An Experimental Comparison of Kernel Clustering Methods

Maurizio FILIPPONE [a] Francesco MASULLI [b,c] Stefano ROVETTA [b]

[a] *Department of Computer Science, University of Sheffield, Sheffield, United Kingdom*
[b] *Department of Computer and Information Sciences, University of Genoa, and CNISM, Genova Research Unit, Genova, Italy*
[c] *Center for Biotechnology, Temple University, Philadelphia, USA*

Abstract. In this paper, we compare the performances of some among the most popular kernel clustering methods on several data sets. The methods are all based on central clustering and incorporate in various ways the concepts of fuzzy clustering and kernel machines. The data sets are a sample of several application domains and sizes. A thorough discussion about the techniques for validating results is also presented. Results indicate that clustering in kernel space generally outperforms standard clustering, although no method can be proven to be consistently better than the others.

Keywords. Kernel methods, Clustering, Experimental comparison, Fuzzy clustering, Performance indexes for clustering

1. Introduction

In this paper, we compare the performances of some among the most popular kernel clustering methods [7] on several data sets. In particular, we compare the clustering algorithms in feature space, clustering with the kernelization of the metric, Support Vector Clustering, and three standard methods: K-means, FCM-I, and FCM-II. The motivation supporting such experimental comparison lies in the fact that these recent clustering models have not been sufficiently validated in applications by the authors. The data sets included in the present study are well known among the Machine Learning community. All the data sets are labeled. Some of them can be found in the UCI repository [1], while one of them is a bioinformatic data set. We decided to include a variety of data sets differing from cardinality, dimensionality, and number of classes. The comparison is done on the basis of three performance indexes, in particular: misclassifications, normalized mutual information, and conditional entropy.

In the next sections we briefly describe the methods compared, the data sets, and the performance indexes. Section 5 shows the results, and the last section is devoted to a discussion about them.

2. Methods

2.1. K-means

This is the standard K-means clustering algorithm [13], included as a baseline method. The initialization is random, and it is made by selecting the position of the centroids among the patterns to cluster. The only input is the number c of clusters to be found.

2.2. FCM-I and FCM-II

These algorithms are two flavours of Fuzzy c-means, differing in the objective function they address. The fuzzy c-means algorithm (FCM-I) [4] identifies clusters as fuzzy sets. In the original formulation, it minimizes the functional:

$$J(U,V) = \sum_{h=1}^{n}\sum_{i=1}^{c}(u_{ih})^m \, ||\mathbf{x}_h - \mathbf{v}_i||^2 \tag{1}$$

with respect to the membership matrix U and the codebook V with the constraints $\sum_{i=1}^{c} u_{ih} = 1$. The parameter m controls the fuzziness of the memberships and often it is set to two; for high values of m the memberships tend to be equal, while for m close to one we obtain crisp memberships as in K-means. By a Lagrangian approach, the update equations are obtained as follows:

$$u_{ih} = \left[\sum_{j=1}^{c}\left(\frac{||\mathbf{x}_h - \mathbf{v}_i||}{||\mathbf{x}_h - \mathbf{v}_j||}\right)\right]^{(m-1)/2} \qquad \mathbf{v}_i = \frac{\sum_{h=1}^{n}(u_{ih})^m \mathbf{x}_h}{\sum_{h=1}^{n}(u_{ih})^m} \tag{2}$$

The method that hereinafter is referred to as FCM-II is a variation [3][14] where a maximum entropy criterion is introduced in the objective function by adding the penalty term

$$\lambda \sum_{h=1}^{n}\sum_{i=1}^{c} u_{ih}\ln(u_{ih}) \tag{3}$$

and fixing $m = 1$. The centroids are still updated with the previous equation, while the membership update is as follows:

$$u_{ih} = \frac{\exp\left(-\frac{||\mathbf{x}_h - \mathbf{v}_i||^2}{\lambda}\right)}{\sum_{j=1}^{c}\exp\left(-\frac{||\mathbf{x}_h - \mathbf{v}_j||^2}{\lambda}\right)} \tag{4}$$

In FCM-I we have to set the number of clusters c and the fuzziness m. In FCM-II we have to set the number of clusters c and the fuzziness λ.

