A Hierarchical Model for Minimum Entropy Data Partitioning

Wei Wu* Ann B. Lee† David Mumford*

* Division of Applied Mathematics, Brown University, Providence, RI 02912
†Department of Mathematics, Yale University, New Haven, CT 06520

October, 2003

Abstract

We develop an unsupervised general methodology for partitioning data into clusters. The notion of clusters is based on information-theoretic terms for the entropy of the partitions given the data. The model produces a hierarchy of nested clusterings (coarse to fine), where each partition is expressed as a mixture of a set of fixed Gaussian kernel functions. As our model splits clusters solely based on their "connectedness", it can deal with clusters of general shapes and sizes.

1 Introduction

In exploratory data analysis, one often needs to partition high-dimensional data into clusters. The underlying assumption is that these clusters are generated by different classes of stimuli or objects. By partitioning the data, we are able to gain a better understanding of the hidden structures in the data, as well as classify and interpret new observations. Partitioning general data is however a difficult problem due to the high variability in size, density, and shape of naturally occuring clusters. The clusters can, for example, be compact, elongated or ring-shaped. Furthermore, there may exist several sensible ways of dividing the data. The criterion for a "good" partitioning often depends on which scale (coarse or fine) one views the system.

In this paper, we develop a data partitioning model that originates from work by S. Roberts et al. As in [2, 3], we model both the unconditional and the class-conditional probability density functions of the data as Gaussian mixtures. The partitioning of the data is furthermore decided by information-theoretic terms for how well the partitions are separated.

The main contributions of this paper are two: (i) We introduce a clustering criterion that is suitable for clusters of general shapes and sizes. (The original algorithm by S. Roberts et al. favors partitions of equal sizes.) (ii) Instead of producing a single clustering for a fixed number of clusters, we build a multi-scale hierarchy of nested clusterings. The different clusterings are arranged according to ascending "partitioning energy" in a tree (see Fig. 2), where a cluster on a finer scale is a subset of a cluster on a coarser scale. The model selection (of the number of clusters in the final partitioning) is automatically decided by the maximum allowed partitioning energy or connectedness of a cluster.

In Sec. 2 we describe the theoretical framework of our model in detail. In Sec. 3, we show a 2D example of how the hierarchical algorithm can be applied to data with clusters of different sizes and shapes. We discuss our results in Sec. 4, and illustrate with a 1D analytical example in Appendix some of the problems in the original partitioning algorithm of S. Roberts et al.

2 Theory

2.1 Data Partitioning through Penalized Conditional Entropy Minimization

In this section, we consider the problem of partitioning a given set of data points $x_1, x_2, \cdots, x_N \in \Omega$ (Ω is a subset of Euclidean space) into K partitions. Assume that the data points $\{x_i\}$ are generated by a probability density function $\pi(x)$. We write $\pi(x)$ as a linear combination

$$\pi(x) = \sum_{k=1}^{K} \pi(x|k)\pi(k) \tag{1}$$

where $\pi(x|k)$ is the density function conditioned on the kth partition set. The coefficients $\{\pi(k)\}$ satisfy $\sum_{k=1}^{K} \pi(k) = 1$ and are the prior probabilities or weights of the partitions. We say that a data point $x_n \in \Omega$ belongs to the kth partition if the partition posteriors satisfy

$$\pi(k|x_n) \geq \pi(k'|x_n), \ \forall k' \neq k \ . \tag{2}$$

We can formulate data partitioning as an energy minimization problem. An information-theoretic quantity suggested by S. Roberts [2] is the conditional entropy of k given x, i.e.

$$H(k|x) = -\int_{\Omega} \left(\sum_{k=1}^{K} \pi(k|x) \log_2 \pi(k|x) \right) \pi(x)\, dx \tag{3}$$

where $0 \leq H(k|x) \leq \log_2(K)$. For an ideal partitioning of a dataset with well-separated clusters, the partition posteriors are either close to 0 or 1. This implies a small value for the conditional entropy $H(k|x)$. However, *direct* minimization of $H(k|x)$ leads to the trivial solution where all data points are assigned to a single cluster k', and all other clusters $k \neq k'$ are empty; the global minimum of $H(k|x)$ is equal to 0.

