DifFUZZY: a fuzzy clustering algorithm for complex datasets

Ornella Cominetti*

Centre for Mathematical Biology,
Mathematical Institute,
University of Oxford,
24–29 St. Giles', Oxford, OX1 3LB, UK
E-mail: cominetti@maths.ox.ac.uk
*Corresponding author

Anastasios Matzavinos

Department of Mathematics,
Iowa State University,
Ames, IA 50011, USA
E-mail: tasos@iastate.edu

Sandhya Samarasinghe and Don Kulasiri

Centre for Advanced Computational Solutions (C-fACS),
Lincoln University,
P.O. Box 84, Christchurch, New Zealand
E-mail: Sandhya.Samarasinghe@lincoln.ac.nz
E-mail: Don.Kulasiri@lincoln.ac.nz

Sijia Liu

Department of Mathematics,
Iowa State University,
Ames, IA 50011, USA
E-mail: sijialiu@iastate.edu

Philip K. Maini

Centre for Mathematical Biology,
Mathematical Institute,
University of Oxford,
24–29 St. Giles', Oxford, OX1 3LB, UK
and
Oxford Centre for Integrative Systems Biology,
Department of Biochemistry,
University of Oxford,
South Parks Road, Oxford, OX1 3QU, UK
E-mail: maini@maths.ox.ac.uk

Radek Erban

Oxford Centre for Collaborative Applied Mathematics,
Mathematical Institute,
University of Oxford,
24–29 St. Giles', Oxford, OX1 3LB, UK
E-mail: erban@maths.ox.ac.uk

Abstract: Soft (fuzzy) clustering techniques are often used in the study of high-dimensional datasets, such as microarray and other high-throughput bioinformatics data. The most widely used method is the fuzzy C-means (FCM) algorithm , but it can present difficulties when dealing with some datasets. A fuzzy clustering algorithm, DifFUZZY, which utilises concepts from diffusion processes in graphs and is applicable to a larger class of clustering problems than other fuzzy clustering algorithms is developed. Examples of datasets (synthetic and real) for which this method outperforms other frequently used algorithms are presented, including two benchmark biological datasets, a genetic expression dataset and a dataset that contains taxonomic measurements. This method is better than traditional fuzzy clustering algorithms at handling datasets that are 'curved', elongated or those which contain clusters of different dispersion. The algorithm has been implemented in Matlab and C++ and is available at http://www.maths.ox.ac.uk/cmb/difFUZZY.

Keywords: clustering algorithm; fuzzy clustering; diffusion distance; genetic expression data clustering.

Reference to this paper should be made as follows: Cominetti, O., Matzavinos, A., Samarasinghe, S., Kulasiri, D., Liu, S., Maini, P.K. and Erban, R. (2010) 'DifFUZZY: a fuzzy clustering algorithm for complex datasets', *Int. J. Computational Intelligence in Bioinformatics and Systems Biology*, Vol. 1, No. 4, pp.402–417.

Biographical notes: Ornella Cominetti received her Engineering degree from the University of Chile, specialising in Biotechnology. She is currently a DPhil student at the Centre of Mathematical Biology, Mathematical Institute, and the Systems Biology Doctoral Training Centre of the University of Oxford. Her research interests are in cancer and network systems biology, multiscale modelling and data clustering and its applications.

Anastasios Matzavinos received his PhD from the University of Dundee in Scotland. He has held positions at the University of Minnesota and the Ohio State University and is currently an Assistant Professor in the Department of Mathematics at Iowa State University. His research interests include mathematical biology, phase retrieval problems in imaging, applied stochastic processes and data clustering algorithms.

Sandhya Samarasinghe received her PhD from Virginia Tech. She is currently an Associate Professor at the Department of Environmental Management and a Founding Member of the Centre for Advanced Computational Solutions (C-fACS) at Lincoln University in New Zealand. Her research interests are complex adaptive systems research involving neural networks, statistics, computer vision, fuzzy systems and fuzzy cognitive mapping, intelligent computing and soft systems and the application of the above methods for understanding and integrated problem solving in biology, environment and natural resources.

Don Kulasiri received his PhD from Virginia Tech. He is currently a Professor of Systems Biology at the Lincoln University and a Founding Member and Head of Centre for Advanced Computational Solutions (C-fACS). His current research lies in molecular systems biology, stochastic modelling of biological and environmental systems and bio-engineering (biotechnology).

Sijia Liu received her BSc in Mathematics and Applied Mathematics from the University of Science and Technology of China. She is currently a PhD candidate at the Department of Mathematics of Iowa State University. Her research interests are in applied mathematics, applications of spectral graph theory to data clustering and bioinformatics, mathematical biology, and stochastic chemical kinetics.

Philip K. Maini received his DPhil in Mathematics from Oxford University. He is currently a Professor of Mathematical Biology and the Director of the Centre for Mathematical Biology, Mathematical Institute, Oxford. He is also part of the Oxford Centre for Integrative Systems Biology, Department of Biochemistry. His interests are mainly in deterministic models of cell- and tissue-level interactions to signalling cues with applications in developmental biology, cancer growth and wound healing.

Radek Erban received his PhD from the University of Minnesota. He is currently a Research Fellow in the Oxford Centre for Collaborative Applied Mathematics, Mathematical Institute, University of Oxford. His research interests include mathematical biology, multiscale modelling, partial differential equations, stochastic simulation algorithms, gene regulatory networks, mathematical fluid dynamics and applications of mathematics in medicine.

1 Introduction

The need to interpret and extract possible inferences from high-dimensional bioinformatic data has led over the past decades to the development of dimensionality reduction and data clustering techniques. One of the first studied data clustering methodologies is the K-means algorithm, which was introduced by MacQueen (1967) and is the prototypical example of a non-overlapping, hard (crisp) clustering approach (Gan et al., 2007). The applicability of the K-means algorithm, however, is limited by the requirement that the clusters to be identified should be well-separated and 'convex-shaped' [such as those in Figure 1(a)] which is often not the case in biological data. Two fundamentally distinct approaches have been proposed in the past to address these two restrictions.

