Communicated by John Moody and Steve Zucker

Clustering Data by Melting

Yiu-fai Wong
Department of Electrical Engineering, 116-81,
California Institute of Technology, Pasadena, CA 91125 USA

We derive a new clustering algorithm based on information theory and statistical mechanics, which is the only algorithm that incorporates scale. It also introduces a new concept into clustering: cluster independence. The cluster centers correspond to the local minima of a thermodynamic free energy, which are identified as the fixed points of a one-parameter nonlinear map. The algorithm works by melting the system to produce a tree of clusters in the scale space. Melting is also insensitive to variability in cluster densities, cluster sizes, and ellipsoidal shapes and orientations. We tested the algorithm successfully on both simulated data and a Synthetic Aperture Radar image of an agricultural site with 12 attributes for crop identification.

1 Introduction

Clustering is an important problem that can be found in many applications where a priori knowledge about the distribution of the observed data is not available (Duda and Hart 1973; Jain and Dubes 1988). Simply stated, the goal is to partition a given data set into several compact groups. Each group indicates the presence of a distinct category in the measurements. It is widely used for exploratory data analysis in diverse disciplines. The literature is therefore spread among many different fields over many years. It is almost impossible to cite each contribution individually.

One of the early algorithms was invented by Lloyd (1982), which was later extended by Linde *et al.* (1980) for vector quantization. In pattern recognition, the ISODATA algorithm (Ball and Hall 1967) and its sequential version, the *k*-means clustering algorithm, have been extensively used. Other algorithms include the fuzzy techniques (Ruspini 1969; Bezdek 1981; Gath and Geva 1989; Rose *et al.* 1990) and the hierarchical techniques such as the agglomerative and divisive methods (see Wishart 1969).

These algorithms, however, suffer from several difficulties: (a) they are highly sensitive to the initialization; (b) they perform poorly if the data contain overlapping clusters; and (c) they also suffer from the inability to handle variabilities in cluster shapes, cluster densities, and cluster sizes. The most urgent problem is the lack of cluster validity criteria (Bezdek 1981). All the algorithms tend to create clusters even when no natural clusters exist in the data.

In this paper, we examine a fundamental way of looking at the problem of clustering, and derive a new algorithm based on information theory and statistical mechanics. We identify clustering with heating up a thermodynamic system, giving rise to hierarchical clustering in the scale space. Melting can also account for variability in cluster densities, cluster sizes, and cluster shapes (ellipsoids). The algorithm was tested successfully on both simulated data and a Synthetic Aperture Radar (SAR) image of an agricultural land with 12 attributes for crop identification (Wong *et al.* 1992; Wong and Posner 1992).

A main contribution of this paper is that this interdisciplinary approach from information theory, thermodynamics, and nonlinear dynamics can provide a proper formulation for effective clustering and related optimization problems.

2 Scale and Cluster Independence

Intuition tells us that the number of clusters depends on the scale we look at the data. At a very coarse scale, the whole data set is a cluster, whereas at a very fine scale, every datum is itself a cluster. Scale has not been exploited by the other clustering techniques, though the idea of scale space has been around for a long time (Gabor 1946; Koenderink 1984).

Wong (1992) introduced a concept called "cluster independence." To explain it, consider the situation where several people are given the same data and the same rule about clusters. Each is told to stop once a cluster is found. If they do not communicate, it is clear that the assignments of the clusters are independent.

If clusters indeed exist, the information should be present in the data itself. The notion of scale implies that the data points near the cluster centers should give more information while the data points far away should give less. This can be implemented by assigning a cost of having a data point reveal the cluster locations. To make a cluster robust, the information should be spread among the data. If we treat the contributions to the determination of a cluster from all the data points as a probability distribution, this means that this probability distribution should be chosen such that its entropy is maximized subject to a linear cost constraint (Jaynes 1957).