In [3], the authors minimize

$$V' = H(k|x) - H(k) = -MI(k,x) \leq 0 \ , \tag{4}$$

where $H(k) = -\sum_{k=1}^{K} \pi(k) \log_2 \pi(k)$ is the Shannon entropy of the partitions, and $MI(k,x)$ represents the mutual information of k and x. Note that the unconditional entropy $0 \leq H(k) \leq \log_2(K)$, where the maximum value $H(k) = \log_2(K)$ occurs when $\pi(k) = 1/K$ for all partitions k. Our experiments show that the partitioning model defined by Eq. 4 works well for datasets where the "natural" clusters are of approximately equal weights, but fails when the clusters are of *different* sizes/weights. In the appendix, we give an analytical example in one dimension to illustrate this problem.

In this paper, we propose a different criterion for partitioning data. Our proposed model can deal satisfactorily with outliers and variable-sized clusters. The energy function is defined as

$$E = \begin{cases} H(k|x) & \text{if} X_k \neq \emptyset \, \text{for} \, \forall k \in \{1, \cdots, K\} \\ \log_2(K) & \text{otherwise} \end{cases} \tag{5}$$

where $X_k \equiv \{x \in \Omega : \pi(k|x) \geq \pi(k'|x), \forall k' \neq k\}$ denotes the kth partition, and $H(k|x)$ is given by Eq. 3. Note that the penalty term $\log_2(K) \gg H(k|x)$ for most "sensible" partitions. By minimizing the penalized conditional entropy function above we can avoid all empty sets X_k without favoring partitions of equal weights.

2.2 Partitions as Gaussian Mixture Models

Before partitioning the data, we set some initial structure on the data. More precisely, we describe the unconditional probability density function $\pi(x)$ as a Gaussian mixture

$$\pi(x) = \sum_{j=1}^{J} p(x|j)p(j) \ , \tag{6}$$

where the Gaussian kernel functions are given by

$$p(\mathbf{x}|j) = \frac{1}{(2\pi)^{d/2}[det(\Sigma_j)]^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{x}-\mu_j)^T\Sigma_j^{-1}(\mathbf{x}-\mu_j)\right] \tag{7}$$

and the number of Gaussians $J \leq K$. The mixture coefficients $\{p(j)\}$ in the model satisfy $\sum_{j=1}^{J} p(j) = 1$. From Bayes' theorem, the kernel posteriors are given by

$$p(j|x) = \frac{p(x|j)p(j)}{\sum_{j=1}^{J} p(x|j)p(j)} \ , \tag{8}$$

where both $p(x|j)$ and $p(j)$ are assumed to be *known* quantities. (The parameters $\{\mu_j, \Sigma_j, P(j)\}_{j=1}^{J}$ in the Gaussian mixture model can be fit to the data by using standard EM techniques [1].)

As in [2], we furthermore describe the *conditional* density functions $\{\pi(x|k)\}$ as Gaussian mixtures. Assume that the density function of the data, conditioned on partition k, is written as

$$\pi(x|k) = \sum_{j=1}^{J} p(x|j)p_k(j) \tag{9}$$

where the mixture coefficients $\{p_k(j)\}$ sum to unity. Inserting Eq. 9 into Eq. 1 gives

$$\pi(x) = \sum_{k=1}^{K} \pi(x|k)\pi(k) = \sum_{j=1}^{J} \left(\sum_{k=1}^{K} p_k(j)\pi(k) \right) p(x|j) \ . \tag{10}$$

From the equivalence of Eq. 6 and Eq. 10, we have that

$$p(j) = \sum_{k=1}^{K} p_k(j)\pi(k) \quad j = 1, \cdots, J \tag{11}$$

By using Bayes' theorem and the Gaussian mixture model (GMM) in Eq. 9, we can express the unknown partition posteriors $\pi(k|x)$ in terms of the known kernel posteriors $p(j|x)$:

$$\begin{aligned}
\pi(k|x) &= \frac{\pi(x|k)\pi(k)}{\pi(x)} = \frac{\sum_{j=1}^{J} p(x|j)p_k(j)\pi(k)}{\pi(x)} \\
&= \sum_{j=1}^{J} \frac{p(j|x)}{p(j)} p_k(j)\pi(k) \\
&= \sum_{j=1}^{J} W_{kj} p(j|x) \ , \qquad (12)
\end{aligned}$$

where

$$W_{kj} = \frac{p_k(j)\pi(k)}{p(j)} \qquad (13)$$

for $k = 1, \cdots, K$ and $j = 1, \cdots, J$. We have the constraints

$$W_{kj} \in [0,1] \qquad (14)$$

and

$$\sum_{k=1}^{K} W_{kj} = \frac{\sum_{k=1}^{K} p_k(j)\pi(k)}{p(j)} = 1 \qquad (15)$$