2.3. FCM-I-fs and FCM-II-fs

These are kernel methods. They are respectively Fuzzy c-means I in feature space and Fuzzy c-means II in feature space [7].

Clustering in feature space is made by mapping each pattern using the mapping function $\Phi()$ defined (possibly only implicitly) by the kernel $K(\mathbf{a},\mathbf{b}) = \Phi(\mathbf{a})\Phi(\mathbf{b})$, and

computing centroids $\mathbf{v}_i^{\Phi}$ in feature space. Distances can be computed by means of the kernel trick.

In both algorithms, we have to select the number of clusters c and the kernel function along with its parameters. In the following, we will use the Gaussian kernel with standard deviation σ. In FCM-I fs we have to set the fuzziness m, while in FCM-II fs we have to set the fuzziness λ.

2.4. FCM-I-km and FCM-II-km

These kernel methods are respectively Fuzzy c-means I with the kernelization of the metric [15] and the Fuzzy c-means II with the kernelization of the metric.

Methods based on kernelization of the metric look for centroids in input space and the distances between patterns and centroids is computed by means of kernels:

$$||\Phi(\mathbf{x}_h) - \Phi(\mathbf{v}_i)||^2 = K(\mathbf{x}_h, \mathbf{x}_h) + K(\mathbf{v}_i, \mathbf{v}_i) + 2K(\mathbf{v}_i, \mathbf{x}_h) \tag{5}$$

In both algorithms we have to select the number of clusters c and the kernel function along with its parameters. In the following, we will use the Gaussian kernel with standard deviation σ. In FCM-I fs we have to set the fuzziness m, while in FCM-II fs we have to set the fuzziness λ.

2.5. SVC

This is the Support Vector Clustering algorithm [2]. The aim of this approach is to look for an hypersphere centered in $\mathbf{v}$ containing almost all data, namely allowing some outliers. The support vector description of data in the kernel-induced space leads to possibly non-linear surfaces separating the clusters in the original space. A labeling algorithm is necessary to assign the same label to the patterns belonging to the same region.

We have to select the parameter C (or ν) and the kernel function along with its parameters. In the following, we will use the Gaussian kernel with standard deviation σ. We set $C = 1$, in order to avoid outlier rejection that is not handled by the other comparing algorithms. The algorithm will automatically find the number of clusters.

3. Performance Indexes

A common problem in evaluating clustering methods is related to the unsupervised nature of the method. Cluster attribution cannot be univocally determined, as opposed to supervised classification, hence it cannot be summarized by a percentage of correct attributions. Moreover, different algorithms and different runs of a given algorithm may produce cluster differing in composition and sometimes in number, and, even when very similar, clusters may be ordered differently from run to run, making it harder to match two clustering results. The following performance indexes are a collection of criteria to evaluate clustering performance, that also exploit supervised information available for the data sets. Therefore, we will only select datasets for which a target is available. By proposing several indicators, the reliability of the results is somewhat increased.

Let $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ be a labeled data set composed of n patterns. Let's denote the class labels with t_i, belonging to the set of the possible realizations $\mathcal{T} = \{t_1, \ldots, t_b\}$. The

class labels can be considered as the realization of a random variable T. Applying a clustering algorithm to the elements of X, we obtain the cluster labels z_i that can be seen as the realization of a random variable Z. Here z_i belongs to the set of possible realizations $\mathcal{Z} = \{z_1, \ldots, z_c\}$. In this context, it is possible to apply some statistical tools to analyze the dependence between these two random variables.

