Algoritmo de clustering intervalo-valorado difuso

Miguel Pagola, Laura De Miguel, Cedric Marco y Humberto Bustince

Universidad Publica de Navarra,
Campus Arrosadia s/n., 31006 Pamplona, Spain
miguel.pagola@unavarra.es http://giara.unavarra.es/

Resumen En este trabajo proponemos una nueva función objetivo para obtener una partición intervalo valorada difusa tras un proceso de clustering. En la partición intervalo valorada que obtenemos la amplitud de los intervalos de pertenencia depende de la posición de los elementos con respecto a los centros de todos los clústeres de forma que podemos distinguir los outliers.

Keywords: Clustering. Partición intervalo valorada difusa. Outliers

1. Introducción

Los algoritmos de clustering son una herramienta eficaz para extraer información de datos en bruto. Son métodos de aprendizaje no supervisado. El objetivo de los algoritmos de clustering es dividir los datos en clusteres o grupos. Básicamente obtener los clusteres de un conjunto de datos no etiquetados $X = \{x^{(1)}, \ldots, x^{(m)}\}$ es particionar X en C subgrupos $\{1 < C < m)$, de tal que forma que cada uno de ellos represente una subsestrutura de X.

Uno de los métodos más conocidos y ampliamente utilizado es el Fuzzy C-Means [1]. El algoritmo FCM asigna pertenencias entre cero y uno a los elementos, las cuales están inversamente relacionadas con la distancia entre dichos elementos y los centros de los clústeres.

La mayoría de los algoritmos de clustering en la literatura, incluyendo FCM, imponen una restricción, que se le llama condición probabilista, por la cual la suma de las pertenencias, o probabilidades de un elemento a todos los clústeres debe sumar uno. Debido a esta restricción las funciones de pertenencia o distribuciones de probabilidad obtenidas no son capaces de distinguir entre elementos que podrían pertenecer a cualquiera de los clústeres, usualmente llamados inliers, o los elementos que podrían no pertenecer a ninguno de los clústeres, conocidos como outliers.

Existen diferentes aproximaciones para tratar de resolver este problema. La forma más común es, simplemente eliminar la restricción probabilista, de tal forma que la suma de las pertenencias de un punto a todos los clústeres debe simplemente mayor que cero, o que este entre cero y uno. El algoritmo possibilistic c-means [10] y algunas de sus mejoras [9] son algoritmos clásicos y muy referenciados sin la restricción probabilista; más recientemente también existen otros trabajos en esta línea [13]. En este trabajo la idea principal es utilizar pertenencias intervalares de los elementos a los clústeres y a través de la amplitud de los intervalos de pertenencia intentar identificar los outliers y los inliers.

Estos dos conocidos algoritmos [5] y [6] también obtienen particiones intervalares. Ambos algoritmos son la generalización de los algoritmos clásicos FCM y PCM. En ambos casos se aplican sobre un mismo problema el algoritmo FCM o PCM varias veces, con diferentes parámetros y a partir de las diferentes soluciones obtenidas se genera la solución intervalar.

En este trabajo proponemos una extensión del algoritmo FCM, que obtiene una partición intervalo valorada difusa. Proponemos una función objetivo en la cual, al minimizarla, la amplitud de los intervalos de los elementos que son difíciles de asegurar su pertenencia a un cluster es mas grande que la amplitud de los elementos "normales". Ademas queremos que la amplitud de los inliers y los outliers sea diferente, para así poder identificar fácilmente esos elementos después del proceso de clustering.

El trabajo está organizado de la siguiente forma: primero presentamos las definiciones y conceptos básicos. Después de la Sección 3 describimos el problema de la restricción de la partición probabilista con un ejemplo. En la Sección 4 describimos el método propuesto y en la siguiente Sección 5 presentamos unos experimentos. Finalmente acabamos con las conclusiones y los posibles trabajos futuros.

