ELSEVIER

Physica A 302 (2001) 70–79

PHYSICA A

www.elsevier.com/locate/physa

Clustering via Hilbert space

David Horn

School of Physics and Astronomy, Raymond and Beverly Sackler Faculty of Exact Sciences, Tel Aviv University, Tel Aviv 69978, Israel

Abstract

We discuss novel clustering methods that are based on mapping data points to a Hilbert space by means of a Gaussian kernel. The first method, support vector clustering (SVC), searches for the smallest sphere enclosing data images in Hilbert space. The second, quantum clustering (QC), searches for the minima of a potential function defined in such a Hilbert space.

In SVC, the minimal sphere, when mapped back to data space, separates into several components, each enclosing a separate cluster of points. A soft margin constant helps in coping with outliers and overlapping clusters. In QC, minima of the potential define cluster centers, and equipotential surfaces are used to construct the clusters. In both methods, the width of the Gaussian kernel controls the scale at which the data are probed for cluster formations. We demonstrate the performance of the algorithms on several data sets.

Keywords: Clustering; Support vector clustering; Hilbert space; Kernel methods; Scale-space clustering; Schrödinger equation

1. Introduction

Clustering algorithms are part and parcel of the general topic of pattern classification [1]. We will look into clustering algorithms that are unsupervised and driven solely by the information about the location of data in a Euclidean data space of dimension d. A well-known algorithm for such a problem is the k-means algorithm [2], in which one specifies the number, k, of clusters, and proceeds to find a set of centers with which points are associated according to their shortest distances. The algorithms that we are going to discuss are non-parametric, i.e. the number of clusters is not specified a priori. They do, however, possess a parameter q that specifies the distance $\sigma = 1/\sqrt{(2q)}$ that is being used to probe the system. In other words, the pattern of clusters is

E-mail address: horn@post.tau.ac.il (D. Horn).

PII: S0378-4371(01)00442-3

scale-dependent. As such they belong to a family of scale-space methods, such as the clustering algorithm of Roberts [3].

2. Support vector clustering

In the support vector clustering (SVC) [4] algorithm data points are mapped from data space to a high-dimensional Hilbert space using a Gaussian kernel. One looks then for the smallest sphere in the Hilbert space that encloses the image of the data. This sphere is mapped back to data space, where it forms a set of contours which enclose the data points. These contours are interpreted as cluster boundaries. As the width parameter of the Gaussian kernel is decreased, the number of disconnected contours in data space increases, leading to an increasing number of clusters.

2.1. Cluster boundaries

Let $\{\mathbf{x}_i\}$ be a data set of N points in $\mathbb{R}^d$, the data space. A nonlinear transformation Φ defines images of all data points in the Hilbert space. Limiting these images to lie within a sphere of radius R and center $\mathbf{a}$,

$$(\Phi(\mathbf{x}_j) - \mathbf{a})^2 \leqslant R^2 \quad \forall j$$

and searching for the smallest such sphere, can be achieved by a variational calculation applied to the Lagrangian

$$L = R^2 - \sum_{j=1}^{N} (R^2 - (\Phi(\mathbf{x}_j) - \mathbf{a})^2)\beta_j \,, \tag{1}$$

where $\beta_j \geqslant 0$ are Lagrange multipliers. Setting to zero the derivative of L with respect to R and $\mathbf{a}$, respectively, leads to

$$\sum_j \beta_j = 1 \,, \tag{2}$$

$$\mathbf{a} = \sum_j \beta_j \Phi(\mathbf{x}_j) \,. \tag{3}$$

Using these relations we may eliminate the variables R and $\mathbf{a}$, turning the Lagrangian into the Wolfe dual form that is a function of the variables β_j only:

$$W = \sum_j \Phi(\mathbf{x}_j)^2 \beta_j - \sum_{i,j} \beta_i \beta_j \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j) \,. \tag{4}$$

