Learning Pairwise Similarity for Data Clustering

Ana L. N. Fred
Instituto de Telecomunicações
Instituto Superior Técnico
Lisbon, Portugal
afred@lx.it.p

Anil K. Jain
Dept. of Computer Science and Engineering
Michigan State University
East Lansing, USA
jain@cse.msu.edu

Abstract

Each clustering algorithm induces a similarity between given data points, according to the underlying clustering criteria. Given the large number of available clustering techniques, one is faced with the following questions: (a) Which measure of similarity should be used in a given clustering problem? (b) Should the same similarity measure be used throughout the $d-$dimensional feature space? In other words, are the underlying clusters in given data of similar shape?

Our goal is to learn the pairwise similarity between points in order to facilitate a proper partitioning of the data without the a priori knowledge of k, the number of clusters, and of the shape of these clusters. We explore a clustering ensemble approach combined with cluster stability criteria to selectively learn the similarity from a collection of different clustering algorithms with various parameter configurations.

1. Introduction

The goal of clustering is to find a partition of n $d-$dimensional points into k groups [5]. In general, k, the number of clusters, and cluster shapes are not known *a priori*. In the presence of *a priori* information about cluster shapes, model-based approaches, such as Gaussian mixture decomposition, can be used; otherwise, there are hundreds of candidate clustering algorithms to apply, each one assuming or imposing a given structure on the data, leading in general to distinct clustering solutions for the same data set.

Clustering ensemble methods attempt to find a robust data partitioning by combining different partitions produced by a single or multiple clustering algorithms [4, 9, 6, 1]. In most of these studies, each partition is given an equal weight in the combination process and all clusters in each partition contribute to the combined solution. Selection criteria amongst possible combination solutions has been proposed based on the quality of the overall partition. These often involve measures of consistency between combined and individual partitions in the clustering ensemble [9]. By using a fixed combination rule, the same measure of pairwise similarity is applied throughout the $d-$dimensional feature space. Law et al proposed a multiobjective data clustering method based on the selection of individual clusters produced by several clustering algorithms, through an optimization procedure [7]. They choose the best set(s) of objective functions for different parts of the feature space from the results of different clustering algorithms.

We explore a clustering ensemble approach combined with cluster stability conditions, to selectively learn the pairwise similarity. Instead of evaluating the overall performance of a clustering algorithm based on the final partition produced by it, we assume that each algorithm can have different levels of performance in different regions of the $d-$dimensional space. We suggest that meaningful clusters can be identified based on cluster stability criteria and propose to learn pairwise similarity based only on these stable clusters. Thus, only those clusters passing the stability test will contribute towards assessing the pairwise similarity, expressed as an $n \times n$ co-association matrix. We show that this matrix is able to capture the intrinsic similarity between points and thereby extract the underlying clustering structure.

2. Proposed Approach

Let n be the number of objects in a data set. The proposed approach to learn pairwise similarity based on multicriteria clustering is schematically presented in figure 1.

2.1. Learning Pairwise Similarity

For the i-th clustering algorithm, Alg_i, the quality of each cluster, c_j^i, $j = 1, \ldots, k_i$, $i = 1, \ldots, M$, is assessed based on cluster stability, leading to tuples $(c_j^i, stab_j^i)$,

where k_i is the number of clusters in the partition produced by Alg_i. Meaningful clusters are filtered out from these partitions, by selecting only those clusters that exhibit average stability above a given threshold, th. A max rule combines the $n \times n$ sub-matrices $[C]^i$, $i = 1, \ldots, M$ (M being the total number of clusterings) into the learned pairwise similarity in a co-association matrix $\mathbf{C_M}$.

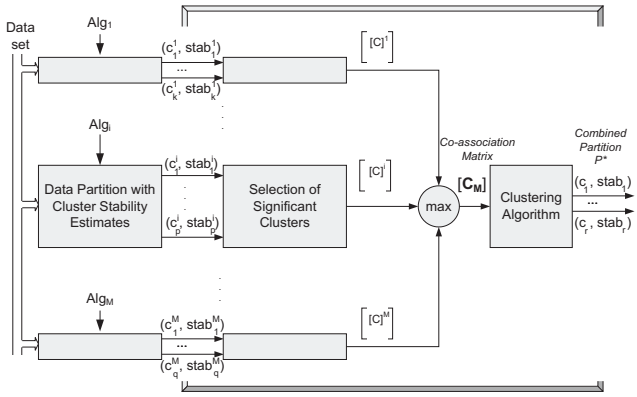


Figure 1. Learning pairwise similarity and robust data clustering with multiple clustering criteria.

