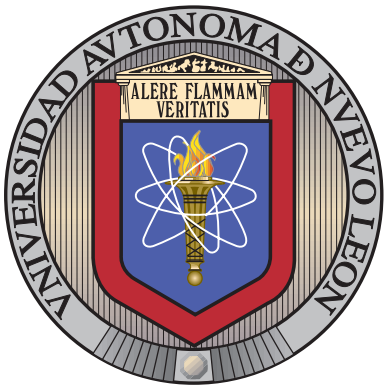
Universidad Autónoma de Nuevo León

Facultad de Ingeniería Mecánica y Eléctrica

División de Estudios de Posgrado



Diseño de Planes Eficientes para la Segmentación de Clientes con Múltiples Atributos

por

Diana Lucia Huerta Muñoz

Tesis

en opción al grado de

Maestro en Ciencias en Ingeniería de Sistemas

San Nicolás de los Garza, Nuevo León Julio 2009

Universidad Autónoma de Nuevo León

Facultad de Ingeniería Mecánica y Eléctrica

División de Estudios de Posgrado

Diseño de Planes Eficientes para la Segmentación de Clientes con Múltiples Atributos

por

Diana Lucia Huerta Muñoz

Tesis

en opción al grado de

Maestro en Ciencias en Ingeniería de Sistemas

San Nicolás de los Garza, Nuevo León Julio 2009

Universidad Autónoma de Nuevo León

Facultad de Ingeniería Mecánica y Eléctrica

División de Estudios de Posgrado

Los miembros del Comité de Tesis recomendamos que la Tesis «Diseño de Planes Eficientes para la Segmentación de Clientes con Múltiples Atributos», realizada por la alumna Diana Lucia Huerta Muñoz, con número de matrícula 1147501, sea aceptada para su defensa como opción al grado de Maestro en Ciencias en Ingeniería de Sistemas.

El Comité de Tesis

Dr. Roger Z. Ríos Mercado
Asesor

Dr. José Arturo Berrones Santos
Revisor

Dr. Jesús Fabián López Pérez
Revisor

Vo. Bo.

Dr. Moisés Hinojosa Rivera
División de Estudios de Posgrado

San Nicolás de los Garza, Nuevo León, Julio 2009

Índice general

Índice de figuras

Índice de tablas

Dedicatoria

A papá por haberme amado y querido tanto. Tus consejos y palabras las llevaré conmigo siempre porque gracias ti aprendí que todo es posible. Ayer, hoy y siempre vivirás en mi corazón.

A mamá por quererme y darme la fuerza que necesito cada día para salir adelante. Por apoyarme para llegar hasta donde me encuentro y sobre todo por darme la vida para poder disfrutarla junto a ti y papá.

Los quiero tanto y siempre estaré agradecida por todo lo que han hecho por mi.

Reflexión

El papel del marketing es hacer innecesaria la tarea de la venta. Su objetivo es llegar a conocer tan perfectamente a los clientes que lo que se ofrezca se venda solo.

Peter Druker

Agradecimientos

Gracias a Dios por darme esa fuerza interna para seguir adelante, por poner en mi camino a aquellas personas que tanto amo, quiero y estimo y que desde pequeña han sido y creado una parte de mi vida.

Mi más eterno agradecimiento a mis padres, Chaguito y Nachita, por darme darme la vida, por brindarme todo su apoyo y paciencia, por aconsejarme, por quererme y desear siempre lo mejor para todos. Por haberme dado la oportunidad de aprender de su sabiduría y poder aplicarla a mi vida para tratar de ser mejor como persona cada vez. Gracias por haberme corregido cuando estaba mal y haberme enseñado que todo se obtiene con dedicación y esfuerzo. Gracias por ser mis padres.

A mis hermanas de quienes recibí siempre apoyo de todo tipo el cual agradezco de corazón. En especial a mi hermanas Verónica y Claudia por estar siempre al pendiente y ayudarme para que pudiera salir adelante en mis estudios.

A mi Isa por estar en todo momento a mi lado y por apoyarme para que pudiera salir adelante en todo lo que me propusiera. Porque gran parte de este camino lo he recorrido tomada de tu mano, te agradezco infinitamente. Gracias por amarme, quererme y permitirme ser parte de tu vida. A tu lado todo es mejor.

Agradezco a la coordinación del Posgrado en Ingeniería de Sistemas (PISIS) por permitirme formar parte del programa. A la Universidad Autónoma de Nuevo León (UANL) y la Facultad de Ingeniería Mecánica y Eléctrica (FIME), por el apoyo otorgado para efectos de inscripción y colegiatura los cuales me fueron de gran ayuda para realizar mis estudios de maestría así como el apoyo mediante el proyecto

UANL-PAICYT CA1478-07 para la presentación de este trabajo en una variedad de congresos. Al Consejo Nacional de Ciencia y Tecnología (CONACYT) por el gran apoyo que me brindó mediante una beca de manutención durante mi estancia como estudiante en el programa y mediante una beca mixta para la realización de una estancia en el extranjero. Agradezco también al CONACYT en conjunto con la Secretaría de Educación Pública por el apoyo otorgado mediante el proyecto SEP-CONACYT 48499-Y.

Agradezco profundamente a mi asesor de tesis, el Dr. Roger Z. Ríos Mercado, por todo su apoyo, consejos y sobre todo su paciencia para que este trabajo se realizara de la mejor manera posible. Gracias por todo el apoyo que me brindó.