Following the approach of support vector machines (SVM) [5] one represents the dot products $\Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j)$ by an appropriate Mercer kernel $K(\mathbf{x}_i, \mathbf{x}_j)$, chosen here as the Gaussian

$$K(\mathbf{x}_i, \mathbf{x}_j) = \mathrm{e}^{-q(\mathbf{x}_i - \mathbf{x}_j)^2} \tag{5}$$

with width parameter q. This turns W into

$$W = \sum_j K(\mathbf{x}_j, \mathbf{x}_j)\beta_j - \sum_{i,j} \beta_i \beta_j K(\mathbf{x}_i, \mathbf{x}_j) = 1 - \sum_{i,j} \beta_i \beta_j K(\mathbf{x}_i, \mathbf{x}_j)\,. \tag{6}$$

This function is being maximized over the space of all $\beta \geqslant 0$ leading to two sets of points, those for which $\beta > 0$, called support vectors (SV) and those for which $\beta = 0$. The latter lie within the sphere in Hilbert space, while the SVs lie on the sphere. Hence the SVs will lie on cluster boundaries in data space while the other points will lie within such boundaries.

The same Eq. (6) can be employed to solve a problem with soft constraints that allow for outliers. In this case, the only difference is [4] that the Lagrange multipliers have an upper limit $\beta < C$. Asymptotically (for large N), the fraction of points that become outliers is $p = 1/CN$. In other words, as C is being reduced, more and more points are labelled as outliers. The latter are defined as those SV for which $\beta = C$, hence they are also called bounded support vectors (BSVs).

3. Examples of SVC

3.1. Example without BSVs

The first example [4] is a data set in which the separation into clusters can be achieved without invoking outliers, i.e., $C = 1$. Fig. 1 demonstrates that as the scale parameter of the Gaussian kernel, q, is increased, the shape of the boundary in data space varies: with increasing q the boundary fits more tightly the data, and at several q values the enclosing contour splits, forming an increasing number of components (clusters). Fig. 1a has the smoothest cluster boundary, defined by six SVs. With increasing q, the number of support vectors increases.

3.2. Example with BSVs

Quite often clusters are not as well separated as in Fig. 1. Thus, in order to observe splitting of contours, one must allow for BSVs. This is demonstrated [4] in Fig. 2a: without BSVs contour separation does not occur for the two outer rings for any value of q. When some BSVs are present, the clusters are separated easily (Fig. 2b).

In the spirit of the examples displayed in Figs. 1 and 2 SVC has to be used iteratively. Starting with a low value of q where there is a single cluster, and increasing it, one expects to observe the formation of an increasing number of clusters, as the Gaussian kernel describes the data with increasing precision. If, however, the number of SVs is excessive, i.e. a large fraction of the data turns into SVs (Fig. 2a), or a number of singleton clusters form, one should increase p to allow these points to turn into outliers, thus facilitating contour separation (Fig. 2b). As p is increased not only does the number of BSVs increase, but their influence on the shape of the cluster