3.1. Simple Match

A simple choice could be the match between the two realizations. In order to do that, we have to take into account two things: in general c and b are not equal and the sets of labels $\mathcal{T}$ and $\mathcal{Z}$ might be different. For these reasons we need to rearrange the cluster labels according to a univoque criterion. A natural choice is to match as much as possible the class labels. In other words, we need to transform the cluster label with a function $\pi_k : \mathcal{Z} \to \mathcal{T}$ such that $\pi_k(z_i) = t_j$. In this way we obtain the new cluster labels vector $\{t'_1, \ldots, t'_n\}$. Now it is possible to compute the match between the two label vectors. We will use the misclassification [12]:

$$\mu = \#\{t'_i \neq t_i\} \tag{6}$$

and the accuracy:

$$\psi = \#\{t'_i = t_i\}/n \tag{7}$$

Among all the permutations π_k, we select the one leading to the minimum value of μ.

3.2. Preliminary definitions for entropy based scores

Let's define the confusion matrix:

		Cluster Labels			
		z_1	z_2	$\cdots$	z_c
Class Labels	t_1	a_{11}	a_{12}	$\cdots$	a_{1c}
	t_2	a_{21}	a_{22}	$\cdots$	a_{2c}
	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
	t_b	a_{b1}	a_{b2}	$\cdots$	a_{bc}

Each entry a_{ij} of the confusion matrix contains the number of times that the clustering algorithm assigned the cluster label z_j to the pattern $\mathbf{x}_i$ having class labels t_i. On the basis of the confusion matrix, the following probabilities can be defined:

$$\begin{aligned} p(t_i) &= \frac{|t_i|}{n} = \frac{\sum_r a_{ir}}{n} \\ p(z_j) &= \frac{|z_j|}{n} = \frac{\sum_r a_{rj}}{n} \\ p(t_i, z_j) &= \frac{a_{ij}}{n} \end{aligned} \tag{8}$$

Entropy for the random variables T and Z is defined as follows:

$$H(T) = \sum_i p(t_i) \log\left(p(t_i)\right) \tag{9}$$

$$H(Z) = \sum_j p(z_j) \log\left(p(z_j)\right) \tag{10}$$

The joint entropy of T and Z is:

$$H(T,Z) = -\sum_{ij} p(t_i, z_j) \log\left(p(t_i, z_j)\right) \tag{11}$$

We will use the following two entropy based scores to assess the quality of the clustering results: Conditional Entropy $H(T|Z)$ and Normalized Mutual Information $I_{\mathrm{N}}(T,Z)$.

3.3. Conditional Entropy

The Conditional Entropy $H(T|Z)$ is a measure of the uncertainty of a random variable T given the value of the random variable Z [6]. It is defined as:

$$H(T|Z) = \sum_j p(z_j) H(T|Z = z_j) = -\sum_j p(z_j) \sum_i p(t_i|z_j) \log\left(p(t_i|z_j)\right) \tag{12}$$

Applying some transformations, it is possible to rewrite the Conditional Entropy:

$$H(T|Z) = H(T,Z) - H(Z) = -\sum_{ij} p(t_i, z_j) \log\left(p(t_i, z_j)\right) + \sum_j p(z_j) \log\left(p(z_j)\right) \tag{13}$$

Intuitively, if the two random variables are identical, knowing the realization of Z gives no uncertainty about T, leading to a null conditional entropy. On the contrary, if the two random variables are independent, there is still uncertainty in the value of T given Z. Formally, in the dependent case, $p(t_i|z_j) = 1$ leading to $H(T|Z) = 0$. In the independent case, $p(t_i|z_j) = p(t_i)$ leading to $H(T|Z) = H(T)$. The Conditional Entropy is zero when each cluster found contains pattern from a single class. This can be useful to check the purity of the cluster labels Z with respect to the class labels T. On the other hand, the method is biased when the number of clusters c is very large. In the extreme case when we assign one pattern per cluster, the Conditional Entropy results $H(T|Z) = 0$.