Cluster independence allows us to consider one cluster at a time. Suppose the cost function is $e(x) = (x - y)^2$ where x is a datum and y is a cluster center. This means that we use the squared distance as a measure of the compactness of a cluster. Let $P(x)$ denote the contribution of datum x to y. Maximizing the entropy

$$-\sum_x P(x)\log P(x) \tag{2.1}$$

subject to the constraint

$$\sum_x P(x)e(x) = C \tag{2.2}$$

one obtains

$$P(x) = e^{-\beta(x-y)^2}/Z \tag{2.3}$$

where $Z = \sum_x e^{-\beta(x-y)^2}$. To make the connection with thermodynamics, we define the "free energy"

$$F = -\frac{1}{\beta}\log Z \tag{2.4}$$

At equilibrium, it is known that a thermodynamic system settles into configurations that minimize its free energy. That is, we want $\partial F/\partial y = 0$, or equivalently,

$$y = \sum_x \frac{xe^{-\beta(x-y)^2}}{\sum_x e^{-\beta(x-y)^2}} \tag{2.5}$$

the weighted mean of the data. We point out that equation 2.5 is very different from that obtained by the maximum-likehood estimate of a Gaussian mixture (Wolfe 1970; Cheeseman *et al.* 1988). Unlike these Bayesian approaches, our method does not assume any particular data distribution.

Without loss of generality, we restrict the notation and the exposition to the case of one-dimensional data. The case of higher dimensional data was treated in Wong (1992). The good news is that the dynamics are essentially the same.

Definition 1. *A* nominal cluster *is* centered at *y if and only if y is a local minimum of the free energy of the thermodynamic system described above.*

Equation 2.5 is only a necessary condition for y to be a cluster center. The sufficient conditions will become clearer as the "melting" process is explained. The details can be found in Wong (1992). Because of that, we will use "cluster" instead of "nominal cluster." Without worrying whether nominal clusters are real clusters, one can ask the following questions:

1. Do clusters exist? This depends on whether the equation has any solutions.

2. How many clusters are there? This depends on the number of solutions the equation has.

3. How do the clusters evolve? The answer is given by the trajectories of the solutions of equation 2.5 as β varies.

The above list could have been longer but it suffices to illustrate the importance of equation 2.5. Since we are concerned only with local minima, this is a great advantage over other applications of physical optimization where global optima are sought.

3 Melting and Its Dynamics

Solutions of equation 2.5 cannot be computed analytically. However, they are identical to the fixed points of the following one-parameter map[1]:

$$y \xmapsto{f} y + \sum_x \frac{(x-y)e^{-\beta(x-y)^2}}{\sum_x e^{-\beta(x-y)^2}} \tag{3.1}$$

This connects our problem with nonlinear dynamics. Figure 1 is a plot of the map for 20 data points along a unit interval with a large β. It is clear that the free energy F acts as the Lyapunov function (Wiggins 1990) for the mapping. The difference between successive ys is $-\frac{1}{2}\partial F/\partial y$. Thus, the ys march down the surface of the free energy and settle down in some local minimum. The mapping 3.1 exhibits no chaotic behavior. Hence, solutions can always be computed iteratively and the convergence is exponentially fast.

One can see that β truly captures the notion of scale. At a very large β, every datum is itself a cluster, while at a very small β, the whole data set is a cluster. The essence of the algorithm is thus as follows. Start with a huge β (fine scale); initialize every datum as a cluster. As β is gradually decreased, the number of clusters decreases due to the merging of the clusters. When two clusters merge, the associated data points are merged as well. Eventually the whole data set is a cluster. Specifically, the *Melting Procedure* is as follows:

1. Choose β_{max}; β_{max} is a number related to the dynamic range and an assumed noise in the observations;

2. set $i = 1$, $\beta_1 = \beta_{\text{max}}$;

3. let every data point be a cluster;

4. iterate according to the mapping 3.1 N times or until the clusters converge. In our simulations, $N = 200$;

5. record the new cluster centers;

[1] We could have instead established a similar equivalence with the differential equation $dy/dt = \sum_x (x-y)e^{-\beta(x-y)^2}$, but the analysis is very similar and the results are the same.

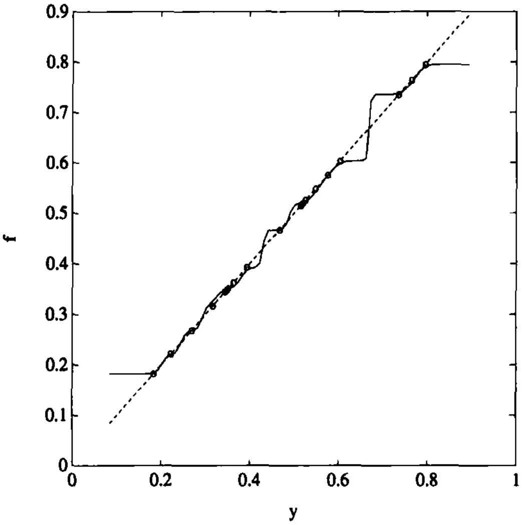

Figure 1: The map for 20 data points along the unit interval.

