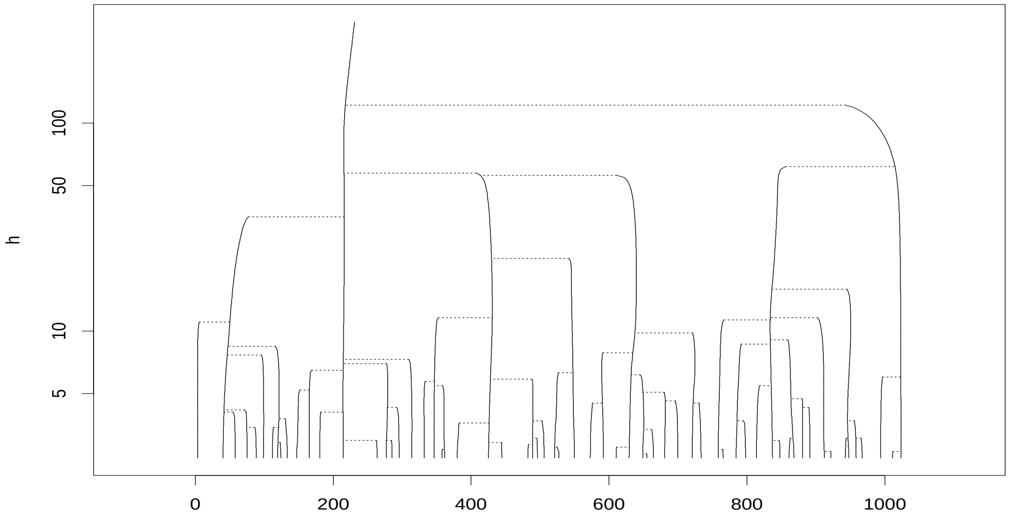

Figure 11: Standard mode tree for the immunocytometry data.

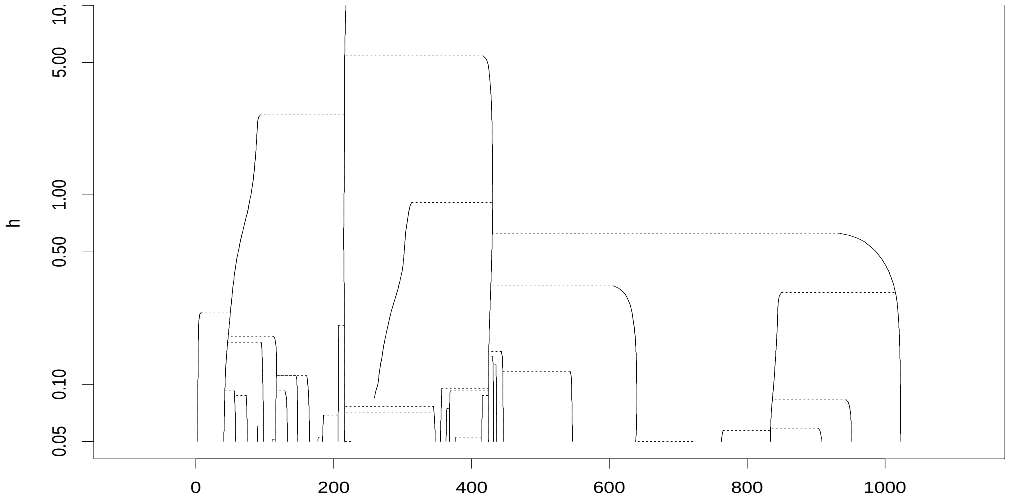

Figure 12: Filtered mode tree for the immunocytometry data.

with the histogram in order to determine the structure in the regions where one is really interested in the modal structure. One solution to this is to first remove all data not from the apparent modes. The filtered kernel provides an alternative to this approach.

We now fit a 5 component mixture to the data. The filtered mode tree is shown in Figure 12. Now we can clearly see the main modes, and the extra structure allowed by smaller bandwidths near the modes allows us to investigate the possibility that the modes indicated by the histogram might split into multiple modes. In order to get this amount of detail near the modes, the standard histogram would have to increase the range of bandwidths used, which would greatly increase the amount of noise from the tails and the between-mode region.

The filtered mode tree clearly indicates the two main modes of interest, and indicates a small number of submodes that might be worth investigating. Consider particularly the two modes at approximately 207 and 216. The standard mode tree becomes hopelessly cluttered before the second mode becomes apparant, which shows clearly one of the strengths of the filtered mode tree. However, this figure also indicates a problem with the filtered mode tree which we did not see in the mode tree. Recall that for normal kernels the number of modes is monotonic in h. The filtered kernel is effectively using a mixture of normals for its kernel (although it is potentially a different mixture for each kernel) and so monotonicity is not guaranteed. Thus we see the mode detected around 300 at about h=1 disappearing as h drops below 0.1. Whether this is a problem sufficient to negate the other positive aspects of the filtered mode tree is a matter of taste. We feel it is not. In regions where a single component of the filtering mixture dominates, the filtered kernel estimator is using essentially normal kernels, and so this would be expected to be less of a problem in these regions. Where the trouble comes in is where there is heavy overlap of the filtering mixture components.

