Possibilistic Clustering in Feature Space

Maurizio Filippone, Francesco Masulli, Stefano Rovetta

Dipartimento di Informatica e Scienze dell'Informazione, Università di Genova, and
CNISM, Via Dodecaneso 35, I-16146 Genova, Italy
{filippone, masulli, rovetta}@disi.unige.it

Abstract. In this paper we propose the Possibilistic C-Means in Feature Space and the One-Cluster Possibilistic C-Means in Feature Space algorithms which are kernel methods for clustering in feature space based on the possibilistic approach to clustering. The proposed algorithms retain the properties of the possibilistic clustering, working as density estimator in feature space and showing high robustness to outliers, and in addition are able to model densities in the data space in a non-parametric way. One-Cluster Possibilistic C-Means in Feature Space can be seen also as a generalization of One-Class SVM.

1 Introduction

In the last few years, some applications of kernel methods [1] to clustering tasks have been proposed. Kernel approach allows us to implicitly map patterns into a high feature space where the cluster structure is possibly more evident than in the original data space. In the literature, kernels have been applied in clustering in different ways. We can be broadly classify these approaches in three categories, which are based respectively on the: (a) *kernelization of the metric* (see, e.g., [9,12]); (b) *clustering in feature space* (see, e.g., [11]); (c) *description via support vectors* (see, e.g., [4]). The first two keep the concept of centroid as a prototype of a cluster as it is in K-Means. Methods based on kernelization of the metric look for centroids in input space and the distance between patterns and centroids is computed through kernels. Clustering in feature space is made by mapping each pattern in feature space and then computing centroids in this new space. The description via support vectors is used in the One-Class Support Vector Machine (One-Class SVM) algorithm [4] that finds a hypersphere with minimal radius in the feature space able to enclose almost all data excluding outliers. When we go back to the input space, this hypersphere corresponds to a non-linear and possibly non-connected surface separating clusters. A labeling procedure is then applied in order to group the patterns lying in the same cluster in data space.

In this paper we present the Possibilistic C-Means in Feature Space (PCM-FS) and the One-Cluster Possibilistic C-Means in Feature Space (One-Cluster PCM-FS) algorithms, which are two novel kernel methods for clustering in feature space based on the possibilistic approach to clustering [5,6]. The proposed algorithms retain the properties of the possibilistic approach to clustering, working as density estimators in feature space and showing high robustness to outliers, and in addition they are able to model densities in the data space in a non-parametric way. Note that previous kernel approaches to possibilistic clustering [12,10,7] are based on the kernelization of the metric.

One-Cluster PCM-FS can be seen also as a generalization of One-Class SVM as it is able to find a family of minimum enclosing hyperspheres in feature space; each of such hyperspheres can be obtained by simply thresholding the memberships.

The paper is organized as follows: Sect.s 2 sketchs the main aspects of the PCM algorithm, in Sect.s 3 and 4 we introduce the PCM-FS and the One-Cluster PCM-FS while in Sect.s 5 and 6 we present some experimental results and the conclusions.

2 Possibilistic C-Means

Let U be the *membership matrix*, where each element u_{ih} ($u_{ih} \in [0, 1]$) represents the membership of the h-th pattern ($h = 1, 2, \ldots, n$) to the i-th cluster ($i = 1, 2, \ldots, c$). In the possibilistic clustering framework [5], memberships u_{ih} can be interpreted of as degrees of typicality of patterns to clusters. To this aim, in the possibilistic clustering framework we do we relax the usual *probabilistic constraint* on the sum of the memberships of a pattern to all clusters (i.e., $\sum_{i=1}^{c} u_{ih} = 1$) that applies, e.g., to the Fuzzy C-Means (FCM) [3], to this minimal set of constraints:

$$u_{ih} \in [0, 1] \quad \forall i, h \tag{1}$$

$$0 < \sum_{h=1}^{n} u_{ih} < n \quad \forall i \tag{2}$$

$$\bigvee_{i} u_{ih} > 0 \quad \forall h. \tag{3}$$

Roughly speaking, these requirements simply imply that clusters cannot be empty and each pattern must be assigned to at least one cluster.

