ISSN 1870-4069

Mejoramiento del agrupamiento de datos mezclados e incompletos mediante algoritmos bioinspirados

Claudia C. Tusell-Rey[1], Yenny Villuendas-Rey[2], Oscar Camacho-Nieto[2], Viridiana Salinas-García[2]

[1] Instituto Politécnico Nacional, Centro de Investigación en Computación, México

[2] Instituto Politécnico Nacional, Centro de Innovación y Desarrollo Tecnológico en Cómputo, México

clautusellrey2014@gmail.com, {yvilluendasr, ocamacho, vsalinasg}@ipn.mx

Resumen. Mejorar los resultados del agrupamiento de datos mixtos (numéricos y categóricos) y con ausencias de información es fundamental para reconocer patrones y tomar de decisiones en diversos ámbitos. Los algoritmos de agrupamiento tradicionales a menudo tienen problemas con tipos de datos heterogéneos e información incompleta, lo que genera grupos subóptimao y conocimientos potencialmente sesgados. Al abordar estos desafíos, las técnicas avanzadas como los algoritmos bioinspirados, pueden proporcionar grupos más precisos y completos. Recientemente se propuso un método que mejora los resultados obtenidos mediante algoritmos de agrupamiento: el PAntSA; pero este sólo fue diseñado y probado para datos numéricos. Por este motivo, este trabajo analiza la influencia de la aplicación del PAntSA en el rendimiento de algoritmos de agrupamiento mezclados e incompletos. Para ello se comparan los resultados de diferentes algoritmos antes y después de aplicar el PAntSA. El análisis estadístico de los resultados proporciona evidencia experimental que respalda que el algoritmo PAntSA mejora la calidad de los grupos obtenidos mediante métodos tradicionales de agrupamiento de datos mixtos e incompletos.

Palabras clave: Agrupamiento bio-inspirado, datos mezclados e incompletos, PAntSA.

Bioinspired Based Improvement of Mixed and Incomplete Data Clustering

Abstract. Enhancing the results of data clustering for mixed (numeric and categorical) and missing data is paramount for robust pattern recognition and decision-making in various fields. Traditional clustering algorithms often struggle with heterogeneous data types and incomplete information, leading to

suboptimal groupings and potentially biased insights. By addressing these challenges, advanced techniques, such as bioinspired algorithms, can provide more accurate and comprehensive clusterings. Recently, a method was proposed that improves the results obtained by clustering algorithms, PAntSA; but this was only designed and tested for numerical data. For this reason, this work analyzes the influence of applying PAntSA on the performance of mixed and incomplete clustering algorithms. To do this, the results of different algorithms are compared before and after applying PAntSA. The statistical analysis of the results provides experimental evidence that supports that the PAntSA algorithm improves the quality of the groups obtained by traditional mixed and incomplete data clustering methods.

Keywords: Bioinspired clustering, mixed and incomplete data, PAntSA.

1. Introducción

El proceso de agrupar un conjunto de objetos físicos o abstractos dentro de clases con objetos similares se denomina agrupamiento. En este proceso se toma una colección dada de datos no etiquetados y se crea un conjunto de grupos de tal manera que los objetos que pertenecen a un grupo sean homogéneos entre sí [1], buscando además que la heterogeneidad entre los distintos grupos sea lo más elevada posible. Esta técnica ha adquirido gran relevancia en los últimos tiempos debido a su aplicación práctica en la solución exitosa de disímiles problemas de la vida real como: el reconocimiento del habla, la segmentación de imágenes y visión por computadora, la recuperación de información y minería de textos, en biología computacional para el análisis de ADN y muchas otras aplicaciones.

La mayoría de los algoritmos de agrupamiento se han diseñado para trabajar sólo con datos numéricos o con datos categóricos, mientras que, en una gran cantidad de ocasiones, es necesario trabajar con datos mezclados, es decir, con atributos de distintos tipos como: numéricos, binarios, discretos y categóricos. También en muchas ocasiones no es posible conocer el valor de un determinado atributo, por lo que se requiere además desarrollar algoritmos para agrupar datos incompletos [2].

El agrupamiento de datos mezclados e incompletos (DMI) ha sido abordado tradicionalmente siguiendo paradigmas clásicos como el jerárquico y particional, aunque también han aparecido propuestas bio-inspiradas que han tenido también un buen desempeño [3-5]. Por otra parte, el agrupamiento obtenido por un algoritmo dado puede ser mejorado mediante otro algoritmo utilizando un índice de validación interno. Empleando esta idea fue publicado el algoritmo bio-inspirado PAntSA [6] (basado en un árbol de hormigas [7,8]) el cual toma los resultados obtenidos por un algoritmo de agrupamiento previo e intenta perfeccionarlos utilizando el índice de la Silueta [9] y la definición de una atracción entre grupos. PAntSA mejora la calidad de los resultados obtenidos por algoritmos de agrupamiento en datos numéricos, particularmente en la clasificación de documentos; sin embargo, hasta donde sabemos, no existe un estudio de la influencia de PAntSA en el agrupamiento de DMI. Por este motivo, en este trabajo nos proponemos analizar la influencia del PAntSA en el mejoramiento de los resultados de los algoritmos de agrupamiento de DMI.