6. if more than two clusters that previously are distinct share the same center, the set of data associated with the new cluster is the union of those with the original clusters;

7. $i = i + 1$, $\beta_i = \beta_{i-1}/1.05$;

8. if there is more than one cluster, go to 4; else Melting is complete.

It is clear that the Melting Procedure generates a strict tree structure in the scale space, analogous to a *dendrogram*. Figure 2 shows an example of the Melting Procedure for a set of one-dimensional data that has two clusters. The graphs are obtained by computing the fixed points of equation 3.1 as scale increases. The horizontal axis indexes i in the Melting Procedure. We merely identify scale with i, which is plotted logarithmically because of the exponential terms in equation 2.5. The original data are plotted as $\star$s at $i = 0$.

The dynamics involved in the merging process can be studied using local bifurcation theory (Wiggins 1990). The necessary condition for bifurcation to occur is $\partial f/\partial y = 1$. That is,

$$2\beta \frac{\sum_x (x-y)^2 e^{-\beta(x-y)^2}}{\sum_x e^{-\beta(x-y)^2}} = 1 \tag{3.2}$$

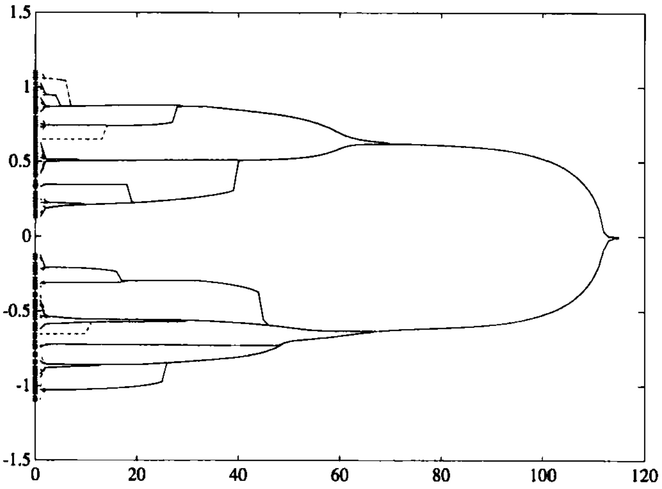

Figure 2: The fixed points versus scale. The leftmost points are the data.

In Wong (1992), two types of bifurcations were identified: pitchfork and saddle-node. Their bifurcation diagrams are shown in Figure 3, which show the trajectories of the fixed points as the parameter β is varied around its critical value. In a pitchfork bifurcation, two clusters continuously merge into a cluster while in a saddle-node bifurcation, a cluster becomes unstable and is siphoned into another cluster. Such bifurcations can be seen in Figure 2.

The interpretation of these two bifurcations for cluster analysis is as follows (Wong 1992):

A pitchfork bifurcation indicates (1) uniformly spaced or nonclustered data, or (2) clustered data but with a high degree of symmetry at certain scales. A saddle-node bifurcation indicates the inhomogeneous spatial distribution present in the data. As one expects, in clustering data, saddle-node bifurcations will be most frequently observed.

It is now clear why we choose to "melt" the system starting from a low temperature, as contrasted with annealing (Kirkpatrick *et al.* 1983). Annealing would fail since saddle-node bifurcation implies that we do not know how much hill-climbing is needed to reach the other local minimum.

We can also find an information-theoretic basis for condition 3.2. The rate distortion function deals with the question of the minimum number of bits needed to encode a source symbol subject to an expected distortion

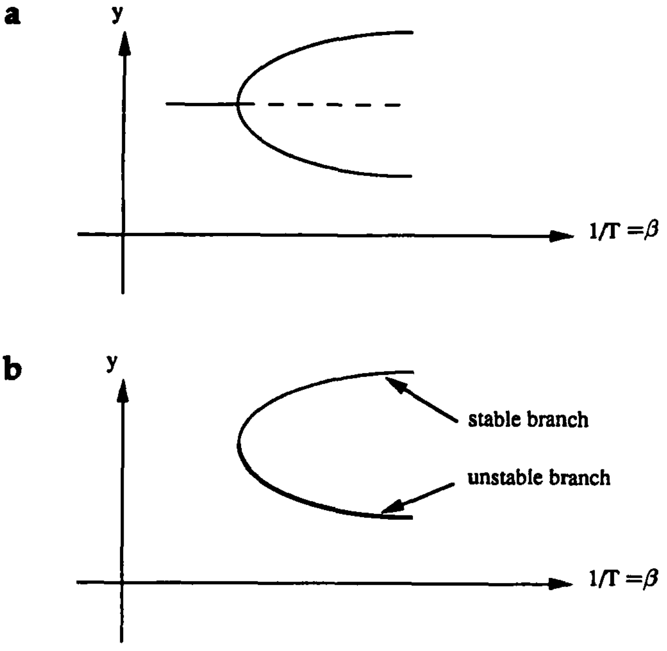

Figure 3: (a) Pitchfork bifurcation in our clustering scheme. (b) Saddle-node bifurcation in our clustering scheme.

