Science The World's Leading Journal of Original Scientific Research, Global News, and Commentary.

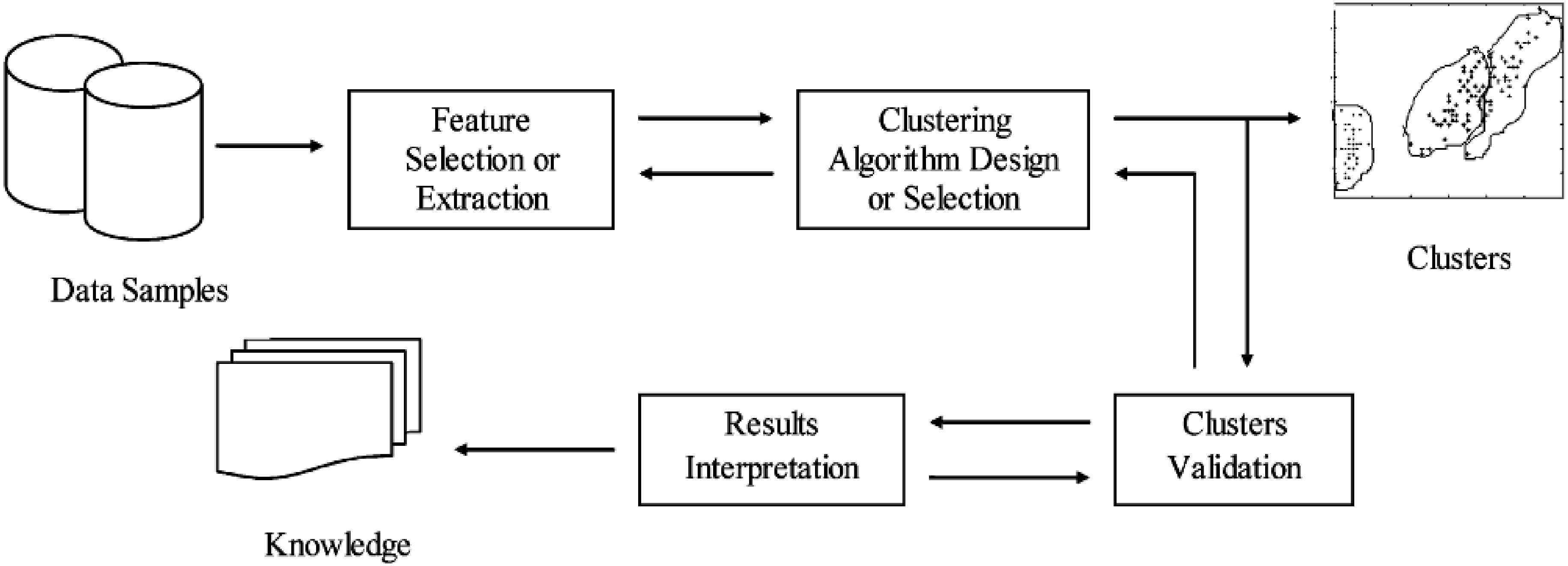
MACHINE LEARNING

Clustering by fast search and find of density peaks

Knowledge Engineering Course

Nov.26.2014

Cluster

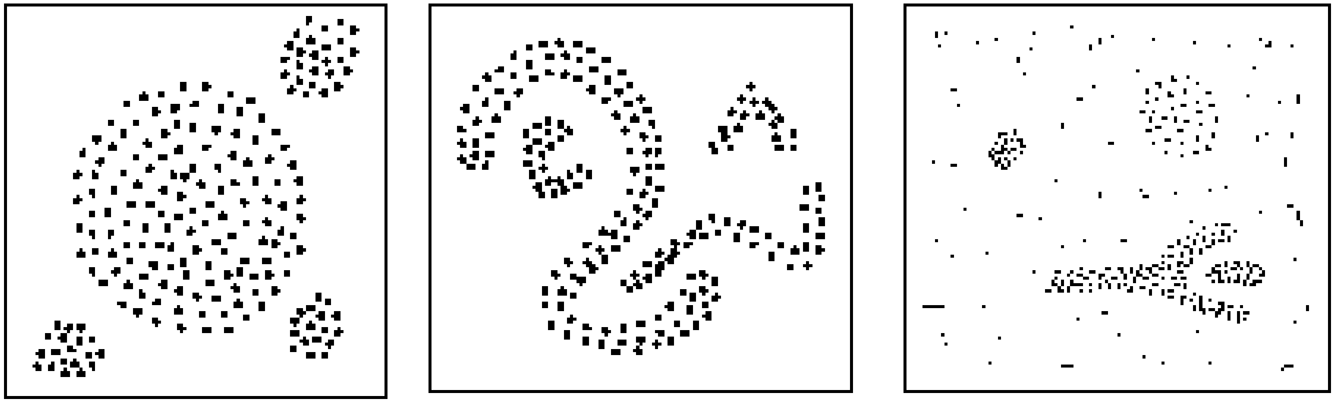


1) Feature selection or extraction.
2) Clustering algorithm design or selection.
3) Cluster validation.
4) Results interpretation.

Clustering Algorithms

A. *Distance and Similarity Measures*
B. Hierarchical -Agglomerative -Divisive
C. *Squared Error-Based*
D. Graph Theory-Based
E. Combinatorial Search Techniques-Based
F. Fuzzy
G. Neural Networks-Based
H. Kernel-Based
I. Sequential Data
J. Large-Scale Data Sets
K. Data visualization and High-dimensional Data
L. How Many Clusters?

Density-Based Clustering



- Each cluster has a considerable higher density of points than outside of the cluster.
- The density within the areas of noise is lower than the density in any of the clusters.
- Two global parameters*:*
  - *Eps*: Maximum radius of the neighbourhood
  - *MinPts*: Minimum number of points in an Eps-neighbourhood of that point

Density-Based Clustering: Background

- Eps-neighborhood of a point:

  The Eps-neighborhood of a point p, denoted by $N_{Eps}(P)$,is defined by

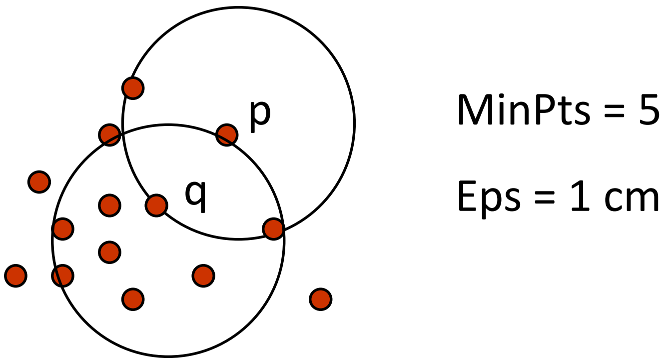
  $$N_{Eps}(P)=\{q \in D \mid dist(p,q) \leq Eps\}$$

- Directly density-reachable:

  A point ***p*** is directly density-reachable from a point ***q*** wrt. ***Eps***, ***MinPts*** if

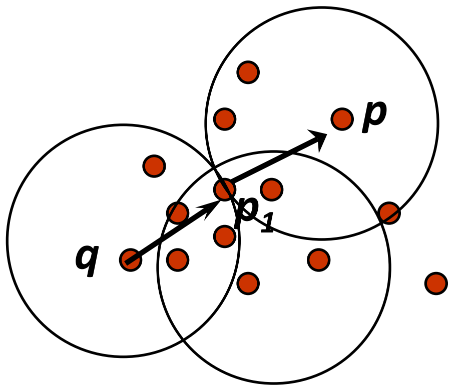
  - 1) $\boldsymbol{p \in N_{Eps}(q)}$
  - 2) $\boldsymbol{|N_{Eps}(q)| \geq MinPts}$

    (core point condition)

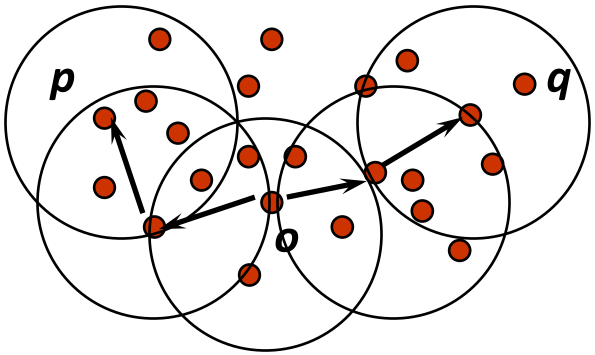


Density-Based Clustering: Background

- Density-reachable:
  - A point p is density-reachable from a point q wrt. *Eps*, *MinPts* if there is a chain of points $p_1, \ldots, p_n$, $p_1 = q$, $p_n = p$ such that p_{i+1} is directly density-reachable from p_i



- Density-connected
  - A point p is density-connected to a point q wrt. *Eps*, *MinPts* if there is a point o such that both, p and q are density-reachable from o wrt. *Eps* and *MinPts*.



Density-Based Clustering: Background

- Cluster:

  Let D be a database of points. A cluster C wrt. Eps and Minpts is a non-empty of D satisfying the following conditions:

  —1)$\forall$p,q:if p$\in$C and q is density-reachable from p wrt.Eps and Minpts, then q C.

  —2)$\forall$p,q$\in$C: p is density-connected to q wrt.Eps and MinPts.

- Noise :

  Let C1, …,Ck be the cluster of the database D wrt. Parameters Epsi and MinPtsi, i=1, …,k. Then we define the noise as the set of points in the database D not belonging to any cluster Ci, i.e. noise={p$\in$D|$\forall$i:p$\notin$Ci}

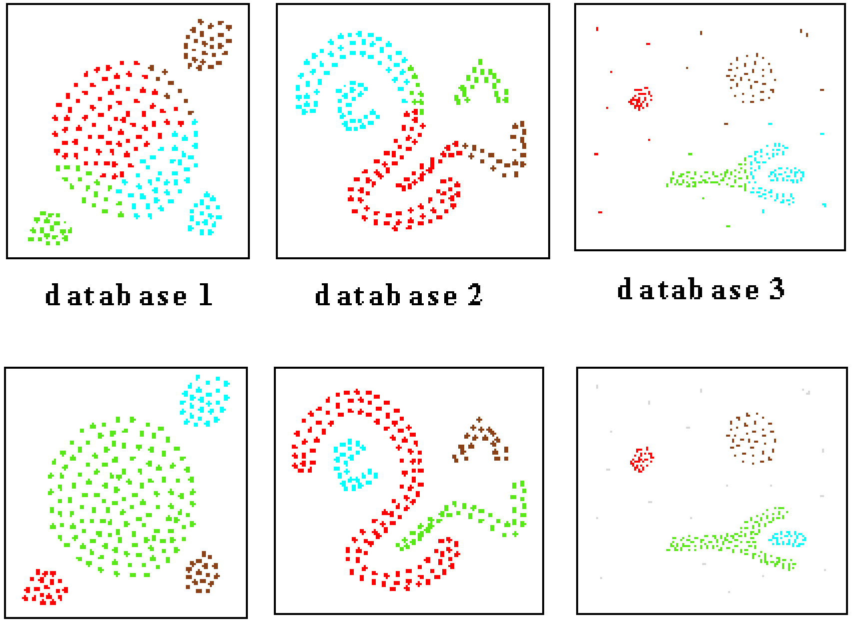
DBSCAN: The Algorithm

- Arbitrary select a point ***p***
- Retrieve all points density-reachable from ***p*** wrt ***Eps*** and ***MinPts***.
- If ***p*** is a core point, a cluster is formed.
- If ***p*** is a border point, no points are density-reachable from ***p*** and DBSCAN visits the next point of the database.
- Continue the process until all of the points have been processed.