A partir de aquí el resto del trabajo está organizado como sigue: En la sección 2 mostramos algunos trabajos relacionados con el agrupamiento de DMI. En la sección 3 describimos el algoritmo PAntSA empleado para mejorar los grupos obtenidos por otros algoritmos. En la sección 4 presentamos un análisis experimental sobre la influencia del PAntSA en el agrupamiento de DMI y los resultados obtenidos. Finalmente, en la sección 5 se ofrecen las conclusiones obtenidas.

2. Trabajos relacionados

Los algoritmos de agrupamiento existentes para DMI, en su mayoría, son producto de extensiones realizadas a métodos para el manejo de tipos de datos homogéneos (numéricos o no numéricos), estos pueden dividirse en varias categorías según el procedimiento que utilizan para agrupar los objetos.

2.1. Algoritmos particionales

La estrategia empleada por los algoritmos particionales para encontrar los grupos es ir reubicando iterativamente objetos entre subconjuntos. En 1967 J. MacQueen propuso un algoritmo clásico que se ha denominado *k-means* el cual se considera arquetipo del modelo particional [10]. El algoritmo *k-means* para su funcionamiento tiene que conocer un número *k*, que es la cantidad de grupos que se desean obtener. La idea es ubicar *k* centroides y agrupar los objetos por su centroide más cercano según una función de distancia definida a priori. Iterativamente, se van actualizando los centroides como el promedio de los objetos que pertenecen a cada grupo, hasta que los centroides dejen de cambiar.

A pesar de su poca complejidad temporal, estos algoritmos presentan muchas desventajas: por ejemplo, los resultados finales dependen de la inicialización de los centroides. Además, son inválidos ante objetos con atributos categóricos por la necesidad de calcular el promedio y resultan inadecuados para detectar grupos no convexos y *outliers* (objetos ruidosos).

Unas de las primeras extensiones realizadas al *k-means* para tratar con DMI fue *k-Prototypes* publicado por Huang en 1997 [11]. Este autor basa su propuesta en la definición de una nueva función se disimilitud entre los objetos, de forma tal que se permite tratar las descripciones mezcladas.

Sea o un objeto, c_i el centro del i-ésimo grupo y X_p el p-ésimo atributo. La disimilitud entre el objeto y el centro se presenta a continuación:

$$d(o, c_i) = \sum_{p \in Rn} (X_p(o) - X_p(c_i))^2 + \gamma_i \sum_{r \in Rc} \Gamma(X_r(o), X_r(c_i)), \quad (1)$$

donde: γi es un parámetro del algoritmo, Rn es el conjunto de rasgos numéricos, Rc el conjunto de rasgos categóricos, y $\Gamma(X_r(O_j), X_r(c_i))$ es una función de disimilitud que es igual a cero si $X_r(O_j) = X_r(c_i)$ y uno en otro caso. Además, realiza una modificación en la forma de obtener los centros de los grupos tomando como valores la media de los atributos numéricos, y la moda de los atributos categóricos.

También Ahmad y Dey propusieron otra modificación al *k-means* [12]. Las modificaciones realizadas consisten en la actualización de la función de disimilitud,

teniendo en cuenta la contribución de cada atributo a cada grupo. Este algoritmo no presenta nombre, solo se enuncia un pseudocódigo bajo el título *modified_kmean_subspace_clustering*, por lo cual decidimos a los efectos de este trabajo, referirnos al algoritmo como AD2011.

2.2. Algoritmos jerárquicos

Los algoritmos jerárquicos, como su nombre indica, construyen una jerarquía de agrupamientos, uniendo o dividiendo los grupos de acuerdo con una cierta función de similitud/disimilitud entre los grupos. En otras palabras, construyen un árbol de grupos llamado dendograma. Tal enfoque permite estudiar los datos con diferentes niveles de granularidad.

Los algoritmos de agrupamiento jerárquicos se categorizan en aglomerativos (*bottom-up*) y divisivos (*top-down*). Un agrupamiento aglomerativo, generalmente, comienza con grupos unitarios (*singleton clusters*) y, recursivamente, une dos o más grupos apropiados. El proceso continúa hasta que se alcanza algún criterio de parada (frecuentemente el número k de grupos).

Entre las ventajas de los algoritmos de agrupamiento jerárquicos se puede mencionar la flexibilidad con respecto al nivel de granularidad, son fáciles de manejar y son aplicables a cualquier tipo de atributo. Entre las desventajas se encuentran el no mejoramiento de los grupos que han sido construidos por la no consideración de los objetos ya asignados y la sensibilidad al ruido. Además, el costo computacional para la mayoría de estos algoritmos es también como mínimo de O(m^2), donde *m* es el número de objetos, lo que limita su aplicación a grandes conjuntos de datos.

En 1999, Reyes-González y Ruiz-Shulcloper propusieron un nuevo algoritmo aglomerativo para DMI [13]. El algoritmo (al que denominaremos AERE) en cada paso forma un nuevo nivel, hasta que todos los objetos se encuentren en el mismo nivel. Un nivel está definido por la cantidad de grupos presentes en el nivel, el valor de β_0 (similitud máxima entre dos grupos del mismo nivel), y el conjunto de particiones posibles. Como elemento distintivo se tiene que es determinista (siempre obtiene la misma solución), y que cada grupo esté formado por elementos que se encuentran en la misma componente β_0 conexa en un grafo de máxima similitud. Finalmente, el algoritmo devuelve todas las estructuraciones posibles (conjunto de particiones) para el nivel deseado *k* de la jerarquía formada.

