XIV ELAVIO, El Fuerte, Sinaloa, México, 9-14 de agosto de 2009

DISEÑO DE PLANES EFICIENTES PARA LA SEGMENTACIÓN DE CLIENTES CON MÚLTIPLES ATRIBUTOS

Diana Lucia Huerta Muñoz[1], Roger Z. Ríos Mercado[1], Elisa Schaeffer[1], Rubén Ruiz[2]

[1]Universidad Autónoma de Nuevo León
San Nicolás de los Garza, Nuevo León, México
{lucia,roger,elisa}@yalma.fime.uanl.mx

[2]Universidad Politécnica de Valencia
Valencia, España
rruiz@eio.upv.es

Palabras Clave: segmentación de clientes, K-medias, GRASP, VNS, heurísticas.

Resumen: *El presente trabajo trata una problemática real de una empresa distribuidora de productos en Monterrey, N.L., México. La problemática consiste en particionar un conjunto de clientes, distribuídos geográficamente, en segmentos de manera que la disimilitud con respecto a cuatro atributos de relevante importancia para la empresa, sea la menor posible. Además se requiere que se encuentren relativamente cercanos. Debido al tamaño de las instancias reales no es posible aplicar métodos exactos para su resolución. Es por ello que se desarrolló una metodología, basada en métodos aproximados, que consiste en obtener una partición inicial utilizando un procedimiento de búsqueda adaptativo, aleatorizado y voraz (GRASP). La mejor solución obtenida es mejorada por un método heurístico iterativo que destruye, y construye la solución de una manera voraz para luego aplicar una búsqueda local basada en una búsqueda de entornos variables (VNS).*

1. Introducción

La segmentación de mercado es un proceso que consiste en dividir el mercado total de un bien o servicio en grupos más pequeños e internamente homogéneos, cuya esencia es conocer realmente a los clientes. Un segmento de mercado representa a un grupo relativamente grande y homogéneo de clientes que tienen deseos, ubicación geográfica, actitudes ó hábitos de compra similares y que reaccionarán de modo parecido ante una determinada estrategia. El comportamiento del cliente suele ser demasiado complejo como para explicarlo con una o dos características, se deben tomar en cuenta varias dimensiones partiendo de las necesidades de los clientes.

En este trabajo, se aborda un caso de estudio de segmentación de clientes de una empresa distribuidora de bebidas de la ciudad de Monterrey, N.L., México. Dado un conjunto de clientes, la empresa desea particionar dicho conjunto en segmentos de manera que la disimilitud con respecto a cuatro atributos, sea la menor posible. Estos cuatro atributos son con respecto a la ubicación geográfica, el volúmen de compra, el tipo de contrato y tipo de establecimiento del cliente. Por semejanza con respecto a ubicación geográfica se entiende que los clientes asignados a un segmento se encuentren relativamente cercanos.

[1] Agradecemos a CONACyT (48499-Y), a UANL, PAICYT (CA1478-07)

La importancia de obtener segmentos de esta manera surge de la necesidad de la empresa en desarrollar e implementar diferentes estrategias de mercadotecnia y poder así satisfacer las necesidades de sus clientes según sus preferencias o necesidades. Además de disminuir, al obtener segmentos compactos, el costo de transportación del producto y posibles inconformidades entre sus clientes al aplicar diferentes estrategias en cada segmento. Debido al tamaño de las instancias reales no es posible aplicar métodos exactos para su resolución. Es por ello que la metodología de solución propuesta en este trabajo se basa en métodos aproximados o heurísticos. Dicha metodología consiste en una fase de construcción de soluciones y una fase de mejora de las mismas. En la primera fase se hace uso del algoritmo *K-medias*, un algoritmo muy utilizado en el área de segmentación por su sencilla implementación computacional y su rapidez para encontrar soluciones. Dado a que una de las desventajas de este algoritmo es que cae rápidamente en un óptimo local, se ha desarrollado un método de post-procesamiento iterativo para mejorar la solución obtenida por el *K-medias* basado en el método desarrollado por (Ruiz y Stützle, 2007) y que fué aplicado originalmente a un problema de secuenciación de tareas en una línea de flujo.