Bezdek et al. (1984) proposed the fuzzy C-means (FCM) algorithm as an alternative, soft clustering approach that generates fuzzy partitions for a given dataset. In the case of FCM the clusters to be identified do not have to be well-separated, as the method assigns cluster membership probabilities to undecidable elements of the dataset that cannot be readily assigned to a specific cluster. However, the method does not exploit the intrinsic geometry of non-convex clusters, and, as we demonstrate in this article, its performance

is drastically reduced when applied to some datasets, for example those in Figures 2(a) and 3(a). This behaviour can also be observed in the case of the standard K-means algorithm (Ng et al., 2001). These algorithms have been very successful in a number of examples in very diverse areas [such as in image segmentation (Trivedi and Bezdek, 1986), analysis of genetic networks (Stuart et al., 2003), protein class prediction (Zhang et al., 1995), epidemiology (French et al., 2008), among many others], but here we also explore datasets for which their performance is poor.

To circumvent the above problems associated with the geometry of datasets, approaches based on spectral graph theory and diffusion distances have been recently devised (Nadler et al., 2006; Yen et al., 2005). However, these algorithms are generally hard clustering methods which do not allow data points to belong to more than one cluster at the same time. This limits their applicability in clustering genetic expression data, where alternative or superimposed modes of regulation of certain genes would not be identified using partitional methods (Dembélé and Kastner, 2003). In this paper, we present DifFUZZY, a fuzzy clustering algorithm that is applicable to a larger class of clustering problems than the FCM algorithm (Bezdek et al., 1984). For datasets with 'convex-shaped' clusters both approaches lead to similar results, but DifFUZZY can better handle clusters with a complex, non-linear geometric structure. Moreover, DifFUZZY does not require any prior information on the number of clusters.

The paper is organised as follows. In Section 2, we present the DifFUZZY algorithm and give an intuitive explanation of how it works. In Section 3, we start with a prototypical example of a dataset which can be successfully clustered by FCM, and we show that DifFUZZY leads to consistent results. Subsequently, we introduce examples of datasets for which FCM fails to identify the correct clusters, whereas DifFUZZY succeeds. Then, we apply DifFUZZY to biological datasets, namely, the Iris taxonomic dataset and cancer genetic expression datasets.

2 Methods

DifFUZZY is an alternative clustering method which combines ideas from fuzzy clustering and diffusion on graphs.[1] The input of the algorithm is the dataset in the form:

$$\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_N \in \mathbb{R}^p \tag{1}$$

where N is the number of data points and p is their dimension, plus four parameters, one of which is external, M, and the rest are the internal and optional parameters γ_1, γ_2 and γ_3. M is an integer which represents the minimum number of data points in the clusters to be found. This parameter is necessary, since in most cases only a few data points do not constitute a cluster, but a set of soft data points or a set of outliers. There are three optional parameters: γ_1, γ_2 and γ_3 whose default values (0.3, 0.1 and 1, respectively) have been optimised and used successfully in all the datasets analysed. A regular user can use these values with confidence. However, more advanced users can modify their values, with the intuitive explanation provided in Section 2.4.

DifFUZZY returns a number of clusters (C) and a set of membership values for each data point in each cluster. The membership value of data point $\mathbf{X}_i$ in the cluster c is denoted as $u_c(\mathbf{X}_i)$, and it goes from zero to one, where this latter case means that $\mathbf{X}_i$

is very likely a member of the cluster c, while the former case ($u_c(\mathbf{X}_i) \sim 0$) corresponds to the situation in which the point $\mathbf{X}_i$ is very likely not a member of the cluster c. The membership degrees of the ith point, $i = 1, 2, \ldots, N$, sum to 1, that is:

$$\sum_{c=1}^{C} u_c(\mathbf{X}_i) = 1. \tag{2}$$

DifFUZZY has been implemented in Matlab and C++ and can be downloaded from: http://www.maths.ox.ac.uk/cmb/diffuzzy. The algorithm can be divided into three main steps, which will be explained in the following Sections 2.1–2.3. The reader who is not particularly interested in understanding the details of the algorithm can skip this part of the paper.

2.1 Identification of the core of clusters

To explain the first step of the algorithm, we define the auxiliary function $F(\sigma) : (0, \infty) \to \mathbb{N}$ as follows. Let $\sigma \in (0, \infty)$ be a positive number. We construct the so called σ-neighbourhood graph where each node represents one data point from the dataset (1), i.e., the σ-neighbourhood graph has N nodes. The ith node and jth node will be connected by an edge if $||\mathbf{X}_i - \mathbf{X}_j|| < \sigma$, where $||\cdot||$ represents the Euclidean norm. Then $F(\sigma)$ is equal to the number of components of the σ-neighbourhood graph which contain at least M vertices, where M is the mandatory parameter of DifFUZZY introduced above.

Figure 1(b) shows an example of the plot of $F(\sigma)$, which was obtained using the dataset presented in Figure 1(a). We can see that $F(\sigma)$ begins from zero, and then increases to its maximum value, before settling back down to a value of 1. The final value will always be one, because the σ-neighbourhood graph is fully connected for sufficiently large σ values, i.e., it only has one component.

