Newtonian Clustering: An Approach based on Molecular Dynamics and Global Optimization

K. Blekas and I. E. Lagaris
Department of Computer Science, University of Ioannina
P.O.Box 1186, Ioannina 45110 - GREECE
E-mail: {kblekas,lagaris}@cs.uoi.gr

Abstract

Given a data set, a dynamical procedure is applied to the data points in order to shrink and separate, possibly overlapping clusters. Namely, Newton's equations of motion are employed to concentrate the data points around their cluster centers, using an attractive potential, constructed specially for this purpose. During this process, important information is gathered concerning the spread of each cluster. In succession this information is used to create an objective function that maps each cluster to a local maximum. Global optimization is then used to retrieve the positions of the maxima that correspond to the locations of the cluster centers. Further refinement is achieved by applying the EM-algorithm to a Gaussian mixture model whose construction and initialization is based on the acquired information. To assess the effectiveness of our method, we have conducted experiments on a plethora of benchmark data sets. In addition we have compared its performance against four clustering techniques that are well established in the literature.

Keywords*: Clustering, Molecular Dynamics, Global Optimization, Order Statistics.*

1 Introduction

With the advent of the Internet and the World Wide Web, scientific data from a wide range of fields, have become easily accessible. The interest of the scientific community for the problem of clustering is reflected by the growing appearance of related monographs [1, 2, 3, 4, 5], journal articles and conferences. This convenience has further raised the interest and expanded the audience of clustering techniques. Clustering can be viewed as the identification of existing intrinsic groups in a set of unlabeled data. Associated methods are often based on intuitive approaches that rely on specific assumptions and on the particular characteristics of the data sets. This in turn implies that the corresponding

algorithms depend crucially on some parameters that must be properly chosen anew for each problem.

A plethora of clustering approaches have been introduced over the past decades. Hierarchical methods, arrange the data in a tree-like structure according to some similarity criteria; an example is the *"Single Linkage Clustering"*. Methods based on partitioning, relocate iteratively the data points into clusters until the optimum position of some cluster representatives (e.g. centers, envelopes, etc.) is found; the popular "K-means" algorithm for instance belongs to this category. Model-based methods, assume that the data are generated from a mixture of probability distributions with each component corresponding to a different cluster [2, 3, 5]; in these methods the *Expectation-Maximization* (EM) algorithm [6, 5] is the most frequent choice for tuning the mixture parameters.

A fundamental issue in clustering is the determination of the number of clusters, denoted from here on by K, in a given data set. The likelihood alone can not be used to determine the number of clusters since it is a monotonically increasing function of K. Most clustering methods assume that this number is a priori known, and then try to place K clusters in the space defined by the data. In the literature quite a few methods have been proposed for determining the number of clusters. A common approach is to apply a clustering technique using a range of values for K and select the solution that performs best according to certain evaluation criteria [7, 8, 9, 10]. Such criteria in common use, are the Akaike Information Criterion [11], the Bayesian Information Criterion [12] and the cross-validation criterion [13]. Under the Bayesian framework, K may be treated as a random variable. This induces a posterior distribution for the model uncertainty that can be used for model selection. Several numerical approximation schemes have been considered, for calculating the posterior distribution, such as the Laplace approximation [14], Markov Chain Monte Carlo (MCMC) methods [15, 16], and variational approaches[17]. Recently, a modified EM scheme, that incorporates a model selection criterion (the minimum message length - MML), has been proposed in [18].

The problem of determining the number of clusters in a set of data points, is still open. Apart from being a challenging problem on its own, it appears as a recuring subproblem in many applications from various fields, for instance in pattern recognition [2, 4], machine learning [3], image analysis [19], bioinformatics [20], etc. We propose in this article a new technique which in the data preprocessing phase, determines both the number of clusters as well as the approximate location of their centers. During this phase in order to "shrink" existing clusters and obtain an estimate of their spatial spread, a "Molecular Dynamics"

(MD) approach is applied, by considering that the data points correspond to point particles interacting via an attractive short range two-body potential, whose construction is of key importance. The path traveled by each particle offers useful information for constructing an associated underlying probability density function (pdf). The superposition of these pdfs, results in a multimodal function, where each maximum corresponds to a different cluster with its center approximately located at the position of the peak. To count the number of clusters and retrieve their associated centers, stochastic global optimization methods, that recover all the local maxima [21, 22, 23, 24, 25, 26] are proper. Having determined the number of clusters and the approximate location of their centers, a local-based clustering technique may be used for further refinement. We have tested our method on a suite of benchmarks, taking in account a variety of cases, with excellent results. Comparisons have been made in two directions. To test the performance in estimating K, we compared with two methods that are well established in the literature, namely the "quantum clustering" [27, 28] and the "MML-EM" [18] technique. We have also made a comparison with the "Greedy EM" [29, 30] and the "K-means-initialized EM" to test the quality of the solution.

In section 2 we present the rationale and the formulation of our approach, in section 3 we lay out an algorithmic description and in section 4 we report results obtained by applying our method to several data sets. Finally in section 5 we summarize and conclude with some remarks.

2 Rationale and Formulation

The clustering problem can be stated in the following manner:
Given a set of M data points $A = \{x_i | x_i \in R^N,\ i = 1, \cdots, M\}$, find subsets $A_k \subset A$, containing points with one or more common properties.
These subsets are called clusters. In this study we consider that the properties of a single cluster, may be described implicitly via a *unimodal* probability distribution, i.e. a distribution with a unique peak. Hence a pdf, that could reproduce a given data set, may be expressed as a linear combination of these unimodal cluster distributions $\phi_j(x)$.

$$P(x) = \sum_{j=1}^{K} \pi_j \phi_j(x), \text{ with } \sum_{j=1}^{K} \pi_j = 1, \text{ and } \pi_j > 0 \ . \tag{1}$$

Note that the number of terms K in the sum, corresponds to the number of clusters, which in general is unknown. This corresponds to a mixture model with K distributions,

where the data generation procedure is performed by first selecting a component j, with probability π_j and then sampling x from $\phi_j(x)$.

Now consider the function

$$f(x) = \sum_{i=1}^{M} \alpha_i \delta(x - x_i) \ , \tag{2}$$

with $\sum_{i=1}^{M} \alpha_i = 1$, and $\alpha_i > 0$, where the sum runs over all the data points x_i, and $\delta(x)$ is the Dirac's delta function. This is an extreme case of a pdf that exactly reproduces the data. Of course it conveys no feature information, since it corresponds to single point clusters, i.e. to $K = M$. The delta functions can be approximated by narrow normal distributions in Eq. (2) so as to have:

$$f(x) = \sum_{i=1}^{M} \alpha_i \mathcal{N}(x|x_i, \Sigma_i) \ . \tag{3}$$

The number of peaks of $f(x)$ (initially M), decreases as the normal distributions become wider, since the Gaussians will start to interfere. The spread of the Gaussians is dictated by the parameters controlling the covariance matrices Σ_i.