El presente trabajo está organizado de la siguiente manera. En la Sección 2, se hace una breve descripción de algunos antecedentes encontrados en la literatura. Posteriormente, en la Sección 3, se describe el problema a tratar. En la Sección 4, se describe la metodología propuesta. En la Sección 5 se muestran algunos resultados. Cerramos, en la Sección 6, con las conclusiones.

2. Antecedentes

En la literatura existe una gran cantidad de métodos desarrollados para el problema de segmentación. Se pueden mencionar trabajos como el de (Caballero et al., 2006) quienes desarrollaron un procedimiento metaheurístico para problemas de agrupamiento multiobjetivo, basado en búsqueda tabú y búsqueda dispersa. Por otro lado, (Negreiros y Palhano, 2005) proponen un algoritmo de dos fases, la primer fase construye una solución usando el algoritmo de Forgy y la segunda consiste en mejorar dicha solución por medio de una búsqueda de entornos variables (VNS). (Sheng y Liu, 2004) proponen una búsqueda local híbrida basada en un algoritmo genético para un agrupamiento de *K*-medioides para conjuntos grandes de datos. (Cano et al., 2002) proponen un procedimiento de búsqueda adaptativo, aleatorizado y voraz (GRASP) cuyas fases de construcción y post-procesamiento están basadas en el algortitmo de Kaufman y *K*-medias respectivamente. Por último, (Hartuv y Shamir, 2000) desarrollaron un algoritmo polinomial para el análisis de segmentos basado en técnicas sobre teoría de grafos, entre otros.

3. Descripción del Problema

Dado un conjunto de clientes *V*, ubicado en una determinada área geográfica, se desea obtener una *p*-partición $X = (X_1, ..., Xp)$, $X \epsilon \Pi$ (Π representa al conjunto de soluciones factibles), de dicho conjunto, de manera que la disimilitud entre clientes que conforman un mismo segmento X_k, $k=\{1,...,p\}$, sea la menor posible (1). Dichos atributos corresponden al volúmen de compra, tipo de contrato y establecimiento del cliente. La razón por la cual se desea particionar dicho conjunto de esta manera, es debido a requerimientos de la empresa para la aplicación posterior de estrategias de mercadotecnia en cada segmento establecido. Una forma de medir esta disimilitud es usando una función ponderada de la forma:

$$\min_{X \in \Pi} f_{disimilitud}(X) = \alpha_1 f_{disp}(X) + \alpha_2 f_{sku}(X) + \alpha_3 f_{cont}(X) + \alpha_4 f_{est}(X) \quad (1)$$

donde $f_{disp}(X)$ representa la medida de disimilitud de una partición X con respecto a la dispersión, $f_{sku}(X)$ con respecto al volúmen de compra, $f_{cont}(X)$ con respecto al tipo de contrato y $f_{est}(X)$ con respecto al tipo de establecimiento, α_1 , α_2, α_3 y α_4 representan el peso que se le asigna a los atributos, los cuales están dentro de un rango [0,1] y la suma de éstos debe ser igual a 1. Una descripción muy breve acerca de como se mide la disimilitud con respecto a los cuatro atributos se menciona a continuación.

Dispersión: para medir la disimilitud con respecto a la dispersión se emplea la suma de distancias intra-grupo.

$$f_{disp}(X) = \sum_{k \in K} \sum_{i<j \in X_k} d_{ij} \tag{2}$$

Volúmen de Compra: la disimilitud entre una pareja de clientes en cuanto al volúmen de compra está dada por la suma de los cuadrados de las diferencias de los volúmenes promedio por producto ó SKU comprado por dicha pareja de clientes.