De igual forma muchas gracias a los miembros del comité, Dr. Arturo Berrones y Dr. Fabián López por el tiempo que me brindaron para aclarar mis dudas siempre que éstas se me presentaban. Gracias también por sus comentarios y observaciones las cuales me fueron de gran ayuda para dirigir mi trabajo hacia un buen camino.

A la Dra. Elisa Schaeffer por toda la ayuda que me brindó mediante consejos y observaciones que me ayudaron a comprender mucho de lo que ha sido este trabajo de tesis. Agradezco infinitamente el tiempo que dedicó para ayudarme.

Gracias al Dr. Rubén Ruiz y a los integrantes del Grupo de Sistemas de Optimización Aplicada por permitirme formar parte del grupo durante mi estancia de investigación en la Universidad Politécnica de Valencia y por las ideas aportadas para este trabajo de tesis.

A mis amigos del equipo de trabajo TDP por brindarme su amistad y hacer que el trabajo en el equipo fuera más ameno cada día.

A mis amigos de generación por hacer cada uno de los días en el PISIS un día mejor. En especial a Angy, Yajaira, Lucero, Vanesa y Juan Carlos por darme su apoyo y sus consejos cuando más lo necesité. Gracias amigos por hacer siempre un día diferente en mi vida.

A todos mis profesores en el PISIS por una enseñanza de calidad que me brindaron durante mis estudios de maestría la cual ha sido una fuente de motivación para querer seguir en este ramo en un futuro no muy lejano.

A todos mis amigos ajenos al PISIS que siempre me apoyaron, aconsejaron y motivaron a seguir siempre adelante en mis estudios. Muchas gracias por preocuparse por mi y apoyarme como lo han hecho.

Resumen

Diana Lucia Huerta Muñoz.

Candidato para el grado de Maestro en
Ciencias en Ingeniería de Sistemas.

Universidad Autónoma de Nuevo León.
Facultad de Ingeniería Mecánica y Eléctrica.

Título del estudio:

Diseño de Planes Eficientes para la Segmentación de Clientes con Múltiples Atributos

Número de páginas: 141.

Objetivos y método de estudio: El presente trabajo de tesis está enfocado a una problemática real de una empresa distribuidora de productos de la ciudad de Monterrey N.L., México. El objetivo de esta tesis es obtener conocimiento necesario sobre el problema de segmentación de clientes y poder aplicar dicho conocimiento para desarrollar una metodología de fácil implementación y aplicación que abarque los aspectos requeridos por la empresa para resolver el caso de estudio en cuestión y problemas con similar estructura.

Debido al tamaño de las instancias reales no resulta práctico aplicar métodos exactos para su resolución. Es por ello que la metodología propuesta en esta tesis

se basa en una combinación de métodos aproximados o heurísticas para resolver el problema.

Esta metodología consiste en una fase de preprocesamiento, una fase de contrucción y una fase de mejora de soluciones. La primera fase consiste principalmente en el aprovechamiento de la estructura del problema real para poder reducir el número de clientes a segmentar por medio de una pre-agrupación de éstos (creación de metaclientes) y de esta manera minimizar los tiempos de cómputo en las fases posteriores. La fase de construcción de soluciones consiste en obtener una partición inicial dado un número conocido de segmentos utilizando un procedimiento de búsqueda adaptativo, aleatorizado y voraz (GRASP), donde la mejor solución obtenida es la que se reporta como la solución final de esta fase y como la solución inicial de la fase de mejora. Esta última, basada en una estructura de entornos variables (VNS), consiste en mejorar la solución obtenida de la fase anterior por medio de movimientos de inserción e intercambio de clientes a otros segmentos.

CONTRIBUCIONES Y CONLUSIONES: En términos generales, la contribución de este trabajo de tesis consiste en el desarrollo de un modelo matemático que representa la problemática de la empresa con respecto a la segmentación de sus clientes, así como el proporcionar una metodología para resolver dicho problema de manera eficiente. La metodología no solo tiene la ventaja de ser de fácil aplicación sino que también es de fácil entendimiento para el usuario dado que se requiere de pocos elementos para aplicarla. Además ésta fue creada con el fin de dar flexibilidad al usuario al permitir el cambio de parámetros en sus fases correspondientes para dar mayor diversidad de soluciones, las cuales pueden tomarse a consideración al momento de tomar decisiones en la selección de aquélla que se ajuste mejor a los requerimientos de la empresa.

Firma del asesor: _____

Dr. Roger Z. Ríos Mercado

Capítulo 1

Introducción

En un entorno cambiante, donde la competencia es un tema de gran interés, la aplicación adecuada de las estrategias de mercadotecnia ayuda a desarrollar y mantener ventajas competitivas que permiten sobresalir en el mundo corporativo. Es decir, una empresa no puede permitir involucrarse en el mercado sin una clara orientación al cliente que le permita desarrollar estrategias inteligentemente. Es por ello que actualmente se han desarrollado diferentes estrategias para enfrentar a sus competidores buscando necesidades que sus productos o servicios puedan satisfacer a grupos específicos y generar de esta manera ganancias mediante dicha satisfacción. Para ello es necesario entender como dirigirse a los clientes y así encontrar grupos de mercado donde puedan ser exitosos. Es en estos casos en los que surge la importancia de segmentar el mercado.