Suppose that y_{jl} denotes the l^{th} point of the j^{th} cluster. Hence the sets $\{x_i, i = 1, \cdots, M\}$ and $\{y_{jl}, j = 1, \cdots, K \text{ and } \ l = 1, \cdots, n_j\}$ n_j being the number of points belonging to the j^{th} cluster, are identical. We may define for convenience a correspondence between the single index $\{i\}$ to the couple of indices $\{j, l\}$ via the equality $x_i = y_{jl}$. Gaussians centered at points that belong to the same cluster should have suitable Σ's so as their superposition to form only one peak. Namely the sum $B_j(x) = \sum_{l=1}^{n_j} \beta_{jl} \mathcal{N}(x, y_{jl}, \Sigma_{jl})$ should have a unique peak and hence the corresponding (j^{th}) cluster may be represented by a pdf proportional to $B_j(x)$. Note that β_{jl}, are the α_i coefficients expressed in the double index notation. By comparing Eqs. (1) and (3) we may readily deduce that

$$\phi_j(x) = \frac{1}{\sum_{l=1}^{n_j} \beta_{jl}} \sum_{l=1}^{n_j} \beta_{jl} \mathcal{N}(x|y_{jl}, \Sigma_{jl}) \ . \tag{4}$$

This result is similar to that obtained by the Parzen window approach [31]; it simply restates that the cluster pdf may be expressed as a linear combination of Gaussians. The important additional information gained by the above analysis is that this combination should be unimodal.

Finding proper values for the Σ parameters is crucial. If the Σ's are chosen so that the Gaussians are too wide, one will end up with only one cluster, since the overlap of the Gaussians will be significant. On the other hand, too narrow Gaussians will lead to too many clusters since even closely lying points will contribute to different peaks. Note that a similar problem exists in the quantum clustering [27, 28] approach, where the σ parameter governs the performance of the algorithm. If the proper Σ' s were known, then the number of clusters would be determined by the number of peaks of $f(x)$ in Eq. (3).

2.1 Shrinking the clusters via Molecular Dynamics

In order to overcome the above problem we apply a dynamic technique, inspired from the MD simulation approach used in Physics and Chemistry to study many-body classical systems. (See for instance D. W. Heermann's book [32]). Consider that the data points correspond to particles of unit mass, interacting via a two-body attractive, short-range potential. Let the potential between particles located at points r_i and r_j be of a simple Gaussian form given by:

$$V_{ij} = -e^{-\frac{1}{2}||r_i - r_j||_\sigma^2}, \quad \text{where} \quad ||r||_\sigma^2 = r^T \Lambda^{-2} r, \text{ and } \Lambda = diag\{\sigma_1, \sigma_2, \cdots, \sigma_N\}. \tag{5}$$

The shrinking process is governed by σ_k which is taken to be the averaged (over all points) k^{th} component of the vector between a point and its m^{th} nearest neighbor, and m is chosen in a way to be explained in section 2.2.

Newton's equations of motion are:

$$\frac{d^2 r_i(t)}{dt^2} = -\nabla_i \sum_{\substack{j=1 \\ j \neq i}}^{M} V_{ij} \equiv F_i, \ \forall i = 1, 2, \cdots, M \ . \tag{6}$$

The initial positions are taken to be the data points, i.e. $r_i(t=0) = x_i$ $(\forall i = 1, \ldots, M)$ and the initial velocities ($v_i \equiv \frac{dr_i}{dt}$) are set to zero. We integrate the equations of motion in small time steps δt, considering that the forces F_i remain constant during this short time interval. At each step we reset the velocities to zero in order to avoid artifacts due to "heating". Hence we obtain the following scheme:

$$r_i(t + \delta t) = r_i(t) + \frac{1}{2} \delta t^2 F_i \ . \tag{7}$$

Since the interaction is attractive, after a time period T, the particles belonging to the same cluster will concentrate around the cluster center. So an initially spread–out cluster

is being "shrunk" as a result of the MD preprocessing. The simulation terminates, when the steps become too small and further iterations hardly make any difference. The following criterion is used:

$$\frac{\sum_{i=1}^{M} |r_i(t+\delta t) - r_i(t)|}{\sum_{i=1}^{M} |r_i(t+\delta t) - x_i|} < \eta \ , \tag{8}$$

η being a small positive number ($\eta = 0.01$ is being used in our experiments). The absolute values of the components of the vector $r_i(T) - x_i$ are used to construct a diagonal covariance matrix Σ_i for each sample point, i.e. $[\Sigma_i]_{kk} = |r_i(T)_k - x_{ik}|$, $\forall\ k = 1, \ldots, N$. The objective function is constructed using the processed data $r_i \equiv r_i(T)$ in Eq. (3), instead of the original x_i, and by choosing $a_i = \frac{\sqrt{|\Sigma_i|}}{\sum_{j=1}^{M} \sqrt{|\Sigma_j|}}$. Therefore, apart from a constant overall normalization factor, $f(x)$ is given by:

$$f(x) \propto \sum_{i=1}^{M} \exp\{-\frac{1}{2}(x - r_i)^T \Sigma_i^{-1} (x - r_i)\} \ . \tag{9}$$

The positions of the maxima of $f(x)$ are the estimates for the cluster centers, while their multitude determines the number of clusters. We mention in passing, that the value of $f(x)$ at a maximum, can distinguish outlier groups from real clusters, since the value for an outlier group will be significantly smaller.

2.2 Determining the range of the potential

The range of the potential is crucial for the success of the method and has to be chosen carefully. Sparse data sets require a longer range than dense data sets. Hence the range depends on the data set. A measure for sparsity is offered by the various nearest–neighbor (NN) distances. Setting σ equal to an m^{th} order NN–distance averaged over all points, is a way to take into account the density of the data set.
Let $d_j^{(i)}$ be the distance between a point at x_i and its j^{th} NN. Clearly we have:

$$d_1^{(i)} < d_2^{(i)} < \cdots < d_{M-1}^{(i)}, \quad \forall i = 1, 2, \cdots, M-1 \ . \tag{10}$$

The mean and the mean squared j^{th} NN–distance in a set of M points are given by:

$$< d_j >= \frac{1}{M} \sum_{i=1}^{M} d_j^{(i)}, \ \ \forall\ j = 1, 2, \cdots, M-1 \ , \tag{11}$$

and

$$< d_j^2 >= \frac{1}{M}\sum_{i=1}^{M}(d_j^{(i)})^2, \quad \forall\ j = 1, 2, \cdots, M-1\ . \tag{12}$$