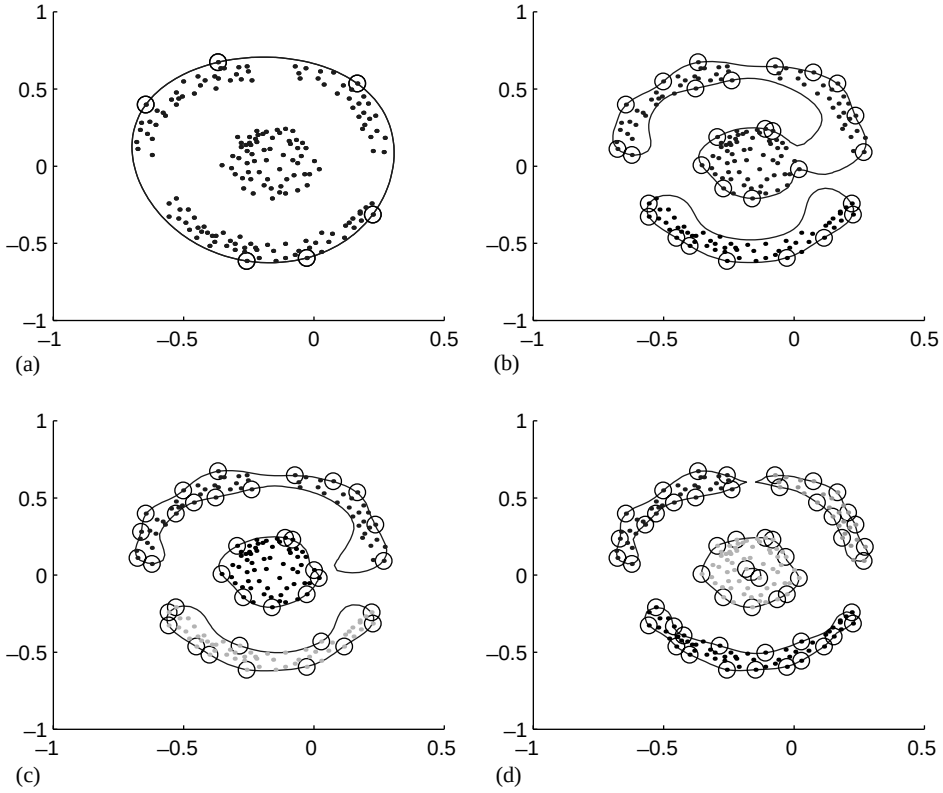

Fig. 1. Clustering of a data set containing 183 points using SVC with $C = 1$. Support vectors are designated by small circles, and cluster assignments are represented by different gray scales of the data points. (a) $q = 1$, (b) $q = 20$, (c) $q = 24$, (d) $q = 48$.

contour decreases. The number of support vectors depends on both q and p. For fixed q, as p is increased, the number of SVs decreases since some of them turn into BSVs and the contours become smoother (see Fig. 2).

3.3. The Iris data

SVC was tried [4] on the Iris data set [7], which is a standard benchmark in the pattern recognition literature, and can be obtained from the UCI repository [8]. The data set contains 150 instances each composed of four measurements of an Iris flower. There are three types of flowers, represented by 50 instances each. Clustering of this data in the space of its first two principal components is depicted [4] in Fig. 3. One of the clusters is linearly separable from the other two by a clear gap in the probability distribution. The remaining two clusters have significant overlap, and were separated at $q = 4.2$, $p = 0.55$ with four misclassifications.