3.4. Normalized Mutual Information

The mutual information between two discrete random variables T and Z is [6]:

$$I(T,Z) = \sum_{ij} p(t_i, z_j) \log\left(\frac{p(t_i, z_j)}{p(t_i)p(z_j)}\right) \tag{14}$$

The mutual information measures the information shared by two discrete random variables: it measures how much knowing one of these variables reduces our uncertainty about the other. Intuitively, if the two random variables are independent, knowing the realization of one of them does not give any information about the other and viceversa; their mutual information is zero. If the two random variables are identical, the realization of one of them determines the value of the other and viceversa. As a result, the mutual information is the same as the uncertainty contained in either one of the random variables, that is their entropy. Formally, if they are uncorrelated, it is possible to factorize the joint probability $p(t_i, z_j) = p(t_i)p(z_j)$ leading to $I(T,Z) = 0$. If they are identical, $I(T,Z)$ reduces

to the entropy $H(T) = H(Z)$, since $p(x,y) = p(x) = p(y)$. These considerations show that the mutual information is dependent on the data set; in other words, the upper bound is not independent from the considered problem. It is possible to normalize $I(T,Z)$ in the interval $[0,1]$ using the following [6]:

$$I_{\mathrm{N}}(T,Z) = \frac{I(T,Z)}{\sqrt{H(T)H(Z)}} \tag{15}$$

In this way, a value of $I_{\mathrm{N}}(T,Z)$ close to one means high correlation between cluster and class labels, a value near zero means independence.

4. Data Sets

As noted earlier, all data sets are labelled, so that we can exploit the supervised information for ease of cluster evaluation.

4.1. Iris

(150 patterns of dimension 4, 3 classes.) This is one of the most popular data sets studied by the Machine Learning community [8,5]. The data set contains three classes of 50 patterns each; each class refers to a type of iris plant. One class is linearly separable from the other two that are overlapped. The features are four: sepal length, sepal width, petal length, and petal width.

4.2. Breast

(683 patterns of dimensio 9, 2 classes.) The Breast Cancer Wisconsin (Original) Data Set was obtained by the University of Wisconsin Hospitals, Madison from Dr. William H. Wolberg [16]. The samples were analyzed in different moments, since they were received periodically. The data set is composed by 699 nine-dimensional patterns, labeled as benign or malignant. Since there are some missing values, we decided to remove the corresponding patterns, obtaining 683 patterns. The class distribution is 65% for the benign class and 35% for the Malignant class.

4.3. Ecoli

(336 patterns of dimension 7, 8 classes.) Contains the protein localization sites of a E. coli [11]. The 336 patterns are described by seven features, and are classified in eight classes. Three of these classes contain less than five patterns.

4.4. Glass

(214 patterns of dimension 9, 6 classes.) This data set contains 214 patterns related to the analysis of types of glass. The nine features describing each pattern are the refractive index and the concentration of eight chemical elements (Na, Mg, Al, Si, K, Ca, Ba, and Fe). The type of glass can be one among six: building windows float processed, building windows non float processed, vehicle windows float processed, containers, tableware, and headlamps.

4.5. Lung

(32 patterns of dimension 54, 3 classes.) The data set was published in Ref. [10]. It contains 32 54-dimensional patterns that can belong to one out of three types of pathological lung cancers. The Authors give no information about the individual variables.

5. Results

The methods presented in Section 2 have been tested on the data sets described in Section 4. The number of classes can give some guidelines on the selection of the number of clusters. It is worth noting, however, that in general the number of clusters and the number of classes might be not related to each other. A typical example is the Iris data set, where the two overlapped classes are very likely to be identified as one cluster by a clustering algorithm. In other situations, it is possible to use some prior information about the number of clusters. To perform a fair comparison among the methods, we used the same number of clusters for all of them. Some algorithms find the natural number of clusters given a particular set of parameters. In this case, we set the parameters in order to have a selection of the wanted number of clusters by the algorithm. We tested the methods varying all the parameters in a wide range; we report the results for the selection giving the best performances. For the algorithms starting with a random initialization, the results are averaged over 20 runs; in Table 1 each score is reported along with its standard deviation.