También para el agrupamiento de DMI, en 2005, fue propuesto un algoritmo jerárquico denominado HIMIC (*HIerarchical MIxed type data Clustering algorithm*) [14]. El algoritmo se basa en el uso de una función de disimilitud entre dos grupos C_i, C_j que considera el conjunto de valores categóricos posibles D_p la cual viene dada por la siguiente ecuación:

$$d(C_i, C_j) = \sum_{p=1}^{n} S_p(C_i, C_j), \tag{2}$$

donde:

2	1	2	1	1	2	1	2	1	2

Fig. 1. Ejemplo de un individuo. En este caso, se tienen 10 objetos, agrupados en dos grupos. Cada posición codifica el grupo al que pertenece dicho objeto:

$$S_p(C_i, C_j) = \begin{cases} 1 - \left| \frac{1}{|C_i|} \sum_{o \in C_i} X_p(o) - \frac{1}{|C_j|} \sum_{o \in C_j} X_p(o) \right| & si\ p\ es\ numérico, \\ \sum_{l=1}^{|D_p|} \frac{|\{o \in C_i | X_p(o) = v_l\}|}{|C_i|} * \frac{|\{o \in C_j | X_p(o) = v_l\}|}{|C_j|} & si\ p\ es\ categórico. \end{cases}$$

Posteriormente se aplica un método aglomerativo tradicional, utilizando como criterio de parada la obtención del número deseado de grupos *k*.

2.3. Algoritmos basados en técnicas de optimización

El problema de agrupar datos puede verse como un problema de optimización que localiza los centroides óptimos de los grupos o encuentra la partición óptima de un conjunto de objetos. Por este motivo se han empleado con éxito diferentes técnicas de optimización que ayudan a encontrar la mejor solución o al menos una solución lo suficientemente buena para un problema en un espacio de búsqueda. En este sentido destacan las metaheurísticas como algoritmos aproximados que intentan resolver estos problemas, sacrificando la garantía de encontrar el óptimo a cambio de encontrar una "buena solución en un tiempo razonable", razón por la cual han sido utilizadas para el agrupamiento de grandes conjuntos de datos.

Se puede decir que una metaheurística es una estrategia de alto nivel que usa diferentes estrategias para explorar el espacio de búsqueda y por su fácil adaptación, simplicidad y eficiencia están entre los métodos aproximados más ampliamente usados. A continuación, describiremos tres propuestas de algoritmos de agrupamiento de DMI que se basan en metaheurísticas.

El algoritmo AKGA propuesto en [15] consiste en la utilización de un Algoritmo Genético (GA) para obtener grupos sin necesidad de aplicar exhaustivamente un algoritmo de agrupamiento. La aplicación de un GA permite salir de óptimos locales, y buscar por un óptimo global, sin un costo computacional demasiado elevado. Como esquema de representación, cada solución consiste en un individuo, representado por una cadena de tamaño igual a la cantidad de objetos, y donde cada elemento i-ésimo de la cadena representa el grupo al cual el i-ésimo objeto está asignado (ver Fig. 1).

En el caso de los centros de los grupos, los autores utilizan el esquema de Ahmad y Dey de 2011 [12]. También utilizan la función de disimilitud de los objetos a los centros de [12] como parte de la función de optimización del Algoritmo genético.

Una metaheurística que ha sido empleada con éxito en el agrupamiento de datos numéricos es ABC (*Artificial Bee Colony*) inspirada en el comportamiento natural de las abejas y propuesta por Karaboga [16] para optimización numérica. Una extensión a esta para agrupar DMI lo constituye BECA (*BEe based Clustering Algorithm*) [4]. Este algoritmo genera n agrupamientos iniciales de forma aleatoria que constituyen las

fuentes de alimento. Luego genera nuevas fuentes empleando la estrategia de mutación definida en [15] y utilizando como disimilitud la HEOM [17] que permite tratar con DMI. Para la evaluación de las fuentes de alimento emplea como función objetivo un índice de validación interno, en este caso el índice de Dunn [9] para obtener grupos compactos y bien separados. Finalmente, después de un número de iteraciones definido a priori se devuelve la fuente de alimento (agrupamiento) que optimizó la función objetivo.

2.4. Índices de validación internos

La Silueta (*Silhouette*) [9] es un índice de validación interno que combina dos elementos claves para la calidad de un agrupamiento dado: compacidad y separabilidad. Este índice calcula el promedio, para todos los grupos, del ancho de la silueta de sus puntos. Si *x* es un objeto del grupo c_i y n_i es el número de objetos en c_i, entonces la silueta de *x* se define por la siguiente ecuación:

$$S(x) = \frac{b(x) - a(x)}{max\{a(x), b(x)\}} \ , \tag{3}$$

donde a(x) es la distancia promedio a todos los otros objetos en c_i, y b(x) es el mínimo del promedio de las disimilitudes de *x* y los objetos en los otros grupos:

$$a(x) = \frac{1}{n_i - 1} \sum_{\substack{y \in c_i \\ y \neq x}} d(x, y) \ , \tag{4}$$

$$b(x) = \min_{\substack{h=1..k \\ h \neq i}} \left\{ \frac{1}{n_h} \sum_{y \in c_h} d(x, y) \right\} , \tag{5}$$

