ERASMUS UNIVERSITEIT ROTTERDAM

Highest Density Regions

For Uni- and Bivariate Densities

Name: Fadallah, A.
Student number: 309040
E-mail: a.fadallah@inventreal.com
Study: Econometrics and Operations Research
Thesis: Bachelor (FEB 23100)
Rotterdam, 6 July 2011

Table of contents:

1. Introduction

This paper presents a novel method for determining a bivariate highest density region (HDR) estimate. Highest density regions are often the most appropriate subset to use to summarize a region, and are capable of exposing the most striking features of the data than most alternative methods (Hyndman, 1996). However, there has been very little research so far on HDR, while bivariate boxplots and contours of estimates of densities have been studied extensively. Bivariate HDR's have also been treated, but these studies have all been based on either kernel estimates of densities or boxplots. In this paper we present a completely new approach for determining a bivariate highest density region (HDR) estimate.

In areas such as finance and marketing, decision making is often supported by the use of software packages that allow extensive use of the available quantitative information. This information, whether collected on regular basis, or every split second through real-time information systems, can be usefully exploited by *summarizing the relevant data information* by means of an econometric model. Such an econometric model allows the business manager to "understand the relation between economic and business variables and also to analyze the possible effects of decisions" through forecasting (Heij et al., 2003). However, if the subsequent decisions by the business manager are to be based on the analysis performed through forecasting, the forecast itself would need to be analyzed for its accuracy. Highest density regions are well suited for this purpose. For example, a common way to summarize forecast accuracy is through the use of forecast regions. However, as Hyndman (1995) notes, forecast regions "usually consist of an interval symmetric about the mean of the forecast", and that this is especially the case "when the forecast densities are normal, as with ARIMA models" (see Brockwell and Davis, 1991). This means that "symmetric intervals may not be appropriate forecast regions when the forecast density is not symmetric and unimodal" and he explains this by pointing to the problems with obtaining the forecast regions when the econometric "time series models, such as those which are non-linear or have non-normal error terms" has forecast densities that are asymmetric or multimodal. Thus, Hyndman argues, "forecast regions symmetric around the mean can be quite misleading", and as a solution he suggests that "forecast regions need to be constructed so as to convey the shape of the forecast density."

Hyndman further argues that highest-density regions (HDR) are a "more effective summary of the forecast distribution than other common forecast region" because of its flexibility "to convey both multimodularity and asymmetry in the forecast density". He outlines a method for estimating HDR's and other forecast regions, and discusses the graphical representation for this purpose in his article. While our approach is different from Hyndman, it is not farfetched to think that our method may be used for the same purpose for bivariate forecast regions.

One of the most distinctive property of HDR's is that of all possible regions of probability coverage$1-\alpha$, the HDR has the smallest region possible in the sample space. "Smallest" mean with respect to some simple measure such as the usual Lebesgue measure[1]; in the one-dimensional continuous case that would be the shortest interval, and in the two-dimensional case that would be the smallest area of the surface. In Bayesian analysis, we find a similar approach, except that in

[1] See James E. Gentle in "A Companion in Mathematical Statistics" (2011): http://mason.gmu.edu/~jgentle/csi9723/MathStat.pdf. Last visited: 30 July 2011.

standard Bayesian terminology it is called the “highest posterior density region (HPD)” or “Bayesian confidence sets” and the posterior density is used as a measure.

In this paper, we draw a distinction between two categories of problems: problems with convex contour shapes, and problems with non-convex contour shapes such as ridges. We focus our attention solely to providing an approach that tackle’s convex contour shaped problems.

This paper is organized as follows. In chapter 2, we start off with discussing some of the fundamental properties of highest density regions, and point to some of its peculiarities. Then in paragraph 2.1 we will provide a heuristic to determine the HDR for univariate densities. In the subsequent paragraph, we will develop two heuristics that are capable of tackling unimodal regions (section 2.2.1) and multimodal regions (2.2.2). In chapter 3 we show and discuss the experimental results that follow after applying the heuristics on synthetic data. And in the final chapter, we provide a conclusion and discuss possible extensions.

2. Modeling Highest Density Regions

In the introduction we quoted Hyndman for writing that HDR's are a "more effective summary of the forecast distribution than other common forecast regions" and we mentioned some of the properties of HDR's. In this section we will go in more depth of HDR's: we will give a more formal definition of HDR's, and discuss some of its properties through illustrations. We shall also provide a brief description of some of the types of problems, which we will discuss in the next section.

What makes HDR distinct from other statistical methods that summarize the sample space for a given probability coverage $1 - \alpha$, is that it represents the shortest interval, or sets of intervals, possible in the one-dimensional sense, and the smallest region(s) possible in the two-dimensional case.

The purpose behind summarizing a probability distribution by a region of the sample space is to identify a relatively "small set which contains most of the probability, although the density may be nonzero over infinite regions of the sample space". This is also the idea "underlying prediction regions and boxplots". However, there are possibly "infinite number of ways to choose a region with given coverage probability". And if we want to stick to a single method, that we could use consistently, it would become necessary to decide what the properties of the underlying data and its region must be. In this paper, we will use the following criteria from Hyndman (1996):

1. The region covering the sample space for a given probability 1-α, should have the smallest possible volume.
2. Every point inside the region should have probability density at least as large as every point outside the region.

The above criteria are equivalent to that of Box and Tiao (1973), and such regions are called highest density regions (HDR's). The advantages of HDR are illustrated in Figure 1, where we see a Normal mixture density and five different statistical methods to summarize the probability distribution by a 75% probability region. Notice that not only does the HDR occupies the smallest region; it is also the only method that makes the bimodality of the graph visible. We define mode to be a local maximum that can be attained in a set of values. For example, the probability density function in figure 1 has two distinct maxima, and thus two modes. From the properties and example above, and as can be seen in figure 1, it follows that the number of disjoint intervals in a highest density region can *never* exceed the number of modal groups in the density function.

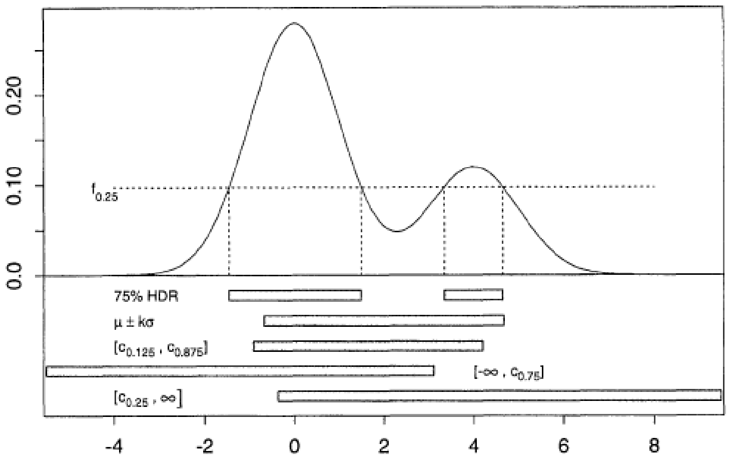

Figure 1. Five Different 75% Probability Regions From a Normal Mixture Density. Here, c_q denotes the qth quantile, μ denotes the mean, and σ denotes the standard deviation of the density.

2.1 HDR's for Univariate Densities

In this paragraph we provide a method for determining the HDR's of a multimodal univariate density function. We start by presenting an appropriate exploratory method for investigating the number of modal groups within the sample. Afterwards, we provide a simple method for determining the HDR's of uni- and multimodal univariate density functions.

Suppose we have a set of measurements of a certain population. Let us assume that the set of measured data points are a sample from an unknown probability density function. To investigate the number of modes manifested by the sample, we need to use a density estimator for a graphical exploration and presentation of the data. There are methods available for this purpose. "The oldest and most widely used density estimator is probably the histogram". However, we will confine ourselves to the use of a kernel density estimator, because, as Fisher (1989) and Silverman (1986) point out, the histogram can be "(positively) misleading indicator for determining the number of modes" in the sample. While on the other hand, the kernel density estimator proves to be "*the most useful graphical technique"* for this purpose (Silverman, 1986, and Fisher et al., 1994).

The kernel density estimator (Rosenblatt, 1956) provides a "visual means of representing the distributional structure of a data set" of univariate observations $X_1, X_2, \ldots, X_n$. It is defined by

$$\widehat{f_h}(x) = n^{-1} \sum_{i=1}^{n} K_h(x - X_i),$$

where $K_h(.) = K(\frac{\cdot}{h})/h$; K is called the kernel function and h the bandwidth. If we assume $\widehat{f_h}$ to constitute a random sample drawn from some probability density $f(x)$, we could also test whether modes of $\widehat{f_h}(x)$ can be concluded to reflect modes of $f(x)$, instead of being due to sample variability. Silverman (1986) and Fisher et al. (1994) provide many important applied aspects for testing the significance of a mode, which are based on kernel density estimation.