For convenience, we introduce the unconstrained variables $\{\theta_{kj}\} \in (-\infty, \infty)$ and let

$$W_{kj} = \frac{\exp(\theta_{kj})}{\sum_{k'} \exp(\theta_{k'j})} \ . \qquad (16)$$

The energy function in Eq. 5 is a function of the variables $\{\theta_{kj}\}$. For a given data set $\{x_1, \cdots, x_N\}$, and a fixed number of partitions K, we want to find the values $\{\theta_{kj}\}$ that minimize

$$E(\{\theta_{kj}\}) = \begin{cases} H(k|x_n) & \text{if } \{n : \pi(k|x_n) \geq \pi(k'|x_n)\} \neq \emptyset \, \text{for} \, \forall k = 1, \ldots, K \\ \log_2(K) & \text{otherwise} \end{cases} \qquad (17)$$

where

$$H(k|x_n) = -\frac{1}{N} \sum_{n=1}^{N} \left(\sum_{k=1}^{K} \pi(k|x_n) \log_2 \pi(k|x_n) \right) \qquad (18)$$

and

$$\pi(k|x_n) = \sum_{j=1}^{J} \frac{\exp(\theta_{kj})}{\sum_{k'} \exp(\theta_{k'j})} p(j|x_n) \ . \qquad (19)$$

2.3 Model Selection through Nested Binary Trees

So far we have only discussed the problem of assigning data to different partitions when the number of partitions is known. In this section, we consider the model selection part of data clustering.

As mentioned, there often exist many sensible ways of dividing a data set. We here propose a hierarchical algorithm that builds a binary tree of partitions at different scales. At each level of the tree, we divide an existing cluster into two new clusters using the methods above. The energy function E in Eq. 17 (with $K = 2$) should be small if the two new clusters are well-separated. We choose some suitable positive number E_0 as an upper threshold for the partitioning energy, and only split clusters if $E \leq E_0$. The final partitioning of the data set is given by the leaves of the tree.

We initialize the tree by choosing $X = \{x_n\}_{n=1}^N$ as the initial cluster. For binary trees, we set the number of partitions K to 2. The probability density function of the data points in the cluster is given by a Gaussian mixture with kernel posteriors $pp = \{p(j|x_n)\}_{j=1,n=1}^{J,N}$ and mixing coefficients $p = \{p(j)\}_{j=1}^J$.

The algorithm for building a nested tree of clusters or partitions can be stated as follows:

```
**Data_Partitioning**$(X, p, pp, K)$

1. $N$=length of $X$;   $J$=length of $p$.
2. Set the threshold $E_0$.
3. Randomly choose $\{\theta_{kj}\}_{k=1,j=1}^{K,J}$ from a standard normal distribution.
4. Minimize the energy function $E(\{\theta_{kj}\})$ by simulated annealing. Let

       $\{\hat{\theta}_{kj}\} = \text{argmin}_{\theta_{kj}} E(\{\theta_{kj}\})$ ;  $\hat{W}_{kj} = \frac{\exp(\hat{\theta}_{kj})}{\sum_{k'} \exp(\hat{\theta}_{k'j})}$
       $\hat{\pi}(k|x_n) = \sum_{j=1}^J \hat{W}_{kj} p(j|x_n)$ ;  $\hat{\pi}(k) = \frac{1}{N} \sum_{n=1}^N \hat{\pi}(k|x_n)$      (20)
       $\hat{E} = -\frac{1}{N} \sum_{n=1}^N \left( \sum_{k=1}^K \hat{\pi}(k|x_n) \log_2 \hat{\pi}(k|x_n) \right)$

   where $k = 1, \cdots, K$, $j = 1, \cdots, J$ and $n = 1, \cdots, N$.
5. If $\hat{E} > E_0$
       • Return $X$
```

Else

- Let

$$\begin{array}{rcl} G_k & = & \{j : \hat{W}_{kj} > 0\} \\ X_k & = & \{x_n \in X : \hat{\pi}(k|x_n) > \hat{\pi}(k'|x_n), \forall k' \neq k\} \\ \hat{p}_k & = & \{\hat{p}_k(j)\} = \left\{ \frac{\hat{W}_{kj} p(j)}{\hat{\pi}(k)} \right\} \ , j \in G_k \\ pp_k & = & \{p(j|x_n)\} \ , j \in G_k \ , x_n \in X_k \end{array} \qquad (21)$$

  where $k = 1, \cdots, K$.