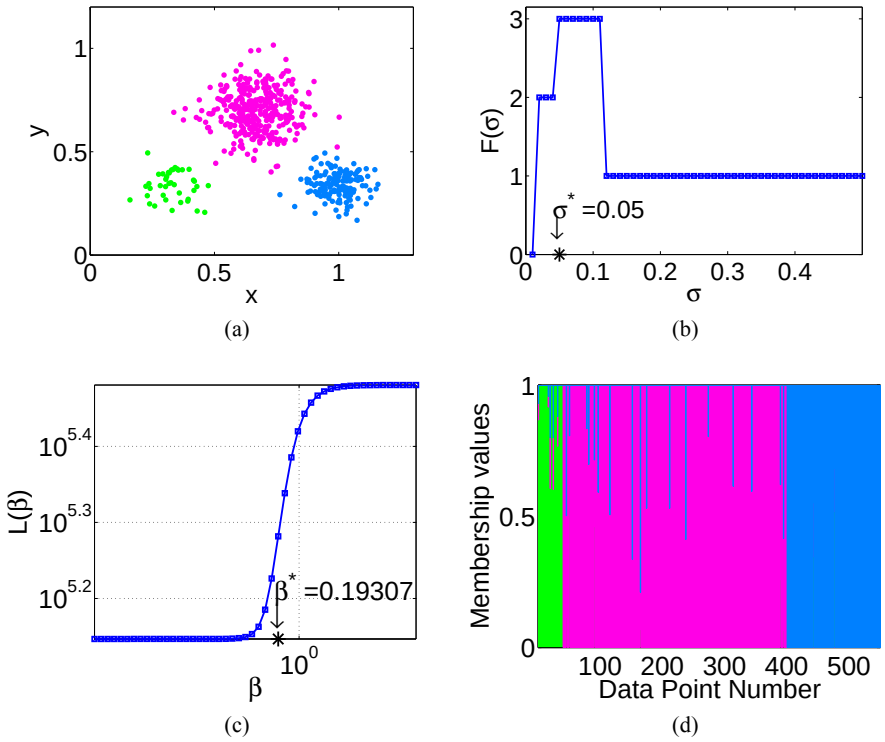
DifFUZZY computes the number, C, of clusters as the maximum value of $F(\sigma)$, i.e.,

$$C = \max_{\sigma \in (0,\infty)} F(\sigma).$$

For the example in Figure 1(b), we have $C = 3$, which corresponds to the three clusters shown in the original dataset in Figure 1(a).

In Figure 1(b), we see that there is an interval of values of σ for which $F(\sigma)$ reaches its maximum value C. As the next step DifFUZZY computes σ^*, which is defined as the minimum value of σ for which $F(\sigma)$ is equal to C. Then the σ^*-neighbourhood graph is constructed. The components of this graph which contain at least M vertices will form the 'cores' of the clusters to be identified. Each data point $\mathbf{X}_i$ which lies in the cth core is assigned the membership values $u_c(\mathbf{X}_i) = 1$ and $u_j(\mathbf{X}_i) = 0$ for $j \neq c$, as this point fully belongs to the cth cluster. Every such point will be called a hard point in what follows. The remaining points are called soft points. Since we already know the number of clusters C and the membership functions of hard points, it remains to assign a membership function to each soft point. This will be done in two steps. First we compute some auxiliary matrices in Section 2.2 and then we assign the membership values to soft points in Section 2.3.

Figure 1 (a) 'Globular clusters' dataset, (b) $F(\sigma)$ for the dataset in (a)*, (c) $L(\beta)$, given by equation (4) plotted on a logarithmic scale, for the dataset in (a)**, (d) DifFUZZY membership values for this dataset***



Notes: *For this dataset, we determined the number of clusters C to be 3, and $\sigma^* = 0.05$, for the parameter M = 35.
**$\beta^* = 0.19307$ was obtained using equation (5).
***Each data point is represented by a bar of total height equal to 1 [from equation (2) (M = 35).
Colour code: green, red and blue correspond to the membership value of the data points in the three clusters, with the corresponding colour code as in (a). This representation will be used in Figures 2 and 3.

2.2 *Computation of auxiliary matrices W, D and P*

In this section, we show the formulae to compute the auxiliary matrices W, D and P, whose definition can be intuitively understood in terms of diffusion processes on graphs, as explained in Section 2.4. We first define a family of matrices $\widehat{W}(\beta)$ with entries:

$$\widehat{w}_{i,j}(\beta) = \begin{cases} 1 & \text{if } i \text{ and } j \text{ are hard points} \\ & \text{in the same core cluster,} \\ \exp\left(-\frac{||\mathbf{X}_i - \mathbf{X}_j||^2}{\beta}\right) & \text{otherwise,} \end{cases} \tag{3}$$

where β is a positive real number. We define the function $L(\beta) : (0, \infty) \to (0, \infty)$ to be the sum:

$$L(\beta) = \sum_{i=1}^{N} \sum_{j=1}^{N} \widehat{w}_{i,j}(\beta). \tag{4}$$

The log-log plot of function $L(\beta)$ is shown in Figure 1(c) for the dataset given in Figure 1(a). We can see that it has two well-defined limits:

$$\lim_{\beta \to 0} L(\beta) = N + \sum_{i=1}^{C} n_i \left(n_i - 1\right) \quad \text{and} \quad \lim_{\beta \to \infty} L(\beta) = N^2,$$

where n_i corresponds to the number of points in the ith core cluster. As explained in Section 2.4, we are interested in finding the value of β which corresponds to an intermediate value of $L(\beta)$. DifFUZZY does this by finding β^* which satisfies the relation:

$$L(\beta^*) = (1 - \gamma_1) \left(N + \sum_{i=1}^{C} n_i \left(n_i - 1\right) \right) + \gamma_1 N^2, \tag{5}$$

where $\gamma_1 \in (0, 1)$ is an internal parameter of the method. Its default value is 0.3. Then the auxiliary matrices are defined as follows. We put:

$$W = \widehat{W}(\beta^*). \tag{6}$$

The matrix D is defined as a diagonal matrix with diagonal elements:

$$D_{i,i} = \sum_{j=1}^{N} w_{i,j}\ , \quad i = 1, 2, \ldots, N, \tag{7}$$

where $w_{i,j}$ are the entries of matrix W. Finally, the matrix P is defined as:

$$P = I + [W - D] \frac{\gamma_2}{\max\limits_{i=1,\ldots N} D_{i,i}}, \tag{8}$$

where $I \in \mathbb{R}^{N \times N}$ is the identity matrix and γ_2 is an internal parameter of DifFUZZY. Its default value is 0.1.