La *segmentación de mercado* es un proceso que consiste en dividir el mercado total, generalmente heterogéneo, de un bien o servicio, en grupos más pequeños y homogéneos en cuanto a sus deseos, necesidades y posibilidades. Su principal esencia es conocer realmente a los clientes que lo conforman para cubrir dichos requerimientos. Un *segmento de mercado* representa a un grupo de clientes que se pueden identificar dentro de un mercado y cuyas necesidades, poder de compra, ubicación geográfica o actitudes son similares y que además reaccionarán de modo parecido ante una determinada estrategia. El comportamiento del cliente suele ser demasiado complejo como para explicarlo con una o dos características, por lo que se deben tomar en cuenta varias dimensiones partiendo de las necesidades del cliente.

1.1 Descripción

En esta tesis, se aborda un caso de estudio de segmentación de clientes de una empresa embotelladora de bebidas de la ciudad de Monterrey, N.L., México. La problemática que esta empresa enfrenta, es que, dado un conjunto de clientes, se desea particionar dicho conjunto en segmentos de manera que la disimilitud con respecto a cuatro atributos, de relevante importancia para la empresa, sea la menor posible. Estos cuatro atributos son: a) la ubicación geográfica, b) el volumen de compra, c) el tipo de contrato y d) el tipo de establecimiento del cliente. Dado que uno de los atributos es con respecto a la ubicación geográfica se desean encontrar, además, segmentos compactos (clientes que conforman un mismo segmento se encuentren relativamente cercanos). La importancia de obtener segmentos de esta manera surge de la necesidad de la empresa de desarrollar e implementar diferentes estrategias de mercadotecnia y poder así satisfacer las necesidades de sus clientes según sus preferencias o necesidades. Además el obtener segmentos compactos disminuye el costo de transportación del producto y posibles inconformidades entre sus clientes al aplicar diferentes estrategias en cada segmento.

1.2 Motivación

Según Bowen [12], uno de los conceptos de estrategia más importantes contribuidos por la disciplina de la mercadotecnia es el de segmentación de mercados. La importancia de descubrir segmentos, cada uno con características un tanto diferentes, es lo que permite a las empresas ofrecer productos que atiendan las necesidades de los clientes. El encontrar grupos de clientes con necesidades similares hace que éstos sean más fáciles de analizar, dando paso hacia la solución de otros tipos de problemas que no pueden ser identificados en un mercado completamente heterogéneo y variado el cual puede ocultar información interesante y necesaria al momento de tomar decisiones. Es entonces que la segmentación apropiada hace la tarea más fácil.

Además, al combinar la segmentación de clientes con un buen diseño territorial de los mismos podemos obtener resultados que pueden proporcionar un buen soporte a problemas de aplicación real muy comunes en la actualidad, como lo son los problemas de distribución de productos en una área determinada, la determinación del precio de sus productos y como consecuencia, la mejora de atención a las necesidades del cliente, por mencionar algunos.

Es por ello que la motivación en esta tesis es la de desarrollar una metodología sencilla y eficiente para obtener una buena segmentación que cumpla con los requerimientos establecidos por la empresa en un tiempo de cómputo razonable y de esta manera contribuir en el proceso de toma de decisiones al momento de desarrollar e implementar estrategias de mercadotecnia para el mejor posicionamiento del producto en el mercado.

1.3 Objetivos

Los objetivos principales de esta tesis son: (a) adquirir y generar el conocimiento necesario para solucionar el problema en cuestión, (b) desarrollar y proveer un modelo matemático que represente apropiadamente el problema planteado por la empresa, (c) desarrollar un método de solución eficiente para el problema abordado, y (d) evaluar el desempeño del mismo en base a una evaluación empírica. Con el cumplimiento de estos objetivos se logra aportar una herramienta valiosa para el apoyo de la toma de decisiones en el problema segmentación de clientes abordado.

1.4 Alcance

En la presente tesis se trabaja un caso de estudio de una problemática real de una empresa distribuidora de producto. El modelo combinatorio propuesto está dado como un modelo determinista mono-objetivo, el cual pretende encontrar de un conjunto de particiones factibles del conjunto de clientes del problema, la partición cuya

disimilitud (suma ponderada de cuatro atributos considerados importantes para la empresa) sea la mínima.

Para medir la distancia entre un par de clientes se considera la distancia euclídea. Dado que se pretende solucionar un caso real, para fines de esta tesis, se aplicará el método a una muestra de datos reales proporcionada por la empresa. En base a esta instancia real se extraen instancias de diversos tamaños para evaluar los diferentes componentes de la metodología.

1.5 Estructura de la Tesis

Esta tesis se encuentra estructurada de la siguiente manera:

- En el Capítulo 2, se introduce brevemente al lector con temas relacionados al trabajo de tesis, así como algunas de las aplicaciones de la segmentación de mercado encontradas en la literatura.
- En el Capítulo 3, se describe el problema tratado el cual pertenece a una aplicación real de una empresa distribuidora de productos así como la modelación matemática desarrollada.
- En el Capítulo 4, se describe la metodología de solución propuesta para el problema en cuestión la cual consta de una fase de preprocesamiento con base de correlación estadística y una fase de construcción y mejora de soluciones usando métodos heurísticos avanzados.
- En el Capítulo 5, se presenta una evaluación empírica de la metodología propuesta.
- En el Capítulo 6, se presentan las conclusiones de este trabajo de tesis, las contribuciones y algunas recomendaciones para trabajo a futuro de la misma.