There are two formulations of the Possibilistic C-Means (PCM) algorithm [5], [6]. Here we consider the latter which attempts to minimize the following functional:

$$J(U, V) = \sum_{h=1}^{n} \sum_{i=1}^{c} u_{ih} \|\mathbf{x}_h - \mathbf{v}_i\|^2 + \sum_{i=1}^{c} \eta_i \sum_{h=1}^{n} (u_{ih} \ln(u_{ih}) - u_{ih}) \tag{4}$$

with respect to U and the set of centroids $V = \{\mathbf{v}_1, \ldots, \mathbf{v}_c\}$. The first term of $J(U, V)$ is the expectation of distortion, while the latter is an entropic term which allows us to avoid the trivial solution with all memberships equal to zero.

Setting the gradient of $J(U, V)$ with respect to the u_{ih} and $\mathbf{v}_i$ to zero we obtain:

$$u_{ih} = \exp\left(-\frac{\|\mathbf{x}_h - \mathbf{v}_i\|^2}{\eta_i}\right) \tag{5}$$

$$\mathbf{v}_i = \frac{\displaystyle\sum_{h=1}^{n} u_{ih}\mathbf{x}_h}{\displaystyle\sum_{h=1}^{n} u_{ih}} \tag{6}$$

To perform the optimization of $J(U, V)$ we apply the Picard iterations method, by simply iterating Eq.s 5 and 6. Each iteration consists of two parts: in the first one the centroids are kept fixed and the memberships are modified using Eq. (5), while in the second one we keep the memberships fixed and update the centroids using Eq. (6). The iteration ends when a stop criterion is satisfied, e.g., memberships change less than an assigned threshold, or when no significant improvements of $J(U, V)$ are noticed. The constraint on the memberships $u_{ih} \in [0, 1]$ is satisfied given the form of Eq. 5.

The parameter η_i regulates the tradeoff between the two terms in Eq. 4 and is related to the width of the clusters. In [5,6], the authors suggest to estimate η_i using a weighted mean of the intracluster distance of the i-th cluster:

$$\eta_i = \gamma \frac{\displaystyle\sum_{h=1}^{n} u_{ih}\|\mathbf{x}_h - \mathbf{v}_i\|^2}{\displaystyle\sum_{h=1}^{n} u_{ih}} \tag{7}$$

where the parameter γ is typically set to one. The parameter η_i can be updated at each step of the algorithm or can be fixed for all iterations. The former approach can lead to instabilities since the derivation of the algorithm have been obtained considering η_i fixed. In the latter a good estimation of η_i can be obtained only when starting from a preliminary solution of the clustering solution, given, e.g., from an algorithm based on the probabilistic constraint, such as the FCM. For this reason often the PCM is usually applied as a refining step for a clustering procedure.

Note that the lack of of competitiveness among clusters due to the relaxation of the probabilistic constraints makes the PCM approach equivalent to a set of c independent estimation problems that can be solved one at a time through c independent Picard iterations of Eq. 5 and Eq. 6, i.e., one for each cluster.

The main drawback for the possibilistic clustering, as well as for most central clustering methods, is its inability to model in a non-parametric way the density of clusters of generic shape (parametric approaches such as Possibilistic C-Spherical Shells [5], instead, have been proposed).

3 Possibilistic clustering in feature space

In order to overcome this limit, we propose the Possibilistic C-Means in Feature Space (PCM-FS) algorithm. It is based on a kernelization of the PCM obtained by applying a mapping Φ from the input space S to a high dimensional feature space $\mathcal{F}$ ($\Phi : S \to \mathcal{F}$) to the patterns, and applying the PCM to them in the new space $\mathcal{F}$. The objective function to be minimized becomes:

$$J^{\Phi}(U, V^{\Phi}) = \sum_{h=1}^{n} \sum_{i=1}^{c} u_{ih} \|\Phi(\mathbf{x}_h) - \mathbf{v}_i^{\Phi}\|^2 + \sum_{i=1}^{c} \eta_i \sum_{h=1}^{n} \left(u_{ih} \ln(u_{ih}) - u_{ih} \right). \quad (8)$$

Note that the centroids $\mathbf{v}_i^{\Phi}$ of PCM-FS algorithm lie in the feature space. We can minimize $J^{\Phi}(U, V^{\Phi})$ by setting its derivatives with respect to $\mathbf{v}_i^{\Phi}$ and u_{ih} equal to zero, obtaining:

$$\mathbf{v}_i^{\Phi} = \frac{\sum_{h=1}^{n} u_{ih} \Phi(\mathbf{x}_h)}{\sum_{h=1}^{n} u_{ih}} = b_i \sum_{h=1}^{n} u_{ih} \Phi(\mathbf{x}_h), \qquad b_i \equiv \left(\sum_{h=1}^{n} u_{ih} \right)^{-1} \quad (9)$$

$$u_{ih} = \exp\left(- \frac{\|\Phi(\mathbf{x}_h) - \mathbf{v}_i^{\Phi}\|^2}{\eta_i} \right). \quad (10)$$