2. Preliminares

En esta sección presentamos las definiciones, teoremas y algoritmos que son necesarios para comprender el resto del trabajo.

Definition 1 (Conjunto Difuso). *Un conjunto difuso A en un universo finito $U \neq \emptyset$ es una función $A\colon U \to [0,1]$.*

Denotamos con $\mathcal{FS}(U)$ el conjunto de todos los conjuntos difusos en U.

2.1. Conjuntos Intervalo-valorados difusos

Denotaremos con $L([0,1])$ al conjunto de todos los subintervalos cerrados en $[0,1]$:

$$L([0,1]) = \{\mathbf{x} = [\underline{x}, \overline{x}] | 0 \leq \underline{x} \leq \overline{x} \leq 1\}.$$

Entonces $L([0,1])$ es un conjunto parcialmente ordenado respecto a la relación de orden $\leq_L$ definida de la siguiente forma: dada $\mathbf{x}, \mathbf{y} \in L([0,1])$,

$$\mathbf{x} \leq_L \mathbf{y} \text{ if and only if } \underline{x} \leq \underline{y} \text{ and } \overline{x} \leq \overline{y}\ .$$

Más aún $(L([0,1]), \leq_L)$ es un retículo completo con el elemento más pequeño $0_L = [0,0]$ y el elemento mayor $1_L = [1,1]$. Hay que tener en cuenta que no es un retículo lineal ya que hay elementos que no son comparables entre sí.

La siguiente definición puede encontrarse en [2].

Definition 2 (Conjunto intervalo-valorado difuso). *Un conjunto intervalo-valorado difuso (IVFS) A en un universo finito $U \neq \emptyset$ es una función $A\colon U \to L([0,1])$.*

Denotaremos por W la función que asocia a un subintervalo cerrado $[0,1]$ su amplitud, i.e. $W\colon L([0,1]) \to [0,1]$ con $W([\underline{x}, \overline{x}]) = \overline{x} - \underline{x}$.

Hay que tener en cuenta que dado un IVFS A la pertenencia de cada elemento $u_i \in U$ está representado por un intervalo $A(u_i) = [\underline{\mu_A}(u_i), \overline{\mu_A}(u_i)]$ con amplitud $W(\mu_A(u_i)) = \overline{\mu_A}(u_i) - \underline{\mu_A}(u_i)$.

Un repaso a la historia de los IVFSs y varios ejemplos de aplicaciones, tipos de conectivos y operaciones está publicado en [2]. Denotaremos por $\mathcal{IVFS}(U)$ al conjunto de todos los IVFSs en U.

Existen dos tipos de interpretaciones diferentes de un IVFS [12]:

1. El grado de pertenencia de un elemento a un conjunto corresponde a un valor que está dentro del intervalo considerado. No podemos estar seguros de que valor es, por lo tanto se provee de un intervalo de confianza.
2. El grado de pertenencia es el subintervalo completo, entendiéndolo como un entidad matemática..

En este trabajo vamos a utilizar la primera interpretación, de hecho vamos a asignar intervalos de pertenencia para cada elemento a todos los clústeres. De tal forma que cuanto mayor sea la amplitud del intervalo, mayor será la incertidumbre que tengamos respecto a la pertenencia real de ese elemento a ese cluster.

2.2. Ordenes entre intervalos

En [3] fué introducido la noción de ordenes admisibles en $L([0,1])$. Los autores establecieron que una relación binaria $\preceq$ en $L([0,1])$ es un *orden admisible* si es un orden lineal en $L([0,1])$ refinando $\leq_L$, i.e. si para todo $[a,b],[c,d] \in L([0,1])$ tal que $[a,b] \leq_L [c,d]$ entonces es también $[a,b] \preceq [c,d]$.

El uso de ordenes admisibles nos permite comparar intervalos usando ordenes totales entre ellos. En este ejemplo mostramos unos ordenes admisibles.