$$f_{sku}(X) = \sum_{k \in K} \sum_{i<j \in X_k} q_{ij}^{SKU} \tag{3}$$

donde,

$$q_{ij}^{SKU} = \sum_{s \in S} \left(\frac{a_{is}}{a_i^T} - \frac{a_{js}}{a_j^T} \right)^2 \tag{4}$$

$$a_i^T = \sum_{s \in S} a_{is} \tag{5}$$

Tipo de Contrato y Establecimiento: para los tipos de contrato y establecimiento, por requerimiento de la empresa, el valor de la disimilitud para una pareja de clientes *i,j* será $h_{ij} = 0$ ($g_{ij} = 0$ para el tipos de establecimiento) si los contratos (establecimientos) son iguales ó $h_{ij} = 1$ ($g_{ij} = 1$) en caso contrario. Entonces el cálculo de la disimilitud de una partición *X* estará dado por la suma de las disimilitudes de todos los clientes pertenecientes a cada segmento.

$$f_{cont}(X) = \sum_{k \in K} \sum_{i<j \in X_k} h_{ij} \tag{6}$$

$$f_{est}(X) = \sum_{k \in K} \sum_{i<j \in X_k} g_{ij} \tag{7}$$

4. Metodología Propuesta

La metodología que se propone se compone de dos fases principalmente, construcción de particiones iniciales y mejora de dichas particiones.

4.1 Construcción de Particiones Iniciales

En esta fase se hizo uso del algoritmo *p-medias,* también llamado *K-medias* (Jain, Murty y Flynn, 1999), para construír inicialmente una partición. Este algoritmo es un método muy popular en problemas de agrupamiento (clustering) que consiste en, partiendo de una configuración inicial de *p* centros, tomados al azar, asignar los elementos restantes a su centro más cercano. Una vez formados los segmentos, se recalculan los centros iterando de esta manera hasta satisfacer un criterio de parada. Una limitación de este método es que la calidad de la solución final depende ampliamente de la configuración inicial de centros. Por tal motivo, proponemos un procedimiento de búsqueda adaptativo, aleatorizado y voraz (GRASP), el cual consiste en obtener mejor configuración de centros, reconociendo que la elección de *p* centros dispersos equivale al problema de *p-dispersión* propuesto por (Erkut, Urkusal y Yenycerioglu,

1994). Estos centros se irán refinando al aplicar el algoritmo *p-medias*, el cual se emplea como fase de post-procesamiento del GRASP.

4.2 Mejora de la Partición

En la fase anterior se desarrolló un GRASP, formado por la combinación de una heurística aplicada en problemas de *p-dispersión* y el algoritmo p-*medias*, para la creación de particiones iniciales de mejor calidad. Para esta siguiente fase, se desarrolló un método basado en un algoritmo voraz iterativo, propuesto por (Ruiz y Stützle, 2007) para un problema de secuenciación de tareas en una línea de flujo (scheduling), el cual ha mostrado excelentes resultados.

El método consiste principalmente en destruír la solución inicial y construirla nuevamente de forma voraz aplicándole posteriormente una búsqueda local (en nuestro caso esta búsqueda local está basada en una búsqueda de entornos variables que aplica movimientos de inserción e intercambio). Si la solución que se obtuvo después de aplicar la búsqueda local es mejor a la actual, se actualiza y se verifica si dicha solución es la mejor encontrada hasta el momento. Si es así, se actualiza también. Por otro lado, si la solución es peor a la actual, puede aún aceptarse con cierta probabilidad con la finalidad de encontrar mejores soluciones al explorar un subespacio que no hubiera podido ser explorado si solo se tomaran en cuenta soluciones mejores.

5. Experimentación

5.1 Descripción

La metodología se desarrolló en C++ bajo un sistema operativo Ubuntu-Linux 8.10. Se utilizaron cinco instancias para cada tamaño n = {1000, 4000, 7000}. Se ha experimentado para tres diferentes números de segmentos p= {5,10,15} y usando una selección de centros aleatoria (es decir GRASP con parámetro de calidad β=1) y dando el mismo peso a los cuatro atributos ($\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = 0.25$). El algoritmo *p-medias* es iterado 50 veces, el número de elementos involucrados en las fases de destrucción/construcción del algoritmo IGLS es del 15% del tamaño de la instancia. El algoritmo IGLS es iterado cinco veces.