6. Discussion

By inspecting the experimental results, it is possible to see that there are no methods that perform better or worse than the others in general.

Concerning clustering methods using kernels, in general, we can see that the methods in feature space perform better than methods with the kernelization of the metric. Clustering with the kernelization of the metric, in some situations give very poor results, especially when the number of clusters is very high. SVC has been used only with $C = 1$, i.e., without the rejection of the outliers. This fact affected the results that are not very good in general. On the other hand, this choice was necessary to compare its results with the other methods that do not handle an outlier class.

An important result, that is clear from the experimental validation, is that clustering in kernel induced spaces outperform standard clustering algorithms. This is one of the motivations that support the interest of the Machine Learning community for these recent clustering techniques. On the other hand, the methods based on kernels require the tuning of the kernel or the adjacency function. In many applications, we found that the values of the standard deviation of such functions lead to good performances only in a narrow interval.

The results therefore show that it is very difficult to identify the best approach in terms of accuracy, and a similar conclusion (although not considered in this study) applies to computational efficiency.

Iris				
Method	Parameters	ψ (μ)	$I_N(T,Z)$	$H(T\|Z)$
FCM-I-fs	$c = 3$, $\sigma = 0.6$, $m = 1.2$	0.947, 0.000 (8.0, 0.0)	0.845, 0.000	0.172, 0.000
FCM-II-fs	$c = 3$, $\sigma = 0.6$, $\lambda = 0.1$	0.923, 0.017 (11.5, 2.6)	0.810, 0.024	0.214, 0.029
FCM-I-km	$c = 3$, $\sigma = 3$, $m = 2.4$	0.907, 0.000 (14.0, 0.0)	0.766, 0.000	0.260, 0.000
FCM-II-km	$c = 3$, $\sigma = 5$, $\lambda = 0.2$	0.913, 0.000 (13.0, 0.0)	0.745, 0.000	0.283, 0.000
SVC	$c = 3$, $C = 1$, $\sigma = 0.35$	0.680, 0.000 (48.0, 0.0)	0.736, 0.000	0.453, 0.000
FCM-I	$c = 3$, $m = 2.4$	0.900, 0.000 (15.0, 0.0)	0.758, 0.000	0.270, 0.000
FCM-II	$c = 3$, $\lambda = 5.4$	0.913, 0.000 (13.0, 0.0)	0.745, 0.000	0.283, 0.000
K-means	$c = 3$	0.860, 0.083 (21.1, 12.5)	0.733, 0.061	0.309, 0.087

Breast				
Method	Parameters	ψ (μ)	$I_N(T,Z)$	$H(T\|Z)$
FCM-I-fs	$c = 3$, $\sigma = 7.2$, $m = 1.2$	0.972, 0.003 (18.9, 2.2)	0.702, 0.039	0.103, 0.014
FCM-II-fs	$c = 2$, $\sigma = 8$, $\lambda = 0.35$	0.972, 0.000 (19.0, 0.0)	0.814, 0.000	0.116, 0.000
FCM-I-km	$c = 2$, $\sigma = 0.1$, $m = 1.2$	0.653, 0.000 (237.0, 0.0)	0.009, 0.000	0.646, 0.000
FCM-II-km	$c = 2$, $\sigma = 0.01$, $\lambda = 0.02$	0.652, 0.000 (238.0, 0.0)	0.007, 0.000	0.646, 0.000
SVC	$c = 3$, $C = 1$, $\sigma = 3.75$	0.652, 0.000 (238.0, 0.0)	0.018, 0.000	0.646, 0.000
FCM-I	$c = 2$, $m = 1.2$	0.960, 0.000 (27.0, 0.0)	0.748, 0.000	0.166, 0.000
FCM-II	$c = 2$, $\lambda = 400$	0.972, 0.000 (19.0, 0.2)	0.812, 0.002	0.118, 0.001
K-means	$c = 2$	0.960, 0.000 (27.0, 0.0)	0.748, 0.000	0.166, 0.000