Example 1. Let $[a,b],[c,d] \in L([0,1])$:

- $[a,b] \preceq_{L1} [c,d] \Leftrightarrow a < c$ o $(a = c \; y \; b \leq d)$;
- $[a,b] \preceq_{L2} [c,d] \Leftrightarrow b < d$ o $(b = d \; y \; a \leq c)$;
- $[a,b] \preceq_{XY} [c,d] \Leftrightarrow a+b < c+d$ o $(a+b = c+d$ y $b-a \leq d-c)$ (definido pot Xu y Yager en [17]);
- $[a,b] \preceq_{\alpha,\beta} [c,d] \Leftrightarrow K_\alpha(a,b) < K_\alpha(c,d)$ o $(K_\alpha(a,b) = K_\alpha(c,d) \; and \; K_\beta(a,b) \leq K_\beta(c,d))$, siendo $K_\alpha\colon [0,1]^2 \to [0,1]$ una función definida $K_\alpha(a,b) = a + \alpha \cdot (b-a)$ para $\alpha, \beta \in [0,1]$ y $\alpha \neq \beta$.

Definition 3. *Una función de agregación n-aria ($n \in \mathbb{N}, n \geq 2$) es una función no decreciente en cada argumento, $M\colon [0,1]^n \to [0,1]$, cumpliendo que $M(0,\cdots,0) = 0$ y $M(1,\cdots,1) = 1$.*

Bustince et *al.* describen en [3] diferentes métodos de construcción de ordenes admisibles utilizando funciones de agregación y presentan la noción de agregación de funciones intervalares.

2.3. Partición intervalo valorada difusa

El concepto de partición I-difusa, esto es la generalización de la partición difusa fué propuesta por V. Torra y Miyamoto en [14]. Esta generalización fué propuesta para el caso de conjuntos intuicionistas difusos de Atanassov y para conjuntos intervalo-valorados difusos. Si consideremos un conjunto de IVFSs, no es posible que todas las pertenencias del extremo inferior sumen uno y que todas las pertenencias del extremo superior de los intervalos sumen uno al mismo tiempo, salvo que la amplitud de todos los intervalos sea cero (es decir el caso difuso). Por tanto la partición intervalar requiere que solo la suma de los extremos inferiores sume uno.

Definition 4. *Sea X el conjunto referencial. Entonces un conjunto de IVFSs $A = \{A_1, \cdots, A_C\}$ es una partición intervalo valorada difusa si*

1. $\sum_{j=1}^{C} \underline{\mu_j}(x) = 1$ *para todo $x \in X$;*
2. *Existe como mucho un IVFS tal que $\underline{\mu_j}(x) = \overline{\mu_j}(x)$ for all x.*

2.4. El algoritmo Fuzzy c-Means

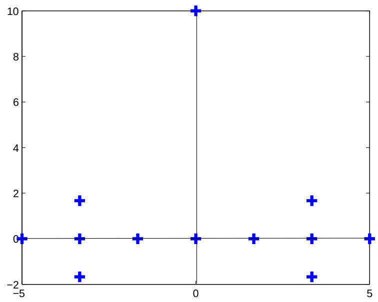
Sea $X = \{x^{(1)}, \ldots, x^{(m)}\}$ un conjunto de datos no etiquetados con m ejemplos , y cada ejemplo $x^{(i)} = (x_1^{(i)}, \ldots, x_n^{(i)})$ un vector de n dimensiones. El objetivo del FCM es encontrar los centros de los clústeres ν_c (centroides) que minimizen la función de disimilitud. La función de disimilitud mide la distancia entre un dato $x^{(i)}$ y el centroide ν_c: $d_{ic}^2 = ||x^{(i)} - \nu_c||_A^2$ siendo $||.||_A$ una norma inducida en $\mathcal{R}^n$ con A una matriz definida positiva $(n \times n)$. La función objetivo es una suma ponderada de las disimilitudes dentro de cada cluster:

$$J(U, \nu) = \sum_{i=1}^{m} \sum_{c=1}^{C} (\mu_c(x^{(i)}))^b d_{ic}^2 \tag{1}$$