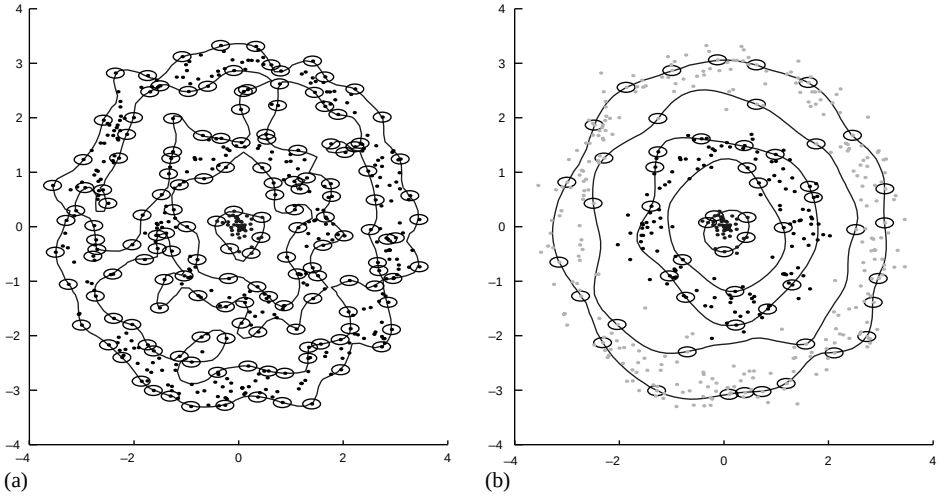

Fig. 2. Clustering with and without BSVs. The inner cluster is composed of 50 points generated from a Gaussian distribution. The two concentric rings contain 150/300 points, generated from a uniform angular distribution and radial Gaussian distribution. (a) The rings cannot be distinguished when $C = 1$. Shown here is $q = 3.5$, the lowest q value that leads to separation of the inner cluster. (b) Outliers allow easy clustering. The parameters are $p = 0.3$ and $q = 1.0$.

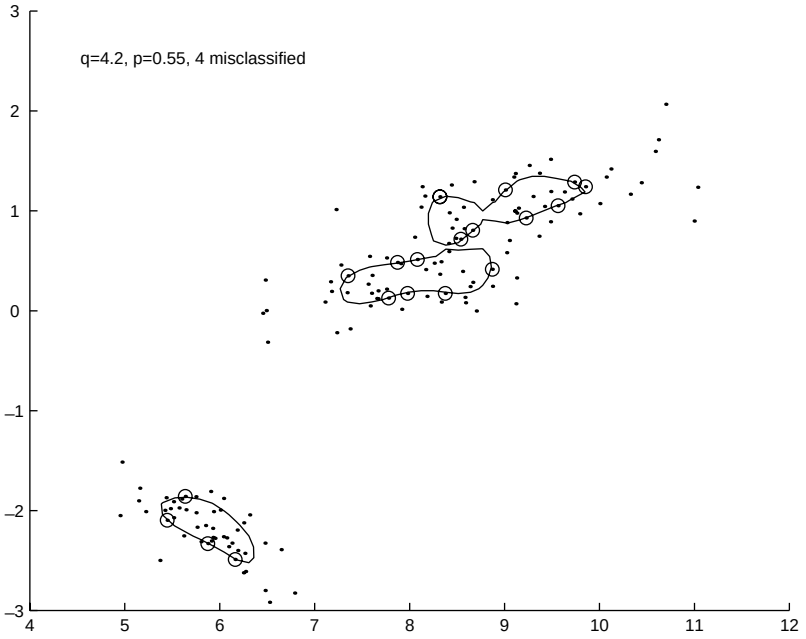

Fig. 3. Cluster boundaries of the Iris data set analyzed in a two-dimensional space spanned by the first two principal components. Parameters used are $q = 4.2$, $p = 0.55$. This resulted in four misclassifications.

These results compare favorably with other non-parametric clustering algorithms: the information theoretic approach of [9] leads to five misclassifications and the SPC algorithm [10] has 15 misclassifications.

4. Scale space algorithms

For each point $\mathbf{x}$ in data-space one may define the distance of its image in feature space from the center of the sphere:

$$R^2(\mathbf{x}) = (\Phi(\mathbf{x}) - \mathbf{a})^2 . \tag{7}$$

This can be rewritten as

$$R^2(\mathbf{x}) = 1 - 2\sum_j \beta_j K(\mathbf{x}_j, \mathbf{x}) + \sum_{i,j} \beta_i \beta_j K(\mathbf{x}_i, \mathbf{x}_j) . \tag{8}$$

Now note that one may define the function

$$P_{svc} = \sum_j \beta_j \mathrm{e}^{-q(\mathbf{x}-\mathbf{x}_j)^2} . \tag{9}$$

Because of its positive definiteness it may serve as the analog of a probability distribution. Up to constants it is the complement of R^2. The points $\mathbf{x}$ where it reaches its local maxima are the locations of the minima of $R^2(\mathbf{x})$. The minima of $R(\mathbf{x})$ are different from zero, since there exists no point in data space whose image in feature space lies at $\mathbf{a}$ (this is proved in the next section). Nonetheless, minimal radius values should correspond to centers of clusters. Hence, maxima of P_{svc} should be regarded as such.