constraint (Pierce and Posner 1980). For a Gaussian source with variance σ^2 and the average distortion $\leq \delta$,

$$R(\delta) = \begin{cases} \frac{1}{2}\log(\sigma^2/\delta) & \text{if } \delta \leq \sigma^2 \\ 0 & \text{if } \delta \geq \sigma^2 \end{cases} \tag{3.3}$$

Thus, when equation 3.2 becomes an equality, $R(\delta) = 0$ signifies that there is no need to waste bits to encode the source. The cluster should either disappear or be merged.

4 What Is a Good Cluster and How Many Are There?

One needs a criterion to decide the good clusters among all the clusters in the scale space. We will briefly outline the ideas in Wong (1992).

Recall that $p(x)$ is the contribution of a data point to a cluster. Thus the quantity *fractional free energy* (FFE) of a nominal cluster

$$M_Q(\beta) = \sum_{x \in Q} P(x) = \frac{\sum_{x\in Q} e^{-\beta(x-y)^2}}{Z} \tag{4.1}$$

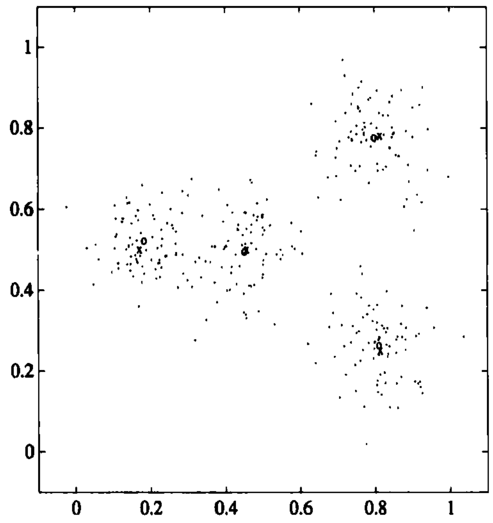

Figure 4: (a) Data and the computed clusters illustrating ability to handle many clusters.

is a measure how good a cluster Q is. A large FFE indicates that most of the contributions come from the data belonging to the cluster itself and vice versa. What is large or small is set by a threshold M_T, which expresses a degree of confidence. Hence, by keeping track of the fixed points and their FFE values, a criterion for deciding "good clusters" was defined in Wong (1992).

We need to select the real clusters among the good clusters. It is very difficult to define a universally accepted criterion because clusters really need to be interpreted in the context of the specific applications. Nonetheless, an attempt to define a scale-based criterion was carried out in Wong (1992), which has found to be applicable in the radar application (Wong and Posner 1992).

If distinct good clusters exist in the data, their FFEs should remain good over a large range of logarithmic scale in β even though the fixed points may vary their positions slightly. In addition, the FFEs of these clusters should start out with very high value, only to drop quickly when they are about to bifurcate, hence, for a good cluster, the longer its FFE remains high, the more robust it is.

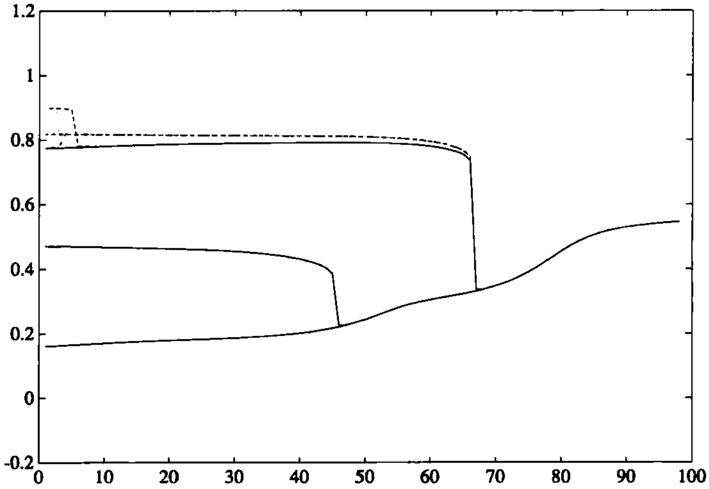

Figure 4: (b) x-components of the trajectories of the cluster centers versus scale.