Siendo U la partición difusa y b el parámetro llamado "grado de fuzzificación". En el proceso de minimización los centros van moviéndose para encontrar la mejor posición de tal forma que la multiplicación de las pertenencias $\mu_c(x^{(i)})$ por las distancias d_{ic}^2 acabe siendo mínima. La optimización aproximada de está basada en la iteración de las condiciones necesarias que fueron derivadas por Bezdek.

3. El problema de las particiones difusas

Muchos autores han descrito los problemas que generan las particiones difusas. En esta sección vamos a comentar un ejemplo que fué propuesto por Pal et.



(a)

	Datos		FCM	
Punto	x	y	$\mu_1(x^{(i)})$	$\mu_2(x^{(i)})$
1	-5,	0	0.9583	0.0417
2	-3.34	1.67	0.9755	0.0245
3	-3.34	0	0.9756	0.0244
4	-3.34	-1.67	0.9318	0.0682
5	-1.67	0	0.9541	0.0459
6	1.67	0	0.0456	0.9544
7	3.34	1.67	0.0245	0.9755
8	3.34	0	0.0245	0.9755
9	3.34	-1.67	0.0683	0.9317
10	5	0	0.0419	0.9581
11	0	0	0.5002	0.4998
12	0	10	0.5000	0.5000

(b)

Figura 1. (a) Puntos representando el conjunto de datos del ejemplo. (b) Conjunto de datos y pertenencias difusas obtenidas por el algoritmo FCM

al. [9]. Si tenemos el conjunto de datos $\{x^{(1)}, x^{(2)}, \cdots .., x^{(11)}, x^{(12)}\}$ de la Figura 1.

Podemos observar que en el conjunto de datos hay claramente dos clústeres de cinco puntos cada uno, pero $x^{(11)}$ y $x^{(12)}$son dos puntos difíciles de asignar a cualquiera de los clústeres. Usualmente $x^{(11)}$ se conoce como inlier (puente) ya que puede pertenecer perfectamente a cualquiera de los dos clústeres, sin embargo $x^{(12)}$ es un outlier porque difícilmente pertenece a ninguno de los dos clústeres. La presencia de estos outliers e inliers afecta a la posición final de los centroides. De hecho, si eliminamos $x^{(11)}$ y $x^{(12)}$ la posición de los centroides cambia, sin embargo el valor de las pertenencias de los elementos a los clústeres permanece prácticamente inalterada. Los datos $x^{(11)}$ y $x^{(12)}$ tienen sus valores de pertenencia alrededor del 0,5 en cada cluster debido a que su distancia es equidistante a los centroides, sin embargo $x^{(12)}$ está mucho más lejos de los clústeres que $x^{(11)}$. Este problema (que los dos puntos sean equivalentes en cuanto a sus pertenencias) esta causado por la restricción de que la suma de las pertenencias debe ser uno. Por lo tanto obtener una partición en la que sea posible diferenciar entre outliers e inliers es un reto interesante.

4. Propuesta de clustering intervalo-valorado

En nuestra propuesta, después del proceso de clustering, queremos obtener una partición intervalo valorada, que satisfaga la definición 4. Vamos a demandar dos condiciones:

1. La amplitud de los intervalos de los elementos que están dentro de los clústeres debe ser pequeña.

2. La amplitud de los elementos que están lejos de todos los clústeres debe ser grande.

Teniendo en cuenta estas consideraciones, y que la partición deberá representar los grupos inherentes en los datos, proponemos esta nueva función objetivo (Ecuación 2) en la cual las pertenencias de los elementos a los clústeres son intervalos.