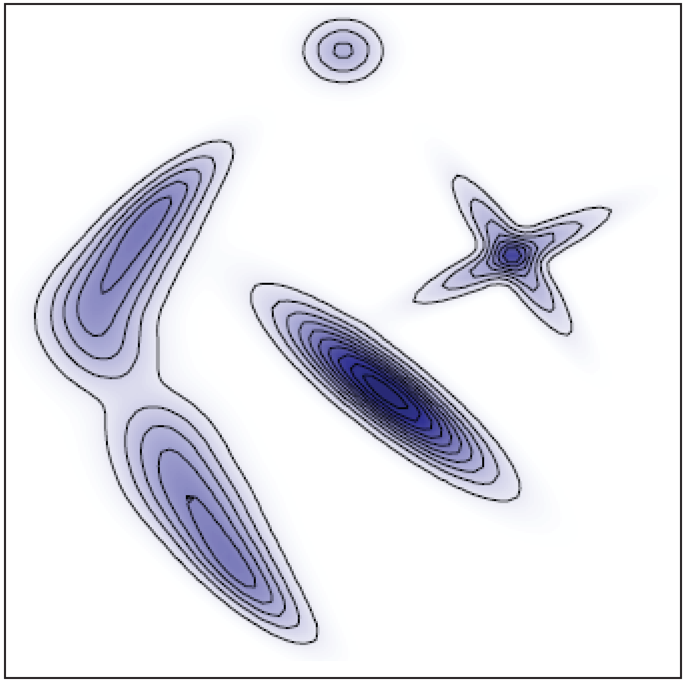
Performance Evaluation

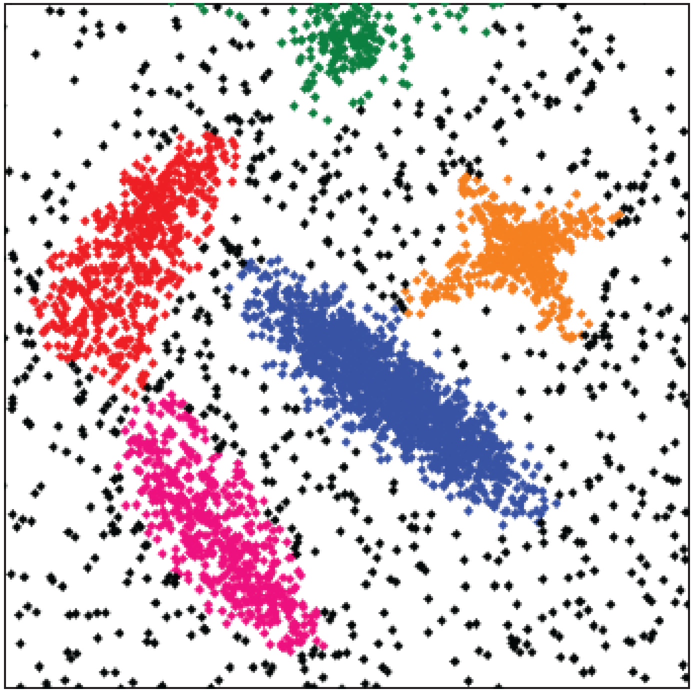
CLARANS:

DBSCAN:



The algorithm





The algorithm

Assumptions

Cluster centers are surrounded by neighbors with lower local density and that they are at a relatively large distance from any points with a higher local density.

Quantities

for each data point i

- Its local density ρ_i
- Its distance δ_i from points of higher density.

Both these quantities depend only on the distances d_{ij} between data points, which are assumed to satisfy the triangular inequality.

local density ρ_i

The local density ρ_i of data point i is defined as

$$\rho_i = \sum_j \chi(d_{ij} - d_c)$$

where $\chi(x)=1$ if $x<0$ and $\chi(x)= 0$ otherwise, and d_c is a cutoff distance. Basically, ρ_i is equal to the number of points that are closer than d_c to point i. The algorithm is sensitive only to the relative magnitude of ρ_i in different points, implying that, for large data sets, the results of the analysis are robust with respect to the choice of $\boxed{d_c}$.

distance δ_i

δ_i is measured by computing the minimum distance between the point i and any other point with higher density:

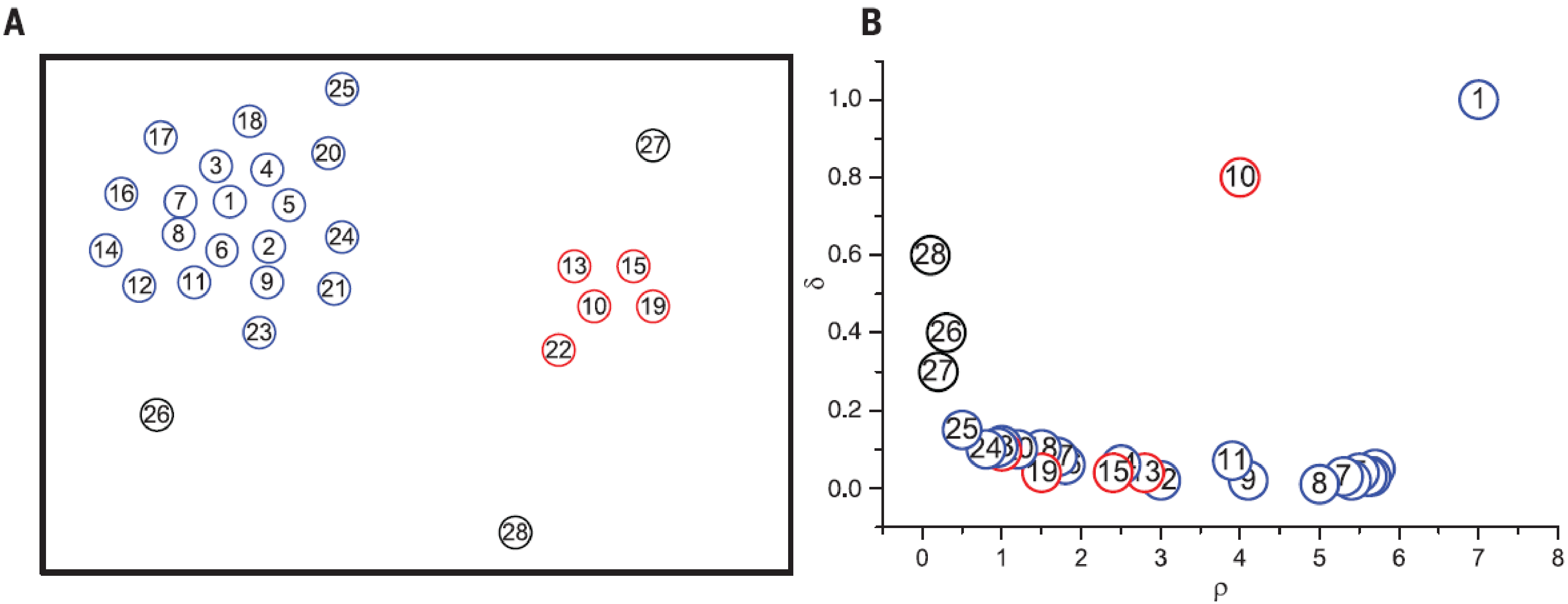
$$\delta_i = \min_{j:\rho_j > \rho_i} (d_{ij})$$

For the point with highest density, we conventionally take $\delta_i = \max_j(dij)$.

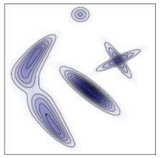
cluster centers

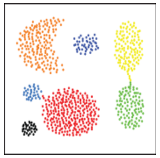
- Note that di is much larger than the typical nearest neighbor distance only for points that are local or global maxima in the density. Thus, cluster centers arerecognized as points for which the value of di is anomalously large.

the algorithm

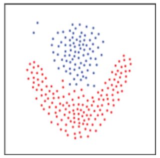


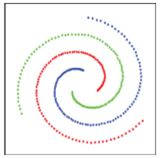
Experiment


Test case


Clustering Aggregation

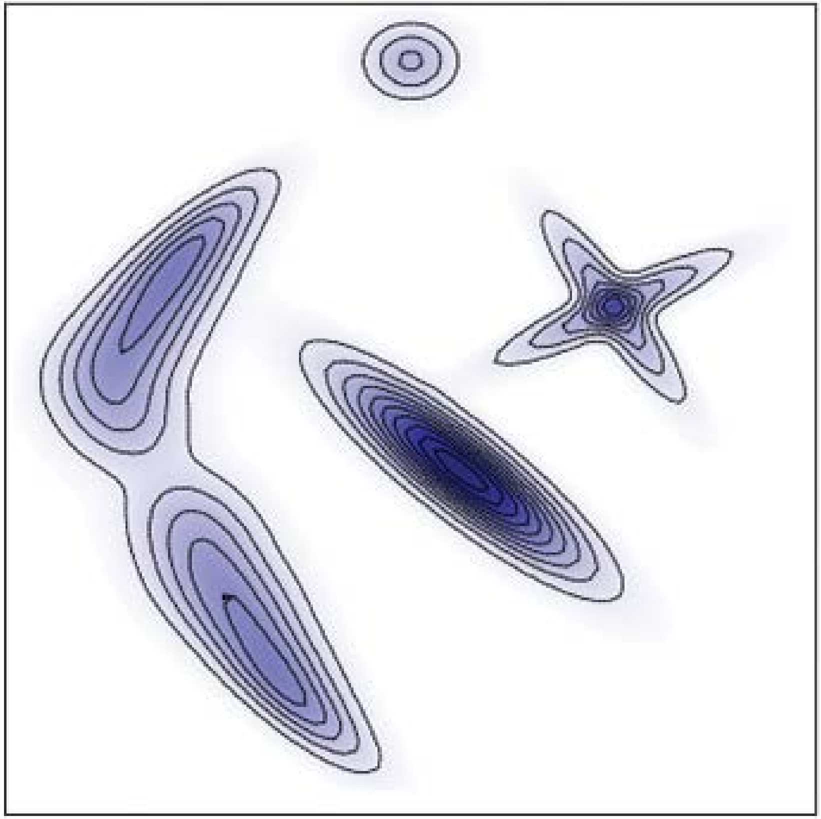

Iterative shrinking method for clustering problems


FLAME


Robust path-based spectral clustering

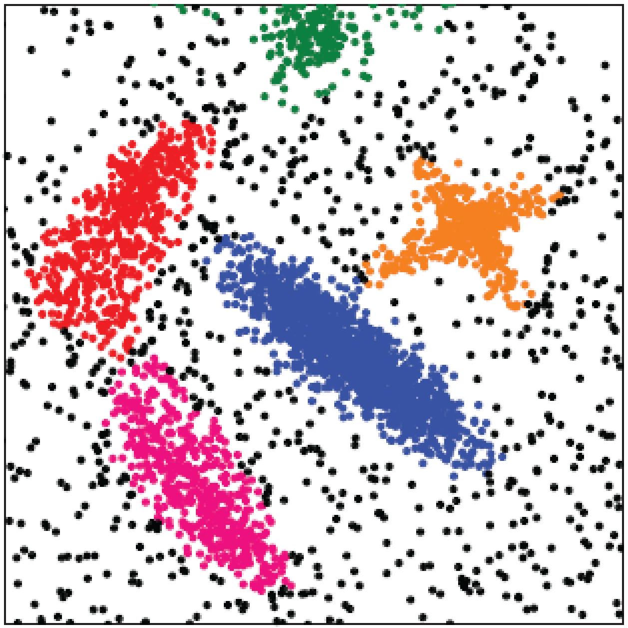

Olivetti Face Database

Test case



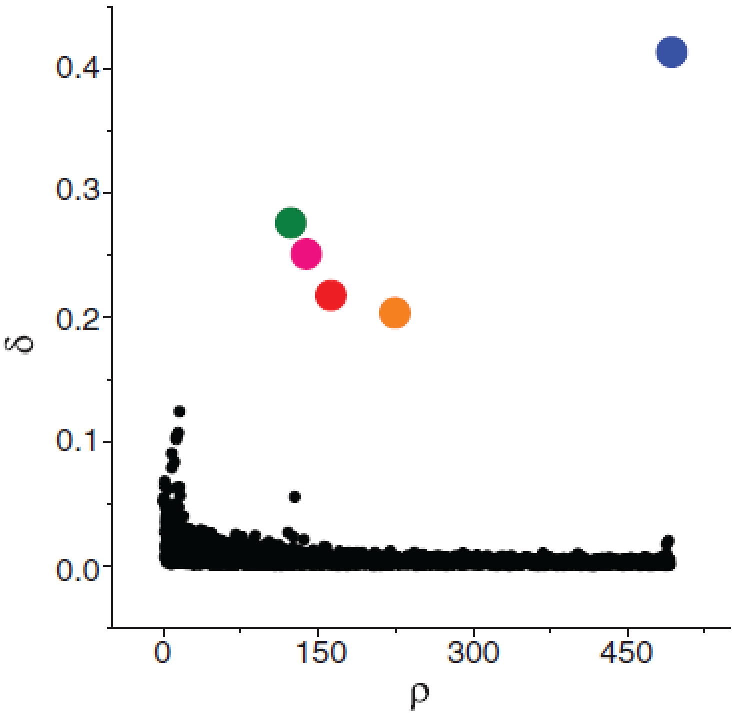
The probability distribution from which point distributions are drawn.

Test case

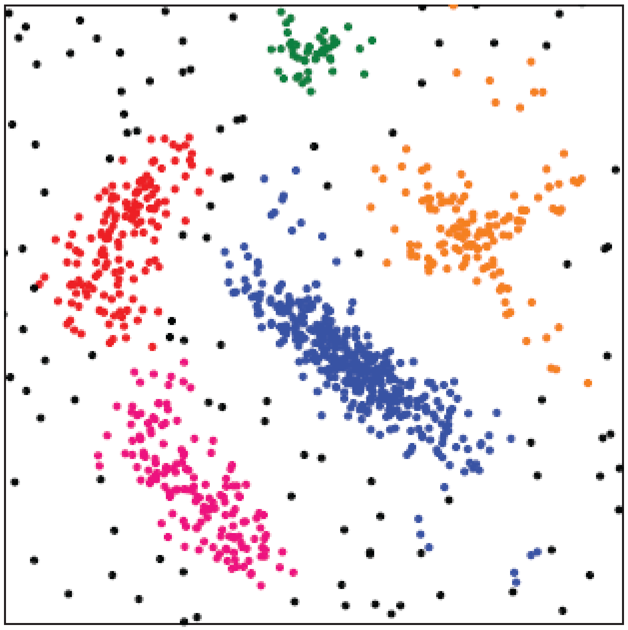


Point distributions for samples of 4000.
Points are colored according to the cluster to which they are assigned. Black points belong to the cluster halos.

The corresponding decision graphs, with the centers colored by cluster.

