ISSN 1870-4069

Análisis de rendimiento de particionamiento por medoides y medianas

María Beatriz Bernábe Loranca[1], Jonathan Ramos Flores[1], Rogelio González Velázquez[1], Gerardo Martínez Guzmán[1], Juan Carlos Duque Cardona[2]

[1] Benemérita Universidad Autónoma de Puebla, Facultad de Ciencias de la Computación, Puebla, México

[2] Universidad EAFIT, Departamento de Ciencias, Colombia

beatriz.bernabe@gmail.com

Resumen. Este trabajo consiste en reportar la experiencia computacional del algoritmo PAM (Particionamiento Alrededor de los Medoides) y de la P-Mediana cuando se contrastan sus propiedades para establecer si existe semejanza algorítmica. Entonces, bajo la conjetura de que ambos algoritmos son equivalentes en cuanto a la función objetivo y la configuración de los grupos que generan, el objetivo de este trabajo se centra en comparar los óptimos y centroides (medoides y medianas) para ambas propuestas. El experimento que permite apoyar el supuesto de equivalencia entre PAM y la P-Mediana se apoya del conjunto de unidades geográficas de la Zona Metropolitana del Valle de Toluca. Una vez que se procesaron los datos para distintos tamaños en los grupos, se registró el tiempo, la calidad de las soluciones y la estructura de los grupos. Se concluyó que PAM y la P-Mediana proporcionan resultados consistentes a la hipótesis planteada.

Palabras clave: Combinatorio, PAM, LINGO, óptimo, ramificación y acotamiento.

Medoid and Median Partitioning Performance Analysis

Abstract. This work consists of reporting the computational experience of the PAM (Partitioning Around Medoids) algorithm and the P-Median when contrasting their properties to establish whether algorithmic similarity exists. Then, under the conjecture that both algorithms are equivalent in terms of the objective function and the configuration of the groups they generate, the objective of this work focuses on comparing the optima and centroids (medoids and medians) for both proposals. The experiment that allows supporting the assumption of equivalence between PAM and P-Median is based on the set of geographic units of the Metropolitan Zone of Toluca Valley. Once the data were processed for different sizes in the groups, the time, the quality of the solutions

and the structure of the groups were recorded. It was concluded that PAM and P-median provide consistent results to the hypothesis.

Keywords: Combinatorial, PAM, LINGO, optimal, branch and bound.

1. Introducción

Es importante destacar la complejidad computacional que implica la solución del problema de Particionamiento [1]. Este problema es considerado como un problema combinatorio NP-duro que exige que su solución sea generada a partir de la utilización de métodos no exactos de optimización.

En términos generales, las metodologías para la solución de este problema pueden ser divididas en métodos exactos y aproximados [2]. Dado que PAM y P-mediada pueden ser resueltos bajo procedimientos exactos, este trabajo centra su atención en analizar su desempeño en tres direcciones: 1) equivalencia en el valor de la función objetivo, 2) comparación de los centroides y 3) costo computacional [3].

La evidencia empírica que se aporta se basa en los 469 AGEBs (Áreas Geoestadísticas Básicas) de la Zona Metropolitana del Valle de Toluca (ZMVT).

El trabajo se encuentra estructurado como sigue; la presente introducción como sección 1. La sección 2 aborda los preliminares de este manuscrito. La sección 3 se centra en los aspectos generales relacionados con el particionamiento y de k-medoides. En la sección 4 se expone el modelo de la P-Mediana y PAM. La sección 5 se ocupa de la experiencia computacional y finalmente se presentan las conclusiones.

2. Preliminares

En términos generales, el agrupamiento se divide en dos tipos:

a) Métodos de particionamiento, entendidos como algoritmos que dividen el conjunto de datos en k grupos, donde el número entero k debe ser especificado por el usuario. El algoritmo debe satisfacer las condiciones de partición clásica que se estudia en conjuntos discretos. Normalmente el usuario ejecuta el algoritmo para un rango de valores k y evaluar distintas pruebas.
b) Métodos jerárquicos, los cuales son algoritmos aglomerativos que producen una jerarquía completa de agrupaciones del conjunto de datos. Esta clase de algoritmos no se abordan en este artículo.

De acuerdo a la descripción anterior, estamos colocados en el primer tipo debido a la evidente pertenencia de PAM a la categoría de clasificación por particiones. Sin embargo, aunque la P-Mediana no puede ni debe ser etiquetado como un algoritmo de particionamiento, muchos autores han implementado la P-Mediana usando particionamiento cuando el algoritmo de ramificación y acotamiento que usan optimizadores comerciales, se ven limitados para resolver la P-Mediana.

Incluso, un problema de particionamiento geográfico se puede modelar como un problema de la P-Mediana.

Es importante señalar que el presente documento se ocupa de obtener particiones ejemplares y exactas ya que las aproximaciones metaheurísticas no se incluyen en este análisis debido a su naturaleza aleatoria. En este aspecto, es oportuno comentar la complejidad computacional del particionamiento. El problema de particionamiento consiste en hallar la partición $P \in P_k$ que maximiza la inercia inter-clases $B(P)$, es decir, maximizar la intensidad de la separación entre los centros de los grupos. La complejidad computacional de este problema es del tipo $NP - hard$ [2].