A question now arises as to whether the second mode near 200 indicated in the filtered mode tree is real or not. The only practical way to investigate this is to parse the data into subsets. Minnotte and Scott (1993) use the bandwidths indicated in the mode tree to test. As indicated in

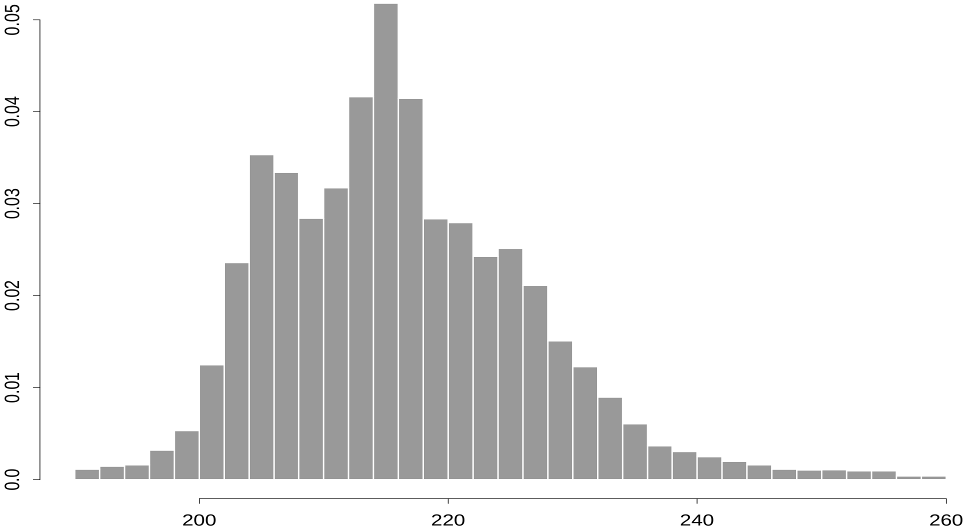

Figure 13: Histogram of immunocytometry data restricted to the range [190,260].

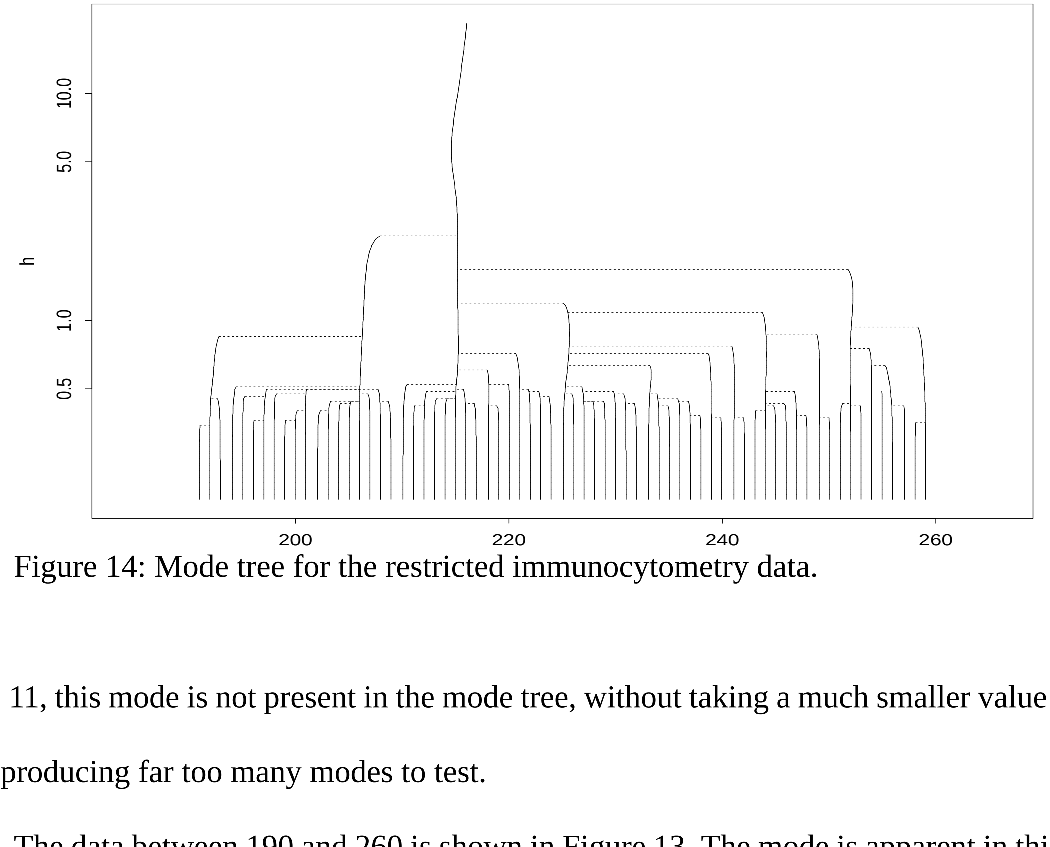

Figure 14: Mode tree for the restricted immunocytometry data.