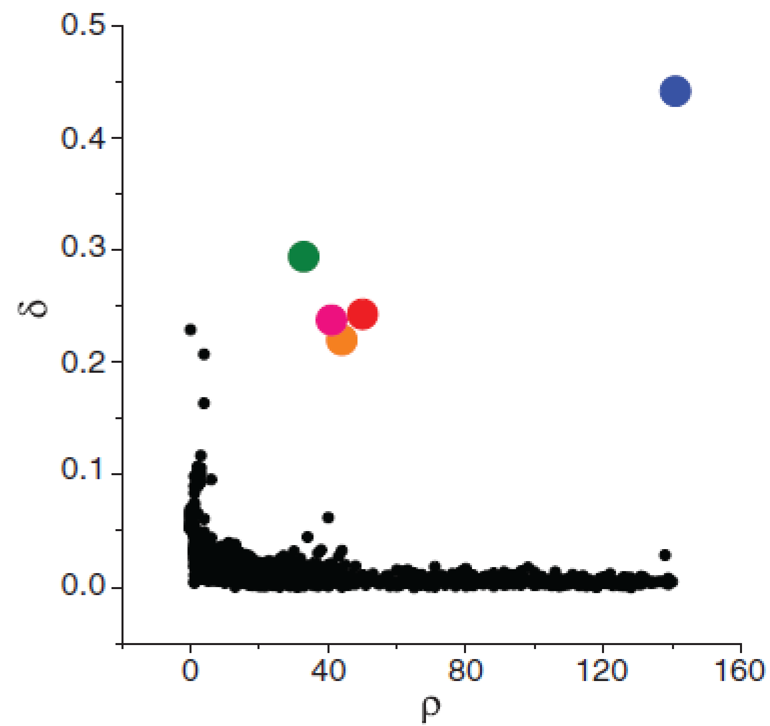


Test case

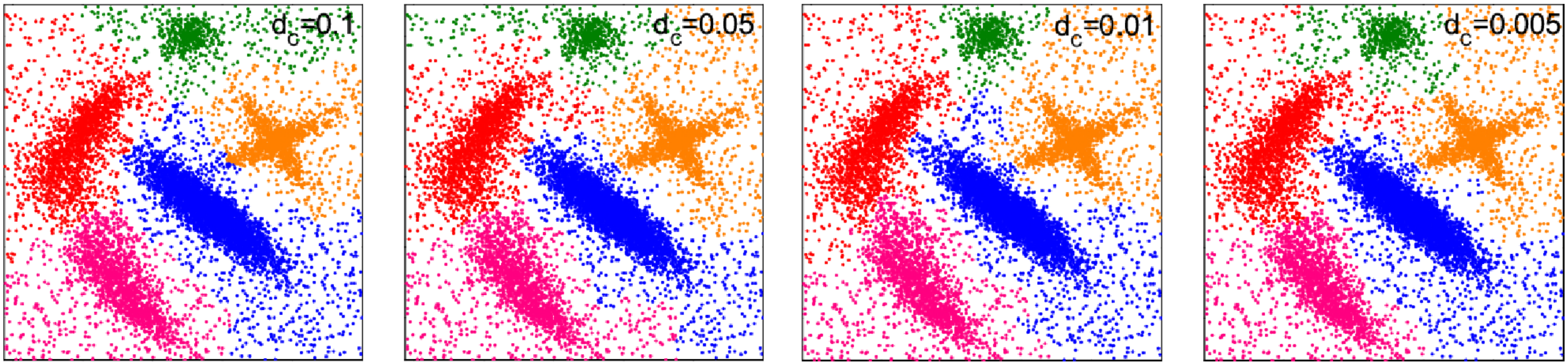


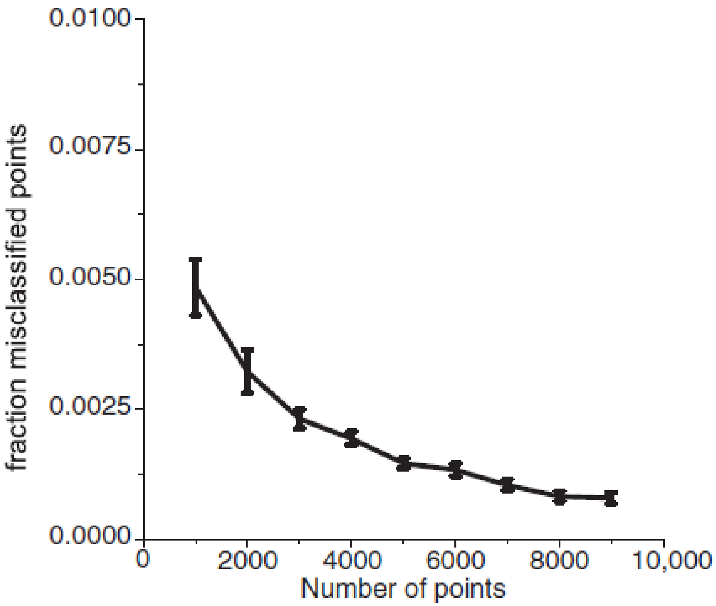
Point distributions for samples of 1000 points
Points are colored according to the cluster to which they are assigned. Black points belong to the cluster halos.

The corresponding decision graphs, with the centers colored by cluster.



d_c and n

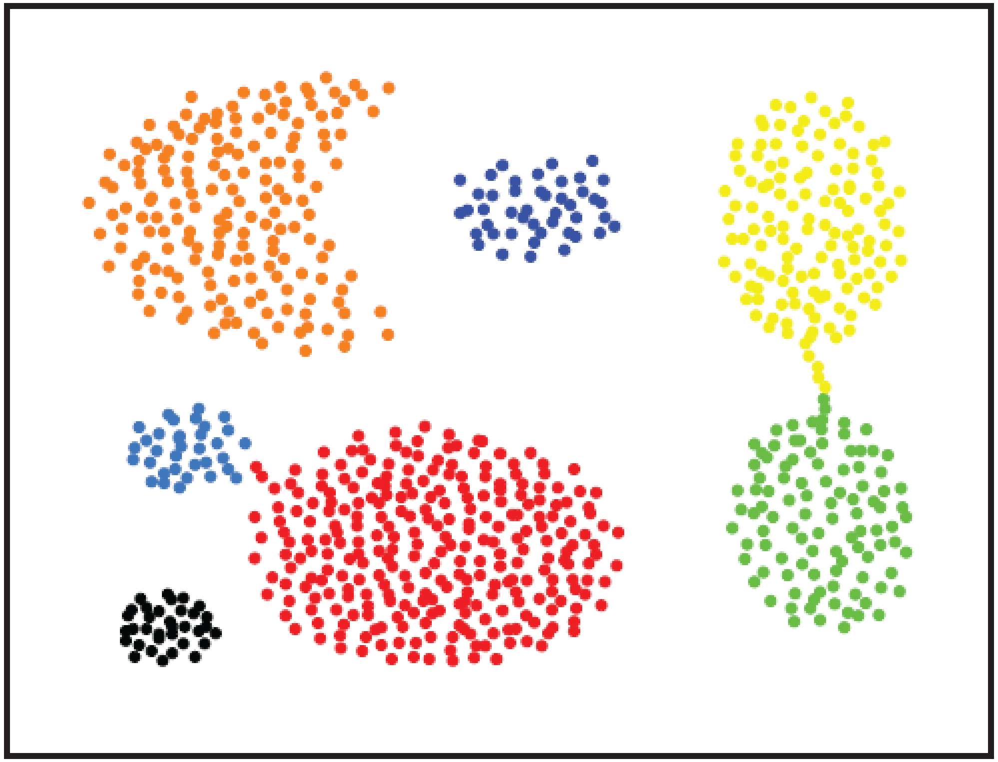




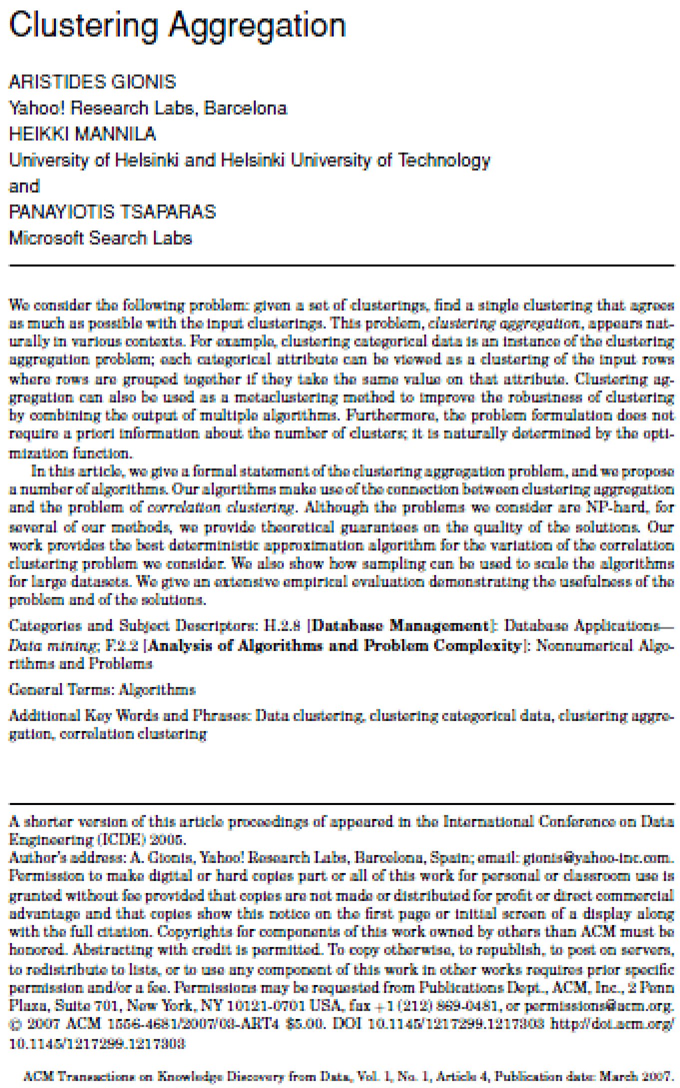
Comparison of the assignation for several values of d_c. Although the average number of neighbours varies between 11 % and 0.2 %, the assignations are very similar.

As a function of the size of the reduced sample, the fraction of points assigned to a cluster different than the one they were assigned to in the reference case.

Clustering Aggregation



A. Gionis, H. Mannila, P. Tsaparas, Clustering aggregation. *ACM Trans. Knowl. Discovery Data* ***1, 4, es (2007).***



Clustering Aggregation

ARISTIDES GIONIS
Yahoo! Research Labs, Barcelona
HEIKKI MANNILA
University of Helsinki and Helsinki University of Technology
and
PANAYIOTIS TSAPARAS
Microsoft Search Labs

We consider the following problem: given a set of clusterings, find a single clustering that agrees as much as possible with the input clusterings. This problem, *clustering aggregation*, appears naturally in various contexts. For example, clustering categorical data is an instance of the clustering aggregation problem; each categorical attribute can be viewed as a clustering of the input rows where rows are grouped together if they take the same value on that attribute. Clustering aggregation can also be used as a metaclustering method to improve the robustness of clustering by combining the output of multiple algorithms. Furthermore, the problem formulation does not require a priori information about the number of clusters; it is naturally determined by the optimization function.

In this article, we give a formal statement of the clustering aggregation problem, and we propose a number of algorithms. Our algorithms make use of the connection between clustering aggregation and the problem of *correlation clustering*. Although the problems we consider are NP-hard, for several of our methods, we provide theoretical guarantees on the quality of the solutions. Our work provides the best deterministic approximation algorithm for the variation of the correlation clustering problem we consider. We also show how sampling can be used to scale the algorithms for large datasets. We give an extensive empirical evaluation demonstrating the usefulness of the problem and of the solutions.