El tamaño combinatorio de este problema puede verse con el siguiente ejemplo. Si denotamos por $S(n,k)$ y B_n el número de particiones de Ω en k clases no vacías y el número total de particiones de Ω respectivamente, entonces por ejemplo $S(60,2) \approx 0.58 \times 10^{18}, S(60,5) \approx 0.72 \times 10^{40}, S(100,5) \approx 0.66 \times 10^{68}$, mientras que $B_{10} = 115975, B_{15} \approx 0.14 \times 10^{10}$ y $B_{40} \approx 0.16 \times 10^{36}$.

En un problema de particionamiento donde Ω tenga 100 objetos y el número de clases sea $k = 5$, si existiera sobre la Tierra un computador tan veloz que fuese capaz de calcular $B(P)$ para cada una de las particiones P de Ω en un tiempo de 10^{-10} segundos en busca de un máximo global, al computador le tomaría algo más de 2×10^{48} siglos en completar el análisis de todas las particiones del problema [2].

2.1. Particionamiento K-medias: inicios y generalidades

A pesar de que no se compara ni analiza a k-medias en este artículo, para exponer la situación en términos razonablemente amplios, se describe de breve manera a K-medias dada su popularidad, relevancia y características básicas que otros algoritmos han tomado y respetado pero que a la vez han mejorado para rebasar las debilidades.

Es decir, se han planteado propuestas similares, pero bajo distintas formas e hipótesis e incluso mejorando sus aspectos teóricos, algorítmicos, así como modificaciones del método (por ejemplo, particionamiento sobre medoides).

Con la publicación de Sokal 1963, el análisis clúster surge como una importante área de estudio [4]. Con Lerman 1970, se inicia la publicación de un compendio de libros y artículos que sentaron las bases de esta área de estudio, expuestos cronológicamente que además profundizan en la naturaleza matemática del problema del agrupamiento [5, 6, 7].

Por otra parte, la publicación de Hartigan 1975 impacta en el mundo académico por su estudio en aspectos computacionales de los algoritmos, lo cual implicó que los problemas y métodos básicos del agrupamiento se hicieron ampliamente conocidos por la comunidad científica [8].

El algoritmo k medias trata de minimizar el criterio clásico de distancia de cada una de las observaciones de la muestra a la media de su grupo al cuadrado. Dicho algoritmo presenta una serie de extensiones y generalizaciones entre las que destacan el caso borroso- o fuzzy-, el de máxima verosimilitud, y aquellos criterios basados en convexidad, entre otros.

Uno de los principales enfoques de las técnicas de agrupamiento se basa en el criterio de la suma de los cuadrados de la varianza conocido como k-medias. Entre las desventajas de K medias se tienen. 1) Encontrar el valor K es una tarea difícil, 2) No es efectivo cuando se usa con clúster global, 3) Si se seleccionan diferentes particiones iniciales, puede variar el resultado de los clústeres, 4) El algoritmo no maneja clústeres de diferente tamaño y densidad.

3. Particionamiento

Los algoritmos de partición han sido durante mucho tiempo algoritmos de clustering, clasificación o agrupamiento por particiones muy populares antes de la aparición de la minería de datos.

Informalmente, el particionamiento se entiende como sigue: Dado un conjunto D de n objetos en un espacio d-dimensional y un parámetro de entrada k, un algoritmo de partición organiza los objetos en k clústeres de tal manera que la desviación total de cada uno de ellos respecto a su centro de clúster o a una distribución de clústeres se minimiza. La desviación de un punto puede calcularse de forma diferente en distintos algoritmos y se denomina más comúnmente función de similitud.

3.1. K-medoids

Dentro de los algoritmos de particionamiento, destaca el particionamiento por medoides (medoids). A diferencia del algoritmo k-means, el método k-medoids utiliza el objeto más céntrico (medoids) de un clúster para ser el centro del clúster en lugar de tomar el valor medio de los objetos de un clúster. Por ello, el método k-medoids es menos sensible al ruido y a los valores atípicos. Sin embargo, esto implica un mayor costo computacional. El siguiente algoritmo describe los aspectos generales de los algoritmos k-medoides:

3.1.1 Algoritmo de reubicación iterativa (RI)

El algoritmo de reubicación iterativa generalizada se compone de los siguientes pasos:

Entrada: El número de clúster k, y una base de datos que contiene n objetos.

Salida: Un conjunto de k clústeres que minimiza una función de criterio E.