Capítulo 2

Marco Teórico

En este capítulo se pretende dar una breve introducción a algunos conceptos teóricos relacionados con el tema a tratar en esta tesis. Se mencionan, además, algunas aplicaciones de la segmentación no solo en problemas de la misma índole sino también en problemas de diferentes áreas de trabajo encontrados en la literatura.

2.1 Antecedentes

La mayoría de las organizaciones, cualquiera que sea su tamaño, buscan tener éxito. Este éxito depende de muchos y diversos factores como la estrategia escogida, la ejecución de dicha estrategia y los sistemas de información existentes, entre otros. Sin embargo, actualmente toda empresa con éxito comparte el hecho de estar centrada en el cliente y su orientación al mercado. Dedican gran parte de su tiempo a identificar y satisfacer las necesidades de los clientes y a desarrollar productos competitivos de alta calidad que proporcionen altos niveles de satisfacción. La mercadotecnia es la función empresarial que más se centra en los clientes para proporcionar valor y satisfacción a sus mercados.

2.1.1 Breve Introducción a la Mercadotecnia

La mercadotecnia (*marketing*) es la ciencia y el arte de explorar, crear y entregar valor para satisfacer las necesidades de un *mercado objetivo* y obtener así una

utilidad. Ésta identifica necesidades y deseos insatisfechos, define, mide y cuantifica el tamaño del mercado y su potencial de utilidad [50].

La definición de mercadotecnia tiene su punto de partida al existir una necesidad, es decir, una carencia de un bien básico, la cual no puede ser creada sino que existe en la misma naturaleza humana, por ejemplo, el alimento, vestido, seguridad, aceptación y autorrealización, entre otros. Las necesidades pueden satisfacerse con algo en específico; sin embargo, ese *algo* no siempre puede encontrarse. Es entonces cuando surge un nuevo concepto llamado *deseo*, es decir, la carencia de algo específico que satisface las necesidades básicas. Por ejemplo, una bebida refrescante sería el deseo de una persona para satisfacer la sed.

Un *producto* es todo aquello que puede ser ofrecido para satisfacer una necesidad o un deseo y usualmente se relaciona con un objeto físico. A todo aquello que satisface una necesidad o un deseo y no puede ser relacionado con algo físico se le llama *servicio*. Las *demandas* son un producto específico, en función de una capacidad de adquisición determinada [51]. Es decir, los deseos se convierten en demandas cuando existe un poder de adquisición. Siguiendo con el ejemplo anterior, una persona puede desear una bebida refrescante pero solamente algunas pueden adquirirla.

Aunque una persona tenga necesidades y deseos no implica que exista mercadotecnia. Ésta comienza una vez que la persona decide satisfacer dichas necesidades y deseos mediante el *intercambio*. El *intercambio* [51] es el acto de obtener un producto deseado de otra persona ofreciéndole algo a cambio. El comprender las necesidades, los deseos y demandas de las personas proporciona información fundamental para diseñar buenas estrategias para que dicho intercambio ocurra, es decir, para dar comienzo al proceso de mercadotecnia.

Estrategia de Mercadotecnia

Otro concepto ampliamente utilizado en el área de la mercadotecnia es el de *estrategia* (vocablo que proviene del griego *estrategas*) cuyo propósito final es el logro de determinados *objetivos*, los cuales se consideran como los resultados de mercado esperados una vez que se apliquen las acciones que se tomarán en dicha área. Es un conjunto de acciones a través de las cuales la empresa espera conseguir una ventaja sobre sus competidores, atraer a los compradores y a hacer el mejor uso de los recursos disponibles. Una estrategia de mercadotecnia debe ser capaz de trazar con precisión el enfoque básico que será utilizado en determinado producto con el fin de que el mismo logre los objetivos de mercado previstos.

Atributos

Otro concepto es el de *atributo*, el cual se puede definir como una característica comercial de un bien o servicio, por ejemplo, grado de alcohol de un vino, sabor de un refresco, tipo de envase de un producto, etc. Ballestero [5] define tres clases de atributos:

- *Atributos incorporados al producto:* Por ejemplo, en una lavadora, peculiaridades de los programas de lavado con los que cuenta. Si se habla de refrescos, el sabor, el nivel de gasificación del líquido, etc.
- *Atributos no incorporados al producto pero que forman parte de su entorno:* Para este tipo de atributos se puede mencionar, por ejemplo, la amabilidad de los vendedores del producto, la presentación en cajas individuales, etc.
- *Precio del producto y condiciones de financiamiento:* El precio y las condiciones de financiamiento son considerados como atributos de una clase especial. Obviamente, el precio es una característica del producto que atrae el interés del comprador. Lo mismo pasa con las facilidades de pago.

Durante la mayor parte de esta tesis, se menciona en varias ocasiones la palabra SKU (acrónimo de *Stock Keeping Unit*) la cual se define como el número de referencia o identificador usado en el comercio que permite el seguimiento sistemático de los productos y servicios ofrecidos a los clientes. Cada SKU se asocia con un objeto, producto, marca, servicio, cargo, etc. En esta tesis se hace uso del término SKU para identificar a un producto en específico que cuenta con determinadas características o atributos.