We have studied the quantity

$$\tilde{\sigma}^2_{M,m} = \frac{1}{m}\sum_{k=1}^{m}\left(< d_k^2 > - < d_k >^2\right) \quad \forall\ m = 1, 2, \cdots, M-1\ , \tag{13}$$

using *order statistics* and, in the case of a single cluster we find that:

$$\tilde{\sigma}^2_{M,m} = \alpha(M)(m+1)^2 + \beta(M)(m+1)\ . \tag{14}$$

This result will be used to determine a proper value m^* that serves our purpose. When two or more clusters exist, $\tilde{\sigma}^2_{M,m}$ is given by a superposition of translated quadratics as illustrated in Fig.1(b). Since $\frac{\tilde{\sigma}^2_{M,m}}{m+1}$ is linear in m, its second difference vanishes. Hence in our experiments m^* is taken to be the smallest value of m for which the second difference of $\dfrac{\tilde{\sigma}^2_{M,m}}{m+1}$ with respect to m vanishes. In practice we search for the smallest m such that:

$$|\frac{\tilde{\sigma}^2_{M,m+1}}{m+2} + \frac{\tilde{\sigma}^2_{M,m-1}}{m} - 2\frac{\tilde{\sigma}^2_{M,m}}{m+1}| < \epsilon\ |\frac{\tilde{\sigma}^2_{M,m}}{m+1}|\ . \tag{15}$$

To elucidate the analysis, consider $\tilde{y}_1, \tilde{y}_2, \cdots, \tilde{y}_{M-1}$ to be an i.i.d. sample from $\rho(\tilde{y}),\ \tilde{y} \geq 0$. If $y_1 \leq y_2, \leq \cdots, \leq y_{M-1}$ denotes the ordered sample, that is the *order statistics* of the original sample, then y_k is distributed according to the following pdf:

$$\mathcal{P}_{M-1,k}(y) = (M-1)\binom{M-2}{k-1}R(y)^{k-1}(1-R(y))^{M-1-k}\rho(y)\ , \tag{16}$$

where $R(y) = \displaystyle\int_0^y \rho(t)dt$, is the cumulative distribution function.
The first two moments are given by:

$$< y_k >\equiv \int_0^\infty y\mathcal{P}_{M-1,k}(y)dy \quad \text{and} \quad < y_k^2 >\equiv \int_0^\infty y^2\mathcal{P}_{M-1,k}(y)dy\ . \tag{17}$$

Given $\rho(y)$ one then can calculate $\tilde{\sigma}^2_{M,m}$. We have calculated $\tilde{\sigma}^2_{M,m}$ for several choices of $\rho(y)$ (performing the required integrations numerically). In all cases the functional dependence on m was in agreement with Eq. (14). Note that the coefficients $\alpha(M)$ and $\beta(M)$ in Eq. (14) depend on $\rho(y)$. If $\rho(y)$ is the uniform pdf in $(0, 1)$ it is possible to integrate

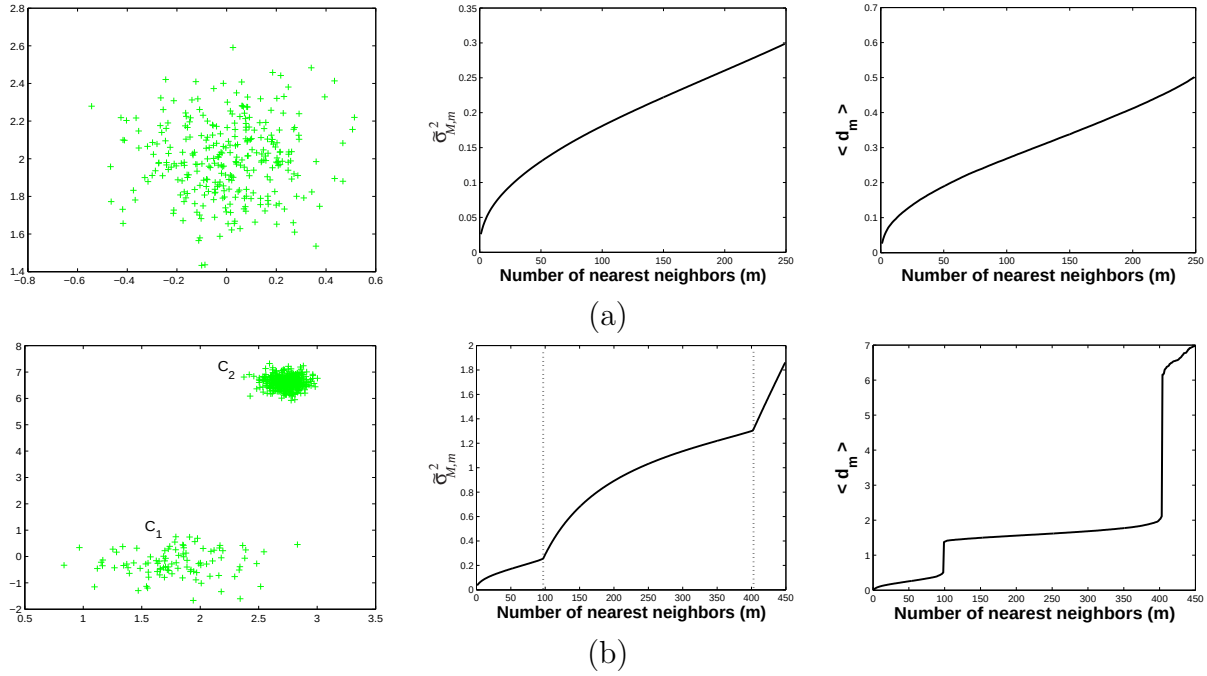


Figure 1: Two data sets along with plots of $\tilde{\sigma}^2_{M,m}$ and $< d_m >$.

analytically. After some algebra and using the result $\int_0^1 x^k(1-x)^l dx = \frac{k!l!}{(k+l+1)!}$, one obtains:

$$\alpha(M) = -\frac{1}{3M^2(M+1)} \quad \text{and} \quad \beta(M) = \frac{3M-1}{6M^2(M+1)} \; . \tag{18}$$

As an illustration we plot $\tilde{\sigma}^2_{M,m}$ and $< d_m >$ for the data sets shown in Fig.1(a) and 1(b). For the data set in Fig.1(a) that contains a single cluster, the expected quadratic form is recovered. Note that $< d_m >$ in this case is continuous. In the case of Fig.1(b) we observe two clusters, C_1 and C_2, with populations m_1 and m_2 respectively. Without loss of generality assume that $m_1 < m_2$. Then, as m is approaching m_1, we expect a jump in $< d_m >$, since points belonging to C_1 will start having their m^{th}-NN in C_2 which is a distance apart. Again when m approaches m_2 another jump is expected in $< d_m >$, since points in C_2 will have their m^{th}-NN in C_1. This effect is also apparent, not as evident however, in the plot of the cumulative quantity $\tilde{\sigma}^2_{M,m}$.