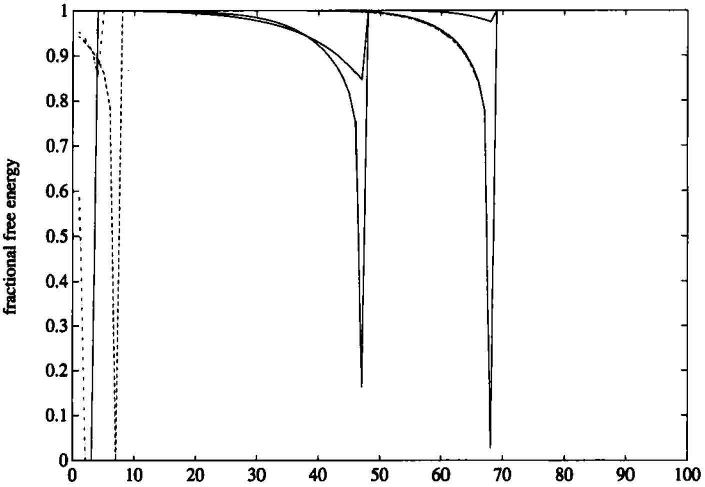

Figure 4: (c) Plots of fractional free energies for the data points in a.

Figure 4b shows the x-component of the trajectories of the clusters for the data shown in Figure 4a.[2] Figure 4c is a plot of the FFEs of the clusters versus scale. One sees that there is a range of scale over which three clusters exist. But there is a longer range of scale over which four clusters exist, which is the correct answer. Here is how we formally define the robustness of a good cluster:

Definition 2. *The* robustness *of a good cluster is defined as the range of logarithmic scales over which its FFE remains above M_T.*

The rule to decide the number of clusters is to pick out the most robust ones until there are no more good clusters left. Here is the Melting Algorithm (Wong 1992):

1. Perform the Melting Procedure;
2. decide the good clusters among the nominal clusters; denote the set by $\mathcal{T} = \{T_1, T_2, \ldots, T_m\}$;
3. compute the robustness of the good clusters;
4. initialize $\mathcal{U}$ to an empty set;
5. while $\mathcal{T}$ is nonempty, do the following:
    a. pick the element T_k in $\mathcal{T}$ with the biggest robustness measure;
    b. put this element into $\mathcal{U}$;
    c. remove T_k and the elements in $\mathcal{T}$ that either are contained in or contain T_k;
6. collect the data points that do not belong to one of the clusters in $\mathcal{U}$ into a set $\mathcal{N}$, which we hope is empty.

Several remarks about the above algorithm: (1) The algorithm actually consists of two parts: melting and determination of the clusters. (2) The Melting Procedure is governed by the scale parameter β only. (3) $\beta_{\max}$ should be chosen such that the number of clusters at $i = 2$ is not significantly less than the number of data points to start with. Otherwise, the initial temperature is too high, which might cause premature partitioning of the data. β_0 can be obtained easily by simple preprocessing. (4) The determination of the clusters is carried out in steps 2–5. We also note that step 5 can be modified to further study the finer structure of the data, such as clusters within clusters.

[2] The data were generated from normal distributions. A "cross" denotes the center of the distribution as seen by the computer. A "circle" denotes the representative of a cluster that is just the arithmetic mean of the data in a given cluster. The horizontal axis indexes scales with fewer than 10 clusters in the Melting Procedure (to avoid too many curves). The same explanations apply to Figure 5.

5 Ability to Handle Clusters of Oriented Ellipsoidal Shapes

Figure 5a shows a data set consisting of four clusters with various orientations and ellipsoidal shapes. Figure 5b shows the trajectories of the clusters. Figure 5c is the plots of the FFEs; it clearly shows that there are four clusters. Figure 5d shows the partition obtained by the algorithm. Note the few data points marked by Os, which get grouped into clusters different from that generated by the computer. Since they are far away from the originating cluster, such grouping is acceptable.

Even without a norm which is biased in the different directions, there is a built-in dynamics in the formulation to handle oriented ellipsoidal shapes with a single β. This was also demonstrated in the radar application (Wong and Posner 1992). Here is a brief explanation.