5.1 Discusión

Los resultados obtenidos se muestran en la Tabla 1. En dicha tabla se comparan los resultados obtenidos usando el algoritmo *p-medias* (*p*-m) y algoritmo de búsqueda local voraz iterativo (IGLS). Como puede observarse el algoritmo IGLS mejoró en hasta un 81% la solución obtenida por el *p-medias* en un tiempo de cómputo razonable.

Inst	n	p	Obj. *p*-m	Obj. IGLS	GAP%	T-*p*-m(s)	T-IGLS(s)
1	1000	5	0.0685	0.0566	20.98	3.32	8.56
1	4000	10	0.0293	0.0229	31.32	56.21	251.45
1	7000	15	0.0182	0.0126	48.02	176.66	504.86
2	1000	5	0.0635	0.0579	9.57	3.45	9.40
2	4000	10	0.0315	0.0228	38.75	57.92	204.83
2	7000	15	0.0181	0.0131	46.61	165.58	500.00
3	1000	5	0.0710	0.0594	19.62	3.33	7.90
3	4000	10	0.0279	0.0229	22.97	54.77	210.69
3	7000	15	0.0180	0.0129	48.00	162.32	512.25
4	1000	5	0.0727	0.0564	31.41	3.46	9.61
4	4000	10	0.0308	0.0221	40.70	57.80	267.71
4	7000	15	0.0177	0.0125	43.39	161.41	540.59
5	1000	5	0.0670	0.0568	18.04	3.43	8.51
5	4000	10	0.0300	0.0221	37.20	52.71	234.25
5	7000	15	0.0088	0.0051	81.37	132.59	384.28

Tabla 1: Resultados obtenidos al aplicar el método propuesto

6. Conclusiones

En este trabajo se ha desarrollado una metodología que con el fin de solucionar de manera aproximada una problemática real de una empresa distribuidora de bebidas de la Cd. de Monterrey, Nuevo León, México. La metodología, desarrollada en el lenguaje de programación C++, consiste en una combinación de métodos aproximados de sencilla implementación computacional y que además resulta de fácil entendimiento para el usuario.

Referencias

Brusco, M. y Stahl, S. (2005), *Branch-and-Bound Applications in Combinatorial Analysis,* Springer, E.U.A.

Caballero, R., Laguna, M., Martí, R. y Molina, J. (2006), "Multiobjective clustering with metaheuristic optimization technology", Reporte Técnico, Departamento de Estadística e Investigación Operativa, Universidad de Valencia, Valencia, España.

Cano, J., Cordón, O., Herrera, F. y Sánchez, L. (2002), "A GRASP algorithm for clustering", *en Proceedings of the 8th Ibero-American Conference on AI: Advances in Artificial Intelligence,* Sevilla, 214 – 223.

Erkut, E., Urkusal, Y. y Yenyerioglu, O. (1994),"A comparison of p-dispersion heuristics", *Computers and Operations Research*, 21(10): 1103-1113.

Hartuv, E. y Shamir, R. (2000), "A clustering algorithm based on graph connectivity", *Information Processing Letters*, 76: 175-181.

Jain, A., Murty, M., y Flynn, P. (1999), "Data clustering: a review", *ACM Computing Surveys*, 31 (3): 264-323.

Negreiros, M. y Palhano, A. (2005), "The capacitated centred clustering problem", *Computers and Operational Research,* 10(7): 289-296.

Ruiz, R. y Stützle, T. (2007), "A simple and effective iterated greedy algorithm for the permutation flowshop scheduling problem", *European Journal of Operational Research*, 177(3): 2033-2049.

Sheng, W. y Liu, X. (2004),"A hybrid algorithm for k-medoid clustering of large data set", en *Congress on Evolutionary Computation,* E.U.A. 1, 77- 82.