- Return $\{\{$**Data_Partitioning**$(X_k, \hat{p}_k, pp_k, K)\} \ , k = 1, \cdots, K\}$

3 Experiments

In this section, we apply our data partitioning model to a complex data set that consists of clusters of different sizes and types (ring-shaped, elongated and compact).

3.1 Data set

We use 20 Gaussians to generate a data set with 1000 data points; see Fig. 1. The Gaussian mixture is given by

- 8 Gaussians, each with weight 0.05, that generate data in a ring structure;
- 1 Gaussian with weight 0.1 that creates data in the center of the ring;
- 10 Gaussians, each with weight 0.04, that generate data in two semi-circular clusters;
- 1 isolated Gaussian, with weight 0.1, near the semi-circles.

In this example, we assume that the Gaussian mixture model (GMM) is already known. In more general cases, one can use standard EM techniques to fit a GMM to the data [1].

3.2 Building a Binary Tree of Clusters

The algorithm in Sec. 2.3 with the stopping criterion $E > E_0 = 0.01$ bits leads to a nested binary tree with four levels of clusters (Fig. 2). On the first level (or the root of the tree), we have the original dataset.

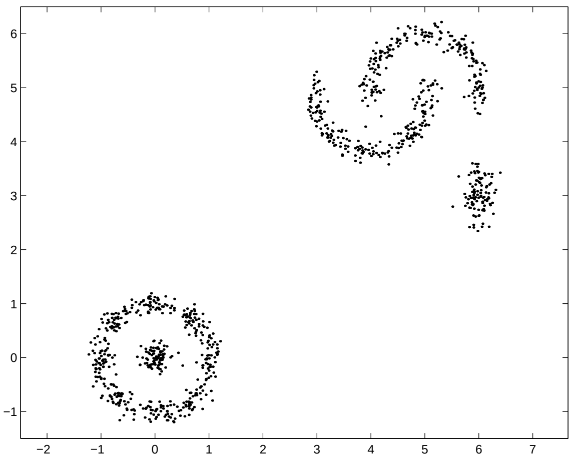

Figure 1: Data set

We denote the initial cluster by **root**. The energy value below the figure represents the minimum partitioning energy E (defined according to Eq. 5) for dividing the dataset into two new clusters. As the energy is less than E_0, we let these two clusters form Level 2 in the tree. We denote the new clusters by **0** and **1**. The partitioning energies of these clusters are also less than E_0. Thus, on Level 3 we have four clusters: **00**, **01**, **10** and **11**, with energies $5 \cdot 10^{-2}$, 1, 10^{-4}, and 1 bits, respectively. As the energy of the cluster **10** is less than E_0, we continue splitting this cluster into two new clusters. On Level 4, we have the new clusters **100** and **101**, which both have partition energies larger than E_0. The final partitioning is defined by the leaves of the tree, that is by clusters **00**, **01**, **100**, **101**, and **11**.

Note that the iterative scheme produces a *hierarchy of nested clusterings*. Each cluster on level $L > 1$ in the tree is a subset of a cluster on level $L' < L$. By definition, we always start by splitting the partition that requires the least energy. Raising the threshold E_0 will produce a finer final partitioning with a larger number of clusters, and lowering the threshold leads to a coarser partitioning.