Método:

1) elegir arbitrariamente k centros/distribuciones como solución inicial;
2) repetir;
3) (re)calcular la pertenencia de los objetos según la solución actual;
4) actualizar algunos/todos los centros/distribuciones de los clústeres según a las nuevas pertenencias de los objetos;
5) hasta que (no haya cambios en E);

El estado de inicialización k-medoids y variantes es igual a RI en el sentido de la selección aleatoria de los k objetos son el centro de los conglomerados (grupos). De las propiedades de k-means, el clustering k-medoids toma el procedimiento de asignar un objeto a su centro más cercano, lo cual se resuelve en el paso 4 del algoritmo RI haciendo que el paso 3 sea redundante, por tanto, K-medoids dista tanto de RI como de k-medias en que como máximo se cambiará un centro en el paso 4 para cada iteración.

Este cambio de centro debe dar lugar a una disminución de la función criterio, que suele ser la función de error al cuadrado en la minimización de distancias entre objetos y centroides y utiliza implícitamente una medida de distancia [9].

Para desarrollar el paso 4, un algoritmo de k-medoides como PAM es considerado [3], entonces se itera a través de todos los k centros de clúster e intenta reemplazar cada uno de ellos con uno de los otros $(n-k)$ objetos. Para cada uno de estos reemplazos, si la función de error cuadrado E disminuye el reemplazo tendrá lugar haciendo que se produzca la siguiente iteración del algoritmo RI.

Sin embargo, si no se encuentra tal reemplazo después de pasar por todos los k clústeres, no habrá ningún cambio en E, entonces el algoritmo termina con un óptimo local. Como PAM intenta reemplazar cada uno de los k centros de clúster con uno de los $(n-k)$ objetos y cada uno de estos intentos resulta en $(n-k)$ operaciones para calcular E, la complejidad total de PAM en una iteración es $O(k(n-k)^2)$.

Para valores grandes de n, este cálculo se vuelve muy costoso [9]. Debido a su complejidad, PAM funciona eficazmente para conjuntos de datos pequeños, pero no se adapta bien a conjuntos de datos grandes. Esta afirmación justifica que se hayan elegido pruebas de tamaño pequeño para el análisis que se persigue en el presente trabajo.

Por otro lado, para tratar conjuntos de datos más grandes, otras estrategias deben ser desarrolladas y en un trabajo futuro se abordarán. Por ejemplo, el problema de grandes cantidades de datos en particionamiento por medoides, fue expuesto por Kaufman y Rousseeuw y desarrollaron un método basado en el muestreo, llamado CLARA (Clustering LARge Applications) [3].

En lugar de tomar en consideración todo el conjunto de datos, se elige una pequeña porción de los datos reales como representante de los mismos. Los medoides se eligen a partir de esta muestra utilizando PAM y la disimilitud media se calcula utilizando todo el conjunto de datos. Si se calcula un nuevo conjunto de medoides que da una disimilitud menor que la mejor solución anterior, se sustituye la mejor solución por el nuevo conjunto de medoides.

3.2. PAM: Aspectos generales y algoritmo

Una evaluación al algoritmo PAM puede plantearse de distintas maneras, sin embargo, la experiencia dicta que no conocer los detalles algorítmicos ni involucrarse directamente con la implementación conducirá probablemente a dudosos resultados, por tanto, el código Java que se programó un par de años atrás se evaluó y adicionalmente, se ha dado revisada literatura clásica sobre K-medoides, por tanto, se describe a modo de resumen en este apartado.

3.2.1 Alcances de K-medoides

El algoritmo K-medoids se utiliza para encontrar medoides en un clúster que es el punto central de un clúster. K-Medoids es más robusto en comparación con K-Means, ya que en K-Medoids encuentra a k como objeto representativo para minimizar la suma de las disimilitudes de los objetos de datos, mientras que K-Means utiliza la suma de las distancias euclidianas al cuadrado para los objetos de datos.

Partiendo de las bondades de K-Medoids que se basa en técnicas de representación de objetos para reducir los inconvenientes del algoritmo K-Means, se utilizó en este manuscrito para los objetivos planteados [10]. Los medoides son el objeto de datos del clúster que se encuentra más centrados, a veces conocidos en un sentido amplio como centroides. Los medoides se seleccionan aleatoriamente de los objetos de datos K_y para

formar el clúster K_y y los demás objetos de datos se colocan cerca de los medoides en un clúster. Luego se procesan todos los objetos de datos del clúster para encontrar nuevos Medoids de manera repetida para representar el nuevo clúster de mejor manera.

Después de encontrar los nuevos medoides se vinculan todos los objetos de datos al clúster. La ubicación de los medoides cambia en consecuencia con cada iteración. Así se forman k_y clústeres que representan n objetos de datos (en este epígrafe, k_y es la notación que se respetó de los autores Mark y Katherine [10]).

3.2.2 Algoritmo K-Medoides

Entrada: K_y: el número de clústeres, : un conjunto de datos que contiene n objetos.

Salida: Un conjunto de k_y clústeres.

Algoritmo:

- Seleccionar aleatoriamente k_y como Medoides para n puntos de datos.
- Encuentre los Medoids más cercanos calculando la distancia entre los puntos de datos n y los Medoids k y asignar los objetos de datos.
    - Para cada Medoids m y cada punto de datos o asociado a m hacer.
    - Intercambiar m y calcular el coste total de la configuración

Seleccionar el Medoids con el menor coste de la configuración.