Categories and Subject Descriptors: H.2.8 [**Database Management**]: Database Applications—*Data mining*; F.2.2 [**Analysis of Algorithms and Problem Complexity**]: Nonnumerical Algorithms and Problems

General Terms: Algorithms

Additional Key Words and Phrases: Data clustering, clustering categorical data, clustering aggregation, correlation clustering

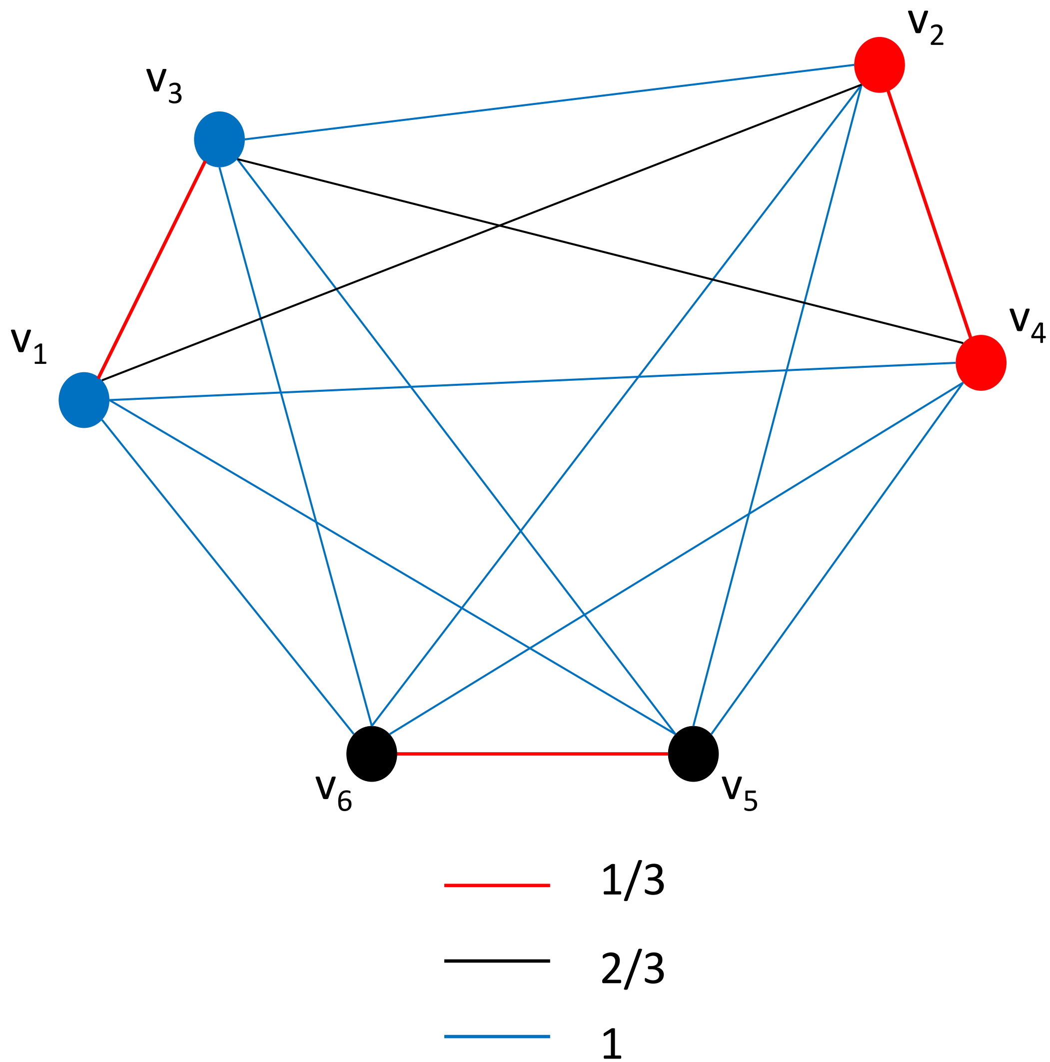
A shorter version of this article proceedings of appeared in the International Conference on Data Engineering (ICDE) 2005.
Author's address: A. Gionis, Yahoo! Research Labs, Barcelona, Spain; email: gionis@yahoo-inc.com.

Clustering Aggregation example

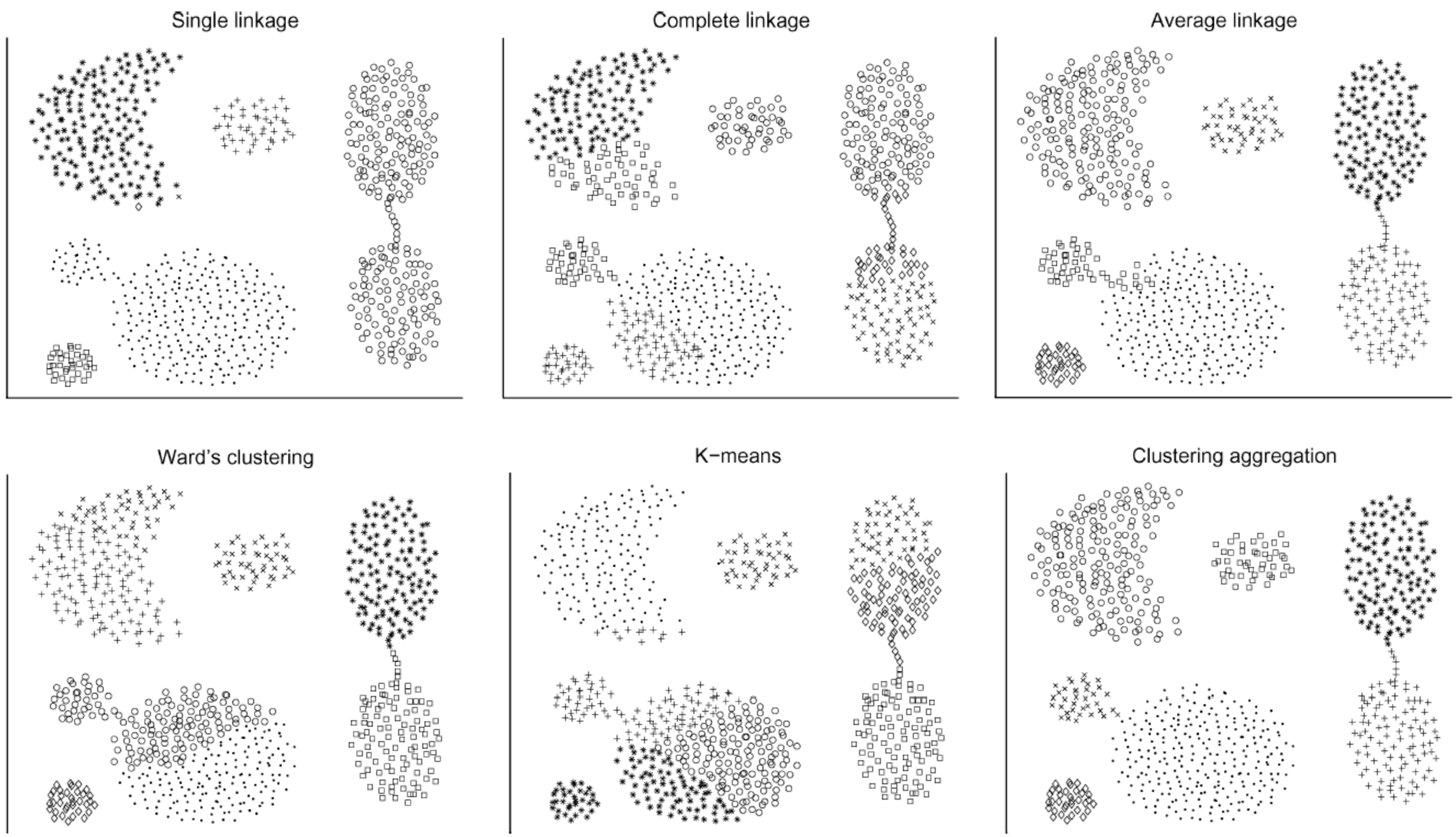
	$\mathcal{C}_1$	$\mathcal{C}_2$	$\mathcal{C}_3$	$\mathcal{C}$
v_1	1	1	1	1
v_2	1	2	2	2
v_3	2	1	1	1
v_4	2	2	2	2
v_5	3	3	3	3
v_6	3	4	3	3

v_1	1	1	1
v_2	1	2	2

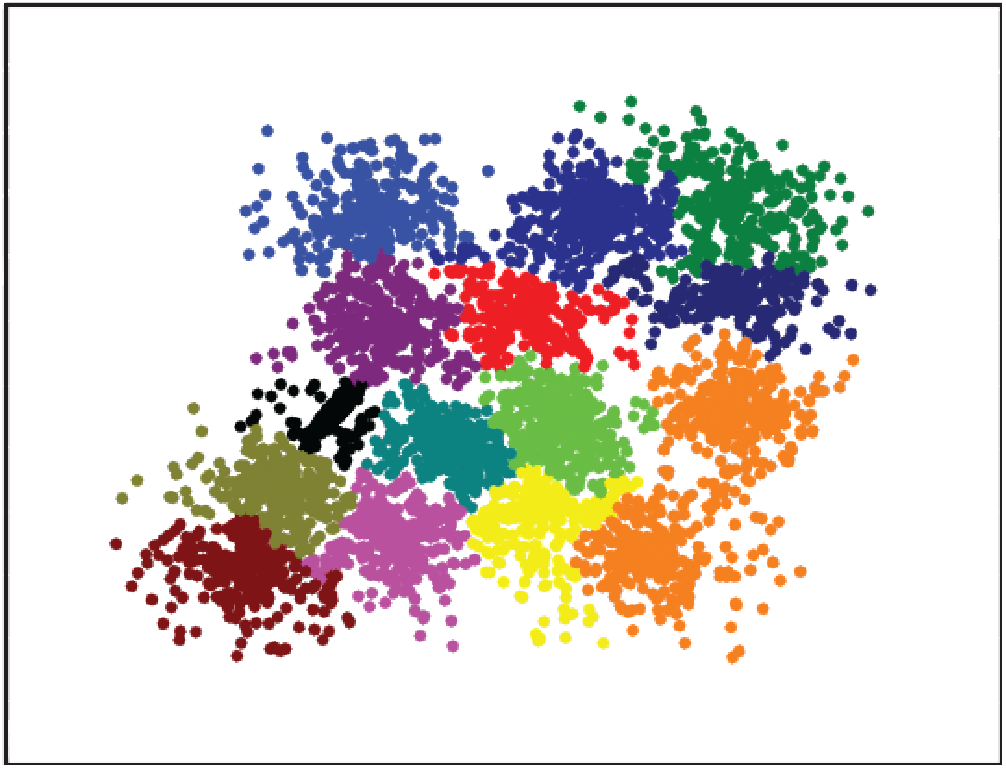
$$W_{12} = \frac{\boxed{\quad}}{\boxed{\quad}} = \frac{2}{3}$$

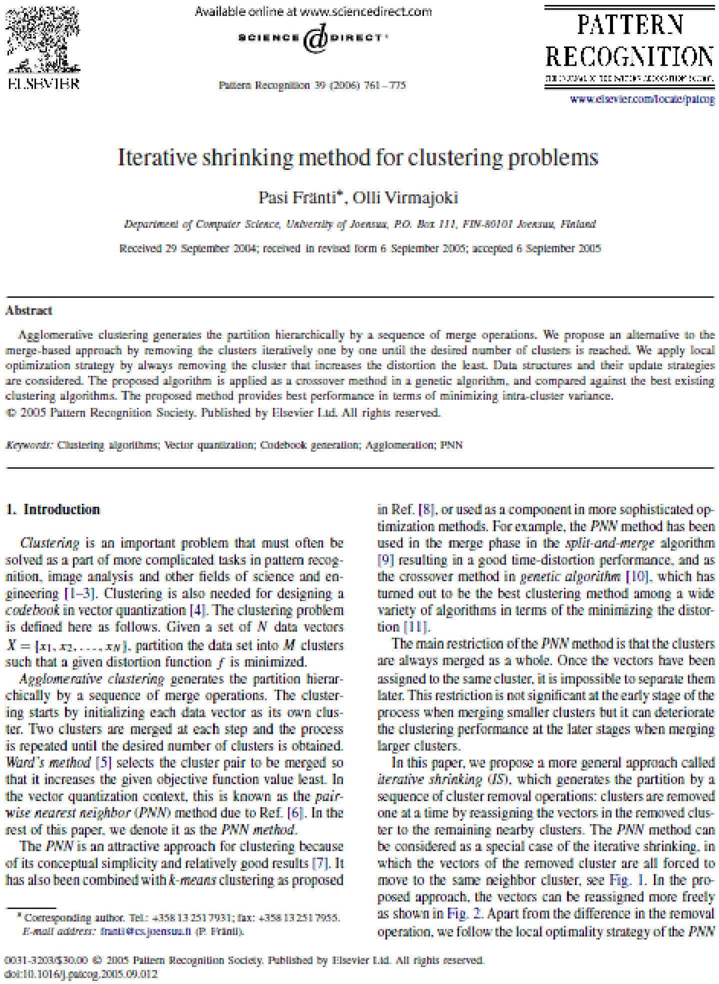


Clustering Aggregation example



Iterative shrinking method for clustering problems





Available online at www.sciencedirect.com

SCIENCE DIRECT

ELSEVIER

Pattern Recognition 39 (2006) 761 – 775

PATTERN RECOGNITION

www.elsevier.com/locate/patcog

Iterative shrinking method for clustering problems

Pasi Fränti*, Olli Virmajoki

Department of Computer Science, University of Joensuu, P.O. Box 111, FIN-80101 Joensuu, Finland

Received 29 September 2004; received in revised form 6 September 2005; accepted 6 September 2005

Abstract

Agglomerative clustering generates the partition hierarchically by a sequence of merge operations. We propose an alternative to the merge-based approach by removing the clusters iteratively one by one until the desired number of clusters is reached. We apply local optimization strategy by always removing the cluster that increases the distortion the least. Data structures and their update strategies are considered. The proposed algorithm is applied as a crossover method in a genetic algorithm, and compared against the best existing clustering algorithms. The proposed method provides best performance in terms of minimizing intra-cluster variance.

Keywords: Clustering algorithms; Vector quantization; Codebook generation; Agglomeration; PNN

1. Introduction

Clustering is an important problem that must often be solved as a part of more complicated tasks in pattern recognition, image analysis and other fields of science and engineering [1–3]. Clustering is also needed for designing a *codebook* in vector quantization [4]. The clustering problem is defined here as follows. Given a set of N data vectors $X = \{x_1, x_2, \ldots, x_N\}$, partition the data set into M clusters such that a given distortion function f is minimized.