$$J(U,\nu) = \sum_{i=1}^{m}\sum_{c=1}^{C} \underline{\mu_c}(x^{(i)})d_{ic}^2 + \sum_{i=1}^{m}\sum_{c=1}^{C} K\frac{\overline{\mu_c}(x^{(i)}) - \underline{\mu_c}(x^{(i)})}{min_{1\leq c\leq C}\{d_{ic}^2\}} \tag{2}$$

El primer termino de la función objetivo es idéntico al del clásico FCM, salvo que el valor de las pertenencias es el extremo inferior del intervalo. Este termino se encarga obtener una partición que represente correctamente los clústeres presentes en los datos (al menos tan bien como el FCM). El segundo termino se encarga de la amplitud de los intervalos. Si la distancia de un elemento al centroide de alguno de los clústeres es pequeña entonces la amplitud del intervalo debe ser pequeña, ya que estamos en un proceso de minimización. Por el contrario, si la distancia de un elemento a todos los clústeres es grande, entonces la amplitud de su intervalo no hace falta que sea tan pequeña.

Hay varias restricciones en el proceso de optimización. Las primeras son debidas a que debemos obtener una partición intervalo valorada, en la cual la suma de los extremos inferiores debe ser uno. Existe una restricción por cada elemento del conjunto de datos.

$$\sum_{c=1}^{C} \underline{\mu_c}(x^{(i)}) = 1 \tag{3}$$

Además las pertenencias deben ser intervalos, es decir, el extremo inferior debe ser menor o igual que el extremo superior, y las dos deben estar en el intervalo unidad. Existe una restricción por todas pertenencias de cada elemento a todos los clústeres.

$$0 \leq \underline{\mu_c}(x^{(i)}) \leq \overline{\mu_c}(x^{(i)}) \leq 1 \tag{4}$$

Los centroides de los clústeres se calculan utilizando extremos inferiores de los intervalos:

$$\nu_c = \frac{\sum_{i=1}^{m} \underline{\mu_c}(x^{(i)})^b x^{(i)}}{\sum_{i=1}^{m} \underline{\mu_c}(x^{(i)})^b} \tag{5}$$

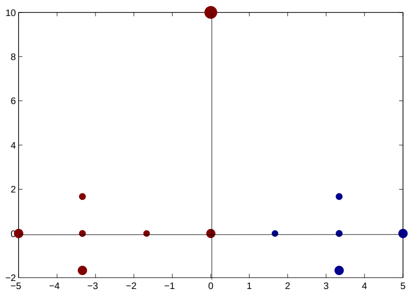
la distancia d_{ic}^2 entre el elemento i-esimo y el cluster c se calcula utilizando la siguiente expresión (similar al FCM): $d_{ic}^2 = ||x^{(i)} - \nu_c||_A^2$. El parámetro K en el segundo termino de la función objetivo (Ecuación 2) se utiliza para controlar la amplitud de los intervalos en la partición final. Si el valor de K es grande, debido a que es un proceso de minimización, dicho segundo término tendrá mucho peso y por tanto las amplitudes serán pequeñas en la partición final. Por el contrario si K es bajo tendremos una partición con amplitudes grandes.

Resolver este problema de optimización no es sencillo y con un gran número de datos se vuelve computacionalmente complejo. El número de restricciones es realmente grande $(m * c + m)$, y crece linealmente con el número de ejemplos y clústeres a identificar. Hemos utilizado la función `fmincon` implementada en la toolbox de optimización de Matlab. Esta función resuelve problemas de optimización no lineal con restricciones basado en el método de puntos interiores.

Queda para un futuro derivar la función objetivo y encontrar las condiciones necesarias y obtener un método de optimización aproximado iterativo para acelerar el proceso de optimización similar al del FCM.