- Si no hay cambios en las asignaciones, repetir los pasos 2 y 3 alternativamente.

4. P-Mediana

La P-Mediana consiste en que dado un conjunto de puntos (o ubicación de consumidores) y una matriz de distancias (o costos) entre todos y cada uno de los puntos, elegir p puntos (o ubicación de las instalaciones) con el propósito de minimizar la suma de las distancias de todos los puntos al punto elegido más cercano.

El desarrollo del planteamiento del problema P-Mediana tuvo lugar en los años 60´s se le puede atribuir a Hakimi en el caso discreto [11]

En 1970 ReValle and Swain presentaron la primera formulación de programación entera para el problema de la P-Mediana citado por [12]. En general el problema de la P-Mediana puede ser expresado matemáticamente como un problema de optimización discreta.

Una amplia cobertura de los métodos para atender el problema de la P-Mediana se encuentra en Reese donde se analiza también un conjunto de artículos sobre el P-Mediana que es clasificado como uno de los principales problemas tipo LAP (Problema de Ubicación Asignación, Location-Allocation Problem) [13].

Este problema determina la ubicación de las instalaciones y asigna los puntos de demanda a una instalación. El modelo de la P-Mediana puede establecerse en términos de grafos como: Sea $G = (V, E)$ un grafo no dirigido donde V es el conjunto de n vértices y E es el conjunto de aristas con un peso asociado que puede ser la distancia entre los vértices $d_{ij} = d(v_i, v_j)$ para toda $i, j = 1, \dots, n$ de acuerdo a determinada métrica, con las distancias se forma una matriz simétrica , encontrar $V_p \subseteq V$ tal que $|V_p| = p$, donde p puede ser o bien variable o fijo, y que la suma de las distancias más

cortas de los vértices en $\{V - V_p\}$ a su vértice más cercano en V_p se reduce al mínimo. Dicho en esta forma el PMP es un combinatorio y pertenece a la clase NP-Duro probado por Kariv y Hakimi en 1979 [14].

4.1. Modelo de la P-Mediana

El problema de la P-Mediana considera la siguiente situación. Se requiere particionar un conjunto finito de objetos en exactamente p grupos. Cada uno de dichos grupos estará caracterizado por uno de los objetos, que es seleccionado como la mediana del grupo, y el subconjunto de objetos asignado a dicha mediana.

Para cada par de objetos se especifica una distancia y se requiere minimizar la suma de distancias entre los objetos y las medianas a las que están asignados.

Sea $N = \{1, \ldots, n\}$ el conjunto de objetos. Para cada par $(i, j), i \in N, j \in N$, d_{ij} denota la distancia (similitud) entre los objetos i y j. Dado el número p, que denota el número de grupos, se necesita particionar el conjunto N en p subconjuntos disjuntos, es decir, $N = \bigcup_{k=1}^{p} N_k$ y $N_r \cap N_s = \emptyset$, para todo $r, s \in \{1, \ldots, p\}, r \neq s$.

A continuación, se considera el siguiente modelo de programación matemática para el problema. Se definen las siguientes variables de decisión:

$$y_i = \begin{cases} 1, & \textit{si el objeto i es selccionado como mediana,} \\ 0, & \textit{en otro caso.} \end{cases}$$

y

$$x_{ij} = \begin{cases} 1, & \textit{si el objeto j se asigna a la mediana, i,} \\ 0, & \textit{en otro caso.} \end{cases}$$

El problema se puede modelar de la siguiente manera:

$$\min \quad Z = \sum_{i \in N} \sum_{j \in N} d_{ij} x_{ij} \quad , \tag{1}$$

$$\text{sujeto a} \quad \sum_{i \in N} x_{ij} = 1 \qquad j \in N \quad , \tag{2}$$

$$\sum_{i \in N} y_i = p \quad , \tag{3}$$

$$x_{ij} \leq y_i \qquad i \in N, j \in N \quad , \tag{4}$$

$$x_{ij} \leq \{0,1\} \quad i \in N, j \in N \quad , \tag{5}$$

$$y_i \leq \{0,1\} \qquad i \in N \, . \tag{6}$$

Las restricciones (2) aseguran que cada objeto se asigna a una de las medianas. La restricción (3) asegura que se seleccionan p objetos como medianas. Finalmente, las

Tabla 1. Pruebas de rendimiento con los centros de cada grupo.