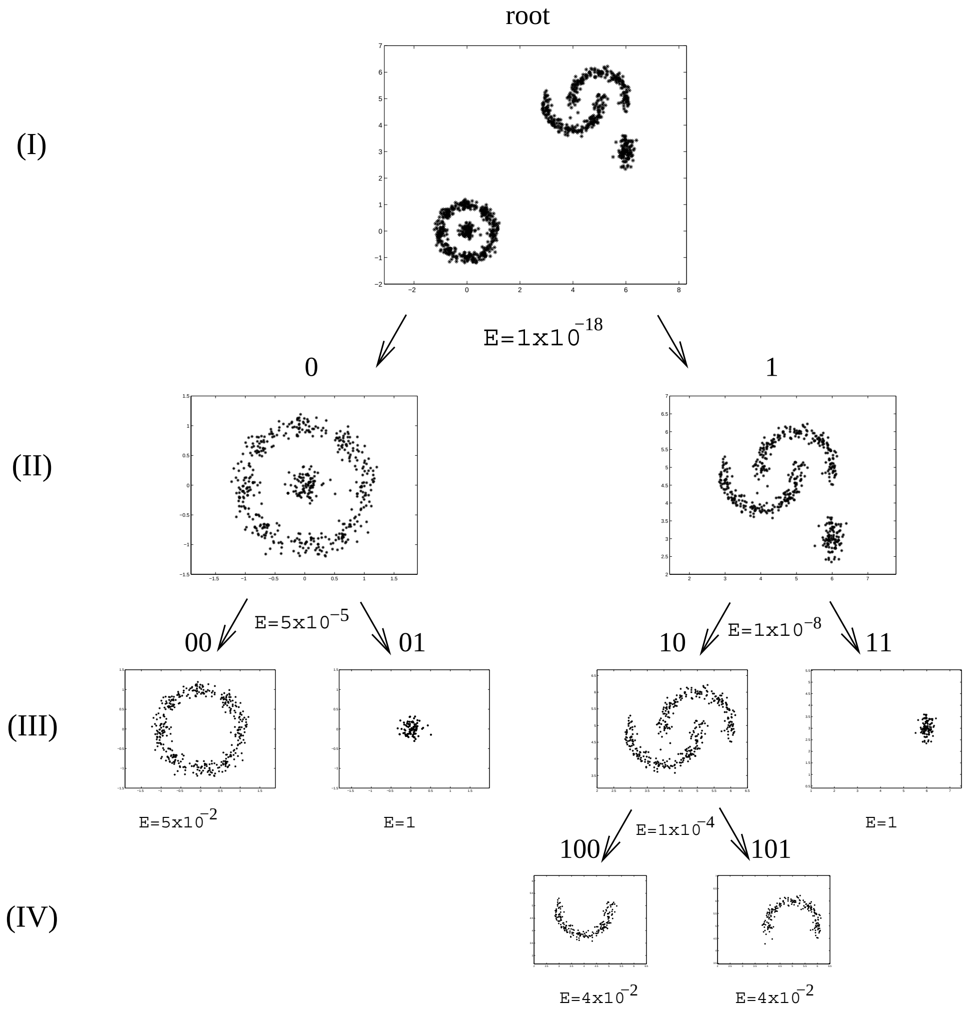

Figure 2: Hierarchical data partitioning. The value below each figure represents the minimum partitioning energy E for dividing the corresponding dataset into two new clusters; the binary numbers above the figure is used to label the clusters. We apply the iterative scheme in Sec. 2.3 until all clusters have a partitioning energy $E > E_0$, for some threshold E_0. The final partitioning of the data consists of the leaves of the tree. In this case, we have the partitioning $\{\mathbf{00}, \mathbf{01}, \mathbf{100}, \mathbf{101}, \mathbf{11}\}$.

4 Conclusions

We have presented a general methodology for partitioning data into clusters. The algorithm which originates in work by S. Roberts et al. is model-free and scales well with the dimensionality of the data space. Each partition is expressed as a linear combination of a set of fixed kernel functions.

In the current implementation, we use the partition entropy conditioned on the data as a criterion for data partitioning. Empty partitions are penalized and thus not allowed in the algorithm. We should point out that, our model splits clusters solely based on their "connectedness". It can therefore deal with clusters of general shapes and sizes. The only requirement on the data is that clusters that belong to different classes are separated by a region with a lower density of points.

For the model selection part, we build a hierarchy of nested clusterings: A cluster on a finer scale is a subset of a cluster on a coarser scale. The final partitioning is given by the leaves of the tree, where a leave is defined as a cluster with a "partitioning energy" that exceeds a certain threshold.

There are many advantages in a hierarchical partitioning model: Often it makes more sense to produce a hierarchy of nested clusterings, rather than a single "optimal" clustering. There may for example exist several reasonable ways of partitioning the given data, depending on which scale (coarse or fine) you are viewing the system. By producing a complete family of clusterings, we also have the choice of incorporating context into the final clustering. An "expert" can, for example, later prune the tree and decide on a partitioning of the data that best fits the specific problem.

Our hierarchical partitioning algorithm is computationally efficient. The total computational cost to build a tree is $O(NJ^2K)$, where N is the number of data points, J is the number of kernel functions, and K is the number of subclusters for each parent cluster in the tree. In our binary tree, $K = 2$ and the computational cost is $O(NJ^2)$. Compare this with model selection algorithms by brute force (e.g. [2]) which require $O(NJ^3)$ operations.