When we know the exact number of modes in our density estimate, we can turn to the problem of determining the HDR's of uni- and multimodal univariate density functions. For tackling this problem, again we use a density estimator, but now we turn to the histogram. The histogram is constructed "by subdividing the axis of measurement into intervals of equal width", over which we construct a rectangle such that the "height of the rectangle is proportional to the *fraction* of the total number of measurements falling in each cell" (Wackerly et al. ,2002). We will leave the number, width, and location of the intervals used to the discretion of the person who's constructing the histogram. We note, however, that the exact method of constructing the histogram will affect the outcome of our problem. Especially, the width of a single interval, will certainly affect the results of our problem, as our objective is to minimize the entire interval while a fraction of, for example, 95% of the total number of data elements is covered. After we established the relative frequency distribution of the data set and the accompanying histogram, we need to represent that through a (1xN) vector A with length N equal to the number of intervals in our histogram, and with a value equal to the fraction of the total number of measurements falling in each cell. Constructing this vector is sufficient for our objectives of developing a method for determining the HDR.

Earlier in this chapter, we established that the number of disjoint intervals in a highest density region can never exceed the number of modal groups in the density function. The number of modes, by definition, and as can be seen from figure 1, is equal to the number of local maxima. From this it follows that the number of modal groups, say k, in the relative frequency distribution represented in vector A, is equal to the number of maxima. This is the same for density estimates of course.

We can now turn to our heuristic for determining the HDR for a univariate density estimate. It is presented in Textbox 1 below. The heuristic has the advantage of providing very fast estimates of the HDR for any number of modes. However, the heuristic does not provide the optimal highest density region, only a relatively good estimate of the optimal region.

Heuristic for determining the HDR for a univariate density estimate

```
1. Let A be a (1xN) vector representing the relative frequency distribution, with length N equal to the number of intervals in our histogram.
2. Let $\alpha$ be the critical value, for example $\alpha$=5%.
3. Sort the vector A in descending order. Let the permutation of A be vector B, and the sorted vector A be vector C.
4. Let S be a scalar valued zero.
5. For j is 1 to N
5a. If S is equal to or greater than $(1-\alpha)$, do
5b. STOP LOOP.
5c. Else, do
5d. Set S = S + C(j);
5e. End.
6. End
7. The HDR can now be constructed using the intervals in B(1) to B(j-1). When we join the direct neighboring subintervals, we will get the HDR with at most k disjoint intervals, with k representing the number of modal groups in the density estimate.
```

Textbox 1: Shows a heuristic for determining the HDR for a univariate density estimate.

2. 2 HDR's for Bivariate Densities

In the previous paragraph we had to deal with univariate sequential data, and we started by first defining the density estimate, and summarizing this information in a vector A. In this paragraph, our data is of a different nature, that is, they consist of coordinates on a Euclidean plane. We must draw a distinction between two categories of problems: problems with convex contour shapes, and problems with non-convex contour shapes such as ridges. In this paper, we restrict our attention to providing an approach that tackle's convex contour shaped problems.

Now because the highest density region is defined to be having the smallest possible region for the sample space for a given probability 1-α, in the two-dimensional space, for convex contour shaped problems, this would translate in constructing the smallest possible convex hull of a sample space for a given probability 1-α. To summarize, determining the highest density region of a given probability for two-dimensional convex contour shaped data means finding the smallest possible convex hull that covers the sample space for the same probability. As we tackle this problem through methods originating of computational geometry and machine learning, we will utilize their terminology where ever necessary.

Now, to determine the smallest possible convex hull covering a given probability of the sample space, we start by analyzing the brute-force method, and improving upon its efficiency in terms of its (worst-case) computational time.

To tackle the problem, we divide the main problem into the following sub-problems:

1. For a given dataset, determining the convex hull and the surface of its area.
2. Determine all possible subsets consisting of 95% of the initial sample space. Determine for each of them its convex hull and their surface.
3. For all these subsets determined above, find the one whose surface of the convex hull has the minimum region.

The brute-force method above, which determines the surface of the convex hull of all possible subsets, is computationally very expensive, even for small data sets. However, there is no other solution known in existing literature. There has been some research in dynamic convex hull algorithms, in particular by Jacob (2002) and Brodal et al. (2002) and Baker et al. (2002). These papers, however, rather focus on updating the convex hull efficiently for the situation of inserting or deleting an element of the sample space. But they do not provide any method for deciding which elements of the sample space would have to be deleted to minimize the surface of the convex hull. And since it's unlikely that there can ever be an efficient polynomial time exact algorithm, we will resort to heuristics that provide sub-optimal solutions with reasonably fast running-time. In particular, we will use a greedy choice heuristic to decide which element of the sample space we need to delete to acquire the minimum size. Obviously, the greedy heuristic has its drawbacks, but we expect this approach to very yield sound results, as the dispersion of the elements in the two-dimensional space will have a high density at the center and a low density surrounding the center, forcing most of the gains through deletions to be made the furthest from the high density center.

2.2.1 Unimodal regions

Basically our proposed method consists of three main parts: (1) to determine the convex hull for a given data set, (2) to compute the surface of a given convex hull, and (3) to minimize/reduce the size of the convex hull as much as is possible in an acceptable running-time. In this section we will tackle all three of these main problems in sequence. We will present an efficient variant of the Graham's Scan algorithm to determine the convex hull in $O(\, n \, log \, n\,)$ time. In addition, we will present a heuristic by Akl and Toussaint that can be used together with Graham's Scan algorithm to speed up the process even more. This acceleration is achieved in linear time. For the second problem we will compute the surface of the convex hull by cutting it in two parts from its left-most coordinate tot its right-most coordinate to get an upper hull and lower hull, where after we compute the surface through a combination of the Riemann sum and some basic geometry. The third main problem is tackled through two different methods, of which the second consists of a combination with the first. Their goal is basically to provide a criterion to choose which of the elements of the convex hull to delete.

The convex hull problem

The planar convex hull problem is, given a data set consisting of n coordination points P in the plane ($n \geq 3$), to determine the boundary points P of the closed convex polygon. We will present a simple variation of the famous Graham's Scan algorithm for determining the convex hull. While the original Graham's Scan uses a counterclockwise examination of points to determine the convex hull, our approach is based on the so called *"incremental construction"* where consecutive points are evaluated one at a time, and "the hull is updated with each new insertion". Points can be added in an arbitrary order, however, this would require us to "test whether points are inside the existing hull or not. "To avoid the need for such a test, we will add points in increasing order of x-coordinate, thus guaranteeing that each newly added point is outside the current hull" (Mount, 2002, and Thanh An, 2007).

The variant of Graham's Scan algorithm makes use of a geometric operation called *orientation*. The orientation operates on points similar in some respects to the relational operations (<, =, >) on numbers. For a given ordered triple of points <p, q, r> in the plane, the orientation is positive if it has a counterclockwise oriented triangle, and the orientation is negative if it has a clockwise oriented triangle, and the orientation is zero if they are collinear. Note that the sign of the orientation of an ordered triple does not change when the points are translated, rotated, and/or scaled (by any positive factor). Formally, the orientation is defined as the "sign of the determinant of the points given in homogeneous coordinates", that is, by prefixing a 1 to each coordinate. For example, for the ordered triple of points <p, q, r> in the plane in figure 2, we define

$$Orient\ (p,q,r) = det\begin{pmatrix} 1 & p_x & p_y \\ 1 & q_x & q_y \\ 1 & r_x & r_y \end{pmatrix}$$

Now that we have explained the orientation, we will briefly explain the algorithm for determining the convex hull. We define a stack U to store the vertices of the hull. The top of the stack contains the most recently stored vertices in the stack. Let first(U) and second(U) denote the top and second element from the most recently added vertices in the stack U, respectively. Note that when we read the stack from top to bottom, the points should have a positive orientation. Thus after we add the last point, if the previous two points do not have a positive orientation; we move them off the stack. And because the orientations of all the remaining points are unaffected, there will be "no need to check any points other than the most recent points and its top neighbors on the stack" (Mount, 2002).