	P-Mediana con Ramificación – Acotamiento (Lingo)			**PAM**		
K	Costo	Tiempo (Segundos)	Medianas	Costo	Tiempo (Segundos)	Medoides
4	11.16330	1383.21	72, 95, 132, 283	11.16329	0	252, 72, 132, 283
8	5.995200	1941.34	1, 21, 29, 46, 70, 90, 113, 314	5.9951	3	21, 196, 1, 314, 307, 46, 113, 90
12	3.934600	2547.67	27, 46, 63, 82, 91, 111, 128, 142, 155, 257, 267, 322	3.9392	60	296, 123, 149, 129, 27, 63, 181, 91, 267, 257, 108, 142
24	1.721400	2165.05	7, 36, 47, 50, 51, 54, 63, 75, 87, 90, 102, 111, 114, 116, 119, 135, 159, 186, 195, 210, 264, 272, 307, 311	1.7219	75	116, 186, 132, 329, 20, 142, 63, 50, 215, 210, 250, 119, 135, 307, 36, 159, 53, 264, 272, 7, 311, 114, 102, 111,

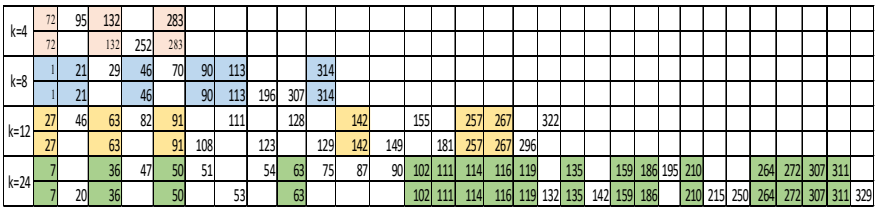

Fig. 1. Medianas vs Medoides.

restricciones (4) a (6) aseguran que los objetos solo puedan ser asignados a las medianas seleccionadas [13].

4.2. Modelo PAM

PAM es un método no jerárquico donde el conjunto de datos es particionado en un número previamente especificado de k conglomerados (grupos), luego iterativamente se asignan las observaciones a los conglomerados hasta que se satisface algún criterio de parada (función a optimizar), por ejemplo, que la suma de cuadrados dentro de los conglomerados sea mínima [3].

La función PAM está basada en la búsqueda de k objetos representativos llamados medoids (medoides) entre los objetos del conjunto de datos. Estos medoids son calculados tal que el total de disimilitud de todos los objetos hacia su medoide más cercano es mínima: es decir, la meta es encontrar un subconjunto $\{m_1 \dots m_k\} \subset \{1 \dots n\}$ que minimiza la función objetivo:

$$\sum_{i=1}^{n} \min_{t=1,\ldots,k} d(i, m_t)$$

.

Cada objeto es asignado al conglomerado correspondiente del centroide más cercano. Esto es, el objeto i es colocado en el conglomerado v_i cuando el medoide m_{vi} está más cercano a i que cualquier otro medoide m_w, o:

$$d(i, m_{vi}) \leq d(i, m_w) \text{ para todo } w = 1, \ldots, k$$

Actualmente el algoritmo de PAM consta de 2 pasos:

1.- Construir los centroides iniciales

m_1 es el objeto con el más pequeño $\sum_{i=1}^{n} d(i, m_1)$

.

.

m_k disminuye el objetivo (1) tanto como sea posible

2.- Intercambio

Repetir hasta lograr convergencia

considerar todos los pares de objetos (i, j) con

$i \in \{m_1, \ldots, m_k\}$ y $j \in \{m_1, \ldots, m_k\}$

y hacer el intercambio $i \leftrightarrow j$ para cualquiera que decrezca más el objetivo.

La función objetivo de PAM depende de las disimilitudes entre los objetos por tanto esta función solamente necesita la matriz de distancias como entrada [3].

5. Experiencia computacional

El propósito de este trabajo es revelar que, bajo mismas condiciones, PAM y P-Mediana son modelos equivalentes. Se puede comprobar este supuesto si se comparan los resultados en dos aspectos: costo de la función objetivo y configuración de la partición resultante.

Es conveniente señalar que la idea para determinar la equivalencia entre Pam y la P-Mediana tuvo lugar cuando una reflexión trivial rodeaba un problema de localización y diseño de territorio que bien se podía atender con la, pero del mismo modo con el particionamiento clásico. Por tanto, se evocó a las metodologías más utilizadas en el particionamiento territorial, donde destacan los modelos de localización-asignación ("location-allocation") y los de particionamiento de conjuntos ("set partitioning").

En el caso particular de diseño de territorio, el particionamiento territorial se modela como un problema de la P-Mediana. Sin embargo, en términos amplios, el particionamiento se puede resolver como un problema de la P-Mediana o la P-Mediana usa el particionamiento para su solución incluso cuando los métodos exactos no son una alternativa.

En este punto, se sostiene que, si existen dos algoritmos equivalentes, la decisión de escoger uno de ellos reside en conocer cual algoritmo genera mejor calidad de las soluciones en mejor tiempo de computo.

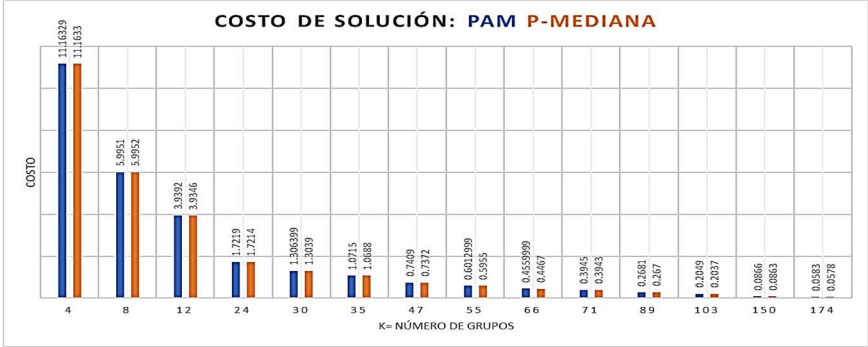

Fig. 2. Gráfica de rendimiento entre PAM y P-Mediana.