References

[1] C. M. Bishop. *Neural Networks for Pattern Recognition.* Oxford University Press, 1995.

[2] S. J. Roberts, R. Everson, and I. Rezek. Maximum certainty data partitioning. *Pattern Recognition*, 33(5):833–839, 1999.

[3] S. J. Roberts, C. Holmes, and D. Denison. Minimum Entropy Data Partitioning using Reversible Jump Markov Chain Monte Carlo. Technical Report PARG-00-7, Robotics Research Group, Department of Engineering Science, University of Oxford, UK, 2000.

A. Appendix

In [3], the authors minimize the energy function

$$V' = H(k|x) - H(k) = -MI(k,x) \leq 0 \; , \tag{22}$$

where

$$H(k|x) = -\int_{\Omega} \left(\sum_{k=1}^{K} \pi(k|x) \log_2 \pi(k|x) \right) \pi(x)\, dx \tag{23}$$

is the conditional entropy of the partitions $k = 1, \ldots, K$ given the data;

$$H(k) = -\sum_{k=1}^{K} \pi(k) \log_2 \pi(k) \tag{24}$$

is the Shannon entropy of the partitions; and $MI(k,x)$ represents the mutual information of k and x. Below we show with a one-dimensional example that minimizing E, or equivalently, maximizing the mutual information $MI(k,x)$, favors partitions with equal weights. The results imply that the energy function in Eq. 22 may not be suitable for datasets with outliers and clusters of many different sizes.

A1. Maximizing the Mutual Information $MI(k,x)$ Favors Partitions with Equal Weights: A 1D Example

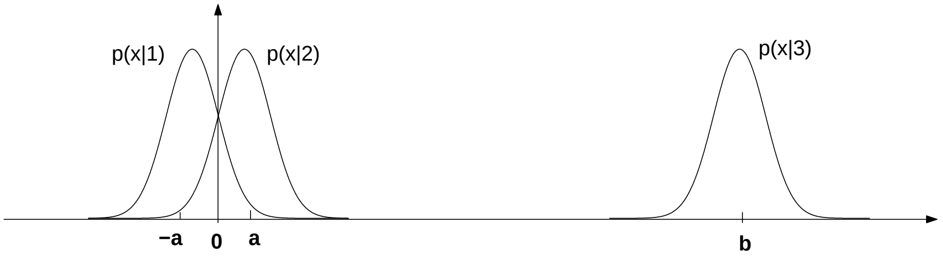

Figure 3: A one-dimensional example where the data is generated by two Gaussian distributions that are overlapping, and a third Gaussian distribution that is far away and of less weight.

Consider a one-dimensional example where the data is generated by two Gaussian distributions that are overlapping, and a third Gaussian distribution that is far away and of less weight; see Fig. 3. The random variable $x \in (-\infty, \infty)$ for the data has a density function

$$p(x) = \alpha \left[p(x|1) + p(x|2) \right] + \beta p(x|3) \ , \tag{25}$$

where

$$\left\{ \begin{array}{lllll} p(x|1) & = & g_1(x) & = & \frac{1}{\sqrt{2\pi}} \exp(-\frac{(x+a)^2}{2}) \\ p(x|2) & = & g_2(x) & = & \frac{1}{\sqrt{2\pi}} \exp(-\frac{(x-a)^2}{2}) \\ p(x|3) & = & g_3(x) & = & \frac{1}{\sqrt{2\pi}} \exp(-\frac{(x-b)^2}{2}) \end{array} \right. \ , \tag{26}$$

$0 < \beta \ll \alpha$, $2\alpha + \beta = 1$, $0 < a < 1$ and $b \gg 1$.

Below we calculate the energy function V' for two different clustering solutions with $K = 2$. In the first case, the two overlapping Gaussians g_1 and g_2 are assigned to one partition, and the third Gaussian g_3 that is far away ($b \gg a$) is assigned to another. We call this "Ideal Partitioning", as this is the most sensible way of dividing the observed data x into two classes. In the second case ("Partitioning with Equal-Sized Clusters"), the first two clusters g_1 and g_2 are assigned to different classes; the third cluster g_3 with weight $\beta \ll \alpha$ is merged with the second Gaussian g_2. This is not a desirable solution, as data samples from g_2 are clearly different from samples from g_3. Furthermore, the first two Gaussians are only weakly separated ($a < 1$).