5. Experimentos

En esta sección presentamos los resultados obtenidos con nuestra propuesta. Primero aplicamos nuestro método en los datos de la Figura 1 (sección 3). Utilizamos la distancia euclidiana, los parametros $b = 1$, $K = 1$, $C = 2$ y la partición inicial es aleatoria. En la Figura 1 mostramos las pertenencias intervalo valoradas obtenidas. Observamos que los intervalos de pertenencia de $\{x^{(1)}, x^{(2)}, \cdots .., x^{(10)}\}$ son altos y tienen amplitudes pequeñas al cluster que pertenencen y pertenencias bajas y amplitudes pequeñas al cluster al que no pertenecen. El extremo inferior de los datos $x^{(11)}$ y $x^{(12)}$ es 0,5 en cada cluster y la amplitud de esos intervalos es mayor que las del resto de elementos. En la Figura 1(a) el área de los puntos es proporcional a la amplitud de los intervalos de pertenencia al cluster 2 (cuarta y quinta columna del Cuadro 1). El color de los puntos depende del cluster al que el elemento ha sido asignado. Para decidir a que cluster pertenece cada elemento utilizamos el siguiente orden admisible para comparar los intervalos de pertenencia $[a, b] \preceq_{L1} [c, d] \Leftrightarrow a < c$ o $(a = c \; y \; b \leq d)$.



(a)

	Cluster 1		Cluster 2	
Pt.	$\underline{\mu_1}(x^{(i)})$	$\overline{\mu_1}(x^{(i)})$	$\underline{\mu_2}(x^{(i)})$	$\overline{\mu_2}(x^{(i)})$
1	0.9427	0.9690	0.0573	0.2600
2	0.9377	0.9640	0.0623	0.1719
3	0.9377	0.9641	0.0623	0.1717
4	0.9245	0.9581	0.0755	0.2756
5	0.9083	0.9431	0.0917	0.1910
6	0.0917	0.1266	0.9083	1.0
7	0.0623	0.0887	0.9377	1.0
8	0.0623	0.0886	0.9377	1.0
9	0.0755	0.1091	0.9245	1.0
10	0.0573	0.0836	0.9427	1.0
11	0.5000	0.5997	0.5000	0.6939
12	0.5000	0.6311	0.5000	0.8762

(b)

Cuadro 1. (a) Gráfico de las amplitudes de los intervalos en la partición final. (b) Pertenencias intervalo valoradas obtenidas por el método propuesto IV-FCM

Por lo tanto en este conjunto de datos de ejemplo comprobamos que hemos conseguido el objetivo. Podemos identificar los otliers y los inliers. El extremo inferior de sus intervalos de pertenencia es 0,5 en ambos casos (inliers y outliers), sin embargo la amplitud de los outliers es mayor que las de los inliers. Podríamos calcular un umbral y decidir que los datos cuya amplitud sea mayor que ese umbral son outliers.

En el siguiente experimento utilizamos un conjunto de datos más complicado que fue propuesto por Chatzis [7] y ha sido utilizado en varios trabajos para comparar algoritmos de clustering. Es un conjunto de datos sintético generado por tres distribuciones gaussianas (100 puntos cada una): $f_a = \mathcal{N}\left(\begin{bmatrix}0\\3\end{bmatrix}, \begin{bmatrix}2 & 0{,}5\\0{,}5 & 0{,}5\end{bmatrix}\right)$, $f_b = \mathcal{N}\left(\begin{bmatrix}3\\0\end{bmatrix}, \begin{bmatrix}1 & 0\\0 & 0{,}1\end{bmatrix}\right)$, $f_c = \mathcal{N}\left(\begin{bmatrix}-3\\0\end{bmatrix}, \begin{bmatrix}2 & -0{,}5\\-0{,}5 & 0{,}5\end{bmatrix}\right)$

Además se añaden 100 puntos, ruido y outliers, que son generados por otra distribución uniforme en el intervalo $[-10, 10]$ (ver Figura 2).