Segmentación de Mercados

El término *segmentación*, como estrategia de mercadotecnia, fué introducido por primera vez en esta área por Smith [75] al considerarla como una alternativa a la estrategia de diferenciación de producto (estrategia que crea una percepción de un determinado producto por parte del cliente que lo diferencía claramente de los de la competencia) de esos tiempos.

Existen diversas definiciones de segmentación de mercado, sin embargo, todas llegan a un fin específico: obtener ganancias a partir de la satisfacción del cliente. La *segmentación de mercado* se define como la subdivisión de un mercado global en varios submercados con respecto a distintos factores (necesidades de los consumidores, estilo de vida, valores, opiniones, entre otros) con el objetivo de ofrecer a cada segmento lo que demanda [6]. La segmentación analiza las diferentes necesidades en el interior de los mercados para tratar de satisfacerlas de la manera más adecuada. Busca establecer segmentos donde los clientes que integran cada uno de ellos tengan necesidades semejantes (veáse Figura 2.1).

Según Barroso González y Alonso Sánchez [6], existen dos conceptos importantes ligados a la segmentación: *la competencia y el posicionamiento*. La *competencia* surge cuando muchas empresas ofrecen el mismo producto dentro de un mismo mercado y rivalizan entre ellas puesto que tienen objetivos comúnes. En general, en cada mercado existe cierto grado de competencia, del cual se pueden identificar dos extre-

Figura 2.1: Ejemplo de una segmentación ideal. Cada segmento es formado por clientes que comparten las mismas características (representados por el mismo color).

mos, *monopolio* y *competencia perfecta*. La primera surge cuando solo existe un único organismo que ofrece un determinado producto o servicio, o bien, cuando habiendo varios organismos o empresas que lo ofrecen, solo una lo domina y marca las pautas a seguir para las demás empresas. La segunda, representa la competencia elevada a su máximo exponente, en donde las empresas no pueden influir en el precio, existe libertad de entrada y salida tanto de compradores como de vendedores y todos los productos son iguales, en un determinado mercado. En cuanto al *posicionamiento*, una vez que la empresa ha analizado su mercado y ha creado segmentos lo más homogéneos posible, se dedica a crear estrategias para cada uno de los segmentos formados de manera que pueda posicionarse en cada uno de éstos apropiadamente obteniendo el mayor provecho de cada uno de ellos.

Para que la segmentación sea significativa, ambos conceptos deben cumplirse, ya que, si no existe un cierto nivel de competencia no interesa segmentar un mercado. Por otro lado, si la competencia existe, surge la necesidad de segmentar, pero la segmentación no es útil sin el posicionamiento adecuado.

2.1.2 Optimización Combinatoria

La *optimización* es el proceso de intentar encontrar la mejor solución posible a un *problema de optimización* [19]. Un *problema de optimización* es un problema en el que existen varias soluciones y una forma de comparación entre ellas. Éste existe siempre y cuando se disponga de un conjunto de soluciones candidatas diferentes que pueden ser comparadas [19].

Un problema de optimización P puede representarse como se muestra en la siguiente formulación:

$$\begin{gathered}\text{Optimizar } f(x)\\ \text{sujeto a}\\ x \in F \subset S,\end{gathered}$$

donde $f(x)$ es la función objetivo que evaluará a una solución x perteneciente a un conjunto F de soluciones factibles contenidas dentro de un espacio de soluciones S, la cual se desea optimizar (maximizar o minimizar). Este tipo de problemas se pueden dividir en dos categorías: aquéllos en los que la solución está dada por valores *reales* o *continuos* y aquéllos cuyas soluciones están dadas por valores *enteros* o *discretos*. Dentro de la segunda categoría se encuentra un tipo particular de problemas denominados *problemas de optimización combinatoria*.

Un *problema de optimización combinatoria* [63] consiste en encontrar un objeto entre un conjunto finito, o al menos contable, de posibilidades el cual puede ser representado por un número natural o conjunto de éstos, una permutación, una estructura de grafo o subgrafo. Algunos ejemplos de este tipo de problemas son el problema del agente viajero, el problema de asignación cuadrática, el problema de secuenciación de tareas, el problema de partición de conjuntos, entre otros.

Existe un algoritmo exacto que permite obtener la solución óptima de los problemas combinatorios llamado *método enumerativo*. Este método, que consiste en explorar todo el conjunto de posibles soluciones, suele ser ineficiente para la mayoría

de los problemas de optimización combinatoria de interés debido a que el tiempo de cómputo requerido para encontrar la solución óptima del problema (la mejor de todas las posibles) aumenta en forma exponencial o desmesurada conforme el tamaño del problema aumenta (véase Figura 2.2). Sin embargo, existen métodos que, aunque no garantizan que la solución encontrada sea la mejor de todas las posibles, tratan de obtener una solución aproximada en un tiempo de cómputo razonable. Estos métodos son comúnmente llamados *heurísticas o métodos aproximados*.