Case 1: "Ideal Partitioning"

We use the same notations as in Sec. 2. Assume that the class-conditional probability density functions are given by

$$\left\{ \begin{array}{l} \pi(x|1) = \gamma p(x|1) + (1-\gamma) p(x|2) \\ \pi(x|2) = p(x|3) \end{array} \right. . \tag{27}$$

From Eq. 1, we then have

$$p(x) = \gamma \pi(1) p(x|1) + (1-\gamma) \pi(1) p(x|2) + \pi(2) p(x|3) \ . \tag{28}$$

Identifying the mixture coefficients above with the coefficients in Eq. 25 gives

$$\pi(1) = 2\alpha, \ \pi(2) = \beta, \ \gamma = \frac{1}{2} \ . \tag{29}$$

As the Gaussian density functions decrease exponentially, we only need to consider data x that are around either 0 or b in the integral in Eq. 23. For x around 0, $\pi(1|x) \approx 1$ and $\pi(2|x) \approx 0$. For x around b, $\pi(1|x) \approx 0$ and $\pi(2|x) \approx 1$. Thus, $H(k|x) \approx 0$ and

$$\begin{aligned} V'_{ideal} &= H(k|x) + \sum_{k=1}^{2} \pi(k) \log_2 \pi(k) \\ &\approx (2\alpha) \log_2 (2\alpha) + \beta \log_2 (\beta) \end{aligned} \tag{30}$$

Case 2: "Partitioning with Equal-Sized Clusters"

Here we assign the first Gaussian to one partition, and the second and the third Gaussians to a different partition. This leads to two approximately equal-sized clusters, as $\beta \ll \alpha$. The class-conditional probability density functions of the data are given by

$$\left\{ \begin{array}{l} \pi(x|1) = p(x|1) \\ \pi(x|2) = \delta p(x|2) + (1-\delta) p(x|3) \ . \end{array} \right. \tag{31}$$

Identification of mixture coefficients as in "Case 1" gives

$$\pi(1) = \alpha, \ \pi(2) = \alpha + \beta, \ \delta = \frac{\alpha}{\alpha + \beta} \ . \tag{32}$$

As before, we use the properties of the Gaussian density function. For values of x around 0, $p(x|3) \approx 0$. Thus,

$$\begin{array}{l} \pi(1|x) = \frac{\pi(x|1)\pi(1)}{p(x)} = \frac{\alpha p(x|1)}{\alpha p(x|1) + \alpha p(x|2) + \beta p(x|3)} \approx \frac{g_1(x)}{g_1(x) + g_2(x)} = \frac{1}{1 + \exp(2ax)} \\ \pi(2|x) = \frac{\pi(x|2)\pi(2)}{p(x)} = \frac{\alpha p(x|2) + \beta p(x|3)}{\alpha p(x|1) + \alpha p(x|2) + \beta p(x|3)} \approx \frac{g_2(x)}{g_1(x) + g_2(x)} = \frac{1}{1 + \exp(-2ax)} \end{array} \tag{33}$$

For values of x around b, $p(x|1) \approx 0$ and $p(x|2) \approx 0$. Thus,

$$\pi(1|x) = \frac{\alpha p(x|1)}{\alpha p(x|1) + \alpha p(x|2) + \beta p(x|3)} \approx 0, \quad \pi(2|x) = 1 - \pi(1|x) \approx 1 \ . \tag{34}$$

This leads to

$$\begin{aligned} H(k|x) &= -\int_{-\infty}^{\infty}\left(\sum_{k=1}^{2}\pi(k|x)\log_2\pi(k|x)\right)\alpha[p(x|1)+p(x|2)]\,dx \\ &\quad -\int_{-\infty}^{\infty}\left(\sum_{k=1}^{2}\pi(k|x)\log_2\pi(k|x)\right)\beta p(x|3)\,dx \\ &\approx -\alpha\int_{-\infty}^{\infty}\left(\frac{g_1}{g_1+g_2}\log_2\frac{g_1}{g_1+g_2}+\frac{g_2}{g_1+g_2}\log_2\frac{g_2}{g_1+g_2}\right)(g_1+g_2)\,dx \\ &= \frac{2\alpha}{\sqrt{2\pi}}I(a) \end{aligned} \tag{35}$$

and

$$\begin{aligned} V'_{equal} &= H(k|x)+\sum_{k=1}^{2}\pi(k)\log_2\pi(k) \\ &\approx \frac{2\alpha}{\sqrt{2\pi}}I(a)+[(\alpha+\beta)\log_2(\alpha+\beta)+\alpha\log_2(\alpha)]\ , \end{aligned} \tag{36}$$

where

$$I(a)=\int_{-\infty}^{\infty}\exp\left(-\frac{(x+a)^2}{2}\right)\log_2(1+\exp(2ax))\,dx\ . \tag{37}$$