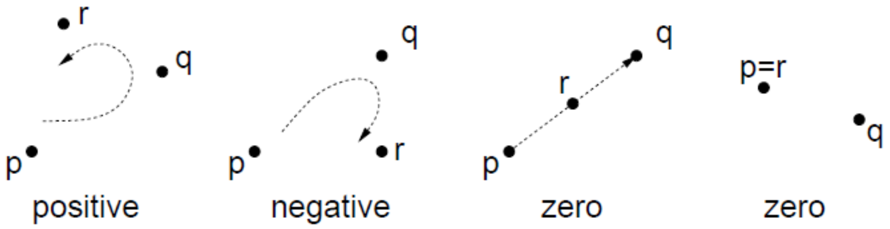

Figure 2: Orientations of the ordered triple (p, q, r).

Graham's Scan Algorithm (variant).

```
1. Sort the points according to increasing order of their x-coordinates, denoted < $p_1$, $p_2$, .., $p_n$ >.
2. Sort the points according to decreasing order of their x-coordinates, denoted < $p_{n+1}$, $p_{n+2}$, .., $p_{2n}$ >.
3. Push $p_1$ and the $p_2$ onto stack U.
4. For j from 3 to 2*n, do:
    4a. While $size(U) \geq 2$ and $Orient(p_j, first(U), second(U)) \leq 0$,
        4a1. Pop U.
    4a2. End.
    4b. Push $p_j$ onto U.
5. End.
```

Textbox 2: Shows a variant of the Graham Scan algorithm for determining the complex hull.

The above algorithm has a worst-case running time of $O(n \, log \, n)$. For proof on running time, see by Mount(2002). While a running time of O(nlog(n)) for Graham's Scan algorithm is relatively fast, we could do much better by combining the first step of the QuickHull algorithm by Akl and Toussaint with that of Graham's Scan. The main benefit of this first step (from hereon we shall call it the Akl and Toussaint heuristic) is that we discard all data points of which we know for certain that they form the interior of the convex hull, thereby reducing the number of data points on the sample space drastically. This would prove very beneficial for very large datasets.

The Akl and Toussaint heuristic

1. Determine the four extreme points of the dataset. These four points are the minimum and maximum X and Y coordinates: say (XMIN,y1), (XMAX, y2), (x1, YMIN), (x2, YMAX), respectively.
2. Identify all points interior to this polygon, and discard them form the sample space.

Textbox 3: Shows the Akl and Toussaint heuristic to speed up the process of determining the complex hull.

Computing the surface

After determining the convex hull, we can calculate its surface with the coordinates of the vertices of the convex hull. To compute the surface of the convex hull, we start by making sure the minimum y-coordinate of the convex hull is nonnegative. If this is not the case, we need to translate all the y-coordinates of the vertices on the convex hull in such a way that they are all positive, or at least zero. We can do this simply by adding the absolute value of the minimum y-coordinate to all the y-coordinates of the vertices. After that, we cut the convex hull into two parts, an upper hull and a lower hull. We make the cut from the left-most vertices' up to the right-most vertices'. From here on we can compute the surface beneath the upper hull and lower hull by making use of the (translated) coordinates of the vertices. The surface of the convex hull is equal to the difference of the surface of the upper hull and the surface of the lower hull. The surface can be computed easily through a combination of the Riemann sum and some basic geometry.

Minimizing the surface heuristically

Now that we have provided the methods to determine both the convex hull, and its surface, we continue with the challenge of reducing the size of the convex hull as much as is possible in an acceptable running-time.

One way to minimize the size of the convex hull to a subset consisting of 95% of the sample space is by using a greedy choice for deciding which element of the sample space to delete to acquire the minimum size. The advantage of this method is that its running-time is linear in the number of evaluations required for each deletion. However the drawback of this method is that while it provides a locally optimal choice at each stage of deletion, this may not result in the global optimum. However, for a medium to large dataset that satisfies the property 1 of HDR, i.e. every point inside the region has probability density at least as large as every point outside the region, we expect this approach to produce very acceptable results, as the dispersion of the elements will have a high density at the center and a low density surrounding the center, forcing most of the gains to be made the furthest from the high density center.

Greedy heuristic

```
1. Let N be the total number of points in the dataset, and let i=0.
2. Let α be the critical value, for example α=5%.
3. For j from 1 to floor(α*N), do:
    3a. Determine the convex hull of the (remaining) points, and its surface.
    3b. Let M be a (1xc) vector with c equal to the number of vertices on the convex hull.
    3c. For k from 1 to c, do:
        3c1. Determine the difference in terms of surface after deletion of the element.
        3c2. Set M(k) equal to that difference in terms of surface.
    3c3. End.
    3d. Identify the index of max(M) on the original dataset.
    3e. Delete the previous element from the dataset.
3f. End.
4. The remaining points on the dataset constitute a subset covering 95% of the sample space.
```

Textbox 4: Shows the greedy heuristic for determining the HDR for unimodal regions of bivariate densities.

A possible alternative to the above method is by adding the demand that no “sharp” corners are allowed to be among the vertices of the convex hull. So if there are any sharp corners present among the vertices of the convex hull, for example: angles smaller than π/2, we will delete the vertices’ that has the sharpest corner. If there are no sharp corners, we use greedy choice to delete the locally optimal vertices.

With this addition to the greedy heuristic we expect two major differences in the outcome. First, we expect the running-time to improve significantly, as the number of surface evaluations per deletion of each point is drastically reduced. Second, in deleting the sharpest corner of the convex hull, we would very likely be deleting a point that is not locally optimal, which as a direct consequence, either takes as closer or further from the global optimum.

Greedy & Angle heuristic

```
1. Let N be the total number of points in the dataset, and let i=0.
2. Let $\alpha$ be the critical value, for example $\alpha$=5%.
3. For j from 1 to floor($\alpha$*N), do:
    3a. Determine the convex hull of the (remaining) points, and its surface.
    3b. Determine the angles of each corner of the convex hull.
    3c. Let M be a (1xc) vector with c equal to the number of vertices on the convex hull.
    3d. If smallest angle is equal to or smaller than the minimal sharpness required, do:
        3d1. Identify the index of corner on the convex hull.
        3d2. Delete the previous element from the dataset.
    3e. Else, do:
        3e1. For k from 1 to c, do:
            3e2. Determine the difference in terms of surface after deletion of the element.
            3e3. Set M(k) equal to that difference in terms of surface.
        3e4. End.
        3f. Identify the index of max(M) on the original dataset
        3g. Delete the previous element from the dataset
    3h. End.
3i. End.
4. The remaining points on the dataset constitute a subset covering 95% of the sample space.
```

Textbox 5: Shows the greedy & angle heuristic for determining the HDR for unimodal regions of bivariate densities.

2.2.2 Multimodal regions

In the previous section we split the original problem in three main parts and presented a heuristic for arriving (in a reasonable running-time) at an acceptable local minimum surface. However, this would not be the case if the data set was not homogeneous, but rather contained a number of clusters whose items differ across the groups, but not so much within the group (cluster). When we apply the greedy heuristic of the previous section to a dataset containing multiple clusters, we will be neglecting the inherent structure of the dataset, and as a consequence, our heuristic would fail to produce an *acceptable* suboptimal solution. In figure 3 this is illustrated for a dataset containing a bimodal region. The origin and details of the graph are discussed later on in chapter 3.

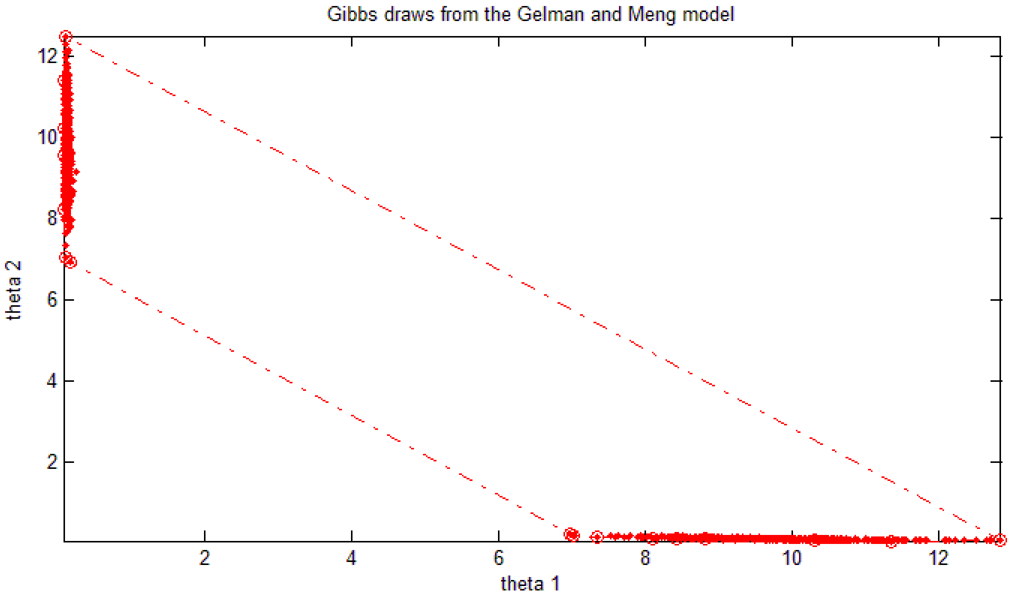

Figure 3: Shows the convex hull of a data set containing bimodal regions. The “o” represent the samples from the dataset, and the dotted line represents the convex hull. The convex hull occupies an enormous empty space that cannot be “freed” by any of our heuristics that we presented in the previous section. This is only possible if we exploit the inherent structure of the dataset.