Agglomerative clustering generates the partition hierarchically by a sequence of merge operations. The clustering starts by initializing each data vector as its own cluster. Two clusters are merged at each step and the process is repeated until the desired number of clusters is obtained. *Ward's method* [5] selects the cluster pair to be merged so that it increases the given objective function value least. In the vector quantization context, this is known as the *pairwise nearest neighbor* (*PNN*) method due to Ref. [6]. In the rest of this paper, we denote it as the *PNN method*.

The *PNN* is an attractive approach for clustering because of its conceptual simplicity and relatively good results [7]. It has also been combined with *k-means* clustering as proposed in Ref. [8], or used as a component in more sophisticated optimization methods. For example, the *PNN* method has been used in the merge phase in the *split-and-merge* algorithm [9] resulting in a good time-distortion performance, and as the crossover method in *genetic algorithm* [10], which has turned out to be the best clustering method among a wide variety of algorithms in terms of the minimizing the distortion [11].

The main restriction of the *PNN* method is that the clusters are always merged as a whole. Once the vectors have been assigned to the same cluster, it is impossible to separate them later. This restriction is not significant at the early stage of the process when merging smaller clusters but it can deteriorate the clustering performance at the later stages when merging larger clusters.

In this paper, we propose a more general approach called *iterative shrinking (IS)*, which generates the partition by a sequence of cluster removal operations: clusters are removed one at a time by reassigning the vectors in the removed cluster to the remaining nearby clusters. The *PNN* method can be considered as a special case of the iterative shrinking, in which the vectors of the removed cluster are all forced to move to the same neighbor cluster, see Fig. 1. In the proposed approach, the vectors can be reassigned more freely as shown in Fig. 2. Apart from the difference in the removal operation, we follow the local optimality strategy of the *PNN*

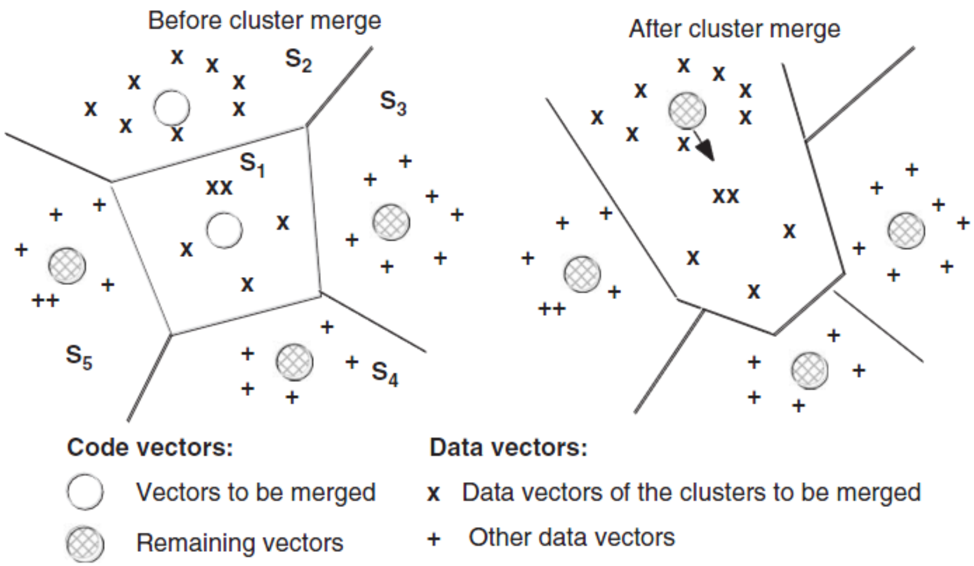
* Corresponding author. Tel.: +358 13 251 7931; fax: +358 13 251 7955.
E-mail address: franti@cs.joensuu.fi (P. Fränti).

0031-3203/$30.00 © 2005 Pattern Recognition Society. Published by Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2005.09.012

P. Fränti, O. Virmajoki, Iterative shrinking method for clustering problems. *Pattern Recognit.* ***39, 761–775 (2006).***

Iterative shrinking method for clustering problems

Pairwise nearest neighbor



PNN(X, M) $\rightarrow$ S

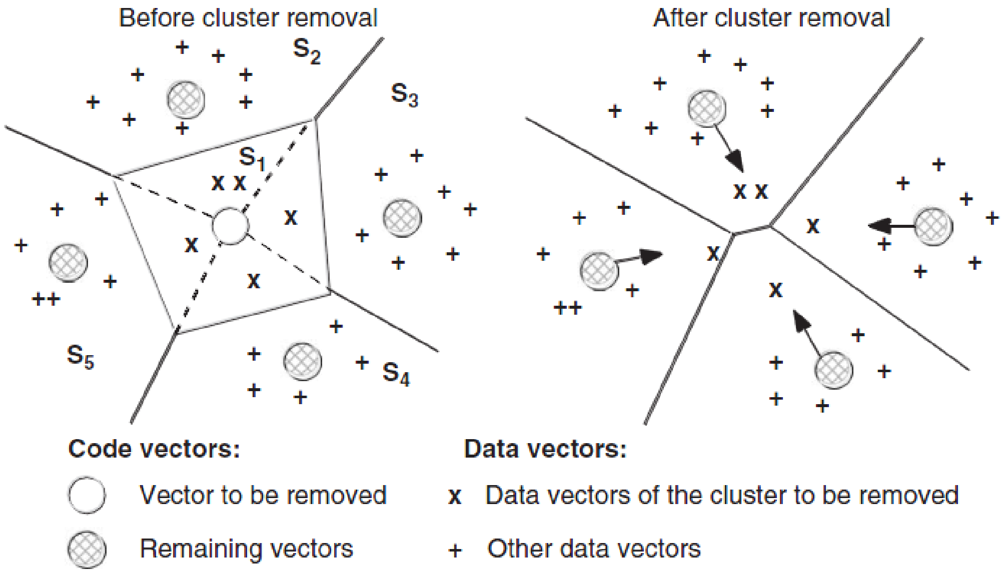
FOR $i \leftarrow 1$ to N DO
    $s_i \leftarrow \{x_i\}$;
REPEAT
    $(s_a, s_b) \leftarrow$ SearchNearestClusters(S);
    Merge(s_a, s_b);
UNTIL $|S|=M$;

$$s_a \leftarrow s_a \cup s_b.$$

$$d_{a,b} = \frac{n_a n_b}{n_a + n_b} . \|c_a - c_b\|^2$$

Iterative shrinking method for clustering problems

Iterative shrinking



IS$(X, M) \rightarrow S$

FOR $i \leftarrow 1$ to N DO

    $s_i \leftarrow \{x_i\}$;

REPEAT

    $s_a \leftarrow$ SelectClusterToBeRemoved(S);

    RemoveCluster(S, s_a);

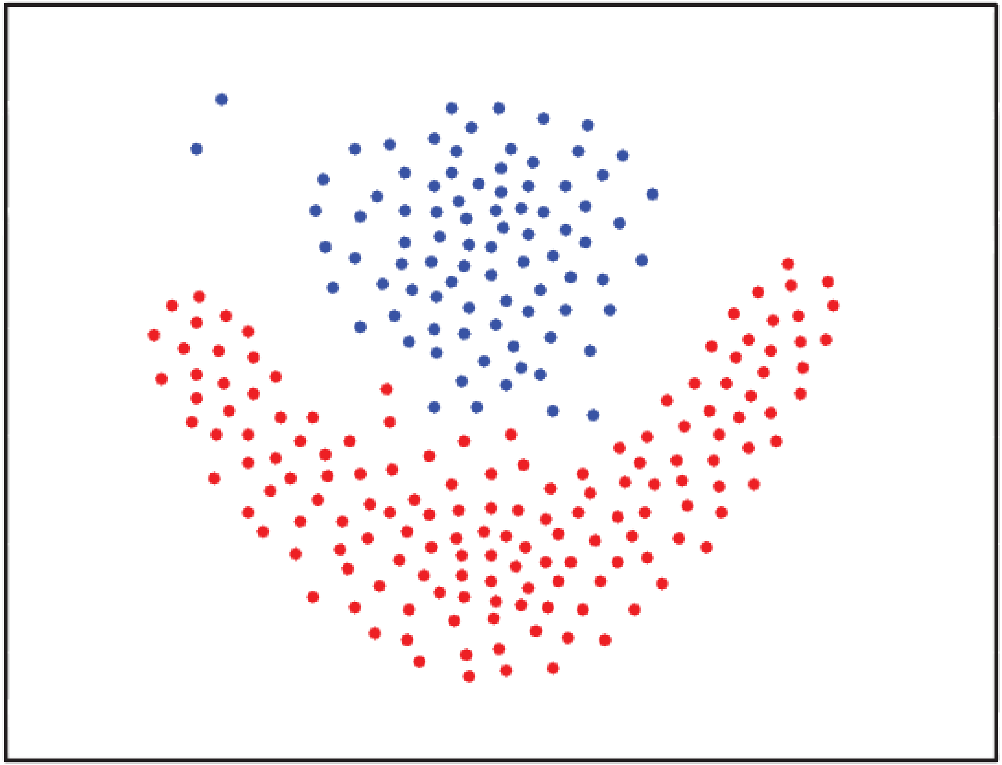
UNTIL $|S|=M$;

$$q_i = \arg \min_{\substack{1 \leqslant j \leqslant m \\ j \neq p_i}} \frac{n_j}{n_j + 1} \|x_i - c_j\|^2$$

$$\Delta D_i = \frac{n_{q_i}}{n_{q_i} + 1} \|x_i - c_{q_i}\|^2 - \|x_i - c_a\|^2$$

$$d_a = \sum_{x_i \in s_a} \Delta D_i$$

Fuzzy clustering by Local Approximation of Membership



L. Fu, E. Medico, FLAME, a novel fuzzy clustering method for the analysis of DNA microarray data. *BMC Bioinformatics* ***8, 3 (2007).***

BMC Bioinformatics

BioMed Central

Methodology article

Open Access

FLAME, a novel fuzzy clustering method for the analysis of DNA microarray data

Limin Fu and Enzo Medico*

Address: Laboratory of Functional Genomics, The Oncogenomics Center, Institute for Cancer Research and Treatment (IRCC), University of Torino, School of Medicine, 10060 Candiolo, Italy.

Email: Limin Fu - limin.fu@ircc.it; Enzo Medico* - enzo.medico@ircc.it

* Corresponding author

Published: 04 January 2007

BMC Bioinformatics 2007, 8:3 doi:10.1186/1471-2105-8-3

Received: 12 June 2006
Accepted: 04 January 2007

This article is available from: http://www.biomedcentral.com/1471-2105/8/3

Abstract

Background: Data clustering analysis has been extensively applied to extract information from gene expression profiles obtained with DNA microarrays. To this aim, existing clustering approaches, mainly developed in computer science, have been adapted to microarray data analysis. However, previous studies revealed that microarray datasets have very diverse structures, some of which may not be correctly captured by current clustering methods. We therefore approached the problem from a new starting point, and developed a clustering algorithm designed to capture dataset-specific structures at the beginning of the process.

Results: The clustering algorithm is named Fuzzy clustering by Local Approximation of MEmbership (FLAME). Distinctive elements of FLAME are: (i) definition of the neighborhood of each object (gene or sample) and identification of objects with "archetypal" features named Cluster Supporting Objects, around which to construct the clusters; (ii) assignment to each object of a fuzzy membership vector approximated from the memberships of its neighboring objects, by an iterative converging process in which membership spreads from the Cluster Supporting Objects through their neighbors. Comparative analysis with K-means, hierarchical, fuzzy C-means and fuzzy self-organizing maps (SOM) showed that data partitions generated by FLAME are not superimposable to those of other methods and, although different types of datasets are better partitioned by different algorithms, FLAME displays the best overall performance. FLAME is implemented, together with all the above-mentioned algorithms, in a C++ software with graphical interface for Linux and Windows, capable of handling very large datasets, named Gene Expression Data Analysis Studio (GEDAS), freely available under GNU General Public License.

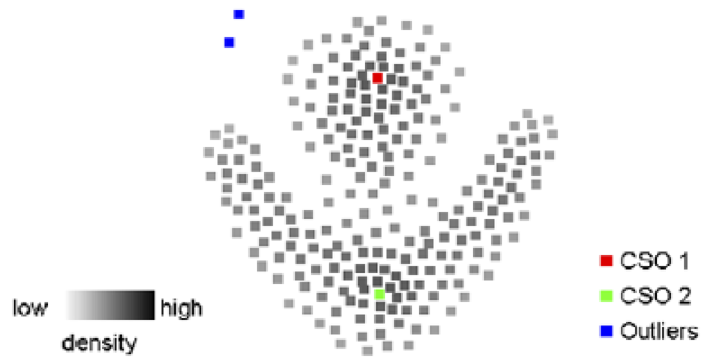
Conclusion: The FLAME algorithm has intrinsic advantages, such as the ability to capture non-linear relationships and non-globular clusters, the automated definition of the number of clusters, and the identification of cluster outliers, i.e. genes that are not assigned to any cluster. As a result, clusters are more internally homogeneous and more diverse from each other, and provide better partitioning of biological functions. The clustering algorithm can be easily extended to applications different from gene expression analysis.