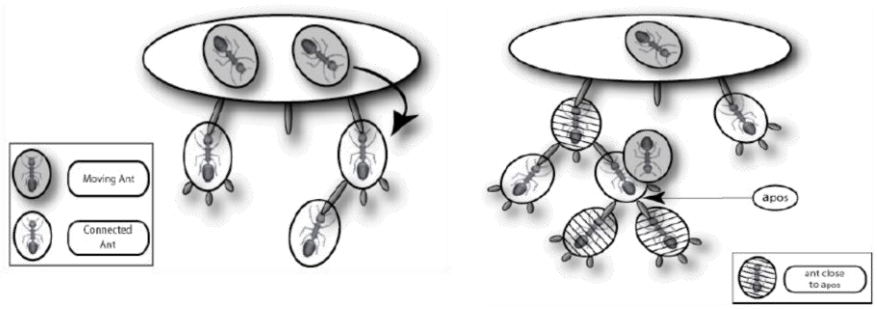
finalmente, la silueta global se define por:

$$S = \frac{1}{k} \sum_{i=1}^{k} \frac{1}{n_i} \sum_{x \in c_i} S(x) \ . \tag{6}$$

Para un objeto dado *x*, el ancho de su silueta varía entre -1 y 1. Si el resultado se encuentra cercano a -1, esto significa que dicho objeto es más similar, como promedio, a un grupo al cual él no pertenece. Por el contrario, si el valor es cercano a 1, quiere decir que la disimilitud promedio entre el objeto que se analiza y los objetos de su grupo, es significativamente menor que la disimilitud promedio con respecto a cualquier otro grupo. Mientras mayor sea la Silueta, más compactos y separados serán los grupos. Debido al rol importante que representa este índice en la validación interna de agrupamientos ha sido empleado en el algoritmo PAntSA para mejorar los resultados un algoritmo de agrupamiento dado.

3. Algoritmo PAntSA

El algoritmo PAntSA [6] está basado en el algoritmo AntTree, propuesto en [7,8]. Para el mejor entendimiento del PAntSA, explicaremos en detalle primeramente el



(a) (b)

Fig. 2. Movimiento de a_i sobre el árbol. (a) Representación del árbol formado por hormigas conectadas y hormigas en movimiento. (b) Posibilidades de movimiento de una hormiga que se encuentra sobre apos. (figuras tomadas y adaptadas de [6]).

funcionamiento del algoritmo AntTree. Este algoritmo es pionero en la aplicación de la modelación de la construcción de nidos por las hormigas, a problemas de Inteligencia Artificial. El AntTree se basa en modelar la habilidad de las hormigas para construir estructuras vivientes con sus cuerpos [8], para descubrir, de forma distribuida y no supervisada, una estructura arbórea que organice un conjunto de datos. Esta estructura jerárquica puede ser interpretada de varias formas: como una partición de los datos o como una estructuración jerárquica de los mismos [8].

El principio fundamental del AntTree es el siguiente: cada hormiga representa un nodo en el árbol que será construido (es decir, los objetos que serán agrupados) y existe una función de similitud entre dos objetos Sim(i,j).

Sobre la base de un nodo raíz ficticio a0, que representa el soporte sobre el que se va a construir el árbol, cada hormiga ai se va a ir fijando paulatinamente al nodo inicial, y sucesivamente a las hormigas ya fijadas, hasta que todas las hormigas están en la estructura. Todos los movimientos y fijaciones en la estructura van a depender del valor de Sim(i,j), y de una vecindad en donde se mueven las hormigas (ver Fig. 2).

Tomando como inspiración el AntTree y particularmente el AntSA, un algoritmo derivado de este, Ingaramo et al. proponen el PAntSA (*Partitional AntSA*) [6], específicamente para el mejoramiento de los resultados de cualquier algoritmo de agrupamiento de textos. En el PAntSA cada hormiga conectada representa un grupo, y las conectadas a esta forman una lista simple.

Así, cuando una hormiga a_i se incorpora al grupo de una hormiga a+ (hormiga más similar a a_i) denotado Ga+, la atracción se implementa simplemente adicionando la hormiga al grupo correspondiente. A continuación, se presenta el pseudocódigo del algoritmo (Fig. 3).

A pesar de que Ingaramo et al. probaron la eficiencia del PAntSA [6] y mostraron que es capaz de mejorar a algoritmos de agrupamiento para datos numéricos, hasta nuestro conocimiento no existe un estudio de si el mismo es capaz de mejorar los resultados de algoritmos de agrupamiento para datos mezclados, lo cual es objetivo de este trabajo. Es importante señalar que la complejidad del PAntSA para el peor caso

Algoritmo PAntSA

Entradas:
T: conjunto de objetos
A: algoritmo de agrupamiento
k: cantidad de grupos

Pasos:
1. Aplicar el algoritmo de agrupamiento *A*, a los objetos *T*.
2. Construir *k* filas (una para cada grupo obtenido en el paso 1) y ordenarlas decrecientemente de acuerdo al índice de la silueta.
3. Crear un grupo para la 1ra hormiga de cada fila, donde dicha hormiga es la representante del grupo, R_j (la hormiga corresponde al objeto con mayor silueta).
4. Unir las filas en una fila *F* tomando iterativamente la 1ra hormiga de cada fila no vacía, hasta que todas las filas estén vacías.
   Para cada hormiga a_i en *F*:
   a. Para cada hormiga R_j:
      i. Hallar el conjunto de hormigas en el grupo de R_j, denotadas por Ga_+
      ii. Calcular la atracción de cada hormiga $att(a_i, Ga_+) = \sum_{a \in Ga_+} Sim(a_i, a)/|Ga_+|$.
   Conectar a la hormiga a_i al grupo con $att(a_i, Ga_+)$ máximo.
5. Devolver los grupos obtenidos.