Obviously, the dynamics of the mapping 3.1 is invariant to the rotation of the coordinate system. Suppose, as is reasonable, that each cluster consists of data coming from a source corrupted with unimodal noise. Due to insufficient sampling or pure random fluctuation, the local density is not monotonically decreasing. However, at a coarser scale, it will be monotone. This implies that sooner or later, a cluster will see that its center cannot be balanced due to the monotonicity. It will try to "swim" toward the gradient until balance is reached, which is possible only in the neighborhood around the true signal.

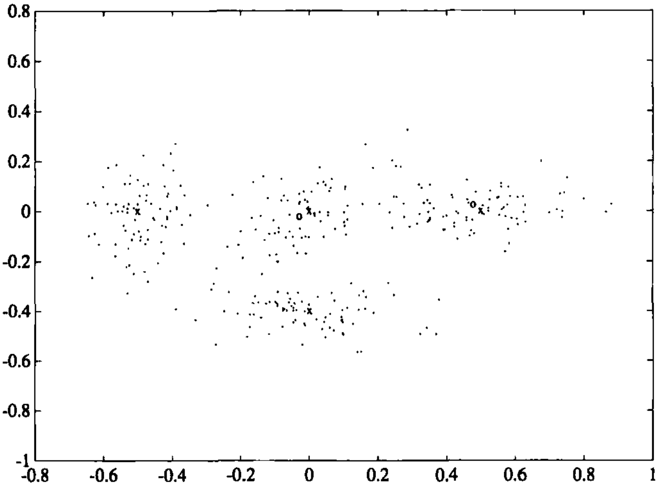

Figure 5: (a) Data illustrating ability to handle many clusters of different shapes and sizes.

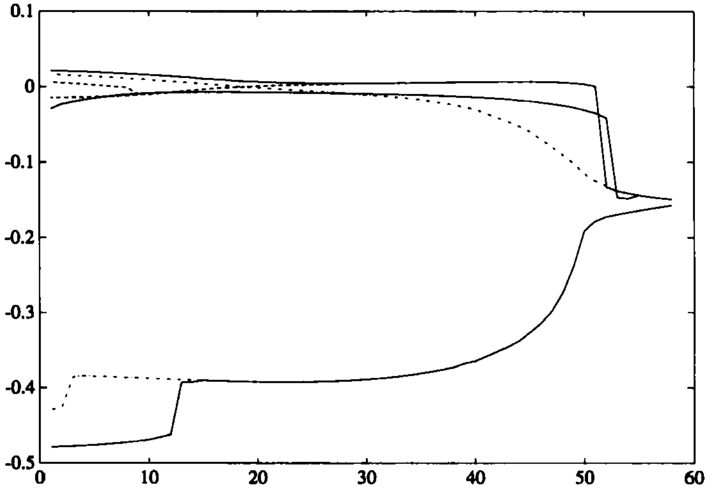

Figure 5: (b) *y*-components of the trajectories of the cluster centers versus scale.

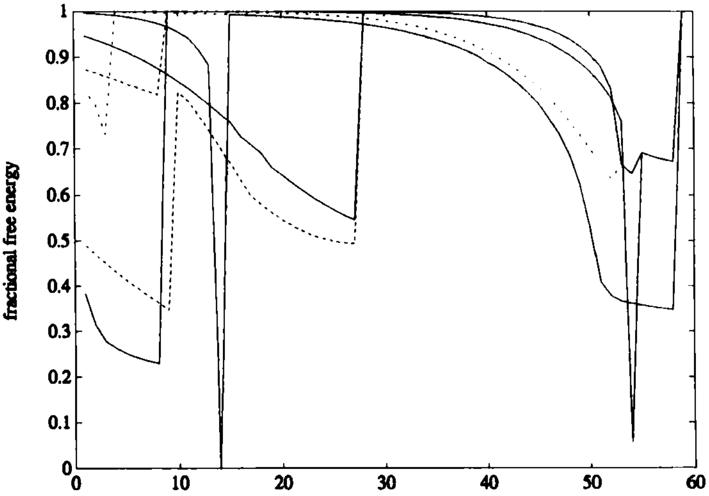

Figure 5: (c) Plots of fractional free energies for the data points in a.