2.3 The membership values of soft data points

Let $\mathbf{X}_s$ be a soft data point. To assign its membership value $u_c(\mathbf{X}_s)$ in cluster $c \in \{1, 2, \ldots, C\}$, we first find the hard point in the c-th core which is closest (in Euclidean distance) to $\mathbf{X}_s$. This point will be denoted as $\mathbf{X}_n$ in what follows. Using the matrix W defined by equation (6), DifFUZZY constructs a new matrix $\overline{W}$ which is equal to the original matrix W, with the sth row replaced by the nth row and the sth column replaced by the nth column. Using $\overline{W}$ instead of W, matrices $\overline{D}$ and $\overline{P}$ are

computed by (7) and (8), respectively. DifFUZZY also computes an auxiliary integer parameter α by:

$$\alpha = \left\lfloor \frac{\gamma_3}{|\log \lambda_2|} \right\rfloor,$$

where λ_2 corresponds to the second (largest) eigenvalue of P and $\lfloor \cdot \rfloor$ denotes the integer part.

Next, we compute the diffusion distance between the soft point $\mathbf{X}_s$ and the cth cluster by:

$$\text{dist}(\mathbf{X}_s, c) = \left|\left| P^{\alpha}\mathbf{e} - \overline{P}^{\,\alpha}\mathbf{e} \right|\right|, \tag{9}$$

where $\mathbf{e}(j) = 1$ if $j = s$, and $\mathbf{e}(j) = 0$ otherwise. Finally, the membership value of the soft point $\mathbf{X}_s$ in the cth cluster, $u_c(\mathbf{X}_s)$, is determined with the following formula:

$$u_c(\mathbf{X}_s) = \frac{\text{dist}(\mathbf{X}_s, c)^{-1}}{\sum_{l=1}^{C} \text{dist}(\mathbf{X}_s, l)^{-1}}. \tag{10}$$

This procedure is applied to every soft data point $\mathbf{X}_s$ and every cluster $c \in \{1, 2, \ldots, C\}$.

2.4 Geometric and graph interpretation of DifFUZZY

In this section, we provide an intuitive geometric explanation of the ideas behind the DifFUZZY algorithm. The matrix P can be thought of as a transition matrix whose rows all sum to 1, and whose entry $P_{i,j}$ corresponds to the probability of jumping from the node (data point) i to the node j in one time step. The jth component of the vector $P^{\alpha}\mathbf{e}$, which is used in (9), is the probability of a random walk ending up in the jth node, $j = 1, 2, \ldots, N$, after α time steps, provided that it starts in the sth node.

In this geometric interpretation we can give an intuitive meaning to the auxiliary parameters γ_1, γ_2 and γ_3. The parameter $\gamma_1 \in (0, 1)$ is related to the time scale of this random walk. $\gamma_1 \sim 1$ corresponds to the case where all the nodes are highly connected, and therefore the diffusion will occur instantaneously, whereas for values of $\gamma_1 \sim 0$, there will be almost no diffusion between cluster cores. Therefore, we are interested in an intermediate point, where there is enough time to diffuse, but where equilibrium has not yet been reached. The parameter $\gamma_2 \in (0, 1)$ ensures that none of the entries of the transition matrix P are negative, which is important, since they represent transition probabilities. It can be interpreted as the length of the time step of the random walk on the graph. For very small values of γ_2 we have $P \sim I$, for which the probabilities of transition between different data points is close to zero, therefore there will not be any diffusion during one time step.

The parameter $\gamma_3 \in (0, \infty)$ is the number of time steps the random walk is going to be run or propagated, capturing information of higher order neighbourhood structure (Lafon and Lee, 2006). Small values of γ_3 give us a few time steps, whereas large values of γ_3 give us a large number of time steps. In the first situation not much diffusion has taken place, whereas in the latter case, when the random walk is propagated a very large number of times, the diffusion process is near to reaching the equilibrium.

The matrix $\overline{P}$ is used to represent a different diffusion process, an equivalent one to the first random walk, but over a new graph, where the data point $\mathbf{X}_s$ has been moved to the position of the data point $\mathbf{X}_n$. This matrix then corresponds to the transition matrix for this auxiliary graph.

3 Results

In Section 3.1, we present three computer generated test datasets, designed to illustrate the strengths and weaknesses of FCM. In all three cases we show that DifFUZZY gives the desired result. Then, in Section 3.2, we apply DifFUZZY to datasets obtained from biological experiments.