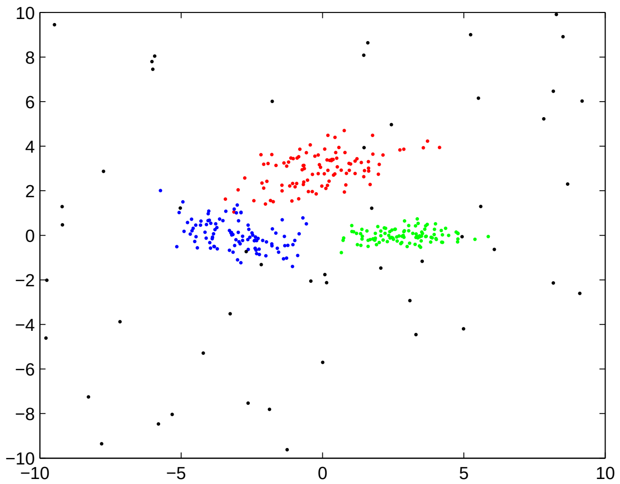


Figura 2. Conjunto de datos

Aplicamos nuestro método utilizando la distancia euclídea, con los parámetros $b = 1$, $K = 1$, $C = 3$ y con la partición inicial aleatoria. En la figura 3 mostramos los resultados en donde el área de los puntos es proporcional a la amplitud de los intervalos. Las propiedades que demandamos a la partición se cumplen, ya que los outliers tienen una amplitud más grande que los puntos que son cercanos a los centroides de los clústeres.

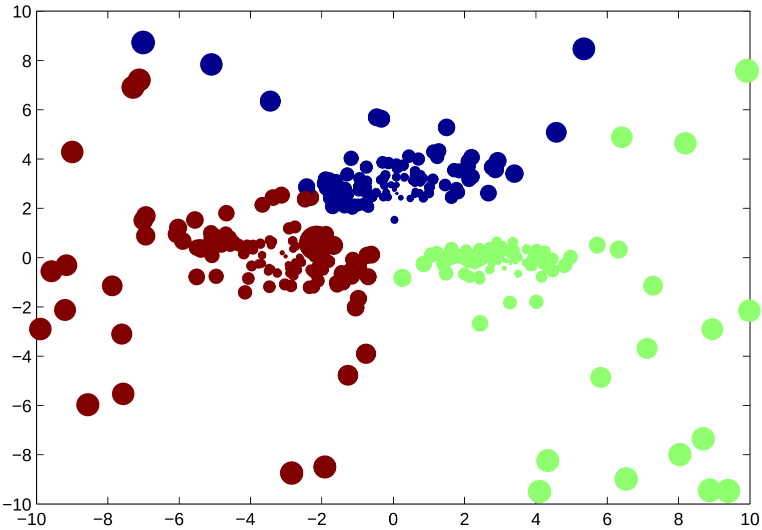


Figura 3. Clústeres y amplitud de los intervalos en el conjunto de datos sintéticos.

Agradecimientos

Este trabajo está parcialmente financiado por el proyecto TIN2013-40765-P.

6. Conclusiones y trabajos futuros

En este trabajo hemos propuesto una nueva función objetivo para obtener una partición intervalo valorada difusa en un proceso de clustering. Representamos la incertidumbre en la pertenencia de los elementos a los clústeres con la amplitud de los intervalos, de tal forma que somos capaces de detectar de forma sencilla los outliers. Como trabajos futuros nos planteamos comparar nuestro método con otros algoritmos de clustering que también sirvan para detectar outliers. Además nos planteamos derivar la función objetivo y encontrar las condiciones necesarias para obtener un algoritmo de optimización aproximado que sea iterativo y así reducir la complejidad computacional cuando hay muchos datos. Otras dos facetas que se pueden investigar es el efecto de utilizar diferentes particiones intervalares y añadir términos a la función objetivo similares a las extensiones del algoritmo FCM existentes.