In the remaining of this section we will (1) provide a revision of both heuristics of the previous section to deal effectively with data containing multimodal regions, (2) provide a method that identifies the data items across multiple clusters, and discuss some of the properties of this method.

Revising the Greedy heuristic and the Greedy & Angle heuristic

Both heuristics of the previous section are useful for tackling unimodal multivariate densities; however, in case we need to deal with multimodal problems, we could still tackle the problem if we could only identify which data elements belong to which clusters. There are two very practical clustering methods we could use for this purpose, i.e. the EM algorithm and the K-means clustering method. We will discuss the these methods further on, but for now, let us suppose that we could determine the number clusters in the data and identify which data items must be grouped together as a cluster. From here on, we still need to determine the HDR for a given probability coverage of (1-α)x100% and there seem to be two different tracks that would lead us to two very different solutions. The first track is to determine for each cluster the HDR separately, as if it was a unimodal multivariate density problem of its own, and to do this for all clusters in the original dataset. This method would require no revision of our heuristics other than adding a few steps for implementation. However, the total of all the separate HDR’s need not to be a HDR on its own, as property 2 of the HDR is violated. The second track is to minimize, for a given probability coverage, the total surface of all clusters to a local optimum, by a number of deletions, where at each stage of deletion, we evaluate the vertices of the convex hulls of all clusters that yield the biggest gains in terms of minimizing the total surface. The second track would, in comparison to first track, evaluates more vertices at each stage of deletion, and thus seems more capable of minimizing the total surface. It also does not violate property 2 of the HDR as opposed to the first proposed track. For the above reasons, our preference goes to the second proposed method.

In textbox 6 & 7 we present the same heuristics as in section 2.2.1, with the only difference that they are now adapted to be able to tackle multimodal regions of bivariate densities.

Determining the HDR through the Greedy heuristic

```
1. For a given dataset, and given number of clusters, identify which data items must be grouped together to form a cluster.
2. For each cluster, determine its convex hull and surface. And compute the sum of all cluster surfaces.
3. Let N be the total number of points in the dataset, and let α be the critical value, for example α=5%.
4. Let cluster=0
5. For k from 1 to floor(α*N), do:
    6. Determine the convex hull of the (remaining) points of each cluster.
    7. Determine the sum of all cluster surfaces.
    8. If cluster==i, with i≠0, and i is the cluster ID, do:
        9. Let M be a (1xc) vector with c equal to the number of vertices on the convex hull of cluster i.
        10. For j=1:c, do:
            11. Determine the difference in terms of surface after deletion of the point.
            12. Set M(j) equal to that difference.
        13. End.
    14. Else, evaluate all points, i.e. repeat step 9-12 for all clusters.
    15. End.
    16. Identify the index and cluster ID of Max(M) on the original dataset. Set i=cluster ID
    17. Delete the previous element from the dataset.
18. End.
19. The convex hull of the remaining points of each cluster constitute the heuristically determined HDR.
```

Textbox 6: Shows the adapted greedy heuristic for determining the HDR for multimodal regions of bivariate densities.

The clustering problem

We are now left with the clustering problem of determining the number of clusters in the dataset, and of identifying which data items belong to which clusters. The first step in dealing with this general clustering problem is to view it as a density estimation problem (Silverman, 1986). In addition, the data is assumed to origin from a mixture model where the cluster identifier is hidden. We can keep track of which data point belongs to which cluster, by assuming that there is an unobserved variable for each data point that indicates its membership to a certain cluster ID. The advantage of using finite mixtures is that they allow a probabilistic model-based approach to clustering (Figueiredo et al.,2002). In general, a mixture model M having K clusters C_{ID}, with ID=1, …, K, assigns to each data point x a probability $\Pr(x|M) = \sum_{ID=1}^{K} W_{ID} * \Pr(x|C_{ID}, M)$, where W_{ID} are called the mixture weights (Bradley et al. 1998). The general clustering problem can thus be tackled by inferring the parameters associated with the mixture model M and identifying the probability distribution of each cluster, that maximizes the likelihood of the data given the model.

We mentioned earlier in this section that the EM algorithm and the K-means clustering method are two very practical clustering methods we could use for this purpose. The EM algorithm is a broadly applicable approach that can be used to estimate the parameters of the general clustering problem. The K-means, however, is more a special case of the EM algorithm that assumes each cluster to have a spherical Gaussian distribution, and that all mixture weights W_{ID} are

Determining the HDR through the Greedy & Angle heuristic

```
1. For a given dataset, and given number of clusters, identify which data items must be grouped together to form a cluster.
2. For each cluster, determine its convex hull and surface. And compute the sum of all cluster surface's.
3. Let N be the total number of points in the dataset, and let $\alpha$ be the critical value, for example $\alpha$=5%.
4. Let cluster=0
5. For k from 1 to floor($\alpha$*N), do:
6. Determine the convex hull of the (remaining) points of each cluster, and the angles of their vertices.
7. Determine the sum of all cluster surfaces.
8. If smallest angle is equal to or smaller than the minimal sharpness required, do:
9. Identify the index of corner on the convex hull.
10. Delete the previous element from the dataset.
11. Else, do:
8. If cluster==i, with $i \neq 0$, and i is the cluster ID, do:
9. Let M be a (1xc) vector with c equal to the number of vertices on the convex hull of cluster i.
10. For j=1:c, do:
11. Determine the difference in terms of surface after deletion of the point.
12. Set M(j) equal to that difference.
13. End.
14. Else, evaluate all points, i.e. repeat step 9-12 for all clusters.
15. End.
16. Identify the index and cluster ID of Max(M) on the original dataset. Set i=cluster ID
17. Delete the previous element from the dataset.
18. End.
19. End.
20. The convex hull of the remaining points of each cluster constitute the heuristically determined HDR.
```

Textbox 7: Shows the adapted greedy & angle heuristic for determining the HDR for multimodal regions of bivariate densities.

equal (Bradley, 1998). Typically, for the k-means, the number of clusters K is assumed to be known, and given as input. On the other hand, the EM algorithm does not need this number of clusters as input, as this issue can be addressed in a formal manner[2]. However, if the number of clusters is known beforehand and used for initialization, the implementation of the EM algorithm would require less sophistication and less iteration to converge to an acceptable local optimum.

For our purposes, in the context of determining the HDR for convex contour shaped data, we expect to encounter different manifestations of the Gaussian mixture distribution. Thus, the k-means clustering algorithm falls short in its ability to adequately represent non-spherical Gaussian distributions. We will therefore use the EM algorithm for finite Gaussian mixture models. For a very readable introduction and discussion of the EM algorithm for Gaussian mixture models, see McLachlan and Peel (2000).

[2] For example, Figueiredo and Jain (2002) propose a formal method that allows the EM algorithm to select the number of clusters and not require careful initialization. Also, Zhang et al. (2003) propose an algorithm that is capable of choosing the necessary number of clusters automatically and efficiently, and also being insensitive to the initial configurations.

3. Experimental results

In this chapter we show and discuss the experimental results that follow after applying our methods on synthetic data. We start with the heuristic for determining the univariate densities. We then continue to discuss our novel methods for determining the HDR estimate for two different variants of the Gelman-Meng model: a bell-shaped example, and a bimodal example. We compare the methods in terms of effectiveness in reducing the surface and in terms of running time.

3.1 Univariate densities

We simulate a two normal distribution. The first has mean 10 and variance 2, N(10,2), the second has mean 18 and also variance 2, N(18,2). We draw 100.000 samples and determine the relative frequency distribution of it, with width value 0.5 units. The graph of the relative frequency distribution can be seen in figure 4a below. When executing the heuristic in textbox 1, we choose alpha to be 30%, and get a HDR estimate, whose graph is plotted in figure 4b. The interval of the highest density region goes from 8.5 to 12 and from 16 to 20; that makes in total a length of (12-8.5)+(20-16)=7.5 units, with only one interuption.