3.1 Synthetic test datasets

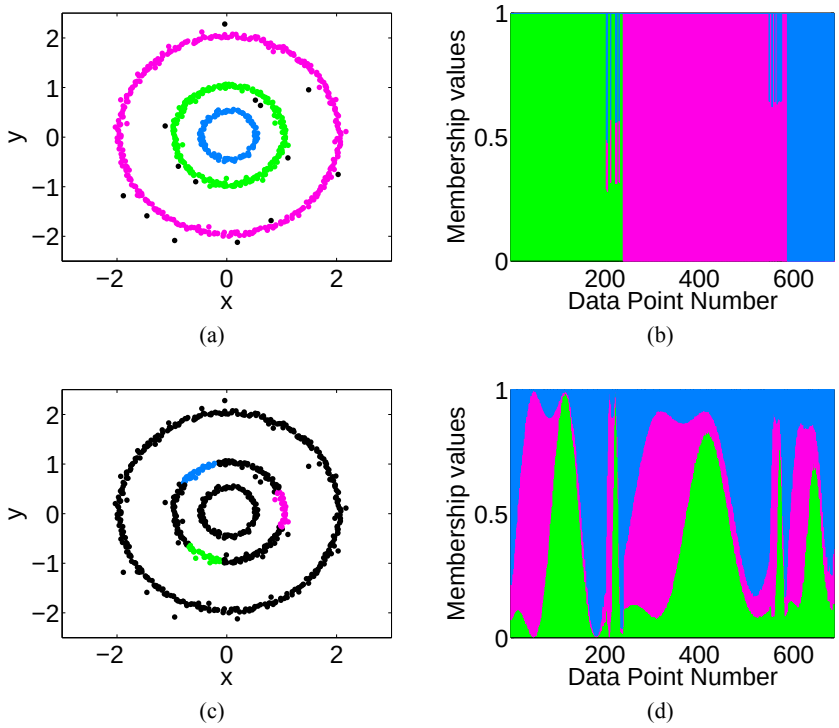
The output of DifFUZZY is a number of clusters (C) and for each data point a set of C numbers that represent the degree of membership in each cluster. The membership value of point $\mathbf{X}_i$, $i = 1, 2, \ldots, N$, in the c-th cluster, $c = 1, 2, \ldots, C$, is denoted by $u_c(\mathbf{X}_i)$. The degree of membership is a number between 0 and 1, where the values close to 1 correspond to points that are very likely to belong to that cluster. The sum of the membership values of a data point in all clusters is always one [see equation (2)]. In particular, for a given point, there can be only one cluster for which the membership value is close to 1, i.e., the point can belong to only one cluster with high certainty.

A prototypical cluster dataset in two-dimensional space is shown in Figure 1(a). Every point is described by two coordinates. We can see that the data points form three well-defined clusters which are coloured in green, red, and blue. Any good soft or hard clustering technique should identify these clusters. However, when we introduce intermediate data points, the clusters are less well-defined, closer together, and some hard clustering techniques may have difficulty in separating the clusters. FCM can successfully handle this problem (see the supplementary material). The same is true for DifFUZZY. In Figure 1(d), we present the results obtained by applying DifFUZZY to the dataset in Figure 1(a). We plot the membership values for all data points. This is a prototypical example of the type of problem for which FCM works and DifFUZZY gives comparable results. Further examples are shown in the Supplementary Material.

A classical example where the K-means algorithm fails (Filippone et al., 2008) is shown in Figure 2(a). This is a two-dimensional dataset formed by three concentric rings. Using DifFUZZY we identify each ring as a separate cluster, as can be seen in Figures 2(a)–2(b). Since fuzzy clustering assigns each point to a vector of membership values, it is more challenging to visualise the results. One option is to plot the membership values as shown in Figure 2(b). A rough idea of the behaviour of the algorithm can also be obtained by making what we call a ‘hard clusters by threshold’ (HCT)-plot defined as follows: a data point is coloured as the points in a given core cluster only if its membership value for that cluster is higher than an arbitrary threshold z. All the other data points are unassigned, and consequently plotted in black. Such a plot is shown in Figure 2(a) for $z = 0.9$. HCT-plots do not show the complete result from applying a given fuzzy clustering method to a dataset, since they contain less information than the complete result (all the membership values), and the HCT-plots depend on the threshold. However, it is illustrative to include them to clearly show how

the results of different algorithms compare. The membership values obtained with FCM are plotted in Figure 2(d). In Figure 2(c), we present the corresponding HCT-plot with a threshold value of 0.9. Comparing Figures 2(a)–2(b) with Figures 2(c)–2(d), we clearly see that DifFUZZY identifies the three rings as different clusters, while FCM fails, and this can be observed for any value of z.

Figure 2 'Concentric rings' test dataset: (a) DifFUZZY HCT-plot, $z = 0.9$ (M = 90), (b) DifFUZZY membership values, (c) FCM HCT-plot, $z = 0.9$, (d) FCM membership values

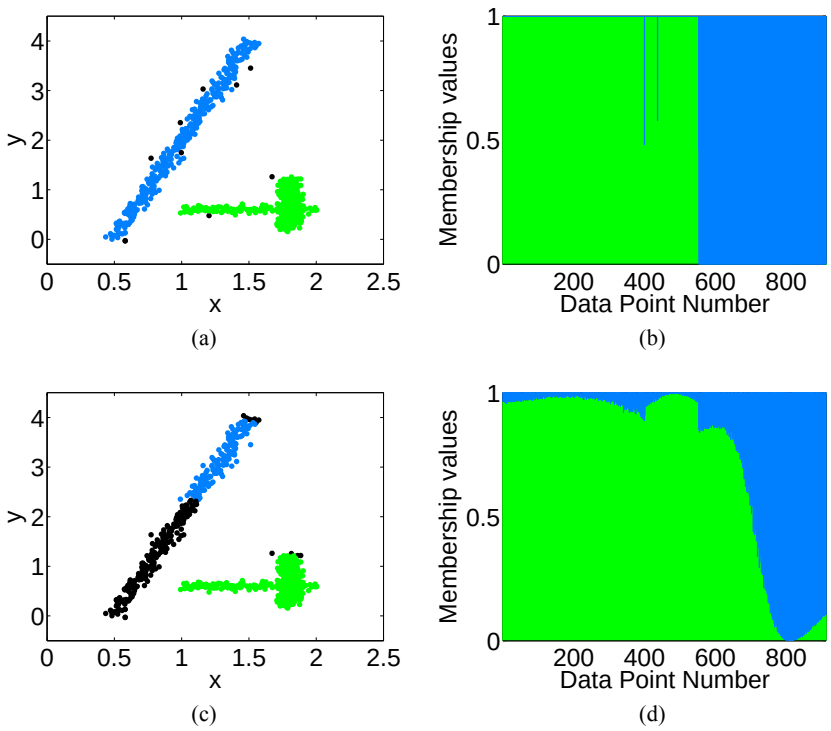


Note: Colour code for (b) and (d) as in Figure 1(b).