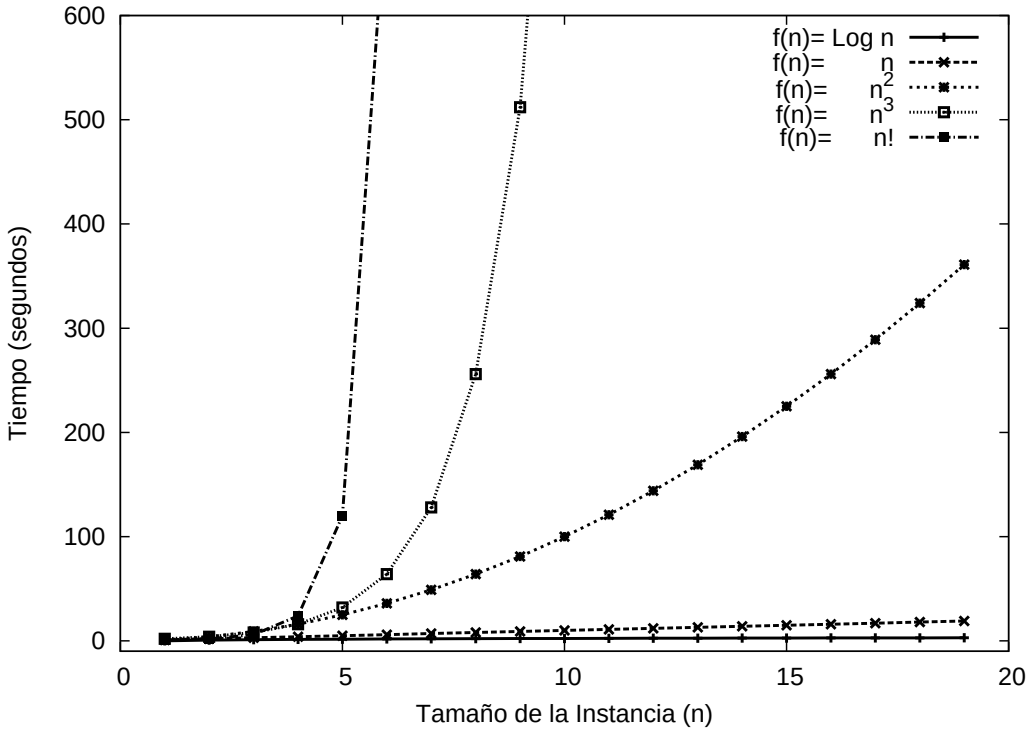


Figura 2.2: Crecimiento del tiempo en función del tamaño de la instancia.

2.1.3 Problemas Fáciles y Díficiles

La teoría de la complejidad computacional fué iniciada por Cook [17], un reconocido científico de la computación quien formalizó la noción de NP-completo. Cook intentó categorizar los requerimientos computacionales de los algoritmos y clasificar los problemas encontrados en la práctica como problemas *fáciles* o *difíciles*. La descripción de cada uno de ellos se menciona a continuación:

Un problema se considera *fácil* si existe o se puede desarrollar un algoritmo que lo resuelva a optimalidad (de forma exacta) en tiempo polinomial conforme aumenta el tamaño del problema [63]. Es decir, un problema se considera *fácil* si existe una manera de obtener la mejor solución (de todas las posibles) en un tiempo de cómputo polinomial. Los problemas de optimización combinatoria tienen la característica de ser usualmente fáciles de describir o representar, pero muy difíciles de resolver.

Un problema de decisión es aquél donde las únicas respuestas posibles son sí o no. Existen varias clases de complejidad, algunas están relacionadas con el espacio (relacionadas con la necesidad de memoria requerida) y otras con el tiempo (relacionadas con la tiempo de cómputo requerido para responder si o no). Una clasificación de ésta última corresponde a las clases de complejidad conocidas como P (de polinomial) y NP (no determinista de tiempo polinomial). De ésta última se puede identificar un subconjunto de problemas denominados NP-completos. Por otro lado existe un conjunto especial de problemas que no pertenecen a la clase NP llamados problemas NP-duros [28, 63].

- **P**: Los problemas de esta clase son aquellos problemas de decisión para los cuales existe un algoritmo polinomial que los resuelve, es decir, que en la práctica estos problemas pueden ser resueltos en un tiempo de ejecución razonable.

- **NP**: Un problema de clase NP es un problema de decisión el cual puede resolverse mediante un algoritmo no determinísta en tiempo polinomial (donde un algoritmo no determinísta es aquél que puede escoger una de varias posibles alternativas existentes). Los problemas P también son NP ya que si un algoritmo determinista resuelve un problema en tiempo polinomial, uno de tipo no determinista tambien puede hacerlo y en igual tiempo.

- **NP-completo**: Los problemas NP-completos [28] son los problemas más difíciles del conjunto de problemas NP. Se piensa que muy probablemente no forman parte de la clase de complejidad P.

- **NP-duro**: Los problemas NP-duros son al menos tan difíciles de resolver que

los problemas NP-completos. Para un problema NP-duro, no puede establecerse que pertenezca a la clase NP. Sin embargo, existe un problema NP-completo que es polinomialmente reducible, dicho de otra manera, transformable a este problema.

De ser demostrada la relación de que P=NP, todos los problemas de NP tendrían también una solución en tiempo polinomial. Demostrar que P=NP o P≠NP es uno de los problemas abiertos más importantes en el campo de la computación teórica, premiado con un millón de dólares. La Figura 2.3 muestra la relación que existe entre los problemas P, NP y NP-completo.