Fig. 3. Comparación del costo computacional.

Entonces, desde esa perspectiva y con el propósito de examinar el comportamiento de PAM y la P-Mediana para determinar su equivalencia, se diseñó un básico experimento que contrasta tres aspectos fundamentales y que garantizan de algún modo, la semejanza planteada: 1) función de costo, 2) centroides 3) costo computacional y configuración de los grupos. Informalmente, esta situación puede expresarse considerando que debido a que PAM crea particiones tal que cada grupo de la partición tiene un "centro" llamado medoide, el cual atrae a los objetos que minimizan su distancia hacia el medoide a través de la función de costo, el desafío consiste en retar a PAM con otro algoritmo cuya semántica algorítmica sea similar.

En este escenario, se escogió el modelo de la P-Mediana, el cual posee un método equivalente si se considera la siguiente situación: Se requiere particionar un conjunto finito de objetos en exactamente p grupos. Cada uno de dichos grupos estará caracterizado por uno de los objetos que es seleccionado como la mediana del grupo y el subconjunto de objetos asignado a dicha mediana. Para cada par de objetos se especifica una distancia y se requiere minimizar la suma de distancias entre los objetos y las medianas a las que están asignados.

De acuerdo a lo anterior, se subraya que la finalidad de este trabajo es evaluar 2 algoritmos ampliamente citados en la literatura para medir características semejantes. Dichos algoritmos son formulados también como métodos y se les conoce en términos amplios como de localización-asignación y de clasificación por particiones (PAM y P-Mediana respectivamente).

El experimento fue diseñado para que el costo de computo fuera razonable y de la misma manera el software elegido para programar la P-Mediana pudiera arrojar soluciones sin rebasar las limitaciones de Lingo. Lingo, resuelve la P-Mediana con Branch and Bound y ofrece un valor exacto en el resultado de la función objetivo e implícitamente produce una partición

Por otra parte, PAM proporciona una partición bajo un esquema exhaustivo-combinatorio y se asume que en estas condiciones PAM y P-Mediana son métodos equivalentes en al menos 3 puntos: 1) utilizan en la entrada de datos una matriz de distancias, 2) generan una partición, 2) usan la misma función de costo, 3) cada centro de los grupos que componen la partición tiene la misma meta (básicamente la equivalencia se centra en el punto 2, los demás puntos son tácitos).

5.1 Experimento

En esta subsección se describe la experiencia computacional para determinar la equivalencia entre PAM y P-Mediana.

Las tablas 1-5, concentran los resultados de las pruebas hechas a la P-Mediana con Lingo, y a PAM con un algoritmo de clasificación por particiones combinatorio. Como puede observarse, los resultados no son idénticos, excepto en la tabla 4. Se insiste que la diferencia en la función objetivo es un "épsilon", el cual se justifica en el cálculo numérico por la aproximación o truncamiento.

En la figura 1, hemos identificado las medianas y los medoides para la P-Mediana y PAM respectivamente. Por ejemplo, para k=4 la mediana=95 y el medoide=252 son los únicos centroides que no coinciden.

De la misma manera para k=24, las medianas 47, 51, 54, 75, 87, 90 son distintos a los medoides 20, 53, 132, 142, 215, 250, 329; el resto de los objetos son iguales. En la gráfica de la figura 2, se distingue la "igualdad" de los óptimos para PAM y la P-Mediana de 4, 8, 12, 24, 30, 35, 47, 55, 66, 71, 89, 103, 150, 174.

En la figura 3 se observa que el costo computacional de PAM es mejor hasta k=9 grupos con respecto a Ramificación y Acotamiento para la P-Mediana en Lingo.

En la tabla 3 se obtuvieron 2 casos de las pruebas realizadas. Se observa que para la instancia de k=4 y k=8, PAM produce un valor óptimo menor a la P-Mediana de Lingo con una diferencia de -0.00001 y -0.00010 respectivamente. Como puede verse para k=12 y k=24, Lingo minimiza mejor el objetivo que PAM (la diferencia sigue siendo una épsilon).

El comportamiento de PAM vs. P-Mediana en Lingo cuya diferencia en la distancia de los óptimos es casi despreciable, se aprecia en la siguiente tabla 4, la cual es una extensión de la tabla 3.

En la tabla 5 se distingue que tanto PAM como la P-Mediana tienen el mismo costo objetivo. En la figura 4, se ha ampliado la escala para que sea evidente la diferencia minúscula de los costos objetivos para los algoritmos en cuestión. A pesar de ello, se mantiene la conjetura de la equivalencia para PAM y P-Mediana. A continuación, la tabla 6 revela el registro de la prueba que alcanzó exactamente el mismo óptimo e incluso los mismos grupos con sus respectivos centros para ambos algoritmos.