Another dataset where K-means algorithms fail is presented in Figure 3(a). This two-dimensional dataset contains two elongated clusters, one in a diagonal orientation and the other a cross-shaped cluster. The results of DifFUZZY and FCM applied over this dataset are summarised in the membership value plots in Figures 3(b) and 3(d), respectively. DifFUZZY can separate the clusters remarkably well, as is clear from Figure 3(a). For this dataset, FCM can not separate the clusters, cutting the left cluster (blue) in two parts as can be seen in the HCT-plot shown in Figure 3(c), using the threshold value $z = 0.9$. If we compare the membership values given by FCM [Figure 3(d)] to the one by DifFUZZY in Figure 3(b), which basically corresponds to the desired membership values of the data points, we see the wrong identification of the data points numbered around 550–700, which in the case of FCM have been given a

higher membership in the green cluster than in the cluster where they originally belong (the blue one).

Figure 3 'Elongated clusters' test dataset: (a) DifFUZZY HCT-plot, $z = 0.9$ (M = 150), (b) DifFUZZY membership values (M = 150), (c) FCM HCT-plot, $z = 0.9$, (d) FCM membership values



Note: Colour code for (b) and (d) as in Figure 1(b).

3.2 *Biological datasets*

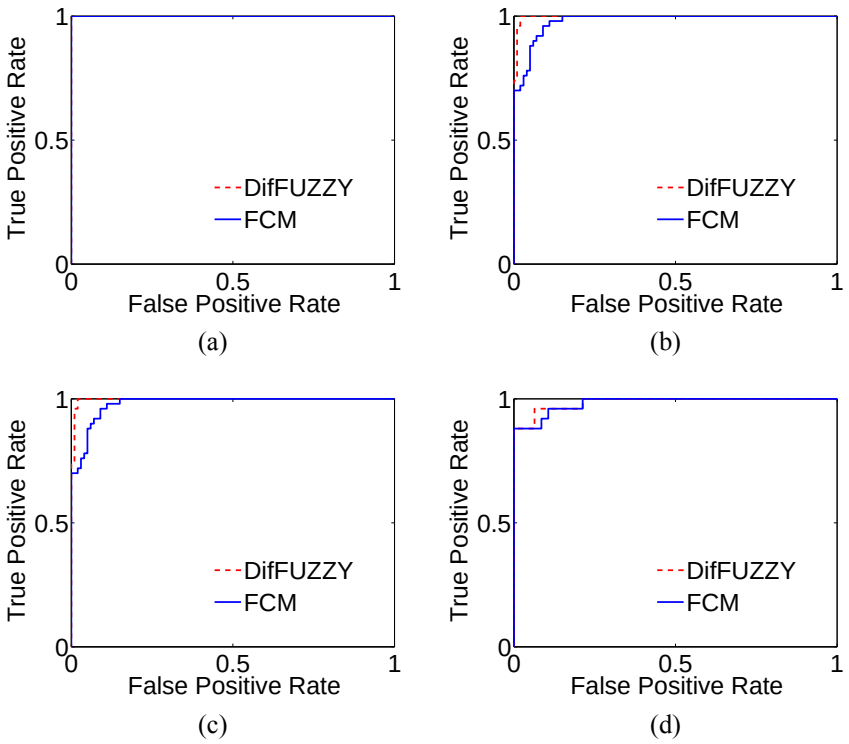
DifFUZZY was tested in two widely used biological datasets: Iris (Fisher, 1936) and Leukemia (Golub et al., 1999). In the supplementary material, we include results of the application of DifFUZZY to more biological datasets.

Iris dataset

This is a benchmark dataset in pattern recognition analysis, freely available at the UCI Machine Learning Repository (Asuncion and Newman, 2007). It contains three clusters (types of Iris plants: Iris Setosa, Iris Versicolor and Iris Virginica) of 50 data points each, of four dimensions (features): sepal length, sepal width, petal length and petal width. The class Iris Setosa is linearly separable from the other two, which are not linearly separable in their original clusters (Fisher, 1936).

We show the results of applying DifFUZZY and FCM over the Iris dataset in the form of ROC (receiver operating characteristic) curves which, in machine learning, correspond to the representation of the fraction of true positive classification (TPR) vs. the rate of false positive assignments (FPR) (Fawcett, 2006). Each data point in the curve represents a pair of values (FPR, TPR) obtained for a given threshold z. The precise definitions of both TPR and FPR are given in the Supplementary Material. A perfect clustering method would give a curve that passes through the upper left corner, presenting a 100% true positive rate for a 0% false positive rate, like the one obtained for DifFUZZY (using the parameter M = 15) and FCM in Figure 4(a), whereas if true positive and false positives are equally likely, the curve would be the diagonal line TPR=FPR. In the supplementary material, we include further information regarding how to compute a ROC curve from the membership values given by a fuzzy clustering method.

Figure 4 DifFUZZY and FCM ROC curves for: (a) the Iris Setosa data (M = 15), (b) the Iris Versicolor data (M = 15), (c) the Iris Virginica data (M = 15), (d) the Leukemia dataset (M = 11)



Figures 4(a)–4(c) show the ROC curves for the Iris Setosa, Iris Versicolor and Iris Virginica data, respectively. The three plots show very good clustering using DifFUZZY. For the Iris Setosa data, the DifFUZZY and FCM ROC curves correspond to perfect classifications, with both curves going through the (0,1) corner; both methods

classify all those points correctly, and do not assign other points to that cluster (zero false positives), but for the Iris Versicolor and Iris Virginica data, DifFUZZY performs better than FCM, since its curves pass closer to the upper left corner.