Fig. 3. Pseudocódigo del PAntSA.

(todos los grupos tienen la misma cantidad de objetos) está acotada por $O(k * (m/k + (m/k)^2))$ donde *m* es la cantidad de objetos, y *k* es la cantidad de grupos.

4. Resultados y discusión

4.1. Configuración experimental

Bancos de datos bajo estudio

Para evaluar el desempeño del PAntSA en datos mezclados se realizó una comparación experimental mediante algoritmos de agrupamiento de DMI reportados en la literatura, aplicando el PAntSA a los resultados obtenidos por cada uno de ellos.

Para los experimentos se utilizaron 10 bases de datos mezclados e incompletos del repositorio de la Universidad de California en Irvine (UCI) [18] (ver Tabla 1). La selección de este conjunto de datos descansa en que se trata de bases de datos etiquetadas.

Ello posibilitó contar con agrupamientos modelo (clases) contra los cuales medir la calidad de los grupos obtenidos por los algoritmos mediante dos índices de validación.

Tabla 1. Descripción de los bancos de datos utilizados.

No.	Banco de Datos	Atributos Categóricos	Atributos Numéricos	Clases	Valores perdidos	Instancias
1	Autos	10	16	7	si	205
2	Colic	15	7	2	si	368
3	dermatology	1	33	6	si	366
4	heart-c	7	6	5	si	303
5	hepatitis	13	6	2	si	155
6	Labor	6	8	2	si	57
7	lymph	15	3	4	no	148
8	sponge	44	0	3	si	76
9	tae	2	3	3	no	151
10	zoo	16	1	7	no	101

Índices de validación utilizados

El primer índice utilizado fue la Entropía, la cual mide el grado de desorden del agrupamiento modelo (AM) en los grupos obtenidos mediante la siguiente ecuación:

$$E = \sum_{k \in AE} \frac{|k|}{N} \left[\frac{1}{log(|AM|)} \sum_{m \in AM} \frac{n_k^m}{|k|} log \frac{n_k^m}{|k|} \right], \quad (7)$$

donde: AE representa el agrupamiento a evaluar, |k| es el total de objetos del grupo *k*, |AM| es el total de grupos del agrupamiento modelo y n_k^m es el total de objetos en el grupo *k* que pertenece al grupo de AM. Mientras menor sea la Entropía mejor será la calidad del agrupamiento.

El segundo índice de validación utilizado fue el Error del Agrupamiento (*Cluster Error*), otro de los índices de validación internos más empleados en comparaciones experimentales. Su funcionamiento se basa en minimizar los objetos que estén asignados a un grupo diferente del agrupamiento modelo (AM) en el agrupamiento a evaluar (AE) y su definición viene dada por:

$$CE = \sum_{k \in AE} \frac{1}{N} \min_{m \in AM} \{P_k^m\}, \quad (8)$$

donde P_k^m es el total de objetos en el grupo *k* que no pertenecen el grupo *m* de AM, mientras menor sea el Error mayor calidad tendrá el agrupamiento.

Algoritmos de agrupamiento evaluados y sus parámetros

En la comparación experimental se emplearon seis algoritmos que permiten el agrupamiento de DMI y se evaluaron los resultados obtenidos por estos antes y luego de aplicado el PAntSA [6]. Así se seleccionaron los métodos particionales kPrototypes [11] y AD2011 [12], el jerárquico HIMIC [14] y el método CEBMDC [19] el cual es un algoritmo que utiliza una combinación de otros métodos. Además, se utilizaron los metaheurísticos AGKA [15] y BECA [4].

Un aspecto importante en el diseño de los experimentos son los parámetros con los que se van a ejecutar los algoritmos (ver Tabla 2). Cabe señalar que el PAntSA no tiene parámetros más allá de la función de disimilitud entre objetos.

Tabla 2. Parámetros utilizados por cada algoritmo.

No.	Algoritmo	Parámetros
1	AD2011	Parámetro de contribución de los grupos: 20
2	AGKA	Número de generaciones: 10 Probabilidad de mutación: 0.05 Tamaño de la población: 10 Probabilidad de cruzamiento: 1
3	BECA	Cantidad de fuentes de alimento: 10 Cantidad de abejas exploradoras: 1 Número de generaciones: 10 Límite de las fuentes de alimento: 10
4	CEBMDC	Umbral de similitud: 0
5	HIMIC	-
6	k-prototypes	-

Como todos los algoritmos requieren conocer la cantidad de grupos a formar, el valor asignado a este parámetro va a coincidir, para cada base de datos, con la cantidad de clases. Con esto se toma las clases como agrupamiento modelo contra el cual evaluar los agrupamientos resultantes de aplicar los algoritmos.

El resto de los parámetros fueron elegidos en base a la existencia de estudios que la recomendasen determinados valores para un mejor desempeño. Además, a los parámetros comunes de los diferentes algoritmos se les suministró el mismo valor. Ello permitió lograr cierta homogeneidad y disminuir un posible desbalance en el desempeño de un algoritmo frente a otro por el empleo de valores diferentes para un mismo parámetro.

En el caso de la disimilitud se utilizó la función HEOM [17] para todos los algoritmos. La razón de su uso fue sus buenos resultados en el tratamiento de DMI.