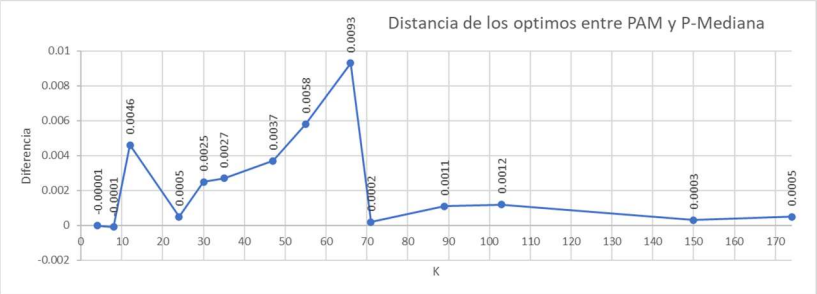

Fig. 4. Limites superior e inferior de los costos para PAM y P-Mediana.

Tabla 5. Comparación entre los dos costos de solución idéntica para k=5.

PAM	Lingo	PAM = Lingo	Resultado
6.88E-02 (0.06879999)	6.88E-02 (0.0688)	0. 068 = 0.068	Iguales

Tabla 6. Grupos con sus respectivos centros.

		PAM		P-Mediana	
K	Clúster	Medoide	Elementos del grupo	Mediana	Elementos del grupo
5	1	3	1,2,4.	3	1,2,4,5.
	2	28	15,19,22,23,26,27.	14	16,17,20,21,24.
	3	14	16,17,20,21,24.	18	11,13.
	4	18	11,13.	25	6,7,8,9,10,12,29,30.
	5	25	5,6,7,8,9,10,12,29,30.	28	15,19,22,23,26,27.

Después de analizar distintas particiones de la P-Mediana en Lingo y PAM, se tienen conclusiones que quizá muchos investigadores han experimentado, por ejemplo, la incapacidad de Lingo para generar soluciones debido a la gran cantidad de variables que genera el algoritmo implícito de Ramificación y Acotamiento.

En este trabajo, la instancia de la ZMVT es de 469 objetos y para que Lingo pudiera competir justamente con PAM, se observó que solo soporta hasta 369 objetos.

6. Conclusiones

Sabemos que un problema es de optimización combinatoria (OC) si en ellos las variables de decisión formadas por vectores X, sólo admiten valores enteros y su espacio de soluciones factibles Ω está formado por permutaciones o subconjuntos de números naturales. PAM pertenece a esta categoría combinatoria.

Por otra parte, la P-Mediana puede ser resuelta con algún software optimizador como Lingo, cuya implementación responde al algoritmo Branch and Bound, el cual produce una solución exacta. No obstante, las restricciones de los optimizadores para resolver la P-Mediana se enfrentan al número de variables que se generan por el algoritmo programado del modelo, que en este caso es entero-binario.

Un dilema que los investigadores tienen en problemas de esta magnitud, consiste en optar por dejar a un lado el software de optimización y programar el modelo de la P-Mediana con algún software de propósito general. Comúnmente la P-Mediana es implementada con las características del particionamiento clásico, siempre que no sea necesario incorporar métodos heurísticos para conseguir una solución aproximada.

En la literatura se encuentran distintas maneras de expresar tanto el particionamiento como la P-Mediana, incluso con distinta notación, pero todas las descripciones tienen la misma intención: crear grupos con un centroide tal que la distancia a cualquier otro objeto del conjunto de datos sea mínima.

Se ha insistido que los centroides en la P-Mediana son las medianas, y tal concepto no debe confundirse con la mediana estadística, dado que la P-Mediana es la generalización de la mediana conocida comúnmente como una medida estadística descriptiva, es decir, la mediana geométrica de un conjunto discreto de puntos o datos de una muestra en un espacio euclídeo, es el punto (centro) que minimiza la suma de las distancias a los puntos de la muestra, lo cual generaliza el concepto de la mediana estadística.

El significado de esta mediana se centra en la propiedad de minimizar la suma de distancias para datos unidimensionales y además proporciona una medida de tendencia central en dimensiones superiores.

Es posible afirmar que la mediana geométrica es un estimador de localización en estadística, incluso también es un indicador estándar en la resolución del problema de localización de instalaciones, donde modela el problema de localizar una instalación para minimizar el costo del transporte [15].

Formalmente para un conjunto dado de *m* puntos $x_1, x_2, \dots x_m$ con cada $x_i \in R^n$ la mediana geométrica se define como $Argmin \sum_{i=1}^{n} \|x_i - y\|_2$, donde argmin significa el valor del argumento $y$que minimiza la suma. En este caso, es el punto y desde donde la suma de todas las distancias euclidianas a x_i es mínima.