Genetic expression dataset

We tested DifFUZZY on the publicly available leukaemia dataset (Golub et al., 1999), which contains genetic expression data from patients diagnosed with either of two different types of leukaemia: acute myeloid leukaemia (AML) or acute lymphoblastic leukaemia (ALL) (Tan et al., 2004). This dataset, composed of 7,129 genes, was obtained from an Affimetrix high-density oligonucleotide microarray. The original data are divided into two sets, a set for training and a test set. Since our method is unsupervised we merged both sets obtaining a set with data from 72 patients: 25 with AML, 47 with ALL.

Before testing our clustering method on the Leukaemia dataset we pre-processed the data as done by Tan et al. (2004). The gene selection procedure consisted of selecting the Q genes with the highest squared correlation coefficient sums (Tan et al., 2004), where Q corresponds to the number of genes to be selected, which for the case of this dataset was set to be 70.

Figure 4(d) shows that DifFUZZY performs better than FCM when clustering the Leukaemia dataset, since for every point of the curve, at the same false positive rate DifFUZZY presents a higher or equal true positive rate than FCM. In the Supplementary Material we provide the plots of the membership values for all the data points. Through this example we are able to show that our method can also handle high dimensional microarray data and it can be successfully used for multi-class cancer classification tasks.

4 Discussion

In this paper and the supplementary material, we showed that the fuzzy spectral clustering method DifFUZZY performs well in a number of datasets, with sizes ranging from tens to hundreds of data points of dimensions as high as hundreds. This includes microarray data, where a typical size of a dataset is dozens or hundreds (number of samples, conditions, or patients in medical applications) and dimension is hundreds or thousands (number of genes on the chip) (Quackenbush, 2004). It is worth noting that the dimension p of the data points in equation (1). is not a bottleneck for this method, since it is only used once when computing the pairwise distances. The dimension of matrices (i.e., the computational intensity) is determined by the number N of data points, which is often smaller than the value p.

One of the issues that should be addressed is the pre-processing of data. This is crucial for some clustering applications. Noisiness of the data and the normalisation used on a given dataset can have a high impact on the results of clustering procedures (Kim et al., 2006; Karthikeyani and Thangavel, 2009). What type of normalisation to use will depend on the data themselves, and when additional information on the dataset is available it should be used in order to improve the quality of the data to be input in the algorithm. In the case of genetic expression datasets [such as the one analysed in Figure 4(d)], different steps of preprocessing commonly used are filtering, thresholding, log normalisation and gene selection (Tan et al., 2004). The latter is done in order to

reduce the dimensionality of the feature space, by discarding redundant information. Another option is to weight the different features in order to make the dimensions of the different features comparable or to augment the influence of features which carry more or better information about the data structure. The use of independent feature scaling has been described in the context of similarity matrices in Erban et al. (2007), where a single value of the parameter β in equation (3) is not necessarily appropriate for all the components (variables), given that these may vary over different orders of magnitudes. Two examples of natural weights that can be used are giving the same weight (equal importance) to the absolute values of each feature, or to rescale each variable in order for them to have the same minimum and maximum values.

A mathematical analysis of the DifFUZZY algorithm will be done in a future publication. As briefly addressed in Section 2.4, it involves an understanding of the mixing time (see, e.g., Levin et al., 2009) of the random walk defined in (8) for specific types of graphs. In particular, for a given dataset, the performance of the developed method relies on the parameter α determining the diffusion distance in (9). Computational experimentation with test datasets reveals that the optimal choice of α tends to be robust for a broad variety of dataset geometries. In order to understand this phenomenon and the underlying mechanics of DifFUZZY, current work in progress focuses on investigating mathematically the asymptotic properties of the random walk in (8) over classes of graphs characterised by specific topologies. In this context, the transition matrix P used in (9), which can be written as $P = I + (W - D)\Delta t$, is essentially a first-order approximation to the heat kernel of the graph associated with $L = D - W$. In particular, for every $\Delta t \geq 0$, the heat kernel $H_{\Delta t}$ of a graph G with graph Laplacian L is defined to be the matrix $H_{\Delta t} = e^{-\Delta t L} = I - \Delta t L + \Delta t^2 L^2/2 - \ldots$.

The importance of $H_{\Delta t}$ is that it defines an operator semigroup, describing fundamental solutions of the spatially discretised heat equation $u_t = (W - D)u$.

Heat kernels are powerful tools for defining and investigating random walks on graphs, and they provide a connection between the structure of the graph, as encoded in the graph Laplacian, and the asymptotic behaviour of the corresponding random walk (Chung, 1997). Work in progress exploits these connections in order to analyse the optimal performance of DifFUZZY for datasets exhibiting specific geometries. We are also extending the applications of DifFUZZY to a variety of clustering problems emerging in bioinformatics and image analysis applications. Fuzzy clustering methods have traditionally been used for image segmentation (Bezdek et al., 1997; Chen and Zhang, 2004; Tziakos et al., 2009), especially in the field of medical imaging. Bezdek et al. (1997) discuss the advantages of fuzzy clustering approaches applied to the specific case of segmenting magnetic resonance images (MRIs). Several variations of the FCM method are commonly employed in this context, and recent research has been focused on images that are characterised by a non-Euclidean structure of the corresponding feature space (Chen and Zhang, 2004). The clustering methodology proposed here is specifically designed to handle non-Euclidean datasets associated with a manifold structure, as it seamlessly integrates spectral clustering approaches with the evaluation of cluster membership functions in a fuzzy clustering context.