Cases where $V'_{equal} \leq V'_{ideal}$

We get a "bad" partitioning with approximately equal-sized clusters when $V'_{equal} \leq V'_{ideal}$. In terms of β and a, this corresponds to

$$g(\beta) \geq \frac{1}{\sqrt{2\pi}}(a)\ , \tag{38}$$

where

$$g(\beta)=\frac{1}{2}\left[1+\log_2(1-\beta)-\frac{1+\beta}{1-\beta}\log_2\left(\frac{1+\beta}{2}\right)-\frac{2\beta}{1-\beta}\log_2\beta\right]\ . \tag{39}$$

The derivative

$$g'(\beta)=\frac{1}{(1-\beta)^2}\log_2\left(\frac{2\beta}{1+\beta}\right)\ . \tag{40}$$

is a strictly decreasing function. We denote the inverse of g by g^{-1}. The condition for $V'_{equal} \leq V'_{ideal}$ can then be rewritten as

$$\beta \leq \beta_0 \equiv g^{-1}(\frac{1}{\sqrt{2\pi}} I(a)) \ . \tag{41}$$

It can be shown analytically that $I(a)$ is also a strictly decreasing function, and that β_0 therefore is a strictly increasing function of a. Numerically, we get the curve in Fig. 4. Note that β does not have to be very small for a bad partitioning with approximately equal-sized clusters to occur. Take for example the case where $a = 0.5$ and $\beta \approx 0.03$.

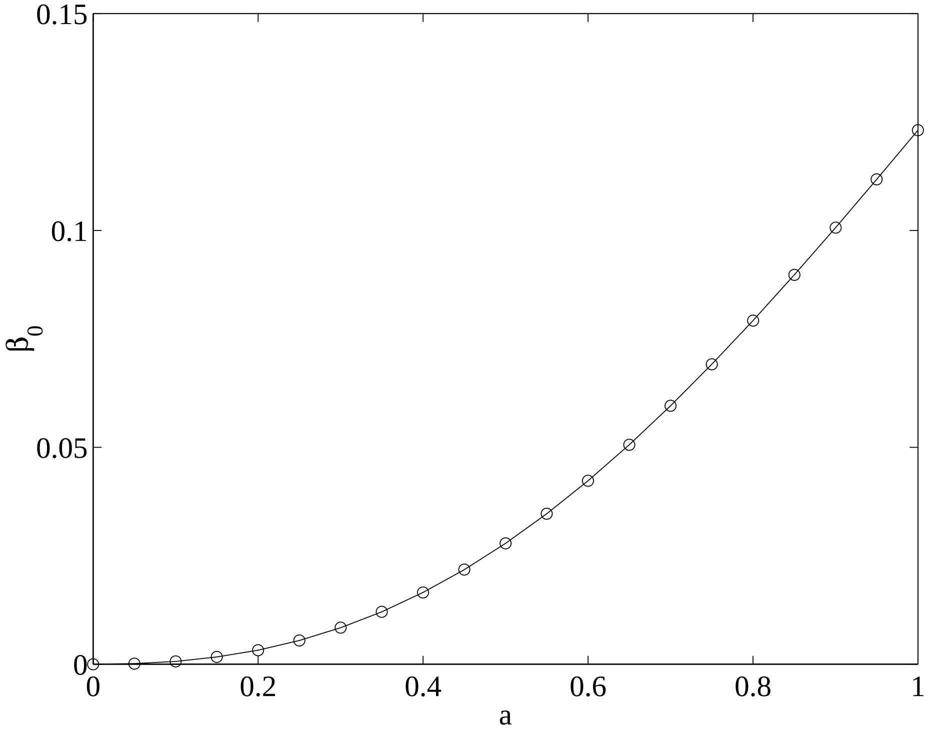

Figure 4: The "mutual information" energy function by S. Roberts et al. favors equal-sized partitions. In the one-dimensional example above, we get a bad partitioning (with approximately equal-sized clusters) of the data if the weight β of the third Gaussian is smaller than β_0. The figure shows the relation between β_0 and a ($a < 1$).