Ecoli				
Method	Parameters	ψ (μ)	$I_N(T,Z)$	$H(T\|Z)$
FCM-I-fs	$c = 7$, $\sigma = 0.6$, $m = 1.6$	0.732, 0.001 (90.0, 0.2)	0.459, 0.001	0.731, 0.002
FCM-II-fs	$c = 7$, $\sigma = 0.8$, $\lambda = 0.09$	0.727, 0.009 (91.8, 2.9)	0.455, 0.012	0.739, 0.022
FCM-I-km	$c = 7$, $\sigma = 0.1$, $m = 1.2$	0.446, 0.000 (186.0, 0.0)	0.046, 0.000	1.446, 0.000
FCM-II-km	$c = 7$, $\sigma = 0.1$, $\lambda = 0.002$	0.443, 0.000 (187.0, 0.0)	0.045, 0.000	1.448, 0.000
SVC	$c = 7$, $C = 1$, $\sigma = 0.22$	0.446, 0.000 (186.0, 0.0)	0.148, 0.000	1.450, 0.000
FCM-I	$c = 7$, $m = 1.6$	0.724, 0.001 (92.8, 0.4)	0.458, 0.004	0.738, 0.007
FCM-II	$c = 7$, $\lambda = 0.06$	0.720, 0.009 (94.1, 3.1)	0.453, 0.015	0.746, 0.025
K-means	$c = 7$	0.705, 0.016 (99.0, 5.4)	0.429, 0.024	0.790, 0.047

Glass				
Method	Parameters	ψ (μ)	$I_N(T,Z)$	$H(T\|Z)$
FCM-I-fs	$c = 6$, $\sigma = 1$, $m = 1.4$	0.623, 0.019 (80.8, 4.1)	0.408, 0.006	0.856, 0.013
FCM-II-fs	$c = 6$, $\sigma = 0.8$, $\lambda = 0.2$	0.624, 0.010 (80.5, 2.2)	0.381, 0.012	0.898, 0.018
FCM-I-km	$c = 6$, $\sigma = 2$, $m = 1.2$	0.463, 0.000 (115.0, 0.0)	0.074, 0.000	1.391, 0.000
FCM-II-km	$c = 6$, $\sigma = 10$, $\lambda = 0.001$	0.393, 0.000 (130.0, 0.0)	0.039, 0.000	1.451, 0.000
SVC	$c = 6$, $C = 1$, $\sigma = 1.3$	0.379, 0.000 (133.0, 0.0)	0.129, 0.000	1.443, 0.000
FCM-I	$c = 6$, $m = 1.8$	0.610, 0.002 (83.4, 0.5)	0.363, 0.001	0.946, 0.0009
FCM-II	$c = 6$, $\lambda = 1.2$	0.614, 0.038 (82.5, 8.2)	0.343, 0.027	0.976, 0.0349
K-means	$c = 6$	0.571, 0.015 (91.7, 3.2)	0.404, 0.022	0.948, 0.026