Acknowledgements

This publication is based on work (RE, AM) supported by Award No. KUK-C1-013-04, made by King Abdullah University of Science and Technology (KAUST) and the Clarendon Fund through the Systems Biology Doctoral Training Centre (OC). PKM was partially supported by a Royal Society-Wolfson Research Merit Award. RE would also like to thank Somerville College, University of Oxford for Fulford Junior Research Fellowship. The research of S.L. is supported in part by an Alberta Wolfe Research Fellowship from the Iowa State University Mathematics department. RE is also supported by the European Research Council Starting Independent Researcher Grant.

References

Asuncion, A. and Newman, D. (2007) 'UCI machine learning repository', available at http://archive.ics.uci.edu/ml/.

Bezdek, J., Ehrlich, R. and Full, W. (1984) 'Fcm: the fuzzy c-means clustering algorithm', *Computers & Geosciences*, Vol. 10, Nos. 2–3, pp.191–203.

Bezdek, J., Hall, L., Clark, M., Goldgof, D. and Clarke, L. (1997) 'Medical image analysis with fuzzy models', *Stat. Methods Med. Res.*, September, Vol. 6, No. 3, pp.191–214.

Chen, S. and Zhang, D. (2004) 'Robust image segmentation using fcm with spatial constraints based on new kernel-induced distance measure', *Systems, Man and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, Vol. 34, No. 4, pp.1907–1916, August.

Chung, F. (1997) 'Spectral graph theory', *American Mathematical Society*, Vol. 92.

Dembélé, D. and Kastner, P. (2003) 'Fuzzy c-means method for clustering microarray data', *Bioinformatics*, May, Vol. 19, No. 8, pp.973–980.

Erban, R., Frewen, T., Wang, X., Elston, T., Coifman, R., Nadler, B. and Kevrekidis, I. (2007) 'Variable-free exploration of stochastic models: a gene regulatory network example', *J. Chem. Phys.*, April, Vol. 126, No. 15, p.155103.

Fawcett, T. (2006) 'Roc analysis in pattern recognition', *Pattern Recogn. Lett.*, June, Vol. 27, No. 8, pp.861–874.

Filippone, M., Camastra, F., Masulli, F. and Rovetta, S. (2008) 'A survey of kernel and spectral methods for clustering', *Pattern Recogn.*, Vol. 41, No. 1, pp.176–190.

Fisher, R. (1936) 'The use of multiple measurements in taxonomic problems', *Annals Eugen.*, Vol. 7, pp.179–188.

French, S., Rosenberg, M. and Knuiman, M. (2008) 'The clustering of health risk behaviours in a western Australian adult population', *Health Promot. J. Austr.*, December, Vol. 19, No. 3, pp.203–209.

Gan, G., Ma, C. and Wu, J. (2007) 'Data clustering: theory, algorithms and applications', *ASA-SIAM Series on Statistics and Applied Probability*.

Golub, T., Slonim, D., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J., Coller, H., Loh, M., Downing, J., Caligiuri, M., Bloomfield, C. and Lander, E. (1999) 'Molecular classification of cancer: class discovery and class prediction by gene expression monitoring', *Science*, October, Vol. 286, No. 5439, pp.531–537.

Karthikeyani, N. and Thangavel, K. (2009) 'Impact of normalization in distributed k-means clusterings', *Int. Journal of Soft Computing*, Vol. 4, pp.168–172.

Kim, S., Lee, J. and Bae, J. (2006) 'Effect of data normalization on fuzzy clustering of dna microarray data', *BMC Bioinformatics*, Vol. 7, No. 134.

Lafon, S. and Lee, A. (2006) 'Diffusion maps and coarse-graining: a unified framework for dimensionality reduction, graph partitioning and data set parameterization', *IEEE Pattern Analysis and Machine Intelligence*.

Levin, D., Peres, Y. and Wilmer, E. (2009) 'Markov chains and mixing times', *American Mathematical Society*.

MacQueen, J. (1967) 'Some methods for classification and analysis of multivariate observations', *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability*, Vol. 1, pp.281–297, University of California Press., Berkeley.

Nadler, B., Lafon, S., Coifman, R. and Kevrekidis, I. (2006) 'Diffusion maps, spectral clustering and reaction', in *Applied and Computational Harmonic Analysis: Special Issue on Diffusion Maps and Wavelets*, p.2006.

Ng, A., Jordan, M. and Weiss, Y. (2001) 'On spectral clustering: analysis and an algorithm', in *Advances in Neural Information Processing Systems 14*, pp.849–856, MIT Press.

Quackenbush, J. (2001) 'Computational analysis of microarray data', *Nat. Rev. Genet.*, Vol. 2, p.418.

Stuart, J., Segal, E., Koller, D. and Kim, S. (2003) 'A gene-coexpression network for global discovery of conserved genetic modules', *Science*, Vol. 302, No. 5643, pp.249–255, October.

Tan, Y., Shi, L., Tong, W., Gene Hwang, G. and Wang, C. (2004) 'Multi-class tumor classification by discriminant partial least squares using microarray gene expression data and assessment of classidication models', *Comput. Biol. Chem.*, July, Vol. 28, No. 3, pp.235–244.

Trivedi, M. and Bezdek, J. (1986) 'Low-level segmentation of aerial images with fuzzy clustering', *IEEE Trans. Syst. Man. Cybern.*, Vol. SMC-16, pp.589–598.

Tziakos, I., Theoharatos, C., Laskaris, N. and Economou, G. (2009) 'Color image segmentation using laplacian eigenmaps', *Journal of Electronic Imaging*, Vol. 18, No. 2, p.023004+.

Yen, L., Vanvyve, L., Wouters, D., Fouss, F., Verleysen, F. and Saerens, M. (2005) 'Clustering using a random-walk based distance measure', in *Proceedings of ESANN'2005*.

Zhang, C., Chou, K. and Maggiora, G. (1995) 'Predicting protein structural classes from amino acid composition: application of fuzzy clustering', *Protein Eng.*, Vol. 8, pp.425–435.

Notes

1 The formulation of the FCM is given in the Supplementary Material.