2.3 Fine Tuning

In the above analysis we have shown a method for estimating the number of clusters K, as well as their centers $\mu_j, j = 1, \cdots, K$, through a global optimization procedure. Although it is not really needed, we apply next a local-based clustering method for fine tuning and for determining the geometrical characteristics of each cluster in the framework of Gaussian mixture models. This final step is useful for reasons of comparison. In particular, we consider the mixture of K Gaussian components:

$$p(x|\Psi_K) = \sum_{j=1}^{K} \pi_j p(x|\mu_j, \tilde{\Sigma}_j) \ , \tag{19}$$

where the parameters $0 < \pi_j \leq 1$ represent the mixing weights satisfying $\sum_{j=1}^{K} \pi_j = 1$, while $\Psi_K = [\pi_1, \ldots, \pi_K, \{\mu_1, \tilde{\Sigma}_1\}, \ldots, \{\mu_K, \tilde{\Sigma}_K\}]$, i.e. the vector of all unknown parameters of the model. To maximize the resulting log-likelihood function we employ the EM algorithm with initial values determined from our solutions.

3 Algorithmic Description

Input: $X = \{x_1, x_2, \ldots, x_M\}$ with $x_i \in R^N$.

1. *Calculate σ and set δt*:

   - If ω_i^m is the position of the m^{th} nearest neighbor of the point located at x_i, calculate the mean and the mean squared m^{th} NN-distance: $< d_m > \equiv \frac{1}{M} \sum_{i=1}^{M} ||\omega_i^m - x_i||$ and $< d_m^2 > \equiv \frac{1}{M} \sum_{i=1}^{M} ||\omega_i^m - x_i||^2$, and finally $\tilde{\sigma}_{M,m}^2$, for $m = 1, 2, \cdots M-1$.
   - Find m^*, the smallest $m \geq 2$ satisfying criterion (15), with $\epsilon = 10^{-3}$. Set $\sigma_k = \frac{1}{M} \sum_{i=1}^{M} | \left(\omega_i^{m^*} - x_i\right)_k |, \quad \forall \ k = 1, 2. \cdots, N.$
   - Set the time step δt to a small value (typically $\delta t = 0.01$) and $r_i(0) = x_i$, $\forall i = 1, \ldots, M$.

2. *MD procedure.*
   Perform the following steps until criterion (8) is satisfied.

   (a) Compute forces according to Eq. (6).

   (b) Update positions using Eq. (7).

3. Let $r_i \quad \forall i = 1, 2, \cdots, M$ be the positions upon completion of the MD procedure. Estimate local diagonal covariances as:

$$[\Sigma_i]_{kk} = (r_i - x_i)_k^2 , \quad \forall\, i = 1, \ldots, M \quad \text{and} \quad \forall\, k = 1, \ldots, N \ .$$

4. Use a global optimization method to find all existing maxima of the function in Eq. (9), in the space defined by the data. The number of maxima determines K. The maxima positions are estimates for the locations of the cluster centers μ_j. In particular, we have used the *"Healed Topographical Multilevel Single Linkage"* (HTMLSL) global optimization method [26] which relies on the *Merlin optimization environment*[1] [33].

5. Apply the EM algorithm to a K-component Gaussian mixture model, with initial parameter values taken from the constructed pdf (Eq. 9).

4 Experiments and Results

Several experiments have been conducted in order to ascertain the effectiveness of our approach. We have considered both simulated multidimensional data sets as well as widely used benchmarks. Our experimental study addresses the following three issues.

1. The stability of the preprocessing phase.

2. The robustness of the method.

3. The competitiveness of the overall approach.

4.1 Testing the preprocessing procedure

The spread parameters σ_k of Eq. (5), that determine the range of the attractive potential used in the MD procedure to shrink the clusters, are quite important in our approach. For instance small values will result in the formation of too many small clusters, while large values will force neighboring clusters to fuse, underestimating so the number of clusters. Our experiments show that the σ_k values obtained using the proposed criterion in Eq. (15), are inside an interval of stability. This is illustrated in Fig. 2 for three simulated data sets, each containing $M = 500$ points in two dimensions.

[1] Available also from: http://merlin.cs.uoi.gr

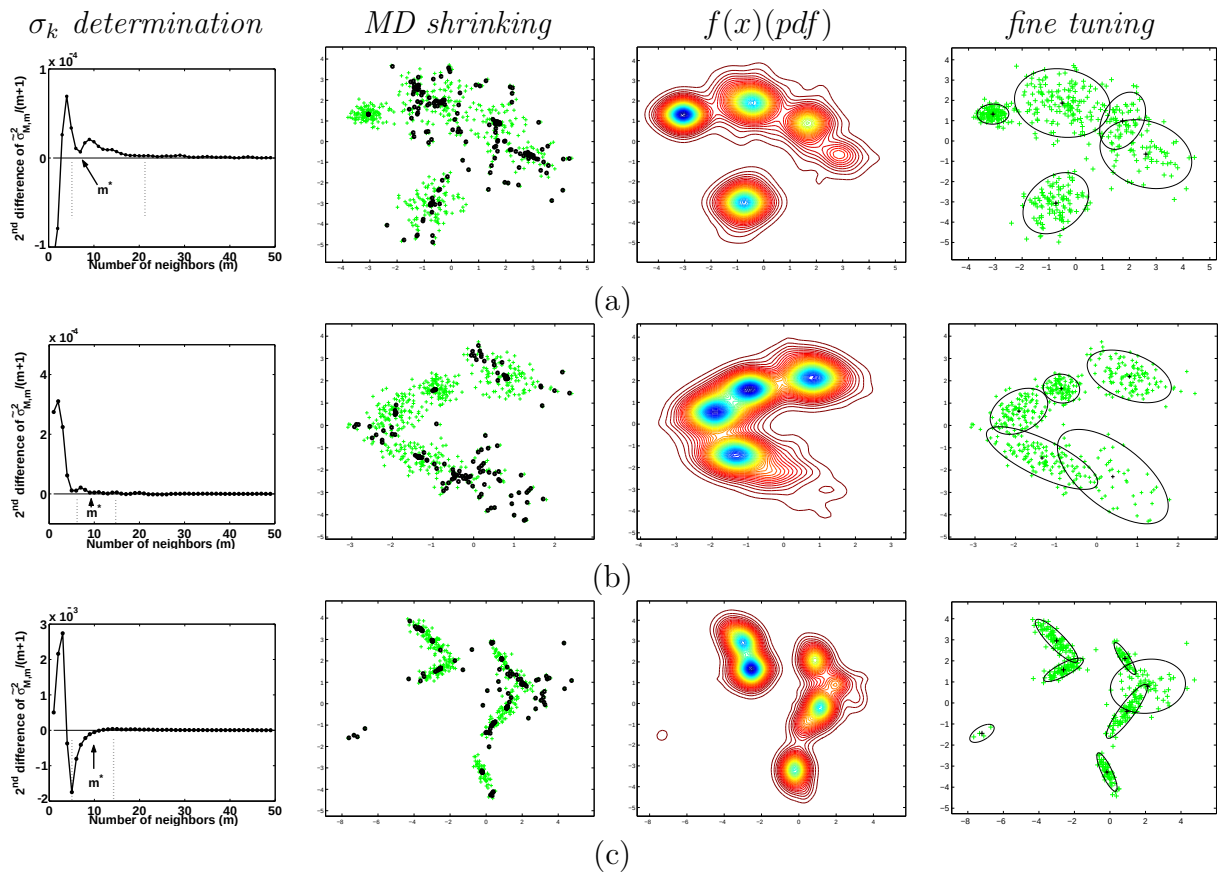
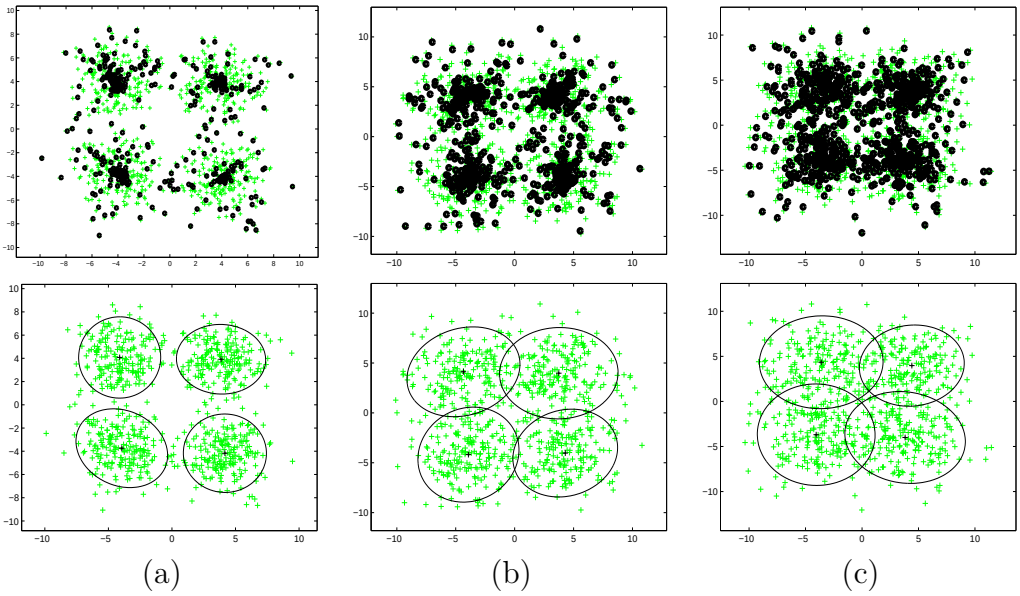