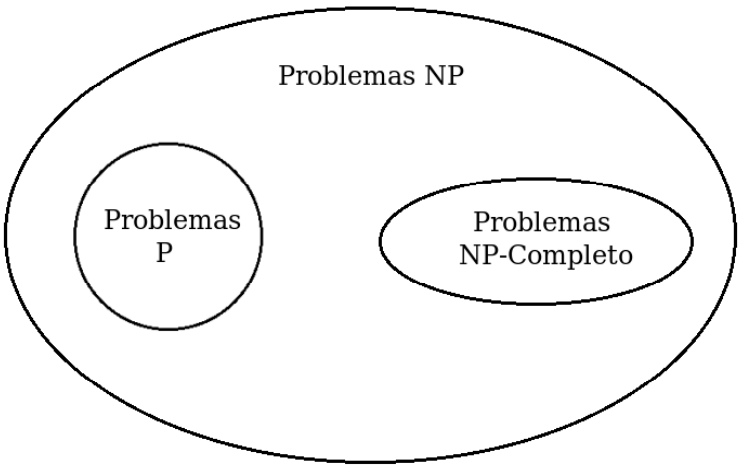


Figura 2.3: Relación entre problemas P, NP y NP-completo [68].

2.1.4 Heurísticas y Metaheurísticas

El término *heurística* proviene del vocablo griego *heuriskein* que significa *encontrar, descubrir o hallar*. Desde un punto de vista científico, el término heurística se debe al matemático Polya [65]. Actualmente, existen muchas definiciones para la palabra heurística. Una de las definiciones más claras e intuitivas es la de procedimientos simples, generalmente basados en el sentido común, con el objetivo de obtener una buena solución (no necesariamente óptima) a problemas difíciles de un

modo sencillo y rápido [19].

Las heurísticas, también conocidas como métodos aproximados, son usualmente utilizadas en problemas de optimización que no pueden ser resueltos a optimalidad en un tiempo de cómputo razonable.

Martí [56] destaca algunas otras razones por las cuales se hace necesaria la aplicación de heurísticas:

- No se conoce algún método exacto para resolver el problema.
- Existe un método exacto para resolver el problema, pero su uso es muy costoso computacionalmente.
- Permite la incorporación de condiciones de díficil modelización que no pueden ser aplicadas a un método exacto.
- Puede utilizarse para proporcionar una buena solución inicial o como paso intermedio de un método exacto.

En los últimos años se han desarrollado métodos aproximados más sofisticados llamados *metaheurísticas*, término que fué dado por primera vez por Glover [29] pretendiendo definirlo como un procedimiento maestro de alto nivel que guía y modifica otras heurísticas para explorar soluciones más allá de la simple optimalidad local.

Una metaheurística es un proceso iterativo que hace uso de una heurística subordinada combinándola con diferentes conceptos de una manera inteligente con el fin de encontrar soluciones cercanas al óptimo de manera eficiente [62].

Algunos ejemplos de metaheurísticas son GRASP, búsqueda tabú (TS), búsqueda dispersa (SS), algoritmos genéticos (GA), búsqueda de entornos variables (VNS) y búsqueda local iterativa (ILS), entre otros. Una sencilla y buena descripción sobre heurísticas y metaheurísticas puede encontrarse en el trabajo de Martí [56].

2.1.5 GRASP

El término GRASP (Greedy Randomized Adaptive Search Procedures) fué introducido por Feo y Resende [24] a final de la década de los ochenta para denominar una nueva técnica metaheurística de propósito general. GRASP es un procedimiento *multiarranque, voraz, adaptativo* y *aleatorizado* para problemas combinatorios, que consiste en la aplicación de una fase de construcción de soluciones y una de post-procesamiento de las mismas (véase Pseudocódigo 1).

En la fase de construcción, GRASP aplica un método que construye iterativamente una posible solución al problema. En cada iteración de esta fase, un elemento es agregado a la solución basándose en una función *voraz*, la cual mide el beneficio de agregar dicho elemento a la solución parcial sin tomar en cuenta que es lo que ocurrirá en iteraciones posteriores. Tal beneficio es actualizado en cada iteración (*adaptativo*). Con la finalidad de ofrecer diversidad de soluciones en cada reinicio del GRASP, y así evitar repetir soluciones en construcciones diferentes, se crea una lista restringida de candidatos (LRC) compuesta generalmente por aquellos k mejores elementos o de más alto beneficio (por cardinalidad), o bien, por aquéllos que se encuentren a un $\alpha\,\%$ del mejor beneficio (por umbral de calidad) seleccionándose uno al azar (*aleatorizado*). Para esta última forma de construcción de la LRC, el parámetro de aceptación de los elementos llamado también *parámetro de umbral de calidad*, denotado por $\alpha \in [0, 1]$, proporciona el rango de aceptación en base al mejor beneficio. Este rango se encuentra entre $[c^{\text{mín}}, c^{\text{mín}} + \alpha(c^{\text{máx}} - c^{\text{mín}})]$, donde $c^{\text{mín}}$ representa el valor del mejor beneficio y $c^{\text{máx}}$ el peor valor (problema de minimización).