In the limit $p \to 1$, P_{svc} is approximately equal to

$$P_R = \frac{1}{N}\sum_j \mathrm{e}^{-q(\mathbf{x}-\mathbf{x}_j)^2} . \tag{10}$$

This last expression is recognized as a Parzen window estimate [1] of the density function (up to a normalization factor). This is the probability distribution proposed by Roberts [3], who suggested looking for its maxima to identify cluster centers. This is known as a scale-space algorithm, providing for different clustering solutions by varying the scale q of the probability estimator.

Returning to SVC we note that, in the high BSV regime, the contours in data space will enclose only a small fraction of the data, hence they should be regarded as cluster *cores*. Such contours correspond to the condition $P_{svc} = \text{const.}$ with the constant chosen as the value of this function on a SV. It seems only natural to ask for the topographic map of P_{svc}, with the cluster core boundaries providing one specific value on this map. But then, in this high p limit, one may also look at the topographic map of P_R and obtain similar results. Thus, SVC-like clustering can be carried out directly with the Parzen window estimator by looking for a density contour $P_R = \text{const.}$, rather than searching for maxima of P_R.

5. Wave-function representation of Hilbert space

The vectors $\Phi(\mathbf{x}_j)$ in the Hilbert space can be represented by wave functions

$$\Phi(\mathbf{x}_j) \equiv c\mathrm{e}^{-q(\mathbf{x}-\mathbf{x}_j)^2}, \tag{11}$$

where c is an appropriate normalization constant guaranteeing that

$$\Phi(\mathbf{x}_i)\cdot\Phi(\mathbf{x}_j) \equiv c^2 \int \mathrm{e}^{-q(\mathbf{x}-\mathbf{x}_j)^2}\mathrm{e}^{-q(\mathbf{x}-\mathbf{x}_i)^2}\,\mathrm{d}\mathbf{x} = \mathrm{e}^{-q(\mathbf{x}_i-\mathbf{x}_j)^2}\,, \tag{12}$$

thus realizing the representation of the dot product by the Mercer kernel.

The center of the SVC sphere **a** becomes

$$\mathbf{a} = \sum_j \beta_j \Phi(\mathbf{x}_j) \equiv c\sum_j \beta_j \mathrm{e}^{-q(\mathbf{x}-\mathbf{x}_j)^2}\,. \tag{13}$$

Two interesting remarks can be made about this representation. First note that this sum of Gaussians cannot be represented by a single Gaussian, hence **a** cannot be the image of a point that lies in data space. Then note that this representation of **a** coincides (up to a constant) with the function P_{svc}. Thus, one may interpret $R(\mathbf{x})$ as measuring the distance between the wave function representing the point **x** and the probability distribution of SVC.

6. Quantum clustering

Let us turn now to a new [11] clustering paradigm that starts with the scale-space probability distribution and regards it as a state in Hilbert space

$$\psi(\mathbf{x}) = \sum_i \mathrm{e}^{-q(\mathbf{x}-\mathbf{x}_i)^2}\,. \tag{14}$$

This is a sum of all data images in Hilbert space of SVC. Now we proceed to search for a Hamiltonian for which this state is an eigenstate:

$$H\psi = \left(-\frac{1}{4q}\nabla^2 + V(\mathbf{x})\right)\psi = E\psi\,. \tag{15}$$

Moreover, we will require it to be the lowest eigenstate, i.e., the ground state of the operator H. Eq. (15) is a rescaled version of the Schrödinger equation in quantum mechanics. Thus a single data point at $\mathbf{x}_1$ leads to $V = q(\mathbf{x}-\mathbf{x}_1)^2$ and $E = d/2$, the analogs of the harmonic oscillator problem in quantum mechanics.