Lung				
Method	Parameters	ψ (μ)	$I_N(T,Z)$	$H(T\|Z)$
FCM-I-fs	$c = 3$, $\sigma = 4$, $m = 1.2$	0.563, 0.000 (14.0, 0.0)	0.300, 0.000	0.760, 0.000
FCM-II-fs	$c = 3$, $\sigma = 6$, $\lambda = 0.1$	0.581, 0.029 (13.4, 0.9)	0.290, 0.028	0.777, 0.024
FCM-I-km	$c = 3$, $\sigma = 70$, $m = 2$	0.553, 0.035 (14.3, 1.1)	0.293, 0.048	0.788, 0.054
FCM-II-km	$c = 3$, $\sigma = 10$, $\lambda = 0.06$	0.603, 0.015 (12.7, 0.5)	0.328, 0.005	0.754, 0.009
SVC	$c = 4$, $C = 1$, $\sigma = 1.9$	0.500, 0.000 (16.0, 0.0)	0.173, 0.000	0.970, 0.000
FCM-I	$c = 3$, $m = 2.2$	0.548, 0.030 (14.5, 0.9)	0.285, 0.061	0.790, 0.065
FCM-II	$c = 3$, $\lambda = 5$	0.633, 0.042 (11.8, 1.3)	0.363, 0.000	0.707, 0.011
K-means	$c = 3$	0.538, 0.024 (14.8, 0.8)	0.279, 0.055	0.796, 0.055

Table 1. Clustering results on all datasets.

Acknowledgments

We thank Giorgio Valentini for the useful suggestions provided while discussing this work.

References

[1] A. Asuncion and D. J. Newman. UCI machine learning repository, 2007.

[2] A. Ben Hur, D. Horn, H. T. Siegelmann, and V. Vapnik. Support vector clustering. *Journal of Machine Learning Research*, 2:125-137, 2001.

[3] G. Beni and X. Liu. A least biased fuzzy clustering method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(9):954-960, 1994.

[4] J. C. Bezdek. Pattern Recognition with Fuzzy Objective Function Algorithms. Kluwer Academic Publishers, Norwell, MA, USA, 1981.

[5] R. O. Duda and P. E. Hart. *Pattern Classification and Scene Analysis*. Wiley, 1973.

[6] Xiaoli Z. Fern and Carla E. Brodley. Random projection for high dimensional data clustering: A cluster ensemble approach. In Tom Fawcett and Nina Mishra, editors, *ICML*, pages 186–193. AAAI Press, 2003.

[7] Maurizio Filippone, Francesco Camastra, Francesco Masulli, and Stefano Rovetta. A survey of kernel and spectral methods for clustering. *Pattern Recognition*, 41(1):176–190, January 2008.

[8] Ronald A. Fisher. The use of multiple measurements in taxonomic problems. *Annals Eugenics*, 7:179–188, 1936.

[9] M. Girolami, Mercer kernel based clustering in feature space, *IEEE Trans. Neural Networks* 13 (3) (2002), pp. 780-784.

[10] Zi Q. Hong and Jing Y. Yang. Optimal discriminant plane for a small number of samples and design method of classifier on the plane. *Pattern Recognition*, 24(4):317–324, 1991.

[11] Paul Horton and Kenta Nakai. A probabilistic classification system for predicting the cellular localization sites of proteins. In *Proceedings of the Fourth International Conference on Intelligent Systems for Molecular Biology*, pages 109–115. AAAI Press, 1996.

[12] Tilman Lange, Volker Roth, Mikio L. Braun, and Joachim M. Buhmann. Stability-based validation of clustering solutions. *Neural Computation*, 16(6):1299–1323, 2004.

[13] S. Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28:129-137, 1982. Reprinted from Bell Laboratories technical note, 1957.

[14] S. Miyamoto and M. Mukaidono, Fuzzy C-Means as a regularization and maximum entropy approach, *Proceedings of the Seventh IFSA World Congress*, Prague, 86-91, 1997.

[15] Zhong D. Wu, Wei X. Xie, and Jian P. Yu. Fuzzy c-means clustering algorithm based on kernel method. *Computational Intelligence and Multimedia Applications, 2003.*

[16] W. H. Wolberg and O. L. Mangasarian. Multisurface method of pattern separation for medical diagnosis applied to breast cytology. *Proceedings of the National Academy of Sciences,U.S.A.*, 87:9193–9196, 1990.