The use of different parameter configurations for each clustering algorithm enables the derivation of similarity between patterns without the *a priori* information about the number of clusters or the tuning of parameter values. With this multi-criteria procedure, the learned similarity matrix is expected to better reflect the underlying clustering structure, thus facilitating a proper partitioning of the data. We use the average link method (AL) with an automatic selection of the number of clusters, based on cluster lifetime criteria [4, 3], to produce the final partitioning. Using the information in matrix $\mathbf{C_M}$, the average stability of each cluster in the final partition is computed, providing a valuable indicator of the quality of the combined solution. This approach will be hereafter referred as *Multi-Criteria Evidence Accumulation Clustering* (Multi-EAC).

2.2. Measuring Cluster Stability

Our basic premise is that a spurious cluster generated by a clustering algorithm is not likely to be stable. Therefore, we explore subsampling techniques to assess cluster stability. Law et al. [7] proposed a method to evaluate the stability of individual clusters within a partition, based on sampling and matching techniques. Here, we propose an approach based on the Evidence Accumulation Clustering (EAC) [4], according to which both the stability of pairwise associations and of individual clusters are assessed – see figure 2.

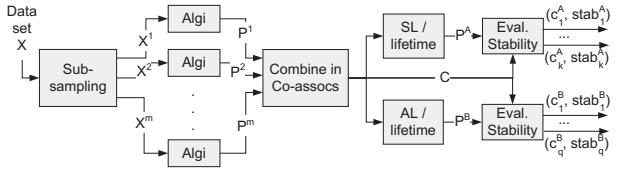
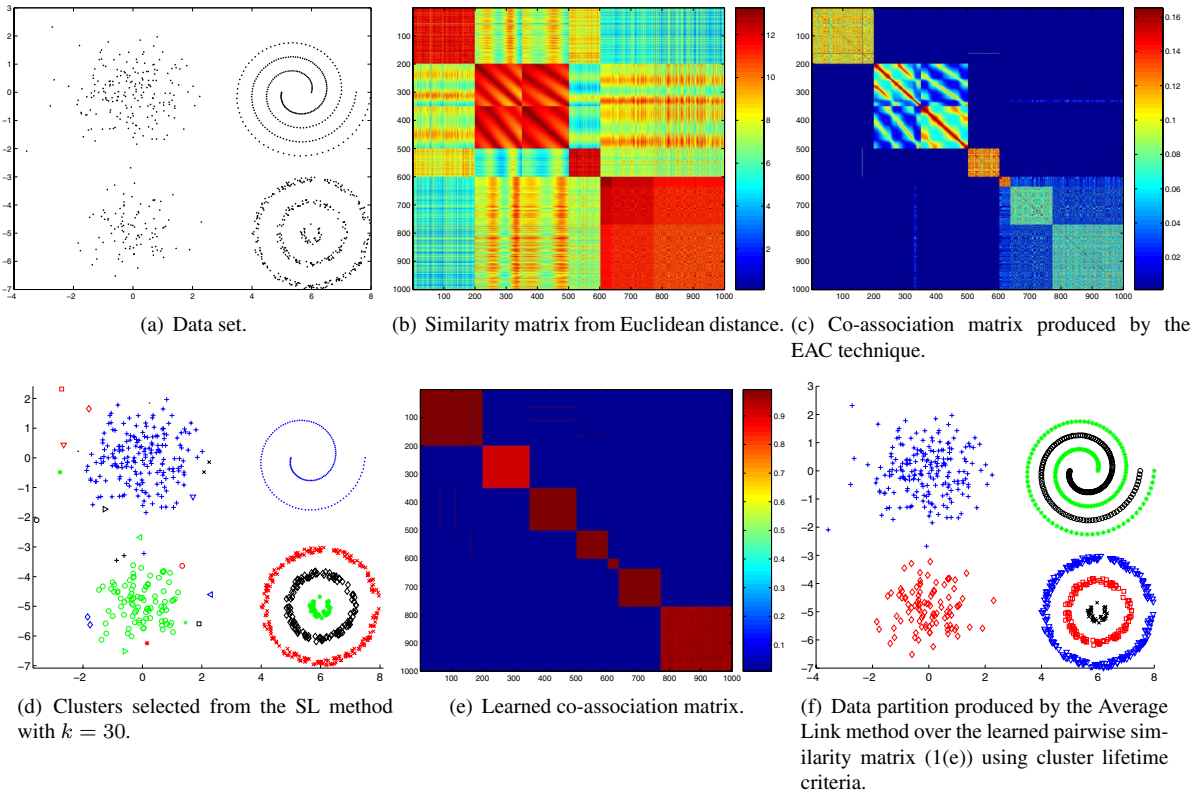


Figure 2. Evaluation of cluster stability under an ensemble approach using EAC.