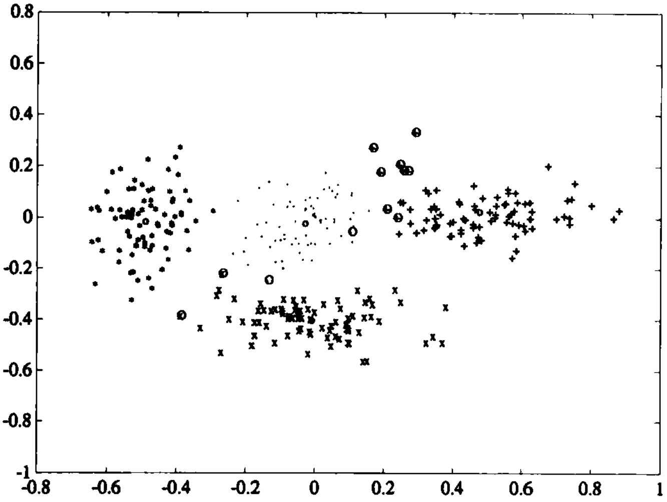

Figure 5: (d) Clustering of the data shown in a. O, misclassified data points.

6 Computational Complexity and Other Observations

Instead of artificial data, we will illustrate the timing on a real application to the clustering and classification of a 12-dimensional SAR image of an agricultural area (Wong and Posner 1992).

Due to the attracting dynamics, the convergence is exponentially fast. It was observed that convergence to a fixed point took an average of 15 iterations at each β. The exact rate of convergence, however, depends on the Jacobian of the map (3.1), which cannot be known a priori, making it impossible to give an upper bound. For the application, the computation took 26 minutes of SPARC-II cpu time to find the clusters. This is very intensive compared to ISODATA (Ball and Hall 1967), which takes 3 sec provided, however, it is given the right initialization.

To compare ISODATA with Melting Algorithm, we note the key point that if we initialize ISODATA wrongly, it will never find the correct clusters. Here, "wrongly" means putting more than one initial cluster in a real cluster. For the application, we have 1397 data points. There are 13 clusters, each with about 106 data points. Suppose that the initial cluster centers are assigned randomly. There are

$$\binom{1397}{13} \approx 1.17 \times 10^{31}$$

choices. Of these, only $106^{13} = 2.13 \times 10^{26}$ initializations give the correct partition; even this is an overestimate since some data are noisy. Hence, the probability of a correct initialization is at most 1.82×10^{-5}. Since ISODATA is 520 times faster than our algorithm, its probability of getting the correct answer is 0.0095 in 26 min, which would have been lower had it not been given the number of clusters. This simple calculation shows that in an obvious sense the Melting Algorithm is at least 105 times (1/0.0095) better than ISODATA. Furthermore, one has to weigh the quality and assurance of the solution obtained by our Melting Algorithm.

The current implementation of the Melting Algorithm does not include any heuristics to speed up its computation though it did use a lookup table of the exponential function. We note the following along these lines:

1. Initially, there are a huge number of clusters. Most clusters will merge quickly since they exist simply because it is too "cold." This effect can be seen in Figure 2. Some preprocessing such as simple grouping would reduce the complexity dramatically.

2. The purpose of the Melting Procedure is to track the trajectories of the clusters in the scale space. Instead of decreasing β by a constant factor, we can also utilize numerical techniques such as continuation (Doebel 1986) and adaptive step size selection to track the bifurcation points more accurately and faster.

3. The algorithm is ideal for parallel implementation because of local calculations and cluster independence.

In addition, it is possible that some partial a priori information will allow one to perform melting over a range of scales. Thus, the computational complexity of the algorithm can be improved significantly with the techniques outlined above and other heuristics that we have yet to investigate. Some preliminary work is reported in Tam (1992).

7 Summary

Clustering is a hard problem. The traditional clustering algorithms suffer from several difficulties. The willingness of existing algorithms to partition any set of data suggests that they may more suitably be named "partitioning" algorithms rather than "clustering" algorithms.

In this work, we have devised a new clustering algorithm that properly exploits the notion of scale. We also introduced the notion of cluster independency, which has not been formally recognized by prior researchers. It permits the natural application of the maximum entropy principle.[3] Cluster centers correspond to the local minima of a thermo-

[3] For related results on clustering using maximum entropy principle, see Rose *et al.* (1990) and the work by J. Buhmann and H. Kühnel in this issue.

dynamic free energy. The system is identical to a one-parameter nonlinear map, which can be rigorously analyzed using bifurcation techniques. Melting the system produces a tree of clusters in the scale space. Melting can also account for variabilities in cluster densities, sizes and shapes (ellipsoidal). We further tested this algorithm on the clustering and classification of a Synthetic Radar Aperture image of an agricultural site with 12 attributes.