Esta afirmación justifica que la mediana geométrica sea la parte medular del modelo de la P-Mediana y también se incorpora como el medoide en el particionamiento sumado a las restricciones; esto es, una partición de un conjunto *A* está formada por los subconjuntos $A_1, A_2, \dots A_n$los cuales deben cumplir:

- Que la unión de todos los subconjuntos sea igual al conjunto dado $A_1 \cup A_2 \cup \dots \cup A_n = A$.
- Que todos los subconjuntos sean disjuntos entre sí y que ningún subconjunto sea vacío.

El particionamiento para un conjunto de datos se plantea matemáticamente como la P-Mediana y viceversa. Ello implica que ambos modelos comparten la función objetivo, excepto que los centros de los grupos son medoides para el particionamiento PAM. Los medoides son centros de grupos y el investigador elige a su criterio la mejor manera de colocar los medoides en el centro de los clústeres, y de acuerdo a la definición de mediana geométrica, ese centro concuerda con el concepto de mediana.

Revisando los costos de los métodos presentados, estos valores son iguales con una discrepancia muy pobre que obedece a la precisión de las cifras significativas aritméticas en el truncamiento y aproximación. Esta situación se puede resolver en PAM con algún método numérico como Newton-Fourier, Bisección o Punto fijo, por

ejemplo, caso contrario, en Lingo desconocemos la manera en que aproximan o truncan los decimales en la función de costo, ahí reside la diferencia del épsilon entre los valores de las funciones objetivo de PAM y P-Mediana en Lingo.

Sin embargo, las normas IEEE en el estándar 754, dicta que los softwares comerciales definan 3 formatos de punto flotante (32, 64, y 128 bits), por tanto, se puede arreglar el problema de aproximación en PAM y de este modo se conseguiría los valores objetivos sean exactamente iguales [16].

Respecto a la diferencia de los centros de los grupos, es más sencillo explicar la situación porque PAM en este artículo ofrece sólo una solución, lo cual no significa que sea la única, es posible que se generasen más óptimos. En otras palabras, en un algoritmo combinatorio como PAM, cuando en el código se comparan soluciones y estas son iguales, el algoritmo escoge cualquiera de ellas porque todas las soluciones óptimas generadas son "igual de buenas", entonces una de ellas coincidirá totalmente con la que produce Lingo.

Referencias

1. Díaz-García, J. A., Bernabé-Loranca, M. B., Luna-Reyes, D. E., Olivares-Benítez, E., Martínez-Flores, J. L.: Relajación lagrangeana para el problema de particionamiento de áreas geográficas. Revista de Matemática: Teoría y Aplicaciones, vol. 19, no. 2, pp. 169–181 (2012) doi: 10.15517/rmta.v19i2.1332
2. Piza-Volio, E., Murillo-Fernández, Á., Trejos-Zelaya, J.: Nuevas técnicas de particionamiento en clasificación automática. Revista de Matemática Teoría y Aplicaciones, vol. 6, no. 1, pp. 51–66 (2009)
3. Kaufman, L., Rousseeuw, P. K.: Clustering by means of medoids. In: Y. Dodge, 405–416 (1987)
4. Sokal, R. R., Sneath, P. H. A.: Principles of numerical taxonomy. W.H. Freeman and Company; 1st edition, pp. 1–359 (1963)
5. Gower, J. C.: Les bases de la classification automatique. Journal of the Royal Statistical Society Series, vol. 20, no. 2, pp. 195–197 (1971) doi: 10.2307/2346469
6. Jardine, N., Sibson, R.: Mathematical taxonomy. Universidad de California, pp. 1–286 (1971)
7. Anderberg, M.: Cluster analysis for applications. Academic Press (1973)
8. Hartigan, J.: Cluster algorithms. John Wiley & Sons (1975)
9. Han, J., Kamber, M., Tung, A.: Spatial clustering methods in data mining: A survey. Geographic Data Mining and Knowledge Discovery, pp. 188–217 (2001)
10. Van der Lear, M., Pollar, K., Bryan, J.: A new partitioning around medoids algorithm. Journal of Statistical Computation and Simulation, vol. 73, no. 8, pp. 575–584 (2003) doi: 10.1080/0094965031000136012
11. Hakimi, S.: Optimum location of switching centers and the absolute centers and medians of a graph. Operations Research, vol. 12, no. 3, pp. 450–459 (1964) doi: 10.1287/opre.12.3.450
12. Church, R.: Cobra: A new formulation of the classic P-Median location problem. Annals of Operation Research, pp. 103–120 (2003)
13. Reese, J.: Solution methods for the P-Median problem: An annotated bibliography. Networks an International Journal, vol. 48, no. 3, pp. 125–142 (2006) doi: 10.1002/net.20128
14. Jiang, H., Zhang, X., Li, M.: Backbone of the P-Median problem. Lecture Notes in Computer Science, vol. 4839, pp. 699–704 (2007) doi: 10.1007/978-3-540-76928-6_78
15. Wolf, G.: Foundations of location analysis. International Journal of Geographical Information Science, pp. 577–578 (2012)

16. https://ieeexplore.ieee.org/document/8766229