4.2. Resultados obtenidos

Los experimentos se condujeron de la siguiente forma: Para cada base de datos, se aplicaron los diferentes algoritmos y se calculó la Entropía. Luego se aplicó el PAntSA [6] al resultado de cada algoritmo obteniéndose un nuevo agrupamiento y se calculó igualmente la Entropía a estos nuevos resultados.

En cada caso, se estableció como cantidad de grupos a la cantidad de clases de cada una de las bases de datos. Este procedimiento se repitió para el Error del Agrupamiento. En las figuras 4, 5 y 6 se muestran los resultados de cada algoritmo, antes y después de aplicar el PAntSA. Como puede observarse, en muchos casos la Entropía disminuye.

En la mayoría de los casos se obtiene una menor Entropía luego de aplicar el PAntSA. Sin embargo, para establecer si estas diferencias son o no significativas desde el punto de vista estadístico, se aplicó la prueba de Wilcoxon para dos muestras relacionadas. En la Tabla 3 se muestra la significación asintótica bilateral obtenida por la prueba.

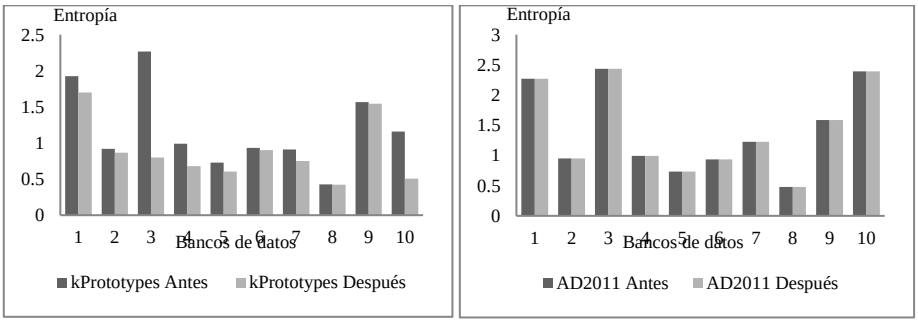


Fig. 4. Resultados de la Entropía antes y después de aplicado el PAntSA para los algoritmos particionales kPrototypes y AD2011.

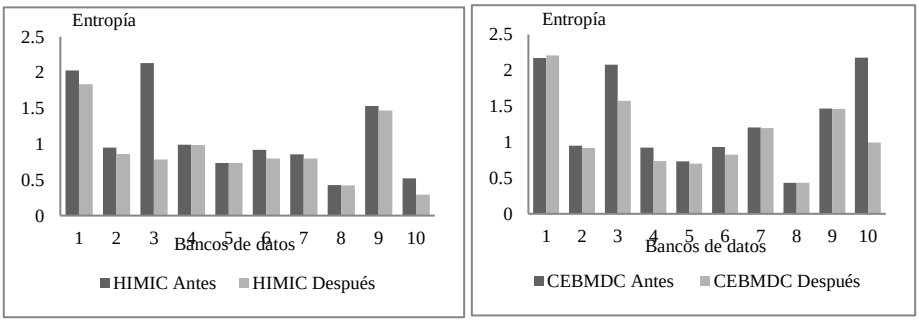


Fig. 5. Resultados de la Entropía antes y después de aplicado el PAntSA para HIMIC y CEBMDC.

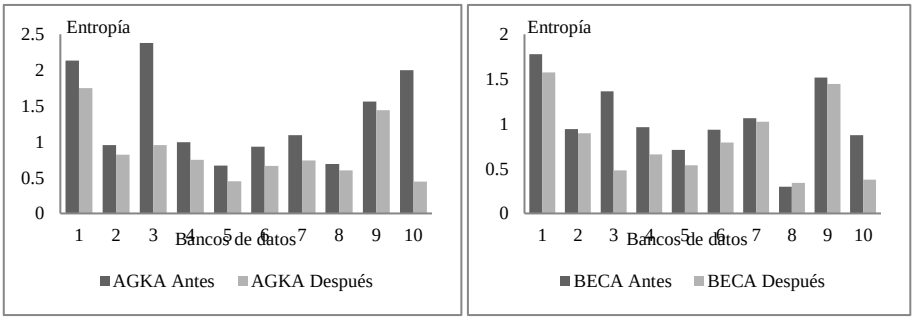


Fig. 6. Resultados de la Entropía antes y después de aplicado el PAntSA para los algoritmos metaheurísticos AGKA y BECA.

El test de Wilcoxon permite esclarecer si efectivamente el PAntSA mejora o no los resultados obtenidos por los métodos de agrupamiento. Para ello, se estableció un nivel de confianza del 95%. Así, para valores de probabilidad mayores que 0.05, se considera que no existen diferencias en la calidad de los grupos luego de aplicar el PAntSA. Para

Tabla 3. Significación asintótica bilateral de la prueba de Wilcoxon para la Entropía de los agrupamientos obtenidos por los algoritmos antes y después de aplicado el PAntSA.

Algoritmos antes y después de aplicar PAntSA	**kPrototypes**	**AD2011**	**HIMIC**	**CEBMDC**	**AGKA**	**BECA**
Significación asintótica	0.005	1.000	0.008	0.037	0.005	0.009
Decisión	☺	😐	☺	☺	☺	☺

Tabla 4. Significación asintótica bilateral de la prueba de Wilcoxon para el Error del Agrupamiento de los resultados obtenidos por los algoritmos antes y después de aplicado el PAntSA.