The $i-$th clustering algorithm, Alg_i, is applied to m data realizations, $X^j, j = 1, \ldots, m$, obtained by subsampling the original data set X, leading to m data partitions, $P^j, j = 1, \ldots, m$. Applying the EAC method, the m partitions are combined into a $n \times n$ co-association matrix C, where $C_{i,j} = \frac{n_{i,j}}{m_{i,j}}$, $n_{i,j}$ is the number of times the pair of objects with indices i and j were grouped together in a cluster, over the m clusterings, and $m_{i,j}$ is the number of data sets X^i where this pair of objects were simultaneously present, $0 \leq m_{i,j} \leq m$. To obtain stable, parameter-free solutions, we apply the single link (SL) and the average link (AL) algorithms, to C to obtain two partitions, respectively, P^A and P^B (possibly with different number of clusters), using the cluster lifetime criteria. The l-th cluster of partition p (p being A or B) is represented as c_l^p; pairwise stability of objects within this cluster is represented by the matrix $stab_l^p$. The stability of individual clusters within each partition is measured as the average stability of the pairs of objects in each cluster: $\overline{stab_l^p} = \sum_{i,j,\in c_l^p} \frac{stab_l^p(i,j)}{|c_l^p|(|c_l^p|-1)}$, where $|c_l^p|$ denotes the number of objects in cluster c_l^p. Average stability values are used to select clusters to learn the pairwise similarity, and also as a measure of the quality of final combined partitions obtained from the learned similarity.

2.3. Illustrative Example

The proposed method is illustrated on the synthetic data set in fig. 3(a). Nineteen different clustering algorithms were applied to this data: K-means (k =7, 9, 20, 30 and 40); single linkage (SL) (forcing k=30, 40); and the spectral clustering algorithm (SC) [8] (k =7, and 30, and σ = 0.1, 0.3, 0.5, and 0.7). The stability of individual clusters is assessed by measuring the average consistency of pairwise associations by sub-sampling (90% of the samples) the data, a total of 100 times. Figure 3(d) illustrates clusters contributing to the learning of similarity using a threshold $th = 0.9$, meaning that only clusters that were consistently produced 90% of the time were selected. The learned pairwise similarity (figure 3(e)) shows a clear separation between clusters (represented as blocks in the diagonal of the co-association matrix) compared to the original Euclidean-based similarity (figure 3(b)) and the co-association matrix produced by the evidence accumulation technique (EAC) [4] (see figure 3(c)). Fig. 3(f) shows the final partition produced by ap-



(a) Data set. (b) Similarity matrix from Euclidean distance. (c) Co-association matrix produced by the EAC technique.

(d) Clusters selected from the SL method with $k = 30$. (e) Learned co-association matrix. (f) Data partition produced by the Average Link method over the learned pairwise similarity matrix (1(e)) using cluster lifetime criteria.

Figure 3. Illustration of the Multi-EAC technique for learning pairwise similarity.

plying the average link (AL) to the learned similarity in fig. 3(e), using the lifetime criteria [4] to determine the final number of clusters.

3. Experimental Results and Discussion

We applied the proposed pairwise similarity learning technique to a number of real data sets: *Iris*, Wisconsin *Breast-Cancer*, *Optical digits* (from a total of 3823 samples, each with 64 features, we used a subset composed of the first 100 samples of all the digits), and *Log Yeast* and *Std Yeast* (consisting of the logarithm and the standardized version, respectively, of gene expression levels of 384 genes over two cell cycles of yeast cell data), available in the UCI Machine Learning Repository [2].

	K-Means & SL: k	SC: k-σ
Iris	3, 5, 10, 12, 15	3-0.1, 12-0.1
Breast-C.	2, 3, 5, 10	2-2.0, 3-2.0
optdigits	10, 15, 20, 30	10-20.0, 20-20.0
log_yeast	5, 7, 10, 20, 30, 40	5-1.0, 7-1.0
std_yeast	5, 7, 10, 20, 30, 40	5-1.0, 7-1.0

Table 1. Parameter values used with the different clustering algorithms.

The clustering algorithms and the associated parameter values used are given in table 1. For each clustering algorithm and each parameter configuration, m=100 data partitions were produced by applying the algorithm to 100 data realizations obtained by sub-sampling (90% of the samples).

Table 2 summarizes the experimental results, and provides a direct comparison between the multi-EAC and the EAC methods. Two sets of experiments were performed: (i) using only the K-Means and the SL algorithms to produce clustering ensembles (columns 2 to 5); and (ii) using all three algorithms (columns 6 to 9). Results with the EAC correspond to applying the EAC technique to the union of the clustering ensembles produced by sub-sampling with the various clustering algorithms and parameter configurations, and applying the average link algorithm over the corresponding co-association matrix; the final number of clusters is either set to the natural number of clusters (columns *(k-known)*), or automatically determined using the lifetime criteria (columns *(lifetime)*). With the Multi-EAC method, a threshold value, th, determines the clusters selected from the partitions obtained with each clustering ensemble. We started with $th = .95$; if the resulting similarity matrix, $\mathbf{C_M}$, had more than 10% samples unassigned (because no cluster with stability above this threshold was found involving these samples), we lowered the threshold by 0.05; the process was repeated, when necessary, until 90% coverage of the data set was achieved or when the threshold reached