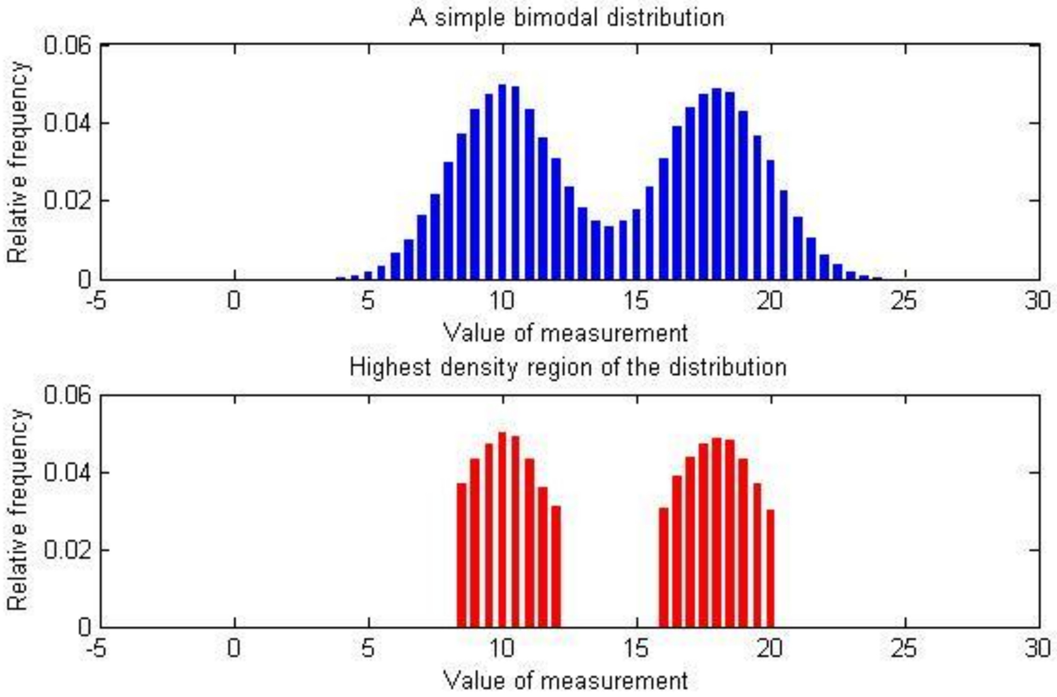

Figure 4: (a) Shows the relative frequency distribution of two Gaussian models. (b) Shows the highest density region estimate of the graph above it.

3.2 Bivariate densities

To illustrate the workings of our methods, we go through two different examples which are based on the model of Gelman and Meng (1991). Suppose we have a joint posterior density of θ_1 and θ_2, which has the following “basic” form

$$p(\theta_1, \theta_2) \propto exp\left[-\frac{1}{2}[a\theta_1^2\theta_2^2 + \theta_1^2 + \theta_2^2 - 2b\theta_1\theta_2 - 2c_1\theta_1 - 2c_2\theta_2]\right]$$

Where a, b, c_1 and c_2 are constants under the restrictions that $a \geq 0$ and if $a = 0$ then $|b| < 1$. The purposes of these restrictions are to ensure that the above joint density is a proper probability density function. The above joint density distribution has the feature that the random variables θ_1 and $\theta_2$are conditionally Normally distributed (Pooter et all., 2006). Moreover, the conditional densities from the above model can be “recognized as normal densities withthe following parameters”

$$p(\theta_1|\theta_2, a, b, c_1, c_2) \sim N\left(\frac{b\theta_2 + c_1}{a\theta_2^2 + 1}, \frac{1}{a\theta_2^2 + 1}\right)$$

$$p(\theta_2|\theta_1, a, b, c_1, c_2) \sim N\left(\frac{b\theta_1 + c_2}{a\theta_1^2 + 1}, \frac{1}{a\theta_1^2 + 1}\right)$$

By choosing different values for the parameters $\theta_1, \theta_2, a, b, c_1$ and c_2, we can “construct joint posterior densities of rather different shapes, while the conditional densities remain the same” (Pooter et all., 2006).

3.2.1 Gelman-Meng: the bell-shaped configuration

Our first example is based on the following configuration: $(a = b = c_1 = c_2 = 0)$, from which it follows that the basic form changes to

$$p(\theta_1, \theta_2) \propto exp\left[-\frac{1}{2}[\theta_1^2 + \theta_2^2]\right]$$

And the conditional densities become standard normal densities. In figure 5a (next page) we provide a scatter plot of one thousand draws from the standard normal densities, and in figure 5b we plot the bell-shaped joint posterior density for θ_1 and θ_2. The draws are obtained from Gibbs sampling (Pooter et al.), where we use a burning-period of 10.000 draws.

Applying the “Greedy heuristic” and the “Greedy & Angle heuristic” on the draws obtained from Gibbs sampling, we get the graphical results in figure 6, and the numerical results in table 1.

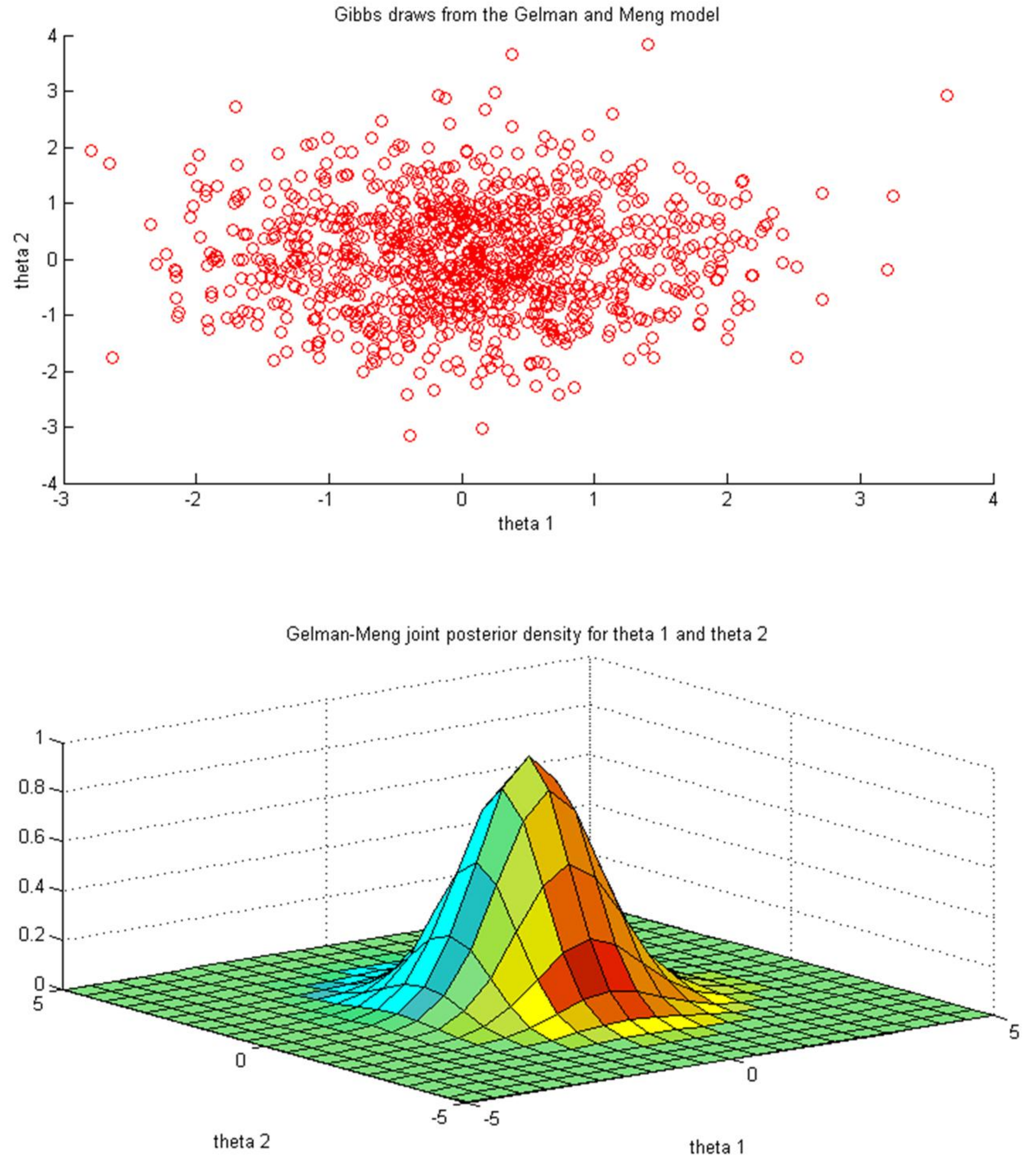

Figure 5: (a) shows a scatter plot of the standard normal densities of θ_1 and θ_2. (b) Shows the Gelman-Meng joint posterior density for the standard normal densities of θ_1 and θ_2, where $a = b = c_1 = c_2 = 0$.