Figure 2: A step-by-step presentation of the proposed Newtonian clustering approach, using three simulated data sets in two dimensions.

In the first column of Fig. 2, we plot the second difference of $\frac{\tilde{\sigma}^2_{M,m}}{m+1}$ versus m. The dotted vertical lines delimit σ's interval of stability, in the sense that any value for σ within this range leads to the same number of clusters. The point m^* selected by the criterion of Eq. (15) is depicted by an arrow. It can be clearly seen that m^* always lies inside the stability range. This is so not only for the presented cases, but for every single data set we dealt with. The second column presents the original data (thin points), together with the processed data (thick points) to illustrate the MD shrinking effect. In the third column, contour plots of the pdfs constructed according to Eq. (9), are displayed. In the last column we present the situation after the application of the EM algorithm, which is the final result.



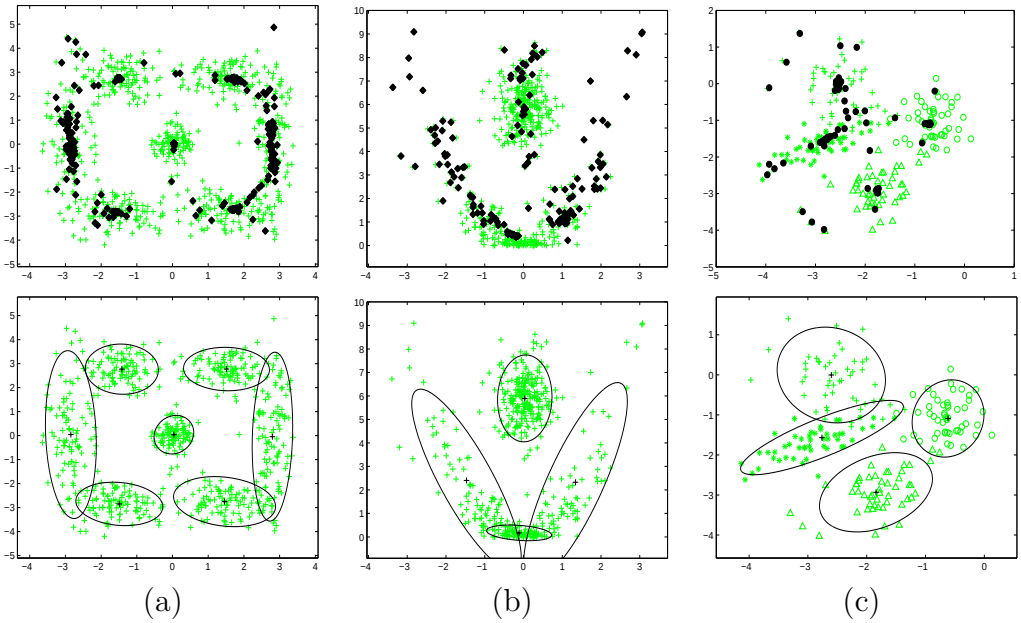
(a) (b) (c)

Figure 3: Three simulated data sets, each containing four clusters differing in the degree of overlap.

4.2 Evaluating the robustness of the method

In order to examine the reliability of our approach, we performed additional experiments. We created three data sets each containing $M = 800$ points, by sampling from a four-Gaussian mixture model, shown in Fig. 3. Starting from a well separated data set (a), we sequentially increased the level of overlap among the clusters producing so, two additional data sets (b) and (c). Note that data set (c), due to substantial overlap, resembles the structure of a single cluster. The MD preprocessing relocates the points towards their associated cluster centers, however in the case of data set (c) this effect is diminished. In spite of this, the information gathered during that phase, i.e. the Σ parameters, is sufficient for the method to distinguish the four clusters. This can be seen in Figure 3, where both the preprocessing and the final phase situations are shown with the original data in the background.

Three more experiments have been performed in two dimensions (see Fig. 4 (a),(b),(c)). The first two (with seven and four clusters, correspondingly) use simulated data sets while the third employs the renowned CRAB data set of Ripley [34], that contains data belonging to four clusters. Original CRAB data are in five dimensions, however here we have selected their projections on the plane defined by the second and third principal components, in order to make a comparison with ref. [28] possible. Figure 4, displays both the MD and



(a) (b) (c)

Figure 4: Experiments with two simulated data sets (a) and (b), and the CRAB data set of Ripley (c). The initial (thin) and the final (thick) points are shown along with the refined cluster shapes.

the fine tuning with the EM results, again with the original data in the background.