Fuzzy clustering by Local Approximation of Membership

Starting data

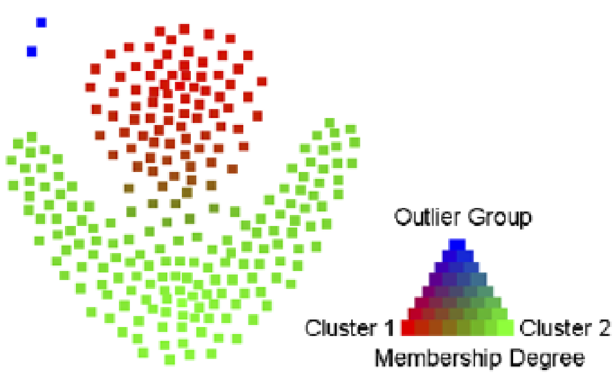
Step One: for each object,

1. Find the k-nearest neighbors and calculate their proximity
2. Use the proximity measurements to calculate object density
3. Use the density to define the object type (CSO/outlier/else)



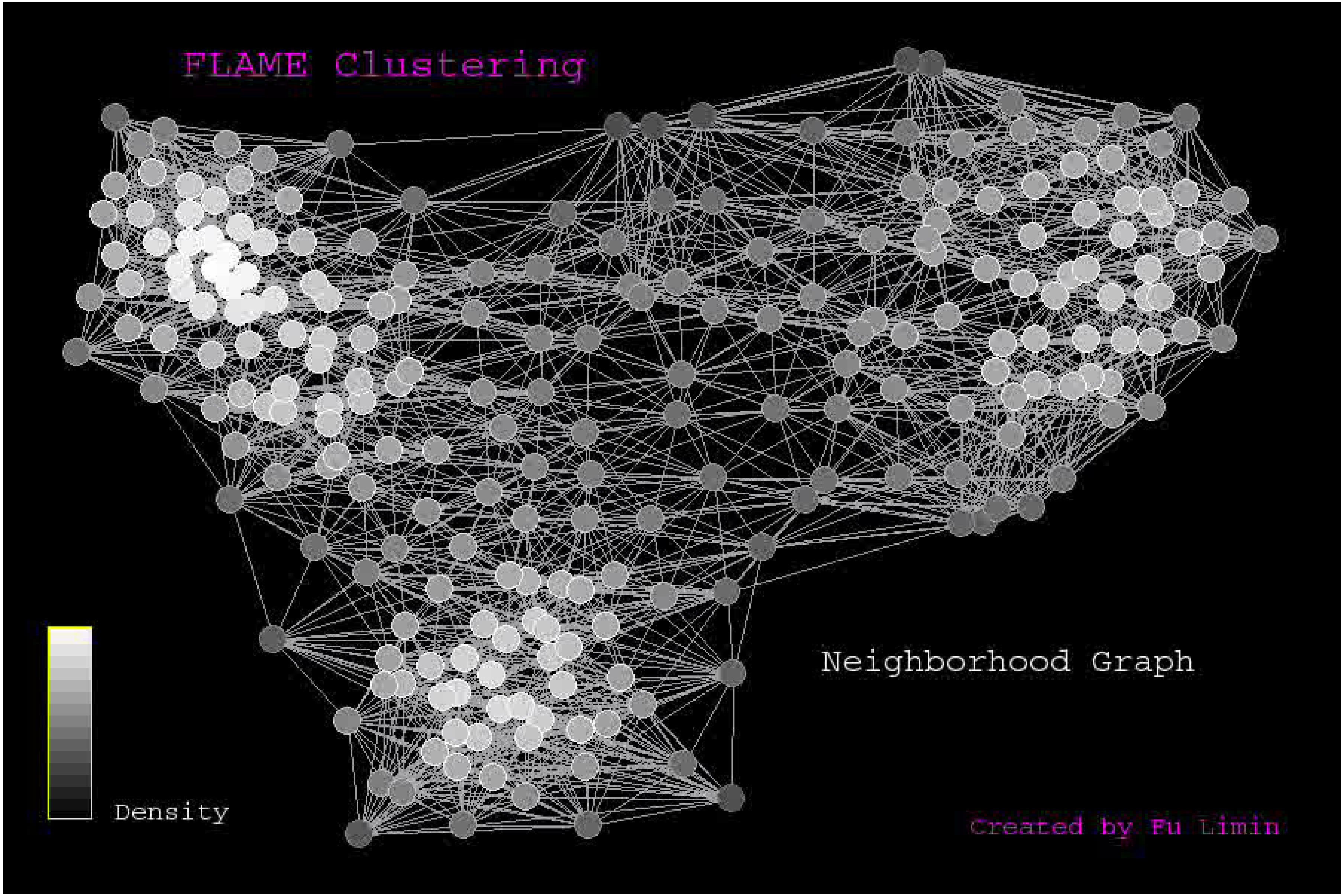
Step Two:

1. Assign initial memberships
2. Local Approximation of Fuzzy Memberships

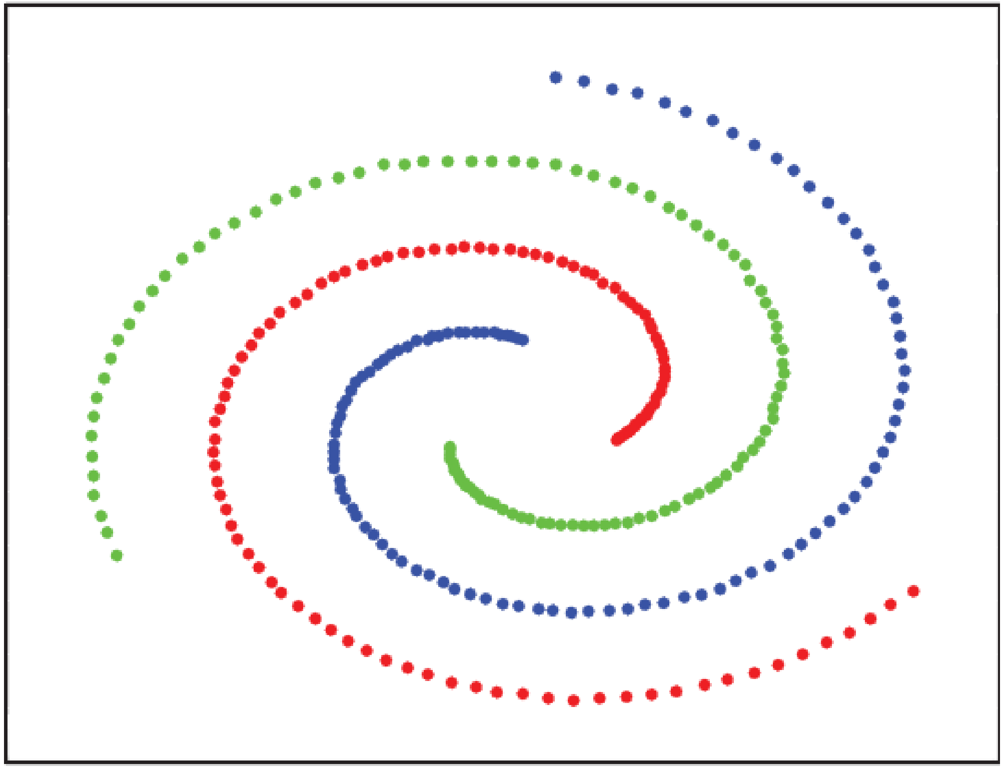


Step Three:
Construct clusters from fuzzy memberships

Fuzzy clustering by Local Approximation of Membership



Robust path-based spectral clustering



H. Chang, D.-Y. Yeung, Robust path-based spectral clustering. *Pattern Recognit.* ***41, 191–203 (2008).***

Available online at www.sciencedirect.com

ScienceDirect

Pattern Recognition 41 (2008) 191–203

PATTERN RECOGNITION

THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/pr

Robust path-based spectral clustering

Hong Chang[a], Dit-Yan Yeung[b,*]

[a] Xerox Research Centre Europe, 6 chemin de Maupertuis, 38240 Meylan, France

[b] Department of Computer Science and Engineering, Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong

Received 3 March 2007; accepted 23 April 2007

Abstract

Spectral clustering and path-based clustering are two recently developed clustering approaches that have delivered impressive results in a number of challenging clustering tasks. However, they are not robust enough against noise and outliers in the data. In this paper, based on M-estimation from robust statistics, we develop a robust path-based spectral clustering method by defining a robust path-based similarity measure for spectral clustering under both unsupervised and semi-supervised settings. Our proposed method is significantly more robust than spectral clustering and path-based clustering. We have performed experiments based on both synthetic and real-world data, comparing our method with some other methods. In particular, color images from the Berkeley segmentation data set and benchmark are used in the image segmentation experiments. Experimental results show that our method consistently outperforms other methods due to its higher robustness.

Keywords: Path-based clustering; Spectral clustering; Robust statistics; Unsupervised learning; Semi-supervised learning; Image segmentation

1. Introduction

Clustering has been among the most active research topics in machine learning and pattern recognition. While many traditional clustering algorithms have been developed over the past few decades [1,2], some new clustering algorithms emerged over the last few years give very promising results on some challenging tasks. Among them are *spectral clustering* [3–6] and *path-based clustering* [7–9], which have demonstrated excellent performance on some clustering tasks involving highly non-linear and elongated clusters in addition to compact clusters.

Despite the promising performance of these algorithms demonstrated on some difficult data sets, there exist some other situations when these algorithms do not perform well. Consider some examples in Fig. 1. Although spectral clustering works perfectly well on the 2-circle data set (Fig. 1(a)), it gives very poor result on the 3-spiral data set (Fig. 1(b)). The poor clustering result is due mainly to the particular choice of the affinity matrix, which is usually defined in a way similar to the Gaussian kernel based on inter-point Euclidean distance in the input space. However, if path-based criteria from path-based clustering are used to define the (dis)similarity between points to form the affinity matrix before spectral clustering is applied, the three clusters in the 3-spiral data set can be found correctly, as shown in Fig. 1(c).

While the combined use of path-based clustering and spectral clustering, referred to as *path-based spectral clustering* here, seems to be very effective, we will show later in the paper that this combined method, like the separate use of spectral clustering or path-based clustering, is not robust enough against noise and outliers which commonly exist in real-world data.

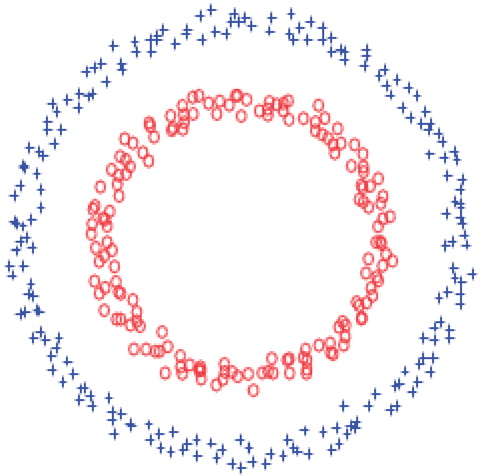
In this paper, based on robust statistical techniques [10], we propose a novel scheme to make path-based (spectral) clustering more robust. Our work is built upon the recent work of Fischer et al. [7–9]. We devise an M-estimator and use it to define a robust path-based similarity measure which takes into account the existence of noise and outliers in the data and hence brings about robustness in the method.

The rest of this paper is organized as follows. Some related work is briefly reviewed in Section 1.1. In Section 2, we propose a robust path-based similarity measure based on robust statistics, with which a robust path-based spectral clustering

* Corresponding author. Tel.: +852 2358 6977; fax: +852 2358 1477.
E-mail address: dyyeung@cse.ust.hk (D.-Y. Yeung).

0031-3203/$30.00 © 2007 Pattern Recognition Society. Published by Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2007.04.010

Robust path-based spectral clustering



Path-based dissimilarity measure

$$D_{ij}^{\text{eff}}(\mathbf{M},\mathbf{D}) = \min_{\mathbf{p}\in\mathcal{P}_{ij}(\mathbf{M})}\left\{\max_{h\in\{1,\ldots,|\mathbf{p}|-1\}}\{D_{\mathbf{p}[h]\mathbf{p}[h+1]}\}\right\}, \text{ where}$$

$$\mathcal{P}_{ij}(\mathbf{M}) = \left\{\mathbf{p}\in\{1,\ldots,n\}^l \,\middle|\, \exists\nu : \prod_{h=1}^{l} M_{p[h]\nu} = 1 \wedge l \le n \wedge p[1] = i \wedge p[l] = j\right\}$$

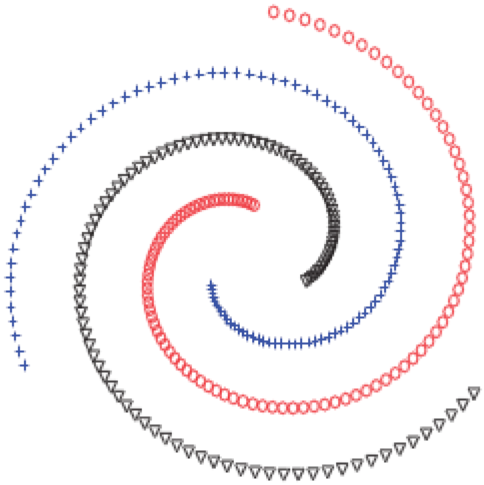
is the set of all paths from o_i to o_j through cluster ν if o_i and o_j belong to cluster ν. If both objects belong to different clusters $\mathcal{P}_{ij}(\mathbf{M})$ is the empty set and the effective dissimilarity is not defined.