Given any ψ we can solve Eq. (15) for V:

$$V(\mathbf{x}) = E + \frac{(1/4q)\nabla^2\psi}{\psi}\,. \tag{16}$$

Let us, furthermore, require that $\min V = 0$. This sets the scale and defines

$$E = -\min\frac{(1/4q)\nabla^2\psi}{\psi}\,. \tag{17}$$

E is the minimal eigenvalue of V since ψ has no node, i.e. since ψ is positive definite over the whole space. All higher eigenfunctions have nodes whose number increases as the energy eigenvalue increases. It is quite easy to prove that

$$0 < E \leqslant \frac{d}{2} . \tag{18}$$

Minima of the potential function V may be identified with centers of attraction in quantum mechanics, hence they are identified here with centers of clusters. As will be shown below this works remarkably well. Clearly, we have also here a sliding scale determined by q. Setting this parameter means that we look for clusters on the scale $\sigma = 1/\sqrt{(2q)}$ in data space.

7. Examples

As an example [11] we display results for the crab data set taken from Ripley's book [6]. These data, given in a five-dimensional parameter space, show nice separation of the four classes contained in them when displayed in two dimensions spanned by the second and the third principal components (eigenfunctions) of the correlation matrix. The information supplied to the clustering algorithm contains only the coordinates of the data points. We display the correct classification to allow for visual comparison of the clustering method with the data. Starting with $q = 1$, or $\sigma = 1/\sqrt{2}$, we see in Fig. 4 that the Parzen probability distribution, or the wave function ψ, has only a

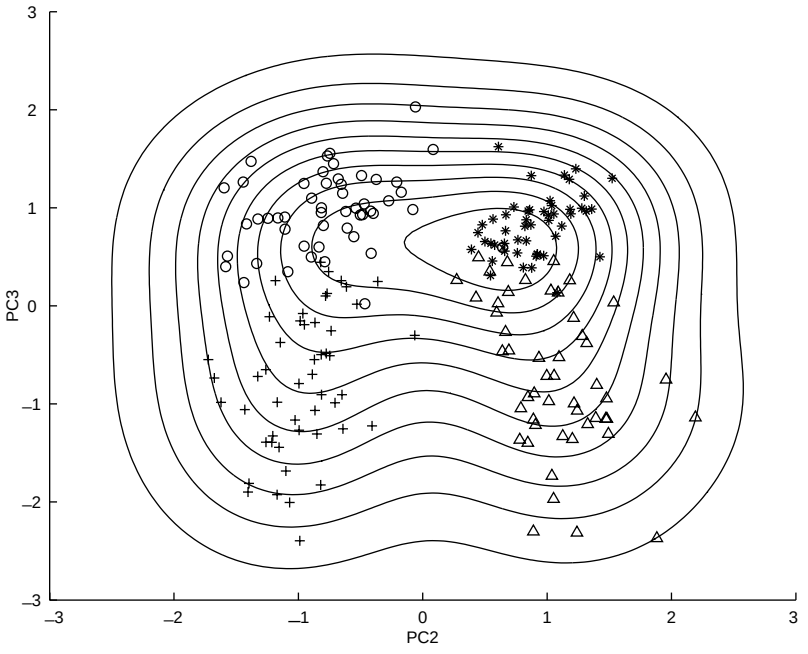

Fig. 4. Ripley's crab data [6] displayed on a plot of their second and third principal components with a superimposed topographic map of the Roberts' probability distribution for $q = 1$.