Additional experiments have been conducted in higher dimensions. We considered (Fig. 5(a)) the well known Fisher-IRIS [35] with $M = 150$ points belonging to three clusters in $N = 4$ dimensions. In addition, four more sets were constructed by simulation using Gaussian mixtures in $N = 5$ dimensions (Fig. 5(b),(c)) with 5 and 4 clusters, and in $N = 10$ dimensions (Fig. 5(d),(e)) with 5 clusters each. Projections on the plane formed by the first two principal components, are shown in Fig. 5, along with the preprocessed via MD points and the final cluster centers as these were recovered by the EM algorithm.

4.3 Competitive comparison

The evaluation of a new method is traditionally obtained through performance comparisons with established methods. In this direction we have conducted tests that allow for a fair comparison. Two kinds of methods have been considered.

1. Methods that require the model order K, as input.

2. Methods that determine the model order K, by themselves.

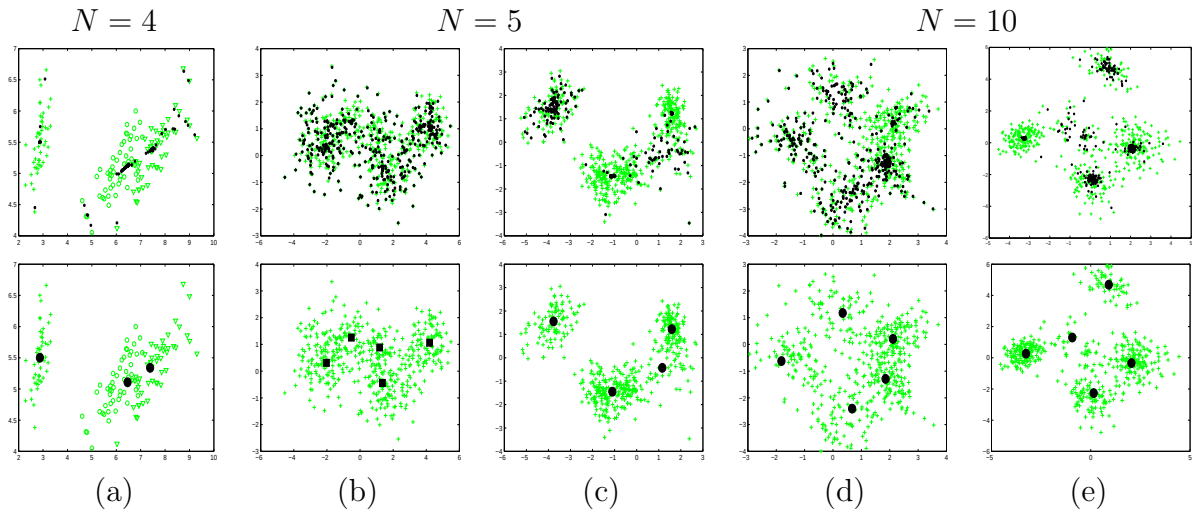


Figure 5: Experiments with high-dimensional data sets. Plotted is a projection of the point positions and the recovered cluster centers on the plane of the first two principal components.

In the first category we have employed two techniques, namely the *K-means-initialized EM* and the *Greedy EM* [29, 30]. In the second category we have again selected two techniques, namely the *Quantum Clustering* [27, 28], and the *MML-EM* described in [18].

4.3.1 Comparison with methods requiring K

For the *K-means-initialized EM*, K-means is first used to determine the centers of K clusters. Subsequently, this information is used to initialize the EM algorithm to treat the data set with a K-order mixture model. Since *K-means* depends on the initial cluster centers that are uniformly selected from the data, 100 runs of this *"K-means-initialized EM"* procedure were performed for each data set. We kept records of the mean value of the log-likelihood function, and the number of steps required for EM to converge. The second technique, i.e. the *Greedy EM*, starts with a single component and adds components sequentially one at a time up to K. It employs an efficient combination of local and global search every time a component is added. *Greedy EM* [2] is deterministic and hence it is run only once for each dataset.

In Tables 1 and 2, we summarize the results from experiments with several data sets that have been presented in previous sections. In the case of the *K-means-initialized EM* we also report the number of cases it managed to recover our result, i.e. the same maximum

[2]The *Greedy EM* software was downloaded from http://staff.science.uva.nl/∼vlassis/software/

Data set of Fig. 4	K	*Newtonian clustering EM*		*K-means-initialized EM*			*Greedy EM*
		log-likelihood	*EM-steps*	*log-likelihood*	*EM-steps*	*success (%)*	*log-likelihood*
(a)	7	−2452.33	40	−2473.05	80	80	-2452.33
(b)	4	−1409.91	56	−1445.85	167	4	-1441.26
(c)	4	−498.91	59	−500.57	67	95	-501.09

Table 1: Comparative results obtained by *"Newtonian Clustering"*, *"K–means-initialized EM"*, and the *"Greedy EM"* techniques, applied on 2-dimensional data sets.

Data set of Fig. 5	K	*Newtonian clustering EM*		*K-means-initialized EM*			*Greedy EM*
		log-likelihood	*EM-steps*	*log-likelihood*	*EM-steps*	*success (%)*	*log-likelihood*
(a)	3	−182.11	37	−187.22	51	79	-183.39
(b)	5	−2955.50	69	−3007.67	127	66	-2955.50
(c)	4	−2286.23	9	−2347.34	27	35	-2286.23
(d)	5	−3908.01	18	−3960.10	39	77	-3908.01
(e)	5	−4217.50	2	−4297.50	19	78	-4217.50

Table 2: Another series of comparative results using higher dimensional data sets.

of the log-likelihood function (denoted as "success" in the 7th column of Tables 1 and 2). It is readily deduced that *"Newtonian Clustering"* performs clearly better than the *K-means-initialized EM*. *Greedy EM* obtains results comparable to ours. However in all cases where the two methods yielded different results, *"Newtonian Clustering"* was the one with the higher log-likelihood value. Note that in the case of the Fig. 4(b) data set, the *K-means-initialized EM* technique has reached the optimum value (found by our Newtonian clustering approach), only in 4 out of 100 runs, while the *Greedy EM* converged to a lower value. Observing the number of the EM steps, we can conclude that the preprocessing not only determines the number of clusters properly, but in addition offers nearly optimal cluster solutions. For example, in the case of data set (e) of Table 2, EM took only 2 steps to converge.

4.3.2 Comparison with methods not requiring K

In this section we compare to two methods that they both determine K, one being deterministic and one stochastic, namely the *"Quantum Clustering"* (QC) [27, 28], and the *"MML-EM"* described in [18].