Algoritmos antes y después de aplicar PAntSA	**kPrototypes**	**AD2011**	**HIMIC**	**CEBMDC**	**AGKA**	**BECA**
Significación asintótica	0.018	0.998	0.028	0.043	0.008	0.018
Decisión	☺	😐	☺	☺	☺	☺

ello se escoge el símbolo 😐 para facilitar la comprensión. Sin embargo, para valores menores a 0.05, es necesario determinar si este método mejora o empeora los resultados obtenidos por el método de agrupamiento, utilizando los símbolos ☺ y ☹, respectivamente.

El símbolo ☺ significa que el PAntSA mejoró significativamente los resultados del algoritmo correspodiente, mientras que el símbolo 😐 significa que no se evidenciaron diferencias entre los resultados del algoritmo antes y después de aplicado el PAntSA. No se encontró evidencias que el PAntSA empeorara los resultados obtenidos. Como se puede apreciar el PAntSA es capaz de mejorar los resultados de los agrupamientos obtenidos por todos los algoritmos a excepción del método AD2011, el cual no evidenció diferencias significativas.

En el caso del Error del Agrupamiento, las figuras 7, 8, y 9 muestran los resultados de los algoritmos comparados antes y después de aplicar el PAntSA.

Como se puede apreciar, al igual que sucedió con la Entropía, en la mayoría de los casos el Error de Agrupamiento disminuye luego de aplicar el PAntSA. Una vez más fue necesario realizar la prueba de Wilcoxon para establecer si existen o no diferencias significativas entre cada algoritmo antes y luego de aplicado el PAntSA. Seguidamente se presentan los resultados de la prueba en la Tabla 4 tomando como nivel de confianza el valor 0.05.

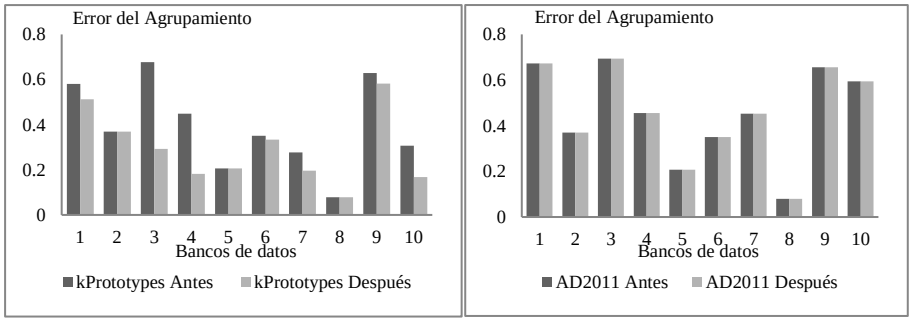


Fig. 7. Resultados del Error del Agrupamiento antes y después de aplicado el PAntSA para los algoritmos particionales.

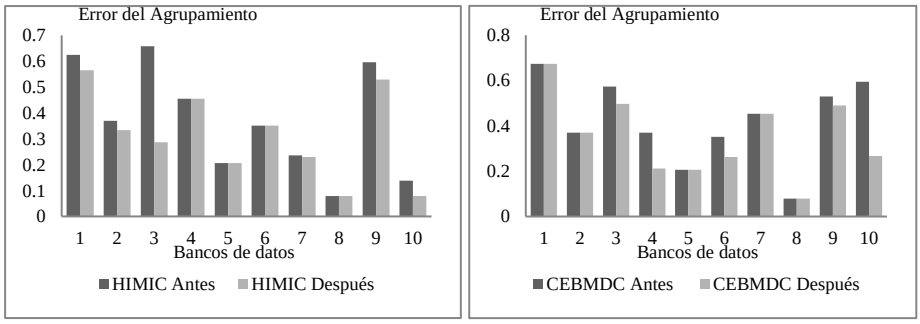


Fig. 8. Resultados del Error del Agrupamiento antes y después de aplicado el PAntSA para HIMIC y CEBMDC.

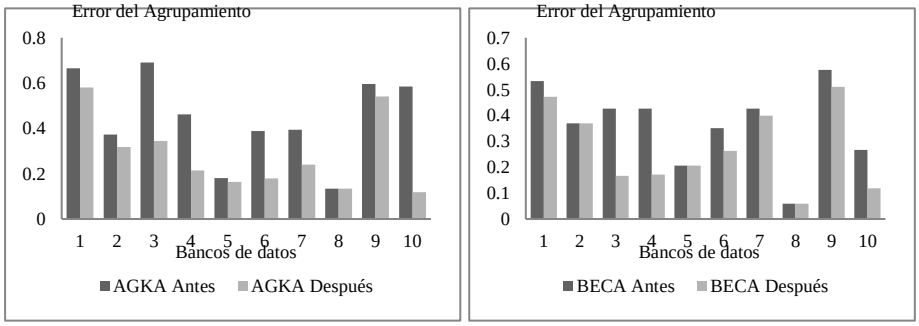


Fig. 9. Resultados del Error del Agrupamiento antes y después de aplicado el PAntSA para los algoritmos metaheurísticos AGKA y BECA.

El comportamiento fue similar a lo ocurrido con la Entropía, todos los algoritmos mostraron diferencias significativas favorables a la aplicación del PAntSA, solo AD2011 no evidenció diferencias dado por su significación de 0.998. De forma general

como se pudo comprobar en las experimentaciones analizadas, PAntSA mostró un buen nivel de efectividad en el mejoramiento de los resultados de algoritmos de agrupamiento de DMI. Solamente no se obtuvieron diferencias utilizando el algoritmo AD2011.