In principle, Eq.s 9 and 10 can be used for a Picard iteration minimizing $J^{\Phi}(U, V^{\Phi})$, but as Φ is not known explicitly, we cannot compute directly them. Despite this, if we consider Mercer Kernels [2] (symmetric and semidefinite kernels) which can be expressed as a scalar product:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j), \quad (11)$$

this relation holds (*kernel trick* [1]):

$$\|\Phi(\mathbf{x}_i) - \Phi(\mathbf{x}_j)\|^2 = K(\mathbf{x}_i, \mathbf{x}_i) + K(\mathbf{x}_j, \mathbf{x}_j) - 2K(\mathbf{x}_i, \mathbf{x}_j). \quad (12)$$

This allows us to obtain an update rule for the memberships by substituting Eq. 9 in Eq. 10:

$$u_{ih} = \exp\left[-\frac{1}{\eta_i} \cdot \left(k_{hh} - 2b_i \sum_{r=1}^{n} u_{ir} k_{hr} + b_i^2 \sum_{r=1}^{n} \sum_{s=1}^{n} u_{ir} u_{is} k_{rs} \right) \right]. \quad (13)$$

Note that in Eq. 13 we introduced the notation $k_{ij} = K(\mathbf{x}_i, \mathbf{x}_j)$. The Picard iteration then reduces to the iterative update of the memberships only using Eq. 13, ending when an assigned stopping criterion is satisfied (e.g., when memberships change less than an assigned threshold, or when no significant improvements of $J^{\Phi}(U, V^{\Phi})$ are noticed).

Concerning the parameters η_i, we can applying in the feature space the same criteria suggested for the PCM (Eq. 7) obtaining in such a way:

$$\eta_i = \gamma\ b_i\ \sum_{h=1}^{n} u_{ih} \left(k_{hh} - 2b_i \sum_{r=1}^{n} u_{ir} k_{hr} + b_i^2 \sum_{r=1}^{n} \sum_{s=1}^{n} u_{ir} u_{is} k_{rs} \right) \tag{14}$$

The parameters η_i can be estimated at each iteration or once at the beginning of the algorithm. In the latter case the initialization of the memberships, that allows to provide a good estimation of the η_i, can be obtained as a result of a Kernel Fuzzy c-Means [11].

Note that if we chose a linear kernel $k_{ij} = \mathbf{x}_i \cdot \mathbf{x}_j$ the PCM-FS reduces to the standard PCM, i.e., using a linear kernel is equivalent to put $\Phi \equiv I$, where I is the identity function. In the following, we will use a Gaussian kernel:

$$k_{ij} = \exp\left(- \frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2} \right) \tag{15}$$

for which

$$\|\Phi(\mathbf{x}_i)\|^2 = \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_i) = k_{ii} = 1. \tag{16}$$

As a consequence, patterns are mapped by the Gaussian kernel from data space to the surface of a unit hypersphere in feature space.

Centroids in the feature space $\mathbf{v}_i^{\Phi}$ are not constrained to the hyperspherical surface as mapped patterns; therefore, centroids lie inside this hypersphere, and due to the lack of competitiveness between clusters (that characterizes the possibilistic clustering framework), centroids of PCM-FS often collapse into a single one, with slight dependency on the value of the cluster spreads η_i.

Note that PCM-FS retains the principal characteristics of PCM, including the capability of estimating hyperspherical densities, this time in the feature space. In the data space this corresponds to the capability to model clusters of more general shape, a significant improvement with respect the original PCM.

4 One-Cluster Possibilistic C-Means in Feature Space algorithm

We propose now the One-Cluster Possibilistic C-Means in Feature Space (One-Cluster PCM-FS) algorithm aimed to model all data points in a single cluster in features space. We assume the presence of a unique cluster in feature space, with no regard to the number of clusters we expect to model in the data space. In the following, we will denote with u_h the membership of the h-th pattern to the cluster. It is made up by three main steps: *Core*, *Defuzzification*, and *Labeling.*

The *Core* step is the "*fuzzy*" part of the algorithm, aimed to producing a fuzzy-possibilistic model of densities (membership function) in the feature space. It is initialized by selecting a *stop criterion* (e.g., when memberships change less than an assigned threshold, or when no significant improvements of $J^{\Phi}(U, V^{\Phi})$ are noticed), setting the value of σ for the Gaussian kernel (in order to define the spatial resolution of density estimation), and initializing the memberships u_h (usually as $u_h = 1$). Then, after estimating the value of η using Eq. 14, we perform the Picard iteration using Eq. 13.