QC is a method based on the properties of the Schrödinger equation, and determines a potential $V(x)$ associated with the pdf that produced the data. The minima of this

potential correspond to the cluster centers. Schrödinger's equation:

$$-\frac{\sigma^2}{2}\nabla^2\Psi(x) + V(x)\Psi(x) = E\Psi(x) \ .$$

contains a free parameter (σ). As discussed in [27], the value of σ is significant for the performance of the method. The proposed way to determine σ is to repeatedly apply QC for various σ values and search for a stability range as far as the number of clusters is concerned. We have applied QC to the same data sets used to evaluate our method[3]. In particular, for each data set we have applied the suggested procedure, using a range of values for the parameter σ with a step of 0.02, and measured each time the number of resulting clusters. Figure 6 illustrates the results obtained for eight data sets, by plotting the number of retrieved clusters as a function of the σ parameter. This analysis reveals the dependence of QC on the value of σ. There are cases, such as the data sets of Fig. 2(a), Fig. 4(a) and Fig. 4(c), with a large interval of stability that renders the decision for the number of clusters easy. Also, there are some other cases of data sets, such as those in Fig. 2(b) and in Fig. 5(b), where the lack of a distinguishably large stability interval, makes the decision dubious. Note that for the data sets of Fig. 2(c), Fig. 4(b) and of Fig.5(a) the proper range of σ values is a narrow interval and hence QC fails to determine the right number of clusters. Our *"Newtonian Clustering"* approach, being parameter–free, has the advantage of allowing for a unique, deterministic estimation of the cluster number.

The technique described in [18], is a stochastic type of method. It uses a modified EM algorithm that incorporates the minimum message length (MML) criterion for model selection. We have repeatedly applied this method[4] 100 times to each data set and measured the number of clusters found. Full-covariance Gaussian densities have been used and K was allowed to vary in the interval $[1, 30]$. The model order with the highest frequency, is considered to be the preferred value for K. Figure 7 shows the frequency of the number of clusters found in eight data sets. In most cases the number of clusters can be clearly deduced. However, there are cases such as the data sets of Fig. 2(c) and Fig. 4(b), where the described procedure for the selection of the optimal model order fails to yield the correct K. There are some failure conditions quoted in [18], and may well be the case that the above data sets have happened to fulfill them.

[3]The QC software was downloaded from the web site http://neuron.tau.ac.il/∼horn/QC.htm
[4]The software was downloaded from the web site *http://www.lx.it.pt/∼mtf/*

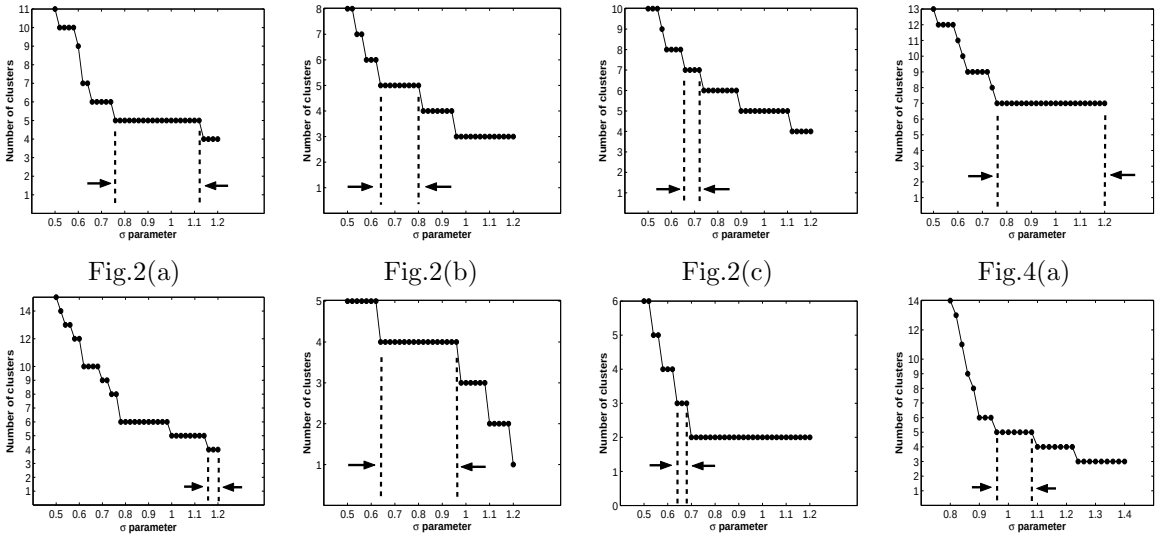


Fig.2(a) Fig.2(b) Fig.2(c) Fig.4(a)

Fig.4(b) Fig.4(c) Fig.5(a) Fig.5(b)

Figure 6: Plots of the number of clusters found using QC, versus σ, for eight experimental data sets. The dotted vertical lines indicate the proper range of σ.

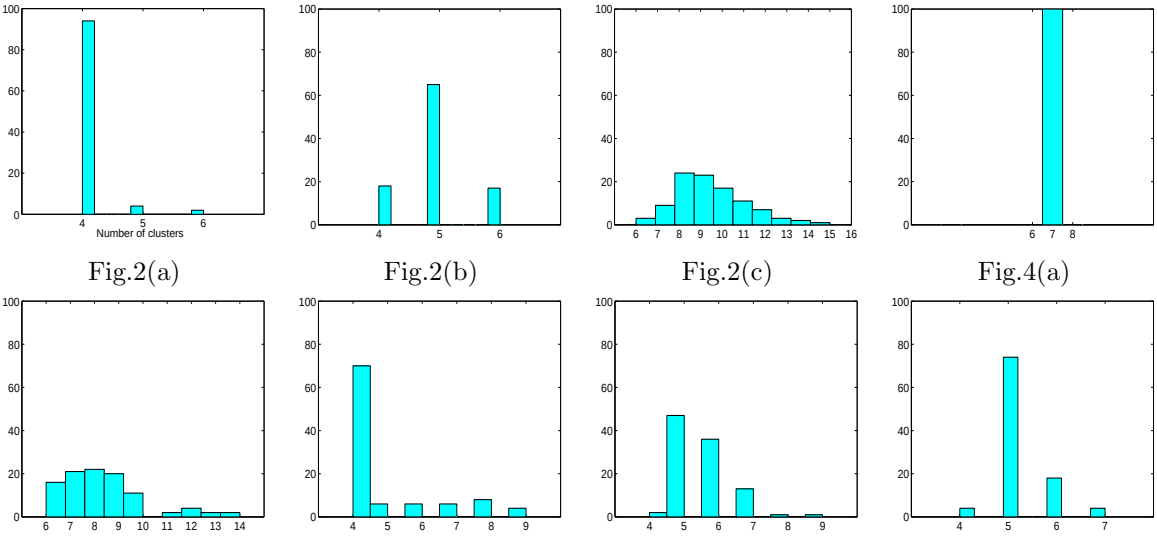


Fig.2(a) Fig.2(b) Fig.2(c) Fig.4(a)

Fig.4(b) Fig.4(c) Fig.5(b) Fig.5(e)

Figure 7: Frequencies of the number of clusters found by applying the MML-EM method on the indicated data sets.