Since clustering is a form of unsupervised learning, we expect this work should provide some new insights for neural network research and optimization theory, too, but we will not discuss that here.

Acknowledgments

I am deeply indebted to my thesis advisor Edward C. Posner for his patience, encouragement, and advice. I benefited much from him. I thank Professor Steve Wiggins at Caltech for clarifying some concepts on the local bifurcations. This work is supported by Pacific Bell through a grant to the California Institute of Technology and by NASA through the Caltech Jet Propulsion Laboratory, as well as a Charles Lee Powell Foundation Graduate Fellowship at Caltech.

References

Ball, G., and Hall, D. 1967. A Clustering Technique for Summarizing Multivariate Data. *Behav. Sci.* **12**, 153–155.

Bezdek, J. C. 1981. *Pattern Recognition with Fuzzy Objective Function Algorithms.* Plenum, New York.

Cheeseman, P. *et al.* 1988. AutoClass: A Bayesian classification system. *Proceedings of the 1988 Machine Learning Workshop.*

Doedel, E. 1986. *AUTO: Software for Continuation and Bifurcation Problems in Ordinary Differential Equations.* Tech. Rep., Applied Mathematics, Caltech.

Duda, R. O., and Hart, P. E. 1973. *Pattern Classification and Scene Analysis.* Wiley, New York.

Gabor, D. 1946. Theory of communication. *J. IEE* **93**, 429–457.

Gath, I., and Geva, A. B. 1989. Unsupervised optimal fuzzy clustering. *IEEE Trans. Pattern Anal. Machine Intell.* **PAMI-11**, 773–781.

Jain, A. K., and Dubes, R. C. 1988. *Algorithms for Clustering Data.* Prentice Hall, Englewood Cliffs, NJ.

Jaynes, E. T. 1957. Information theory and statistical mechanics I. *Phy. Rev.* **106**, 620–630.

Kirkpatrick, S., Gelatt, C. D., and Vecchi, M. P. 1983. Optimization by simulated annealing. *Science* **220**, 671–680.

Koenderink, J. J. 1984. The structure of images. *Biol. Cybern.* **50**, 363–370.

Linde, Y., Buzo, A., and Gray, R. M. 1980. An algorithm for vector quantization. *IEEE Trans. Commun.* **COM-28**, 84–95.

Lloyd, S. P. 1982. Least squares quantization in PCM. *IEEE Trans. Inform. Theory* **28**(129), 137 (reprint of the 1957 paper).

Pierce, J. R., and Posner, E. C. 1980. *Introduction to Communication Science and Systems.* Plenum Press, New York.

Rose, K., Gurewitz, E., and Fox, G. C. 1990. A deterministic annealing approach to clustering. *Pattern Recog. Lett.* **11**, 589–594.

Ruspini, E. 1969. A new approach to clustering. *Inform. Contr.* **15**, 22–32.

Tam, T. K. 1992. Fast and Parallel Implementation of Melting Algorithm for Clustering. Caltech Summer Undergraduate Research Fellowship Report.

Wiggins, S. 1990. *Introductions to Applied Nonlinear Dynamical Systems and Chaos.* Springer-Verlag, New York.

Wishart, D. 1969. Mode analysis: A generalization of nearest neighbor which reduces chaining effects. In *Numerical Taxonomy,* A. J. Cole, ed., pp. 282–308. Academic Press, London.

Wolfe, J. H. 1970. Pattern clustering by multivariate mixture analysis. *Multivariate Behavioral Research* **5**, 329–350.

Wong, Yiu-fai, Peters, K. J., and Posner, E. C. 1992. Unsupervised and hierarchical cluster analysis and classification of SAR images. *Proceedings International Space Year Conference on Earth and Space Science Information Systems,* Pasadena, February, to be published by AIP.

Wong, Yiu-fai, and Posner, E. C. 1992. A new clustering algorithm applicable to multispectral and polarimetric SAR images. *IEEE Trans. Geosci. Remote Sensing,* in press.

Wong, Y. F., and Posner, E. C. 1992. Scale-space clustering and classification of SAR images with numerous attributes and classes. To be presented at 1992 IEEE Workshop on Applications of Computer Vision, November, Palm Springs.

Wong, Yiu-fai 1992. *Towards a Simple and Fast Learning and Classification System.* Ph.D. Thesis, Caltech, Electrical Engineering.

Received 16 January 1992; accepted 12 May 1992.