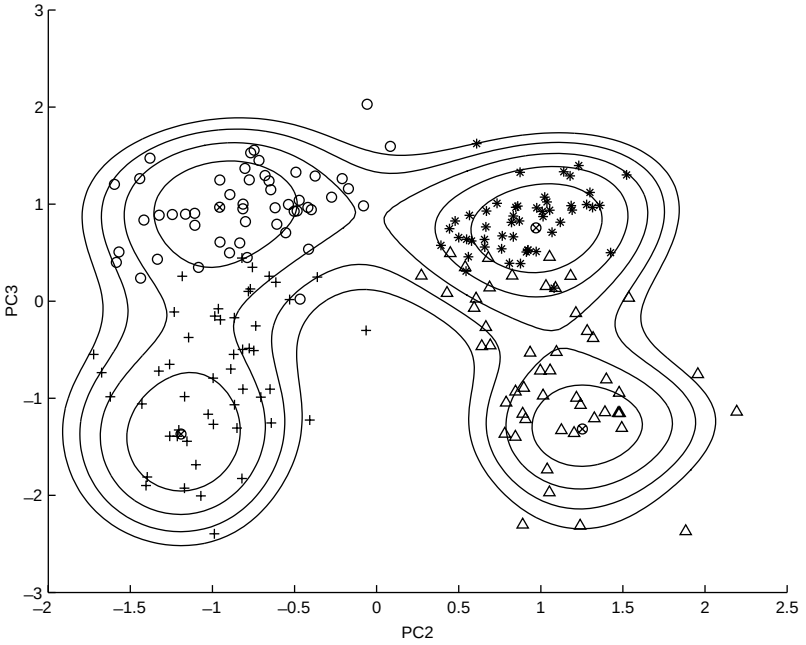

Fig. 5. Same data as in Fig. 4 with a topographic map of the potential for $\sigma = 1/\sqrt{2}$, or $q = 1$, displaying four minima (denoted by crossed circles) that may be identified with cluster centers. The contours of the topographic map are set at values of $V = cE$ for $c = 0.2$, 0.4, 0.6, 0.8, 1.

single maximum. Nonetheless, the potential, displayed in Fig. 5, shows already four minima at the relevant locations. The overlap of the topographic map of the potential with the true classification is quite amazing.

As one increases q, or reduces σ, more maxima appear in P_R and more minima appear in V. As long as the increase of q is moderate, the significant minima of V, i.e. the deep minima, remain the same four minima displayed in Fig. 5. Maxima of P_R develop at all four cluster centers at $q = 4$, but some of them are very weak. Thus, the advantage of studying V is that it has robust low minima, and that they show up at large σ (low q) values that are characteristic of the scale of the clusters that are being looked for.

References

[1] R.O. Duda, P.E. Hart, D.G. Stork, Pattern Classification, 2nd Edition, Wiley, New York, 2001.
[2] J. MacQueen, Some methods for classification and analysis of multivariate observations, Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Vol. 1, 1965, pp. 281–297.
[3] S.J. Roberts, Non-parametric unsupervised cluster analysis, Pattern Recognition 30 (2) (1997) 261–272.
[4] A. Ben-Hur, D. Horn, H.T. Siegelmann, V. Vapnik, A support vector method for clustering, in: T.K. Leen, T.G. Dietterich, V. Tresp (Eds.), Advances in Neural Information Processing Systems 13: Proceedings of the 2000 Conference, MIT Press, 2001, pp. 367–373.
[5] V. Vapnik, The Nature of Statistical Learning Theory, Springer, New York, 1995.

[6] B.D. Ripley, Pattern Recognition and Neural Networks, Cambridge University Press, Cambridge, 1996.
[7] R.A. Fisher, The use of multiple measurements in taxonomic problems, Annu. Eugenics 7 (1936) 179–188.
[8] C.L. Blake, C.J. Merz, UCI Repository of Machine Learning Databases, 1998.
[9] N. Tishby, N. Slonim, Data clustering by Markovian relaxation and the information bottleneck method, in: T.K. Leen, T.G. Dietterich, V. Tresp (Eds.), Advances in Neural Information Processing Systems 13: Proceedings of the 2000 Conference, MIT Press, 2001, pp. 640–646.
[10] M. Blatt, S. Wiseman, E. Domany, Data clustering using a model granular magnet, Neural Comput. 9 (1997) 1804–1842.
[11] D. Horn, A. Gottlieb, Quantum clustering, preprint 2001.