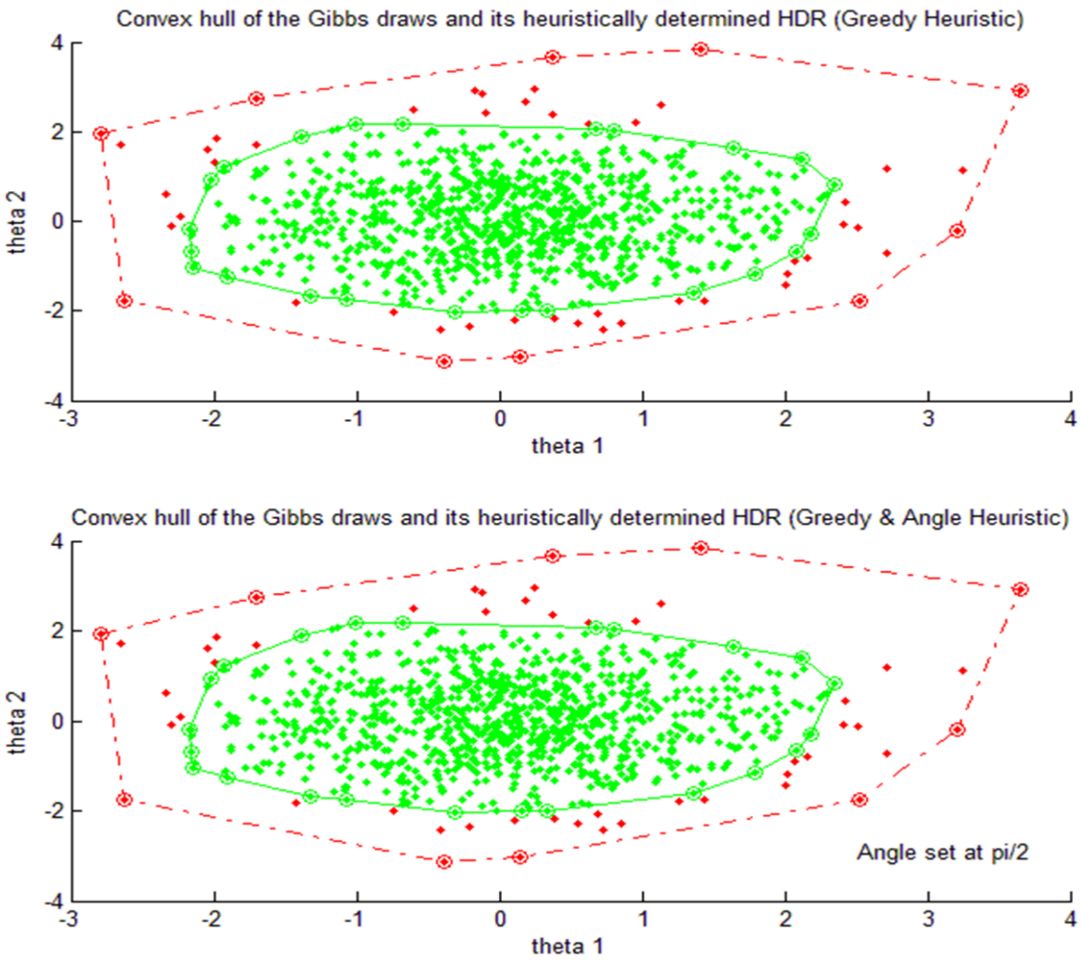

Figure 6: (a) shows the convex hull of the Gibbs draws from the Gelman-Meng bell-shaped configuration and its heuristically determined HDR. The method used here is the Greedy heuristic. The red dots outside the HDR are the deleted points. (b) Like (a), only here the method that was used is the Greedy & Angle heuristic with angle set at pi/2.

Figure 6a and 6b seem very similar to each other. After comparing the numerical results in table 1, we can conclude that they are an exact match. A possible explanation for this is (1) the distribution of the data points is standard normal, causing the dispersion of the data points to be almost circular. And (2) the angle configuration of figure 6b is set at pi/2, which is “too” sharp to permit any points to be deleted. The latter configuration may be changed to suit the need. Despite this seemingly drawback, both methods reduce the surface of the outer convex hull by more than 50%, while only a small fraction (5%) of the data points is deleted.

	Greedy Heuristic	Greedy & Angle Heuristic
Initial surface	33.501	33.501
Removed surface	18.378	18.378
Percentage of surface removed	54.9%	54.9%
# of points deleted with the Greedy method	50	50
# of points deleted with the Angle method	0	0
# of draws from Gibbs sampling	1000	1000
Running time in seconds	14.739	14.739

Table 1: Shows the comparative results after applying both models to the same data set acquired from Gibbs Sampling.

3.2.2 Gelman-Meng: the bimodality configuration

In our second example we set $a = 1$ and $b = 0$, with $c_1 = c_2 = 10$. This transforms the "basic" joint posterior density form of the Gelman and Meng model to

$$p(\theta_1, \theta_2) \propto exp\left[-\frac{1}{2}[\theta_1^2\theta_2^2 + \theta_1^2 + \theta_2^2 - - 20\theta_1 - 20\theta_2]\right]$$

With

$$p(\theta_1|\theta_2, a, b, c_1, c_2) \sim N\left(\frac{10}{\theta_2^2 + 1}, \frac{1}{\theta_2^2 + 1}\right)$$

$$p(\theta_2|\theta_1, a, b, c_1, c_2) \sim N\left(\frac{10}{\theta_1^2 + 1}, \frac{1}{\theta_1^2 + 1}\right)$$

A scatter plot of 1000 Gibbs draws is depicted in figure 7 below. Because the two modes are very far apart, with no probability mass in between, the Gibbs sampler cannot jump from one mode to the other, and thus gets stuck at one of the two modes. To deal with this, we have chosen to draw 500 Gibbs samples with certain starting values, and then another 500 Gibbs samples with other starting values. The results are colored in figure 7. Because figure 7 rather seems to be horizontal and vertical lines instead of two modes of density function, we have plotted the joint posterior density in figure 8. From this figure it is evident to see that the joint posterior density is real, featuring bimodality.

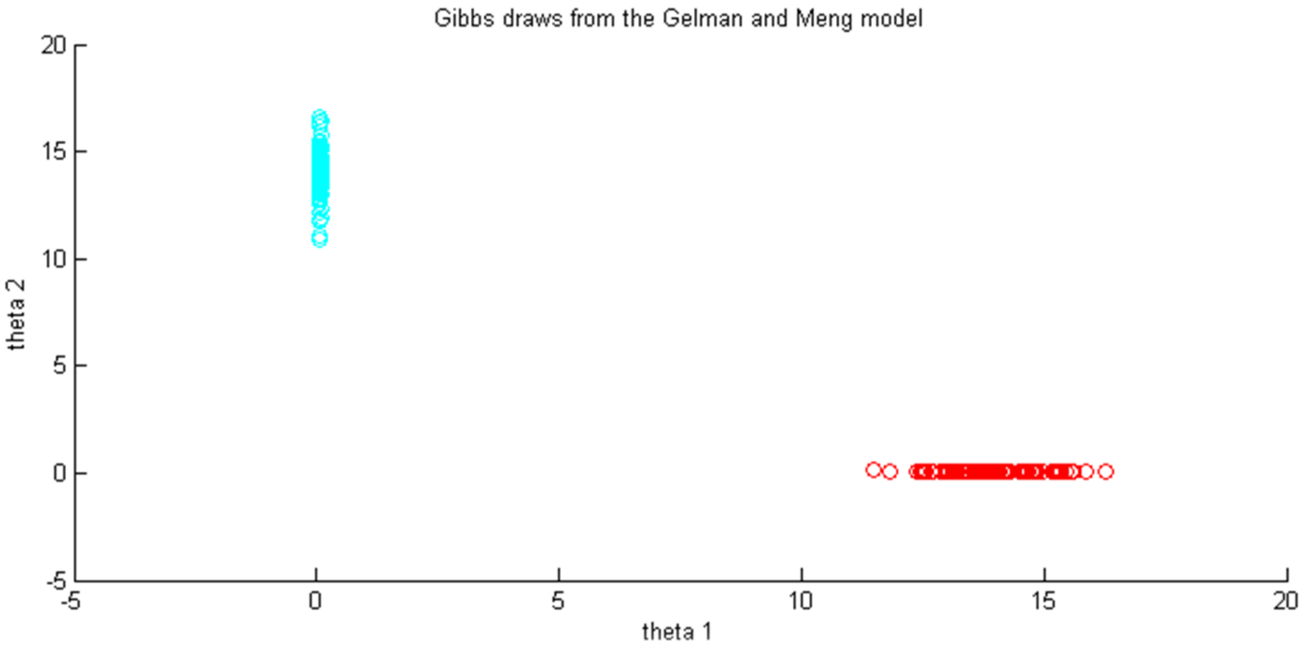

Figure 7: Shows a scatter plot of the Gibbs draws.