5 Conclusions

In this article we have presented the *"Newtonian Clustering"* approach that enumerates and locates the clusters contained in a data set. The method consists of two phases. In the preprocessing phase a dynamical transformation is performed that forces each point to move towards the center of its host cluster. The distance traveled by each point is used to adjust the width of an associated pdf, in a Parzen window approach. The number of peaks of the superposition of these pdfs is shown to correspond to the number of clusters contained in the data set, while their positions offer an approximation for the locations of the cluster centers. In the second phase a local-based refinement is performed using the EM algorithm to a Gaussian mixture model. In the conducted experiments on a suite of benchmark data sets, the performance of *"Newtonian Clustering"* was found to be superior among four tested established clustering methods. Therefore we recommend the use of *"Newtonian Clustering"* on difficult clustering problems that recur in a plethora of real world applications.

Acknowledgments

We wish to thank the anonymous referee for the valuable comments that significantly improved the clarity of the article, and our colleagues, Professors Galatsanos and Likas for illuminating discussions and constructive criticism.

References

[1] J. A. Hardigan, *Clustering Algorithms*. Wiley, 1975.

[2] K. Fukunaga, *Introduction to Statistical Pattern Recognition*. Academic Press, San Diego, 1990.

[3] C. M. Bishop, *Neural Networks for Pattern Recognition*. Oxford Univ. Press Inc., New York, 1995.

[4] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. Wiley-Interscience, New York, 2001.

[5] G. M. McLachlan and D. Peel, *Finite Mixture Models*. New York: John Wiley & Sons, Inc., 2001.

[6] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum Likelihood from incomplete data via the EM algorithm," *J. Roy. Statist. Soc. B*, vol. 39, pp. 1–38, 1977.

[7] G. W. Milligan and M. C. Cooper, "An examination of procedures for determining the number of clusters in a data set," *Psychometrika*, vol. 50, pp. 159 –179, 1985.

[8] R. E. Kass and A. E. Raftery, "Bayes factors," *Journal of American Statistical Association*, vol. 90, pp. 773–795, 1995.

[9] A. Hardy, "On the number of clusters," *Computational Statistics and Data Analysis*, vol. 23, pp. 83–96, 1996.

[10] C. Fraley and A. E. Raftery, "How many clusters? Which clustering method? Answers via model-based cluster analysis," *Computer Journal*, vol. 4, pp. 578–588, 1998.

[11] H. Akaike, "Information theory and an extension of the maximum likelihood principle," in *2nd Intern. Symposium on Information Theory*, (Budapest), pp. 267–281, 1973.

[12] G. Schwarz, "Estimating the dimension of a model," *The Annals of Statistics*, vol. 6, pp. 461–464, 1978.

[13] P. Smyth, "Model selection for probabilistic clustering using cross-validated Likelihood," *Statistics and Computing*, vol. 10, pp. 63–72, 2000.

[14] D. Chickering and D. Heckerman, "Efficient approximations for the marginal Likelihood of Bayesian Networks with hidden variables," *Machine Learning*, vol. 29, no. 2-3, pp. 181–212, 1997.

[15] P. Green, "Reversible Jump Markov Chain Monte Carlo computation and Bayesian model determination," *Biometrika*, vol. 82, pp. 711–732, 1995.

[16] H. Chipman, E. I. George, and R. E. McCulloch, "The practical implementation of Bayesian model selection (with discussion)," *IMS Lecture Notes - Monograph Series*, vol. 38, pp. 67–134, 2001.

[17] A. Corduneanu and C. M. Bishop, "Variational Bayesian model selection for mixture distributions," in *AI and Statistics*, pp. 27–34, 2001.

[18] M. A. Figueiredo and A. K. Jain, "Unsupervised learning of finite mixture models," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 3, pp. 381–396, 2002.

[19] K. Blekas, A. Likas, N. P. Galatsanos, and I. E. Lagaris, "A Spatially-Constrained Mixture Model for Image Segmentation," *IEEE Trans. on Neural Networks*, vol. 16, no. 2, pp. 494–498, 2005.

[20] P. Baldi and S. Brunak, *Bioinformatics: The Machine Learning Approach*. MIT Press, Cambridge, 1998.

[21] C. G. E. Boender, A. H. G. R. Kan, G. T. Timmer, and L. Stougie, "A stochastic method for global optimization," *Mathematical Programming*, vol. 22, pp. 125–140, 1982.

[22] A. H. G. R. Kan and G. T. Timmer, "Stochastic global optimization methods. Part I: Clustering methods," *Mathematical Programming*, vol. 39, pp. 27–56, 1987.

[23] A. H. G. R. Kan and G. T. Timmer, "Stochastic global optimization methods. Part II: Multi level methods," *Mathematical Programming*, vol. 39, pp. 57–78, 1987.

[24] A. Torn and S. Viitanen, "Topographical global optimization using pre–sampled points," *Journal of Global Optimization*, vol. 5, pp. 267–276, 1994.

[25] M. M. Ali and C. Storey, "Topographical Multilevel Single Linkage," *Journal of Global Optimization*, vol. 5, pp. 349–358, 1994.

[26] F. V. Theos, I. E. Lagaris, and D. G. Papageorgiou, "PANMIN: Sequential and parallel global optimization procedures with a variety of options for the local search strategy," *Computer Physics Communications*, vol. 159, pp. 63–69, 2004.

[27] D. Horn and A. Gottlieb, "The method of quantum clustering," in *Advances in Neural Information Processing Systems 14* (T. G. Dietterich, S. Becker, and Z. Ghahramani, eds.), (Cambridge, MA), pp. 769–776, MIT Press, 2001.

[28] D. Horn and A. Gottlieb, "Algorithm for data clustering in Pattern Recognition problems based on Quantum Mechanics," *Physical Review Letters*, vol. 88, no. 1, pp. 018702–4, 2002.

[29] N. Vlassis and A. Likas, "A greedy EM algorithm for Gaussian mixture learning," *Neural Processing Letters*, vol. 15, pp. 77–87, 2001.

[30] J. J. Verbeek, N. Vlassis, and B. Krose, "Efficient greedy learning of Gaussian mixture models," *Neural Computation*, vol. 15, no. 2, pp. 469–485, 2003.

[31] E. Parzen, "On estimation of a probability density function and mode," *Ann. Math. Stat.*, vol. 33, pp. 1065–1076, 1962.

[32] D. W. Heermann, *Computer Simulation Methods in Theoretical Physics*. Springer-Verlag, Berlin, Heidelberg, New York, 1986.

[33] D. G. Papageorgiou, I. N. Demetropoulos, and I. E. Lagaris, "Merlin-3.0: A multidimensional optimization environment," *Comput. Phys. Commun.*, vol. 109, pp. 227–249, 1998.

[34] B. D. Ripley, *Pattern Recognition and Neural Networks*. Cambridge Univ. Press Inc., Cambridge, UK, 1996.

[35] C. J. Merz and P. M. Murphy, "UCI repository of machine learning databases. [http://www.ics.uci.edu/∼mlearn/MLRepository.html]. Irvine, CA.," 1998.