Path-based similarity measure

$$s'_{ij} = \begin{cases} \exp\left(-\dfrac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}\right) & \text{for } i \neq j, \\ 0 & \text{for } i = j, \end{cases}$$

$$s_{ij} = \max_{p\in\mathscr{P}_{ij}}\left\{\min_{1\leqslant h<|p|} s'_{p[h]p[h+1]}\right\},$$

where $p[h]$ denotes the hth vertex along the path from vertex i to vertex j.



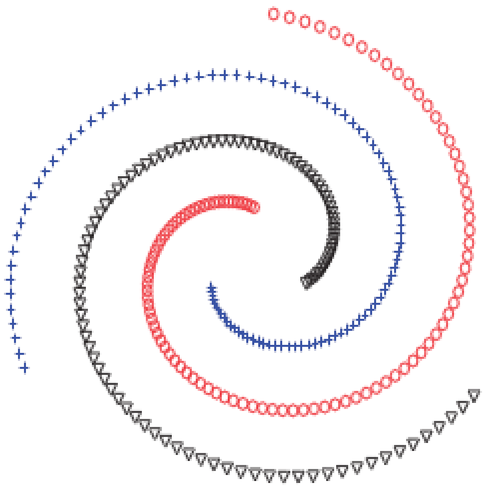
Robust path-based spectral clustering

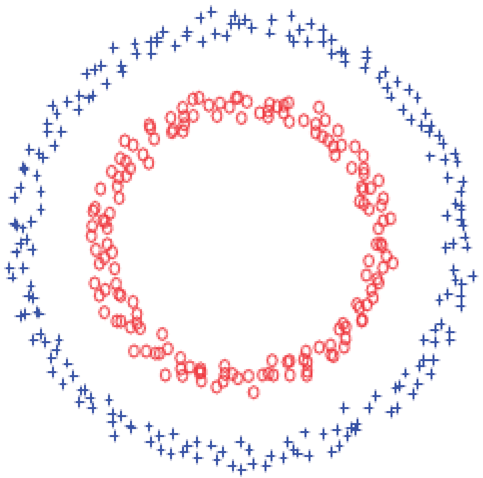
Robust path-based similarity measure

$$w_i' = \sum_{\mathbf{x}_j \in \mathcal{N}_i} a_{ij} = \sum_{\mathbf{x}_j \in \mathcal{N}_i} \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}\right) = \sum_{\mathbf{x}_j \in \mathcal{N}_i} s_{ij}'.$$

$$w_i = w'_i / \max_{xi \in \aleph} w'_i$$

$$s_{ij} = \max_{p \in \mathscr{P}_{ij}} \left\{ \min_{1 \leqslant h < |p|} w_{p[h]} w_{p[h+1]} s'_{p[h]p[h+1]} \right\}$$

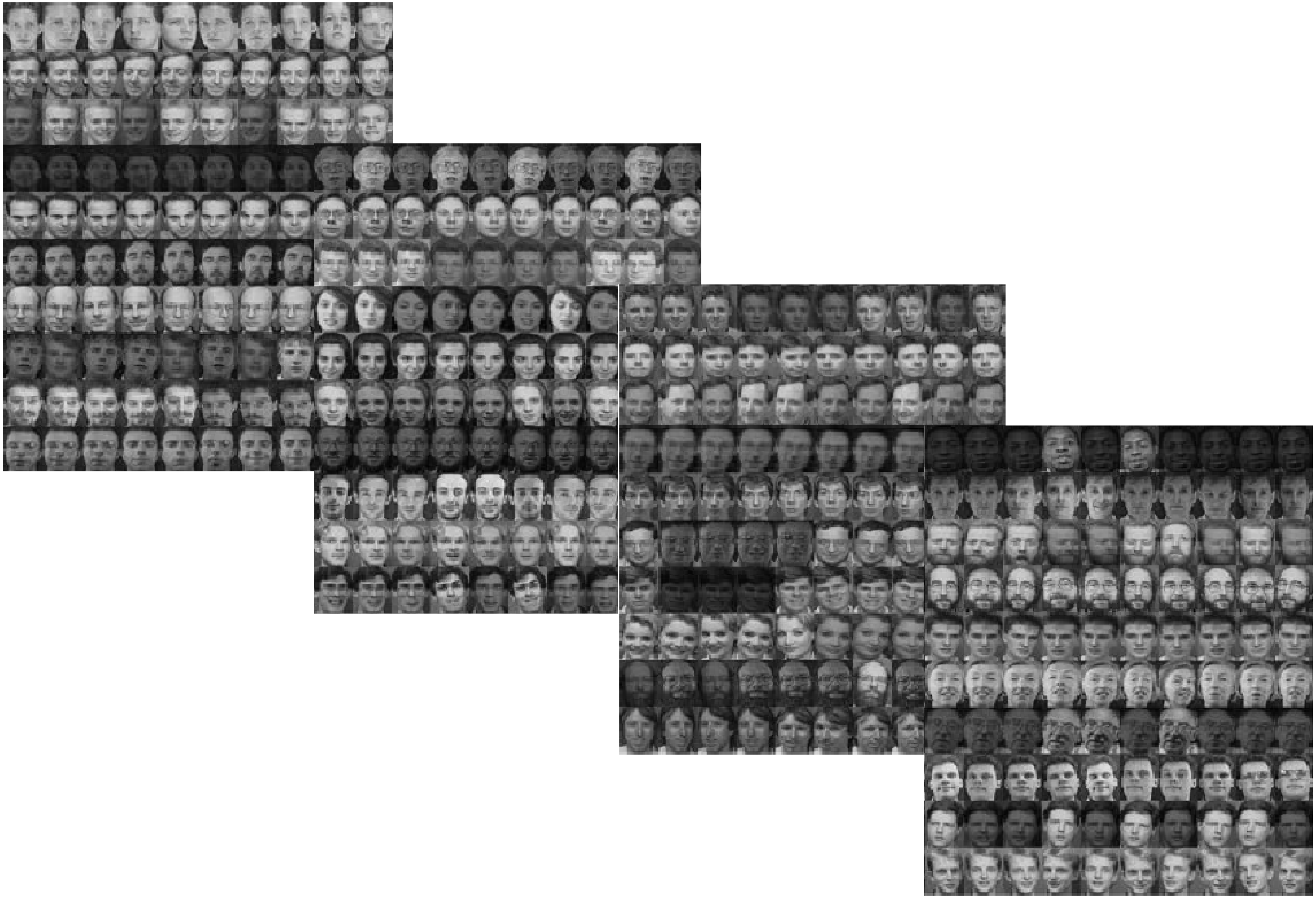




spectral clustering

1) 构建表示对象集的相似度矩阵W；
2) 通过计算相似度矩阵或拉普拉斯矩阵的前k个特征值与特征向量，构建特征向量空间；
3) 利用K-means或其它经典聚类算法对特征向量空间中的特征向量进行聚类。

Olivetti Face Database



Complex Wavelet Structural Similarity

- Structural similarity index (SSIM)

  The index between two image patches x={x_i|i=1, …,M} and y={y_i|i=1, …,M} is defined as

$$S(\mathrm{x},\mathrm{y})=\frac{(2\mu_x\mu_y+\mathrm{C}_1)(2\sigma_{xy}+\mathrm{C}_2)}{(\mu_x^2+\mu_y^2+\mathrm{C}_1)(\sigma_x^2+\sigma_y^2+\mathrm{C}_2)}$$

Where C1 and C2 are two small positive constants, and

$$\sigma_x^2=1\big/M\sum_{i=1}^{M}(x_i-\mu_x)^2,\ \mu_x=1\big/M\sum_{i=1}^{M}x_i,\ \sigma_{xy}=1\big/M\sum_{i=1}^{M}(x_i-\mu_x)(y_i-\mu_y)$$

- Complex wavelet structural similarity index (CW-SSIM)

  c_x={$c_{x,i}$|i=1, …,N} and c_y={$c_{y,i}$|i=1, …,N} are two sets of coefficients extracted at the same spatial location in the same wavelet subbands of the two images being compared, respectively. The CW-SSIM index is defined as

$$\tilde{S}(\mathrm{c}_x,\mathrm{c}_y)=\frac{2\sum_{i=1}^{N}|\mathrm{c}_{x,i}||\mathrm{c}_{y,i}|+K}{\sum_{i=1}^{N}|\mathrm{c}_{x,i}|^2+\sum_{i=1}^{N}|\mathrm{c}_{y,i}|^2+K}\bullet\frac{2\left|\sum_{i=1}^{N}\mathrm{c}_{x,i}\mathrm{c}_{y,i}^*\right|+K}{2\sum_{i=1}^{N}|\mathrm{c}_{x,i}\mathrm{c}_{y,i}^*|+K}=\frac{2\left|\sum_{i=1}^{N}\mathrm{c}_{x,i}\mathrm{c}_{y,i}^*\right|+K}{\sum_{i=1}^{N}|\mathrm{c}_{x,i}|^2+\sum_{i=1}^{N}|\mathrm{c}_{y,i}|^2+K}$$

Complex Wavelet Structural Similarity

$$S(\mathrm{x},\mathrm{y}) = \boxed{\frac{(2\mu_x\mu_y + \mathrm{C}_1)(2\sigma_{xy} + \mathrm{C}_2)}{(\mu_x^2 + \mu_y^2 + \mathrm{C}_1)(\sigma_x^2 + \sigma_y^2 + \mathrm{C}_2)}}$$

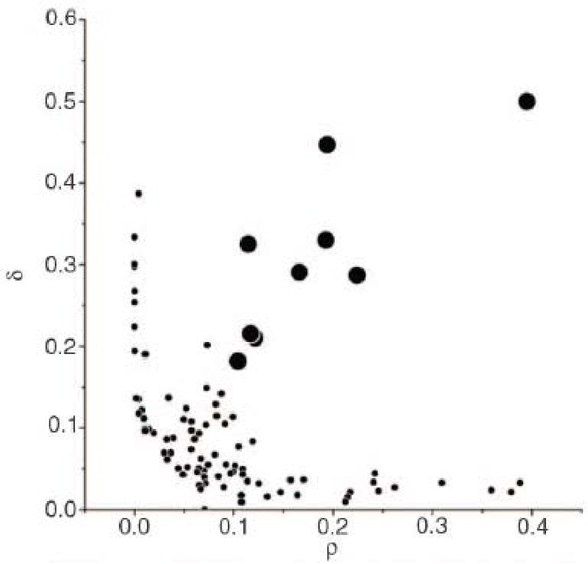
The maximum SSIM index value 1 is achieved if and only if and are identical.

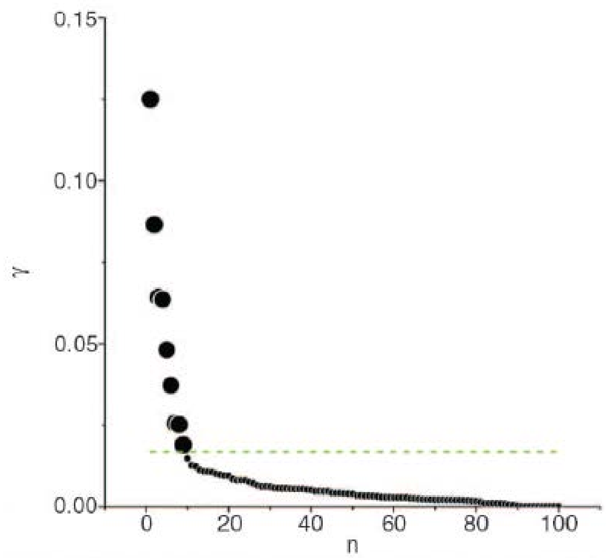
$$\tilde{S}(\mathrm{c}_x, \mathrm{c}_y) = \boxed{\frac{2\sum_{i=1}^{N}\left|\mathrm{c}_{x,i}\right|\left|\mathrm{c}_{y,i}\right| + K}{\sum_{i=1}^{N}\left|\mathrm{c}_{x,i}\right|^2 + \sum_{i=1}^{N}\left|\mathrm{c}_{y,i}\right|^2 + K}} \cdot \boxed{\frac{2\left|\sum_{i=1}^{N}\mathrm{c}_{x,i}\mathrm{c}_{y,i}^*\right| + K}{2\sum_{i=1}^{N}\left|\mathrm{c}_{x,i}\mathrm{c}_{y,i}^*\right| + K}}$$