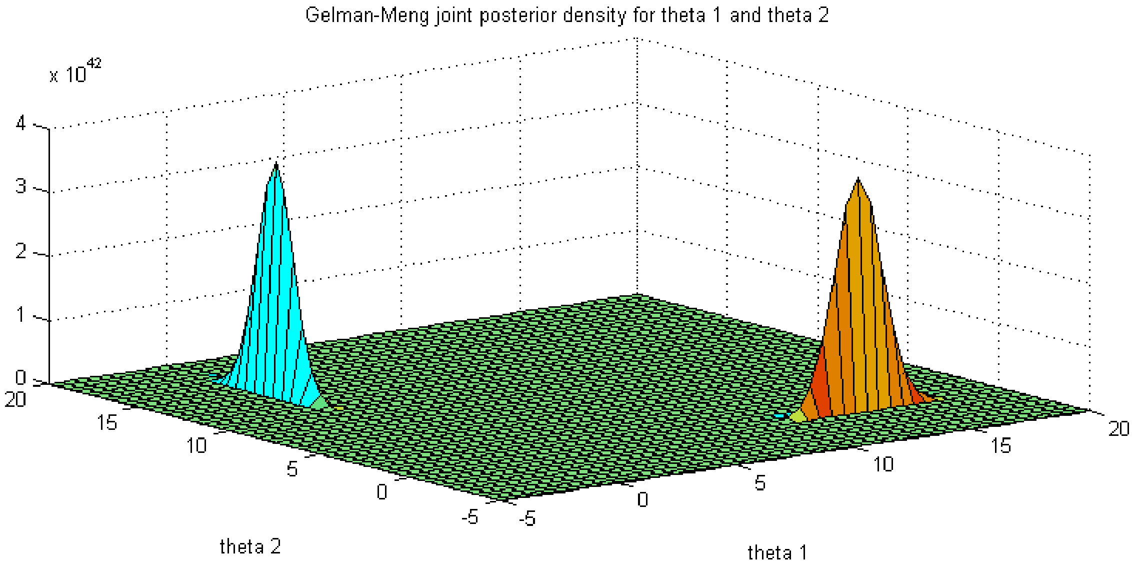

Figure 8: Shows the joint posterior density featuring the bimodality configuration of the Gelman-Meng model.

The two modes of figure 7 and 8 were easily identified by the EM algorithim as two distinct group of clusters, where after we could execute the adapted Greedy heuristic, as was described in textbox 6, and the adapted Greedy & Angle heuristic, as described in textbox 7. The graphical results (with a minor difference in the number of draws, reduced from 500 per mode to 100) can be seen in figure 9 and 10, and the numerical results are shown in table 2.

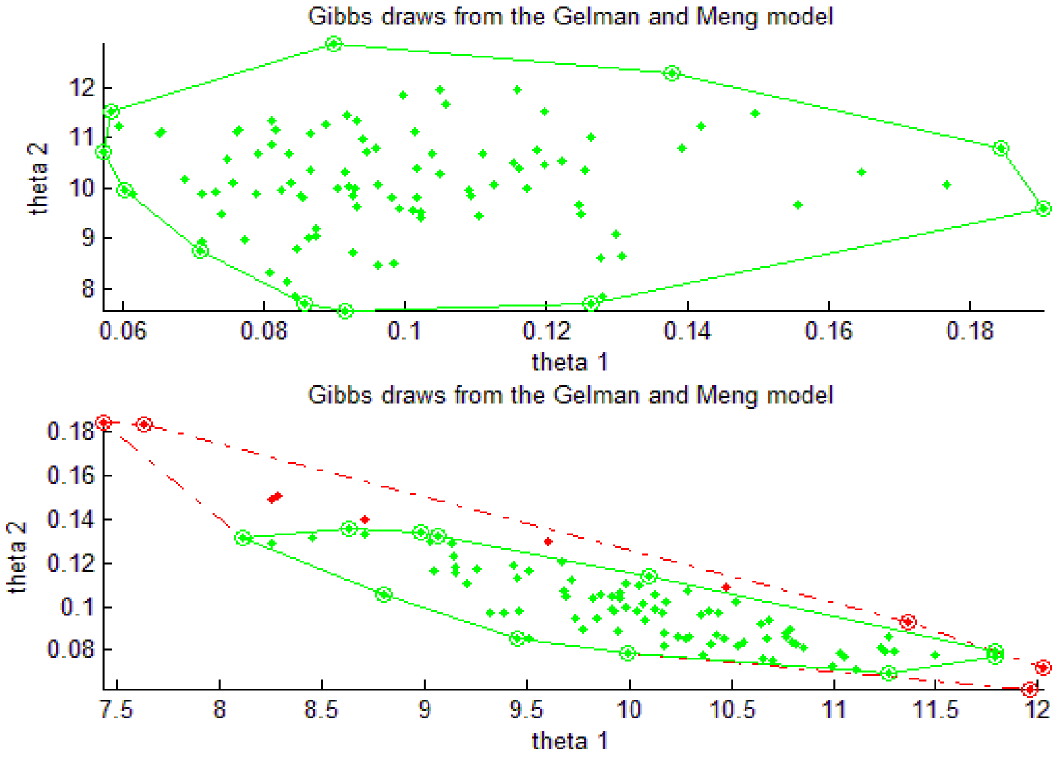

Figure 9: Shows the convex hulls of the draws from the Gibbs sampler for the two different modes, and their HDR. The HDR is determined through the adapted Greedy heuristic.

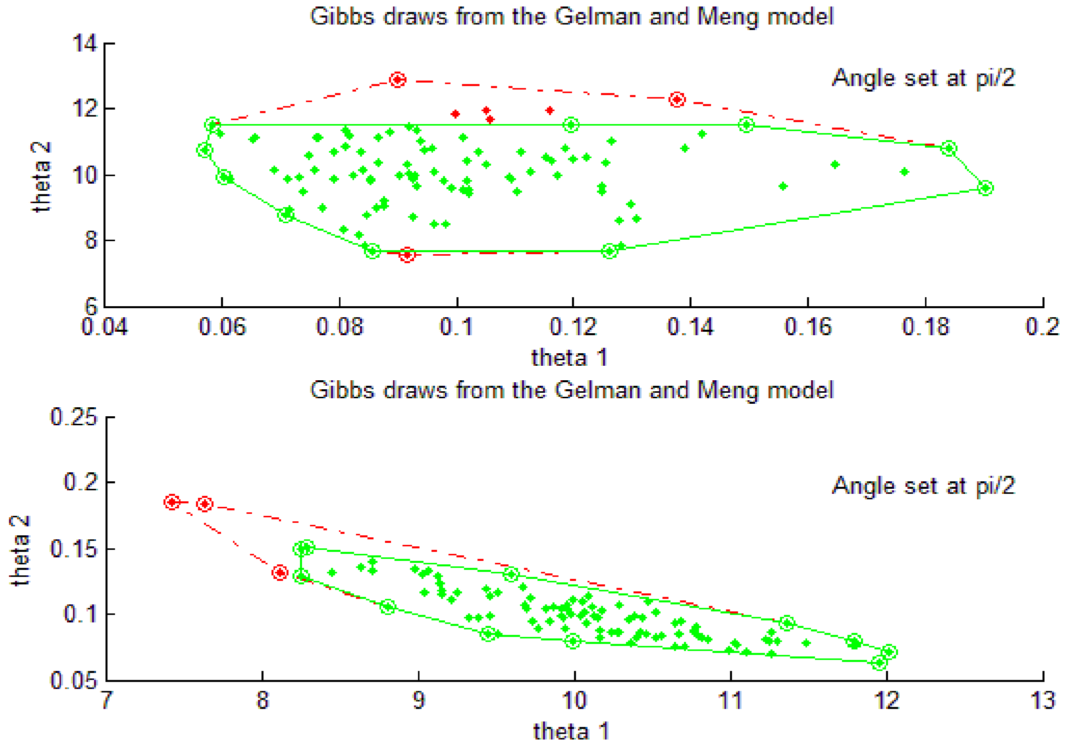

Figure 10: Shows the convex hulls of the draws from the Gibbs sampler for the two different modes, and their HDR. The HDR is determined through the adapted Greedy & Angle heuristic.