Figure 11, this mode is not present in the mode tree, without taking a much smaller value of h, and hence producing far too many modes to test.

The data between 190 and 260 is shown in Figure 13. The mode is apparent in this picture, and Figure 14 shows the standard mode tree on this restricted data. Note that the second mode now shows up clearly and we can do a hypothesis test (as described in Minnotte and Scott, (1993), see also Efron and Tibshirani (1993), pp 227-232) to see if this data is unimodal or bimodal. Recall that the standard mode tree on the entire data set was unable to pick out this second mode. The filtered mode tree was able to look deeper down the tree in regions which warranted the extra depth. This indicates the regions which warrant further investigation, and the standard mode tree can be used to obtain the bandwidths at which to perform the hypothesis test.

The estimated *p* value for this second mode is .08, which gives us some confidence that it is in fact a true mode of the distribution. This example illustrates the way the two different mode trees can be used in concert to produce improved data analysis. The detection of multiple peaks in this region, refered to as aneuploid G1 peaks in the cytometry literature, is of great interest to the pathologist (Schuette, et al. (1983).

6. Discussion

The filtered mode tree gives an alternative to the standard mode tree that allows selective rescaling of the tree in regions of different variance. This can allow the user to better pick out the underlying structure of the data. The filtered mode tree reduces to the standard mode tree when the variances of the normal components are the same. It also elicits structure that the standard mode tree might miss by virtue of having smaller bandwidths in regions in which the scale of the data is small, thus improving the chance that fine detail will be elicited where it is warranted.

As always, when one adds parameters to be tweaked one invites abuse. It is always possible to use a filtering mixture tailored to finding spurious structure. For instance, if one used small variance components in the tails, perhaps fit to outliers, these regions would be given undesired weight in the mode tree, leading to incorrect analysis. It is important to justify the mixture approximation and to check for pathologies such as component variances being driven to zero, or components in regions with little supporting data. It is advisable to be conservative in the choice of mixtures used, unless very good *a priori* information is available.

The filtered mode tree is an alternative to the mode tree, but is not meant to completely supplant it. The mode tree has desirable properties. However, it can be extremely hard to read in certain situations, for example with data whose distribution has long tails, and it is these situations that the filtered mode tree is designed to address. The use of the filtered mode tree to determine which regions are of interest, followed by the standard mode tree in these regions, and combined with hypothesis testing on those regions can give a powerful set of tools for determining the modal structure of the data.

The filtered mode tree is certainly not the only approach. One could use any of a number of variable bandwidth kernel estimators in place of the filtered kernel, however the ease of application and the small number of parameters which need adjusting makes the filtered kernel estimator, and hence the filtered mode tree, quite useful in many situations.

Acknowledgments

This work was performed in part at the Naval Surface Warfare Center. The work of D. Marchette was funded in part by the NSWC Independent Research Program. The work of E. Wegman was funded by the Army Research Office under grant _______ and by the office of Naval Research under grant _____. The authors would like to thank Carey Priebe, Jeff Solka and Rich Lorey for many useful comments and discussions. We would also like to thank Michael Lock and Becton Dickinson Immunocytometry Systems for providing us with the immunocytometry data.

References

Efron, B. and Tibshirani, R. J., 1993, *An Introduction to the Bootstrap*, New York: Chapman and Hall.

Marchette, D. J., Priebe, C. E., Rogers G. W. and Solka J. L., 1994, “The Filtered Kernel Estimator”, Center for Computational Statistics, George Mason University, Tech Rpt No 104, October 1994.

Minnotte, M. C., and Scott, D. W., 1993, “The Mode Tree: A Tool for Visualization of Nonparametric Density Features”, J. Comp. and Graph. Statist., 2, 51-68.

Schuette, W. H., Shackney, S. E., MacCollum, M. A., and Smith, C. A., 1983, “High Resolution Method for the Analysis of DNA Histograms that is Suitable for the Detection of Multiple Aneuploid G_1 Peaks in Clinical Samples”, Cytometry, 3, 376-286.

Scott, D. W., 1992, *Multivariate Density Estimation: Theory, Practice, and Visualization*, New York: John Wiley.

Silverman, B. W., 1981, “Using Kernel Density Estimates to investigate Multimodality”, J. R. Statist. Soc. B, 43, 97-99.

Silverman, B. W., 1986, *Density Estimation for Statistics and Data Analysis*, London: Chapman & Hall.

Titterington, D. M., Smith, A. F. M., and Makov, U. E., 1985, *Statistical Analysis of Finite Mixture Distributions*, Chichester: John Wiley.

Wand, M. P. and Jones, M. C., 1995, *Kernel Smoothing*, London: Chapman & Hall.