The *Defuzzification* steps filters outliers from data points by selecting a threshold $\alpha \in (0, 1)$ and using it to define an *α-cut* (or *α-level set*) on data points:

$$A_\alpha = \{\mathbf{x}_h \in X \mid u_h > \alpha\} \tag{17}$$

Note that given the form of u_h(Eq. 10) the threshold α defines a hypercircle which encloses a hyperspherical cap. A_α is then the set of data points whose mapping in feature space lies on the cap, whose base radius depends on α. Points outside the α-cut are considered to be outliers.

The *Labeling* step separates the data points belonging to the single cluster in feature space, in a number of "*natural*" clusters in data space. It uses a convexity criterion derived from the one proposed for One-Class SVM [4] assigning the same label to a pair of points only if all elements of the linear segment joining the two points in data space belong to A_α.

The *Defuzzification* and *Labeling* steps can be iterated with different values of α, thus performing a very lightweight *model selection*, without involving new runs of the *Core* step. Often, such as in the case of experiments presented in next section, an a-priori analysis of the memberships histogram permits to obtain a good evaluation of α without performing a true model selection. Indeed, the presence of multiple modes in the membership histogram indicates the presence of different structures of data in feature space, and allows us to find several levels of α discriminating the different densities of data in feature space.

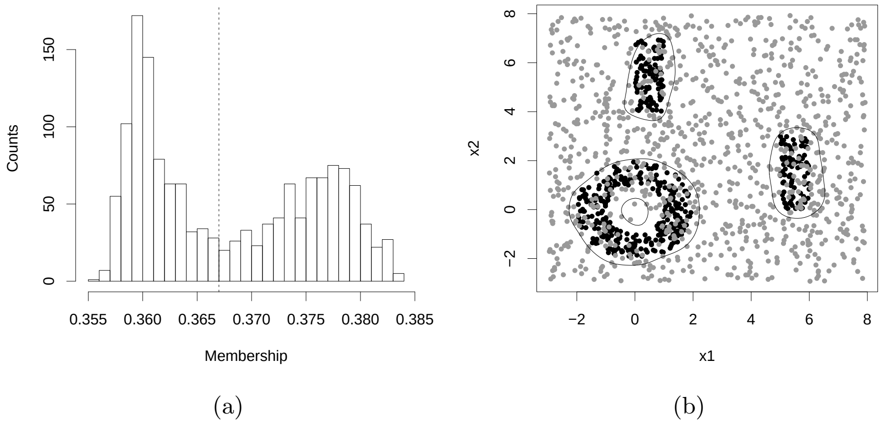
5 Experimental results and discussion

In this section we present some results obtained on a synthetic second data set (Fig. 1) consisting in three disjoint dense regions (black dots) on a 10x10 square: two rectangular regions, each of them corresponding to 1.24 % of the square and composed by 100 patterns uniformly distributed, and a ring shaped region, corresponding to 7.79 % of the square, that contains 300 patterns uniformly distributed. An uniformly distributed noise of 1000 grey dots is superimposed to the square.

We used a Gaussian kernel with standard deviation $\sigma = 0.5$ estimated as the order of magnitude of the average inter-data points distance. The memberships u_h were initialized to 1. The stop criterion was $\sum_h \Delta u_h < \varepsilon$ with $\varepsilon = 0.01$.

In the *Defuzzification* step we evaluated α using the histogram method. As shown in Fig. 1(a), choosing $\alpha = .3667$ that is the value of membership separating the two modes of the histogram, we obtain a good separating surface in the data space (Fig. 1(b)), with no need to perform any iteration for model selection.