Figure 9 and 10 are very different in that their deleted (red) points are selected very differently. And according to the results in table 2, the greedy & angle heuristic beats the less sophisticated greedy heuristic easily! This could be explained from the shape of the graphs, that is, both modes in graph 9 and 10 have “sharp” vertices on their convex hull, and the convex hull itself is almost flat. We noted earlier in section 2.2.1 that “in deleting the sharpest corner of the convex hull, we would very likely be deleting a point that is not locally optimal, which as a direct consequence; either takes us closer to or further from the global optimum.” The results in from this example seem to confirm our expectation, as they have taken us closer to the global optimum.

In addition, we also expected the running-time of the greedy & angle heuristic to be significantly better than that of its counterpart, the greedy heuristic. This again is confirmed from the results in table 2.

	Greedy Heuristic	**Greedy & Angle Heuristic**
Initial surface	0.650	0.650
Removed surface	0.0725	0.131
Percentage of surface removed	11.15%	20.1%
# of points deleted with the Greedy method	10	0
# of points deleted with the Angle method	0	10
# of draws from Gibbs sampling	100	100
Running time in seconds	1.1	0.511

Table 2: Shows the comparative results after applying both models to the same data set acquired from Gibbs Sampling.

4. Conclusion

We have proposed four different methods for determining the highest density region’s for convex contour shaped data. Two of them are only capable of handling unimodal regions of bivariate densities, and the other two are extensions of the latter that are also capable of handling multimodal regions of bivariate densities as long as there is no overlap of the modal groups. Overall, the greedy heuristic seems to be providing a good estimate of the HDR. However, in some cases, as we have seen in the bimodality configuration of the Gelman-Meng model, the greedy & angle heuristic seem to do a better job.

5. Acknowledgements

I would like to thank Dr Wilco van den Heuvel for his directions and comments in connection to this paper.

Figure one was taken from Hyndman (1996), and figure two was taken from Mount (2002).

References

An, P T. “A modification of Graham’s algorithm for determining the convex hull of a finite planar set.” *Annales Mathematicae et Informaticae* 34 (2007): 3-8.

Akl, S G, and G T Toussaint. “A fast convex hull algorithm.” *Information processing letters* 7.5 (1978).

Alsabti, K, S Ranka, and V Singh. “An efficient k-means clustering algorithm.” *http://www.cs.gsu.edu/~wkim/index_files/papers/KMeans98.pdf*. Georgia State University, n.d. Web. 12 May 2011.

Atwah, M M, and J W Baker. “An Associative Static and Dynamic Convex Hull Algorithm.” *http://www.cs.kent.edu/~parallel/papers/atwah02.pdf*. N.p., n.d. Web. 2 June 2011.

Atwah, M M, J W Baker, and S Akl. “an associative implementation of graham’s convex hull algorithm.” *http://www.cs.kent.edu/~parallel/papers/atwah95.pdf*. Kent State University, 3 June 2011. Web. 2 July 2011.

Bohm, C, and H P Kriegel. “Determining the convex hull in large multidimensional databases.” *International Conference on Data Warehousing and Knowledge Discovery* (2001).

Bradley, P S, and U M Fayyad. “Refining initial points for k-means clustering.” *Proceedings of the 15th International Conference on Machine Learning* (1998): 91-99.

Brodal, G S, and R Jacob. “Dynamic Planar Convex Hull.” *The 43rd Annual IEEE Symposium on Proceedings.* (2002).

Box, G. E. P., and Tiao, G. C. “Bayesian Inference in Statistical Analysis.” Reading, MA: Addison-Wesley.

Chazelle, B. “An optimal convex hull algorithm in any fixed dimension.” *Discrete Computational Geometry* 10 (1993): 377-409.

Cheng, C H, et al. “Improved algorithms for k maximum-sums problems.” *Theoretical Computer Science* (2006): 162-170.

Dempster, A P, N M Laird, and D B Rubin. “maximum-likelihood from incomplete data via the EM algorithm.” *Journal of Royal Statistics Society series B* 39 (1977): 1-38.

De Pooter, M D, R Segers, and H K Van Dijk. “On the practice of Bayesian inference in basic economic time series models using Gibbs sampling.” *Tinbergen Institute Discussion Paper* (Apr. 2006): 7-9.

Figueiredo, M A. T., and A K. Jain. “Unsupervised Learning of Finite Mixture Models.” *IEEE Transaction on Pattern Analysis and Machine Intelligence* 24.3 (2002): 381-396.

Fisher, N I. “Smoothing a sample of circular data.” *Journal of structural geology* 11.6 (1989): 775-778.

Fisher, N I, E Mammen, and J S Marron. “Testing for multimodality.” *Computational statistics & data analysis* 18 (1994): 499-512.

Graham, R L. “An efficient algorithm for determining the convex hull of a finite planar set.” *information processing letters* 1 (1972): 132-133.

Gries, D, and I Stojmenovic. “A note on Graham’s vonvex hull algorithm.” *Information processing letters* 25 (1987): 323-327.

Hoogerheide, L F, J F Kaashoek, and H K Van Dijk. “On the shape of posterior densities and credible sets in instrumental variable regresion models with reduced rank: An application of flexible sampling methods using neural networks.” *journal of econometrics* 139 (2007): 154-180.

Hyndman, R J. “Computing and graphing highest density regions.” *The american statistician* 50.2 (1996).

Hyndman, R J. “Highest-density Forecast Regions for Non-linear and Non-normal Time Series Models.” *Journal of Forecasting* 14 (1995): 431-441.

Jacob, R. “Dynamc Planar Convex Hull.” *PhD thesis, BRICS, Dept. Comput. Sci., University of Aarhus* (2002).

Jarvis, R A. “On the Identification of the Convex Hull of a Finite Set of Points in the Plane.” *Information Processing Letters* 2 (1973): 18-21.

Lee, D T. “On Finding the Convex Hull of a Polygon.” *International journal of computer and information sciences* 12.2 (1983).

Mclachan, G, and D Peel. *Finite Mixture Models*. New York: Wiley, 2000.

Mount, D M. “Computational Geometry.” *http://www.cs.umd.edu/~mount/754/Lects/754lects.pdf*. University of Maryland, Fall 2002. Web. 12 May 2011.

Overmars, M H. “Dynamically mainatining configurations in the plane.” *Proc. 12th Annual SIGACT Symp., Los Angeles, CA.* (May 1980).

Overmars, M H, and J Van Leeuwen. “Mainainance of configurations in the plane.” *Journal of Computer Science* 23.2 (1981): 166-204.

Powell, W B. “Merging AI and OR to solve High-Dimensional Stochastic Optimization Problems Using Approximate Dynamic Programming.” *INFORMS Journal of Computing* 22.1 (2010): 2-17.

Powell, W B. “What you should know about approximate dynamic programming.” *http://castlelab.princeton.edu/Papers/NRL%20What%20you%20should%20know%20about%20ADP_Dec162008.pdf*. Princeton University, 16 Dec. 2008. Web. 12 May 2011.

Rosenblatt, M. “Remarks on some non-parametric estimates of a density functio.” *Ann. Math. Statis* 27 (1956): 832-837.

Seidel, R. “Linear Programming and Convex Hulls Made Easy.” *Association for Computing Machinery* (1990).

Silverman, B W. “Density Estimation for statistics and data analysis.” *Monographs on statistics and applied probability* (1986).

Silverman, B W. “Using kernel density estimates to investigate multimodality.” *Journal of the Royal Statistical Society B* 43.1 (1981): 97-99.

Van Deun, K, and P J. F. Groenen. “Majorization Algorithms for Inspecting Circles, Elipses, Squares, Rectangles, and Rhombi.” *Operations Research* 53.6 (2005): 957-967.

Wagelmans, A, S Van Hoesel, and A Kolen. “Economic lot sizing: an O(n log n) algorithm that runs in linear time in Wagner-Within case.” *Operations Research* 40.1 (1992): S145-S156.

Wu, X, and V Kumar. *The Top Ten Algorithms in Data Mining*. N.p.: Taylor & Francis Group, 2009.

Zhang, B, C Zhang, and X Yi. “Competitive EM algorithm for finite mixture models.” *Elsevier* Pattern Recognition.37 (2004): 131-144.

Zivkovic, Z, and F Van der Heijden. “Recursive Unsupervised Learning of Finite Mixture Models.” *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26.5 (2004): 651-656.