5. Conclusiones

La obtención de agrupamientos de elevada calidad en datos mezclados e incompletos reviste especial importancia. El estudio realizado permite aseverar que los resultados obtenidos por métodos de agrupamiento de diversa naturaleza (particionales, jerárquicos, bio-inspirados y otros) pueden ser refinados al aplicar estrategias de post-procesamiento.

El algoritmo PAntSA, en todos los casos mejoró o mantuvo la calidad de los grupos analizados, y en ningún caso su aplicación implicó un detrimento de la misma. Por otra parte, el uso de índices de validación internos, en este caso de la Silueta, abre nuevas líneas de investigación en cuanto a la calidad de los agrupamientos, pues se considera que además de las propiedades de compacidad y separabilidad que tiene en cuenta este índice, otras propiedades pueden ser utilizadas para refinar aún más los agrupamientos de datos mezclados e incompletos.

Las limitaciones de este estudio están dadas por la cantidad de datos y algoritmos estudiados, por lo que en el futuro se pretende realizar un estudio más extenso.

Referencias

1. Jain, A.K., Dubes, R.C.: Algorithms for Clustering Data. Prentice Hall (1988)
2. Ruiz-Shulcloper, J.: Pattern Recognition with Mixed and Incomplete Data. Pattern Recognition and Image Analysis, vol. 18, pp. 563–576 (2008). DOI: 10.1134/S1054 661808040044.
3. González-Patiño, D., Villuendas-Rey, Y., Saldaña-Pérez, M., Argüelles-Cruz, A.J.: A Novel Bioinspired Algorithm for Mixed and Incomplete breast Cancer Data Classification. International Journal of Environmental Research and Public Health, vol. 20, no. 4, pp. 3240 (2023). DOI: 10.3390/ijerph20043240.
4. Cabrera-Venegas, J.F, Chávez-Castilla, Y.: Clustering Mixed Data using An Artificial Bee Colony. In: II Cuba-Flanders Workshop on Machine Learning and Knowledge Discovery (CFWMLKD´11), Cuba (2011)
5. Errecalde, M., Ingaramo, D., Rosso, P.: A New AntTree-based Algorithm for Clustering Short-Text Corpora. JCS&T, vol. 10, no. 1, pp. 1–7 (2010)
6. Ingaramo, D., Errecalde, M., Rosso, P.: A General Bio-Inspired Method to Improve the Short-Text Clustering Task. In: International Conference on Intelligent Text Processing and Computational Linguistics, pp. 661–672 (2010). DOI: 10.1007/978-3-642-12116-6_56.
7. Azzag, H., Venturini, G.: A Clustering Model using Artificial Ants. Université François-Rabelais, Tours-France (2004)
8. Azzag, H., Monmarche, N., Slimane, M., Venturini, G., Guinot, C.: AntTree: A New Model for Clustering with Artificial Ants. In: CEC´03, IEEE Press, Australia, pp. 2642–2647 (2003). DOI: 10.1109/CEC.2003.1299421.

9. Brun, M., Sima, C., Hua, J., Lowey, J., Carroll, B., Suh, E., Dougherty, E.R.: Model-based Evaluation of Clustering Validation Measures. Pattern Recognition, vol. 40, no. 3, pp. 807–824 (2007). DOI: 10.1016/j.patcog.2006.06.026.
10. MacQueen, J.: Some Methods for Classification and Analysis of Multivariate Observations. In: Fifth Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, pp. 281–297 (1967)
11. Huang, Z.: Clustering Large Data Sets with Numeric and Categorical Values. In: Pacific - Asia Conference on Knowledge Discovery and Data Mining (1997)
12. Ahmad, A., Dey, L.: A K-Means Type Clustering Algorithm for Subspace Clustering of Mixed Numeric and Categorical Data. Pattern Recognition Letters, vol. 32, no. 7, pp. 1062–1069 (2011). DOI: 10.1016/j.patrec.2011.02.017.
13. Reyes-González, R., Ruiz-Shulcloper, J.: Un algoritmo de estructuración restringida de espacios. CIARP, Cuba (1999)
14. Ahmed, R.A., Borah, B., Bhattacharyya, D.K., Kalita, J.K.: HIMIC: A Hierarchical Mixed Type Data Clustering Algorithm. Department of Computer Science and Information (2005)
15. Roy, D.K., Sharma, L.K.: Genetic K-Means Clustering Algorithm for Mixed Numeric and Categorical Datasets. International Journal of Artificial Intelligence & Applications (IJAIA), vol. 1, no. 2, pp. 23–28 (2010). DOI: 10.5121/ijaia.2010.1203.
16. Karaboga, D.: An Idea Based on Honey Bee Swarm for Numerical Optimization. Technical Report-TR06. Computer Engineering Department, Erciyes University (2005)
17. Wilson, R.D., Martinez, T.R.: Improved Heterogeneous Distance Functions. Journal of Artificial Intelligence Research, vol. 6, pp. 1–34 (1997). DOI: 10.1613/jair.346.
18. He, Z., Xu, X., Deng, S.: Clustering Mixed Numeric and Categorical Data: A Cluster Ensemble Approach. http://arix.org/abs/cs/0509011 (2005)