As shown in the experiment, One-Cluster PCM-FS shows a high robustness to outliers and a very good capability to model clusters of generic shape in the data space (modeling their distributions in terms of fuzzy memberships). Moreover it is able to find *autonomously* the *natural* number of clusters in the



(a) (b)

Fig. 1. (a) Histogram of the memberships obtained by One-Cluster PCM-FS with $\sigma = 0.5$. The dotted line gets through to membership value .3667 that separates the two modes of the graph; the value of α is then taken as $\alpha = .3667$. (b) Data space: black dots belong to the dense regions and the grey ones are the noisy patterns. The contours correspond to points with membership equal to .3667.

data space. The outliers rejection ability is shared also by the standard PCM, but is limited to the case of globular clusters. The standard PCM shows also a good outliers rejection ability, but it works only with globular clusters.

One-Class SVM [4] is also able to find the "natural" number of clusters of generic shape in the data space, but it doesn't model their distribution. Moreover, One-Class SVM needs a complete *model selection* procedure involving many time consuming runs from scratch of the full algorithm.

In all the runs of One-Cluster PCM-FS the *Core* step, which involves the minimization of $J^{\Phi}(U, V^{\Phi})$ (Eq. 8), resulted to be very fast, as only less than a tenth of iterations of Eq. 13 where enough.

6 Conclusions

In this paper we have proposed the kernel possibilistic approach to clustering and two clustering algorithms, namely the Possibilistic C-Means in Feature Space and the One-Cluster Kernel Possibilistic C-Means in Feature Space algorithms which are novel kernel methods for clustering in feature space based on the possibilistic approach to clustering [5,6]. The proposed algorithms retain the properties of the possibilistic approach to clustering, working as density estimator in feature space and showing high robustness to outliers, and in addition are able to model densities in the data space in a non-parametric way.

One-Cluster PCM-FS can be seen also as a generalization of One-Class SVM as it is able to find a family of minimum enclosing hyperspheres in feature space; each of such hyperspheres can be obtained by simply thresholding the memberships. Note that, after fixed the value of the σ of the Gaussian kernel, the *model selection* does not involve the optimization step of One-Cluster PCM-FS, a.k.a. *Core* step, and can be performed very quickly. Moreover, often this is not necessary, and an analysis of the histogram of memberships can easily permit to find an optimal value for the threshold α, as in the case of the experiment shown.

Acknowledgments

Work funded by a grant from the University of Genova.

References

1. M. Aizerman, E. Braverman, and L. Rozonoer. Theoretical foundations of the potential function method in pattern recognition learning. *Automation and Remote Control*, 25:821–837, 1964.
2. N. Aronszajn. Theory of reproducing kernels. *Transactions of the American Mathematical Society*, 68(3):337–404, 1950.
3. J. C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms.* Kluwer Academic Publishers, Norwell, MA, USA, 1981.
4. A.B. Hur, D. Horn, H.T. Siegelmann, and V. Vapnik. Support vector clustering. *Journal of Machine Learning Research*, 2:125–137, 2001.
5. R. Krishnapuram and J. M. Keller. A possibilistic approach to clustering. *IEEE Transactions on Fuzzy Systems*, 1(2):98–110, 1993.
6. R. Krishnapuram and J. M. Keller. The possibilistic c-means algorithm: insights and recommendations. *IEEE Transactions on Fuzzy Systems*, 4(3):385–393, 1996.
7. K. Mizutani and S. Miyamoto. Possibilistic Approach to Kernel-Based Fuzzy c-Means Clustering with Entropy Regularization. *Second International Conf Modeling Decisions for Artificial Intelligence*, LNCS3558 Springer, pages 144–155, 2005.
8. O. Nasraoui and R. Krishnapuram. Crisp interpretations of fuzzy and possibilistic clustering algorithms. Volume 3., Aachen, Germany (1995) 1312–1318.
9. Z.D. Wu, W.X. Xie, and J.P. Yu. Fuzzy c-means clustering algorithm based on kernel method. *Fifth International Conference on Computational Intelligence and Multimedia Applications*, pages 49–54, 2003.
10. X.H. Wu and J.-J. Zhou. Possibilistic Fuzzy c-Means Clustering Model Using Kernel Methods. *Proceedings of the Int. Conf. Computational Intelligence for Modelling, Control and Automation and Int. Conf.Intelligent Agents, Web Technologies and Internet Commerce* Vol. 2, pages 465–470, 2005.
11. D. Q. Zhang and S. C. Chen. Fuzzy clustering using kernel method. In *The 2002 International Conference on Control and Automation, 2002. ICCA*, pages 162–163, 2002.
12. D. Q. Zhang and S. C. Chen. Kernel-based fuzzy and possibilistic c-means clustering. In *Proceedings of the International Conference on Artificial Neural Networks*, pages 122–125, 2003.