The maximum value 1 is achieved if and only if $|c_{x,i}| = |c_{y,i}|$ for all i

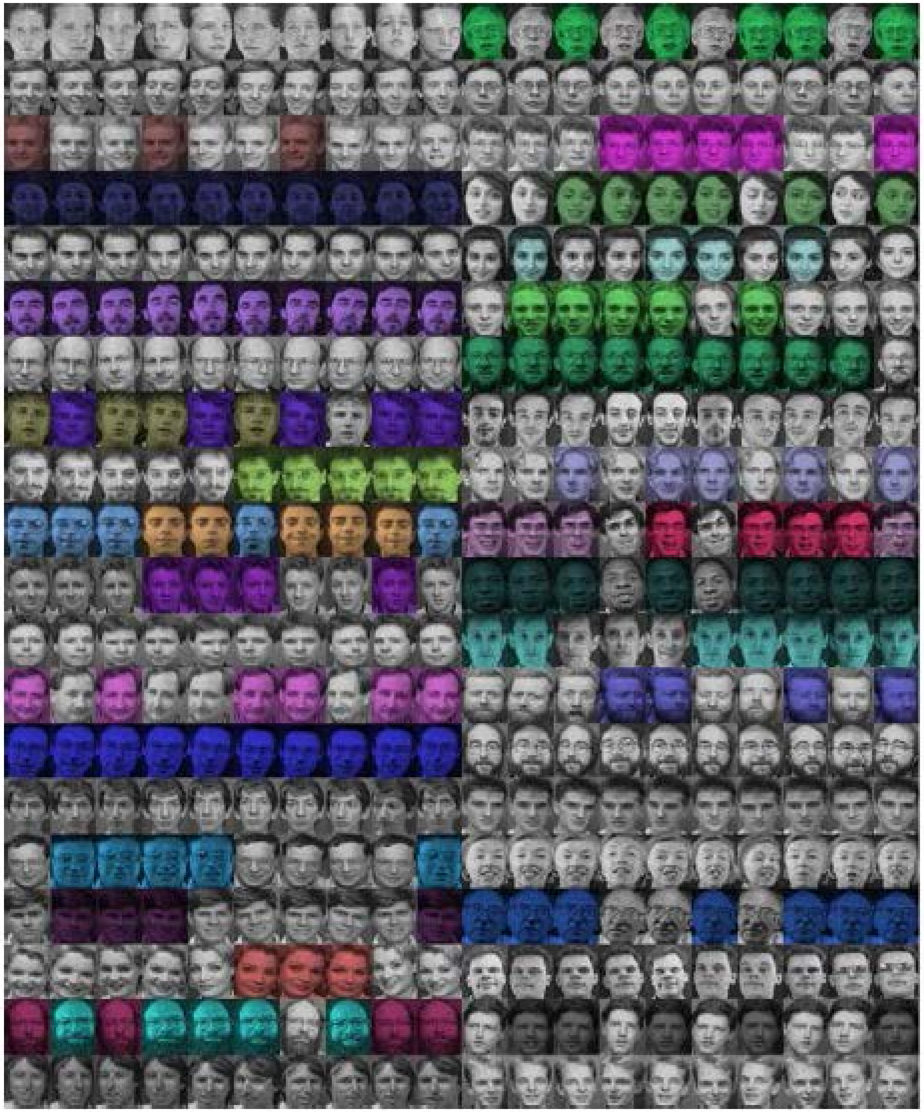
The maximum value 1 when the phase difference between $c_{x,i}$ and $c_{y,i}$ is a constant for all i.

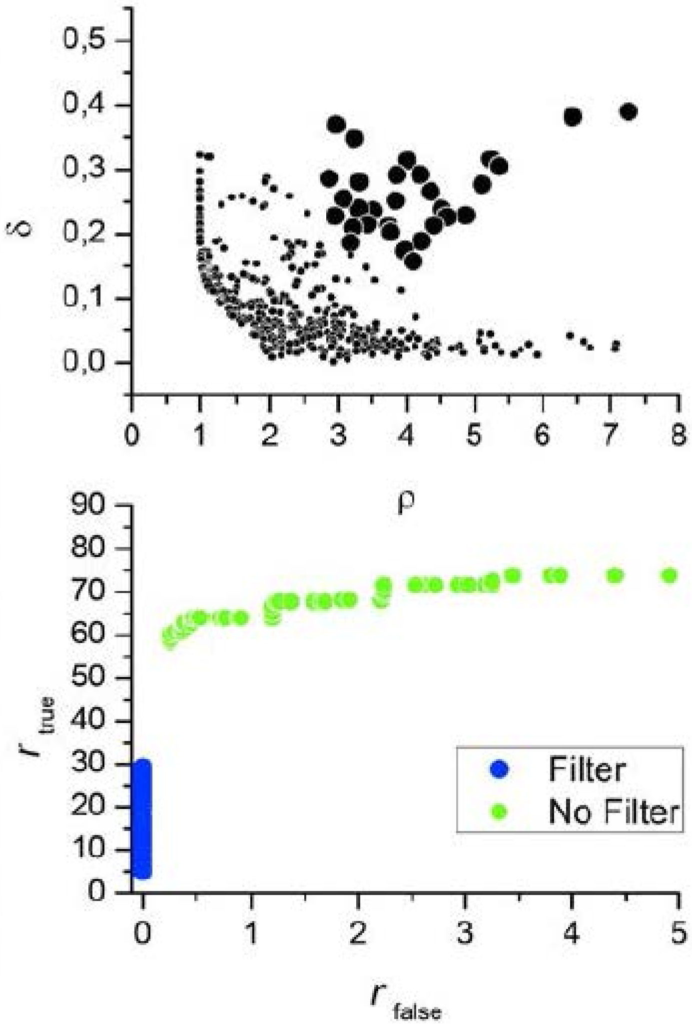
Olivetti Face Database



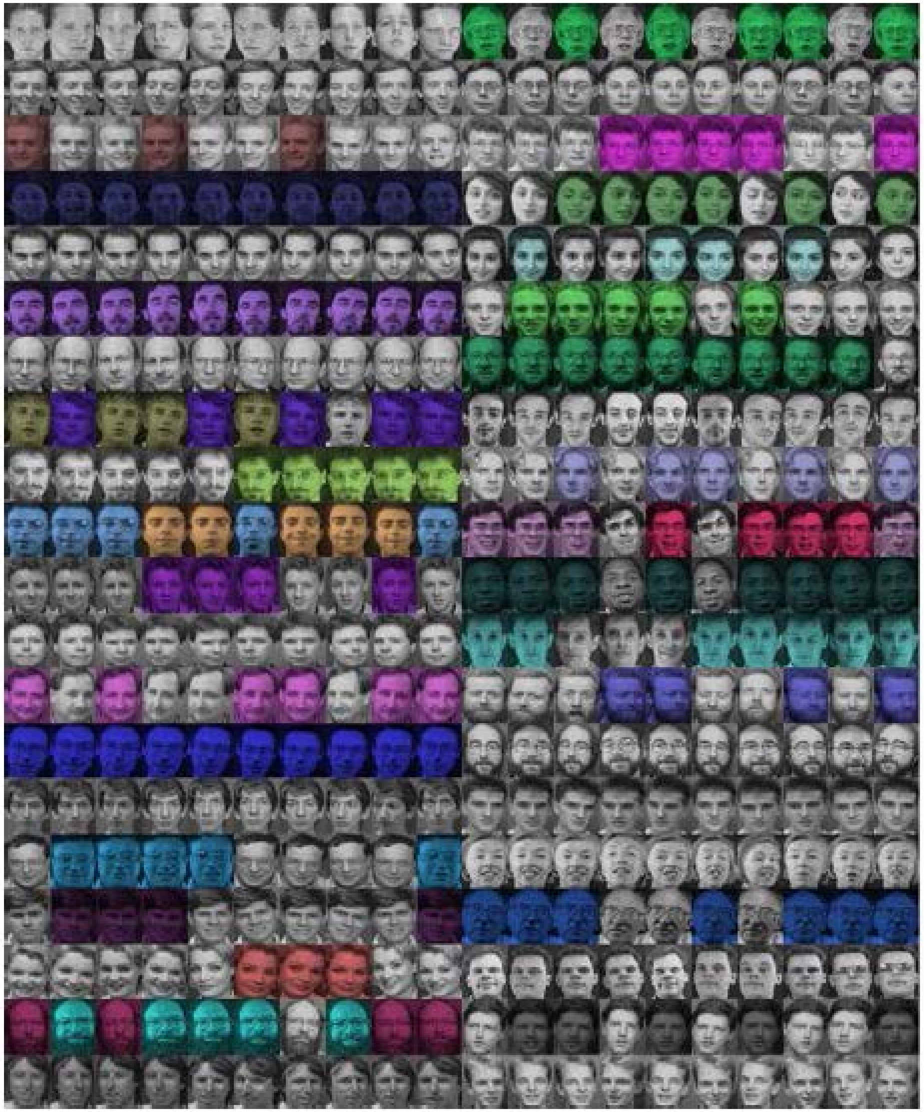


Olivetti Face Database





Olivetti Face Database

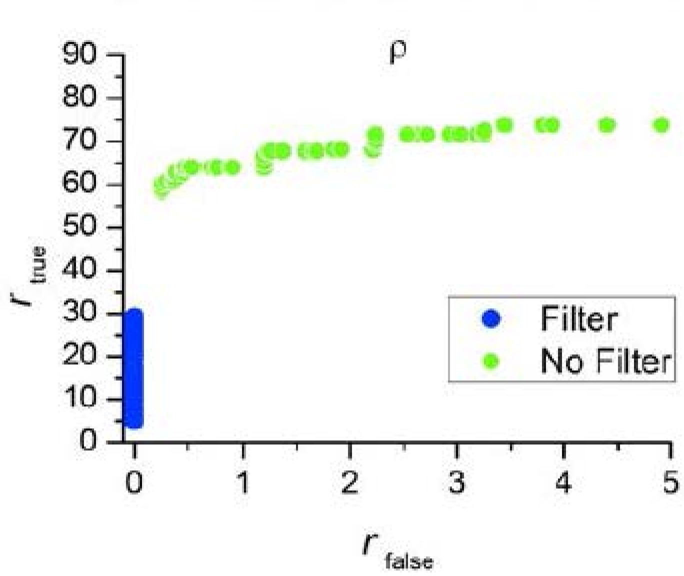


'rate of true association'

the fraction of pairs of images from the same true category that were correctly placed in the same learned category.

'rate of false association'

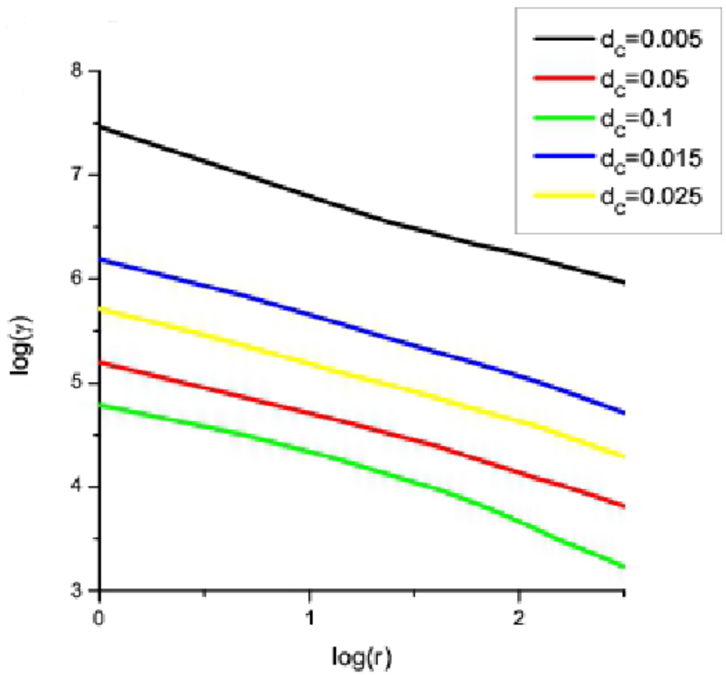
the fraction of pairs of images from different true categories that were erroneously placed in the same learned category.

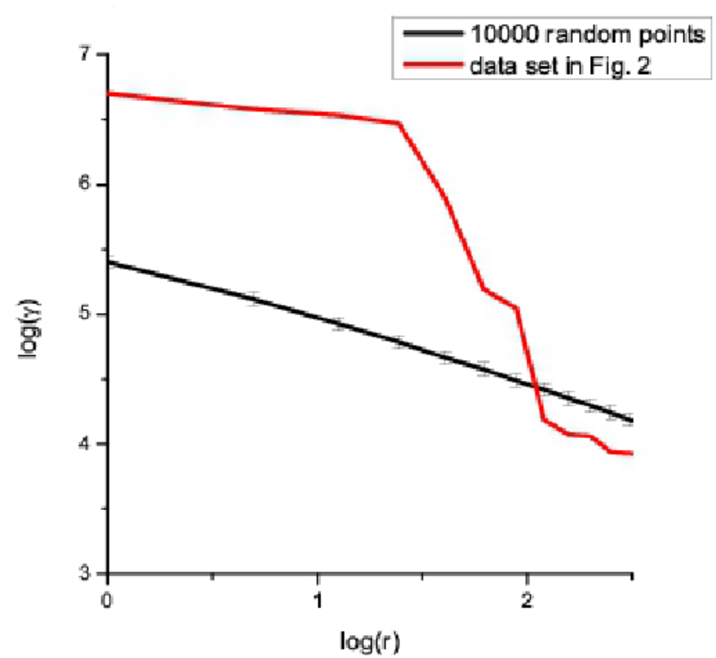


$\Upsilon_i = \rho_i\ \delta_i$

➢For randomly distributed data points, the quantity $\Upsilon_i = \rho_i\ \delta_i$ is distributed with an exponent that depends on the dimensionality of the space in which the points are embedded.

➢This observation may provide the basis for a criterion for the automatic choice of the cluster centers as well as for statistically validating the reliability of an analysis performed with out approach.





END

Thank you

and

questions ?