Referencias

1. Bezdek, J.C.: Pattern Recognition With Fuzzy Objective Function Algorithms. New York: Plenum (1981).

2. Bustince, H., Barrenechea, E., Pagola, M.: Generation of interval-valued fuzzy and Atanassov's intuitionistic fuzzy connectives from fuzzy connectives and from K_α operators: laws for conjunctions and disjunctions, amplitude, *International Journal of Intelligent Systems*, 23(6): 680–714, 2008.
3. H. Bustince, H., Fernandez, J., Kolesárová, A., Mesiar,R.: Generation of linear orders for intervals by means of aggregation functions, *Fuzzy Sets and Systems*, 2012.
4. Bustince, H., Fernandez, J., Hagras, H., Herrera, F., Pagola, M., Barrenechea, E.: Interval Type-2 Fuzzy Sets are generalization of IVFSs: Towards a Wider view on their relationship, in Press *IEEE Transactions on Fuzzy Systems* DOI: 10.1109/TFUZZ.2014.2362149.
5. Hwang, c., Rhee F.C.: Uncertain fuzzy clustering: interval type-2 fuzzy approach to C-means, *IEEE Trans. Fuzzy Systems*, 15:107–120, 2007.
6. Min, J.H., Shim, E.A., Rhee, F.C.H.: An interval type-2 fuzzy PCM algorithm for pattern recognition, *Proceedings of the 18th International Conference on Fuzzy Systems, FUZZ-IEEE'09*, 480–483, 2009.
7. Chatzis, S.P., Tsechpenakis, G.:A possibilistic clustering approach toward generative mixture models, *Pattern Recognition* 45:1819–1825, 2012.
additive fuzzy systems, *IEEE Transactions on Fuzzy Systems*, 13(3): 303–316, 2005.
8. Ichihashi, H., Honda, K., Tani, N.: Gaussian mixture pdf approximation and fuzzy c-means clustering with entropy regularization, In Proceedings of the *Fourth Asian Fuzzy System Symposium*, 217–221, 2000.
9. Pal, N., Pal, K., Keller, J., Bezdek, J.: A possibilistic fuzzy c-means clustering algorithm, *IEEE Transactions on Fuzzy Systems*,13(4):517–530, 2005.
10. Krishnapuram, R., Keller, J.M.: A Possibilistic Approach to Clustering, *IEEE Transactions on Fuzzy Systems*, 1(2):98-110, 1993.
11. Mendel, J. M.; John, R. I., Liu, F.: Interval type-2 fuzzy logic systems made simple, *IEEE Transactions on Fuzzy Systems*, 14(6): 808–821, 2006.
12. Montero, J., Gómez, D. Bustince, H.: On the relevance of some families of fuzzy sets, ,*Fuzzy Sets and Systems*, 158(22): 2429–2442, 2007.
13. Pagola, M., Barrenechea, E., Aránzazu, J., Paternain, D., Bustince, H.: Clustering Based on a Mixture of Fuzzy Models Approach, *Information Processing and Management of Uncertainty in Knowledge-Based Systems Communications in Computer and Information Science*, 443:475-484, 2014.
14. Torra, V., Miyamoto, S.: A definition of I-fuzzy partitions, *Soft Computing*, 15:363–369, 2011.
15. Lu, W., Zhang, L., Liu X., Yang J., Pedrycz W.: A Human-Computer Cooperation Fuzzy c-Means Clustering with Interval-Valued Weights, *International Journal of Intelligent Systems*,30:81–98, 2015.
16. Ji Z., Xia Y., Sun Q., Cao G.: Interval-valued possibilistic fuzzy C-means clustering algorithm, *Fuzzy Sets and Systems*, 253(16):138–156, 2014.
17. Xu, Z.S., Yager, R.R.: Some geometric aggregation operators based on intuitionistic fuzzy sets, *International Journal of General Systems*, 35: 417–433, 2006.