	Clustering Ensembles: K-means, SL				Clustering Ensembles: K-means, SL, SC			
	EAC		Multi-EAC		EAC		Multi-EAC	
	Ci (k-known)	Ci (lifetime)	Ci (lifetime)	th	Ci (k-known)	Ci (lifetime)	Ci (lifetime)	th
Synthetic	84.8	68.0	99.4	0.90	84.8	68.0	100.0	0.90
Iris	68.7	66.7	88.7	0.95	68.0	68.0	88.7	0.95
Breast-C.	65.4	65.4	96.2	0.95	97.1	97.1	96.3	0.95
optdigits	30.6	30.5	76.5	0.75	30.5	30.5	83.1	0.80
log_yeast	35.2	34.9	35.2^*	0.90	35.4	35.2	35.4^*	0.90
std_yeast	36.2	36.2	34.4^*	0.90	36.5	36.2	33.9^*	0.90

Table 2. Combination results. $*$ indicates clusters with average stability below 0.75.

levels below 0.75. The threshold values used with each data set is shown in columns labeled th. The overall quality of the final clusterings produced is measured using the consistency index C_i: $C_i(P^*, P^o)$, obtained by matching the clustering results, P^*, with ground truth information P^o, and counting the percentage of agreement between these labelings.

As shown in table 2, column 4, good clustering results were obtained by using only two clustering algorithms, namely the K-means and the SL, to produce clustering ensembles and learn the pairwise similarity. Without the use of *a priori* information about the number of clusters or tuning of clustering parameters, the Multi-EAC method achieved a better performance than the EAC method, either when using the lifetime criteria (column 3) or assuming the number of clusters is known (column 2). The only situation where the Multi-EAC method gave worse results compared to the EAC method was with the std_yeast data set. However, for this data set, the presence of clusters with low average stability (below 0.5) reveals that the obtained solution is not very reliable, and additional clustering methods should be tested. By including an additional algorithm to produce clustering ensembles (the spectral clustering algorithm), results did not improve significantly except for the optidigits data set.

It is important to note that, in addition to generating a data partition, the Multi-EAC method provides average stability indices for each cluster, thus elucidating the quality of these clusters. The presence of low stability clusters can be used as a warning to investigate additional clustering algorithms that may propose more meaningful clustering solutions to the given data.

4. Conclusions

We have proposed a cluster ensemble approach for learning pairwise similarity between objects. The quality of individual clusters is evaluated based on a stability measure, which is used for the selection of meaningful clusters leading to a delineation of regions of the feature space where the underlying clustering criteria (similarity measure) may contribute to the overall estimation of pairwise similarity. The use of different parameter configurations enables the derivation of similarity between patterns without the use of *a priori* information about the number of clusters or the tuning of parameter values. Furthermore, results of the application of the AL method with lifetime criteria over the learned pairwise similarity on several artificial and real data sets showed the robustness and usefulness of the proposed approach on revealing the underlying clustering structure in the data.

Acknowledgments

This work was partially supported by the grants FCT POSI/EEA-SRI/61924/2004 and ONR N00014-04-1-0183.

References

[1] H. Ayad and M. Kamel. Cluster-based cumulative ensembles. In N. Oza and R. Polikar, editors, *Proc. the 6th International Workshop on Multiple Classifier Systems*, pages 236–245. LCNS 3541, 2005.

[2] C. B. D.J. Newman, S. Hettich and C. Merz. UCI repository of machine learning databases, 1998.

[3] A. Fred and A. K. Jain. Robust data clustering. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2003*, Madison, June 2003.

[4] A. L. Fred and A. K. Jain. Combining multiple clusterings using evidence accumulation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 27(6):835–850, 2005.

[5] A. Jain, M. N. Murty, and P. Flynn. Data clustering: A review. *ACM Computing Surveys*, 31(3):264–323, September 1999.

[6] L. I. Kuncheva and S. Hadjitodorov. Using diversity in cluster ensembles. In *Proc. of IEEE Intl. Conference on Systems, Man and Cybernetics*, pages 1214–1219, 2004.

[7] M. H. C. Law, A. P. Topchy, and A. K. Jain. Multiobjective data clustering. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 424–430, Washington D.C., 2004.

[8] A. Y. Ng, M. I. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In T. G. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems 14*, Cambridge, MA, 2002. MIT Press.

[9] A. Strehl and J. Ghosh. Cluster ensembles - a knowledge reuse framework for combining multiple partitions. *Journal of Machine Learning Research*, 3(Dec):583–617, 2003.