La elección tanto de k como de α determinan que tan restringida será esta lista. Por ejemplo, si tenemos 1000 elementos y si se ha decidido crear la LRC por medio de los k mejores, si k es muy pequeño, como por ejemplo $k = 3$, la calidad podría ser mejor pero no habrá tanta diversidad puesto que solo existen pocos elementos que escoger. Sin embargo, si $k = 999$ prácticamente la LRC estaría compuesta por casi todos los elementos y sería casi igual que implentar un método completamente

aleatorio. Para este último ejemplo existiría gran diversidad pero la calidad sería baja con mayor probabilidad. De igual manera para el *parámetro de umbral de calidad*, para los casos extremos, si $\alpha = 0$ la construcción de la solucón sería completamente voraz, mientra que en el caso contrario ($\alpha = 1$) el comportamiento de construcción sería totalmente aleatorio.

Dado que no se garantiza optimalidad local en la fase de contrucción, la solución obtenida de ella es mejorada en la segunda fase mediante un algoritmo de búsqueda local. Este procedimiento se aplica varias veces (*multiarranque*) hasta cumplir un determinado criterio de parada (puede ser el haber alcanzado un número máximo de iteraciones permitidas), reportándose la mejor de las soluciones como resultado final. En el Pseudocódigo 1 se muestra el esquema general de GRASP, en donde Cons-

Pseudocódigo 1 GRASP(Max_iter, α)

Entrada:

Max_iter := Número máximo de repeticiones de GRASP;

α := Parámetro de calidad;

Salida: X^* := Mejor solución encontrada;

1: $X^* \leftarrow \emptyset$;
2: **Para** $k = 1$ hasta Max_iter **hacer**
3: $X \leftarrow$ Construcción(α);
4: $X \leftarrow$ Post_procesamiento(X);
5: **Si** (X es mejor que X^*) **entonces**
6: $X^* \leftarrow X$;
7: **Fin Si**
8: **Fin Para**
9: **Regresar** X^*;

trucción(α) y Post_procesamiento(X) dependen de las características particulares del problema que se trata de resolver. GRASP es repetido Max_iter veces obteniendo en cada una de ellas una solución con un $\alpha\,\%$ de flexibilidad a la hora de la construcción. Dicha solución es mejorada por un procedimiento de post-procesamiento

en donde usualmente una búsqueda local es implementada. La solución se compara con la mejor encontrada anteriormente y se actualiza en caso de mejora.

2.1.6 Búsqueda de Entornos Variables (VNS)

Sea S el espacio de soluciones, $N_k \subset S$ un conjunto finito de entornos predefinidos y $N_k(x) \subset S$ el conjunto de soluciones en el k-ésimo entorno de x. La búsqueda de entorno variable (*VNS*, por sus siglas en inglés) es una metaheurística reciente para resolver problemas de optimización cuya idea básica es el cambio sistemático de entorno N_k dentro de una búsqueda local [35]. La VNS está basada en tres hechos simples:

- Un mínimo local con una estructura de entornos no es necesariamente un mínimo local con otra.
- Un mínimo global es un mínimo local con todas las posibles estructuras de entornos.
- Para muchos problemas, los mínimos locales con la misma o distinta estructura de entornos están relativamente cercanos entre sí.

El cambio de entorno puede realizarse de manera determinista, estocástica o ambas. Si se realiza un cambio de estructura de forma determinista cada vez que se llega a un mínimo local se obtiene una búsqueda de entorno variable descendente (VND), mientras que si se seleccionan soluciones de $N_k(x)$ de forma aleatoria sin aplicarles un descenso se obtiene una búsqueda de entorno variable reducida o RVNS (útil para instancias muy grandes para las cuales la búsqueda local es muy costosa). Por otro lado, si los cambios de estructuras de entorno combinan estrategias deterministas y aleatorias se obtiene la búsqueda de entorno variable básica o BVNS.

Se pueden encontrar otras extensiones de esta metaheurística como VNS general (GVNS), VNS con descomposición (VNDS), VNS sesgada (SVNS), VNS paralela

(PVNS), así como algunas VNS híbridas [35]. Diversas versiones de la VNS han sido comparadas con heurísticas clásicas usadas en problemas de agrupamiento [7, 27, 58].

2.1.7 Métodos de Agrupamiento

El *Análisis de Conglomerados* es una técnica estadística multivariada cuya finalidad es dividir un conjunto de objetos en grupos de forma que las características entre los objetos de cada uno de ellos sean muy similares entre sí (cohesión interna del grupo) y entre los objetos de grupos diferentes sean distintos (aislamiento externo del grupo).

Existen numerosos métodos utilizados para agrupar o particionar un conjunto de elementos. Entre los muchos tipos de métodos que existen en la literatura cabe destacar los *algoritmos de agrupamiento jerárquico* y los *algoritmos de partición*. Estos algoritmos por lo general hacen uso de una determinada *métrica de distancia* o *medida de asociación* [67] para determinar la distancia o similitud entre dos elementos respectivamente.

Una de las métricas usualmente utilizada en los métodos de agrupamiento es la *distancia euclídea* la cual se representa por la longitud del segmento de recta que une a dos puntos i y j por medio de sus coordenadas geográficas (x, y). La *distancia euclídea* está dada de la siguiente manera:

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}, \tag{2.1}$$

donde (x_i, y_i) y (x_j, y_j) representan las coordenadas geográficas de los puntos i y j respectivamente. La Figura 2.4 ilustra el concepto de esta métrica.

Durante la revisión de la literatura se encontró un amplia gama de métodos utilizados en el área de agrupamiento o segmentación. La Figura 2.5 muestra una breve clasificación de algunos de los métodos de agrupamiento encontrados [8, 43].

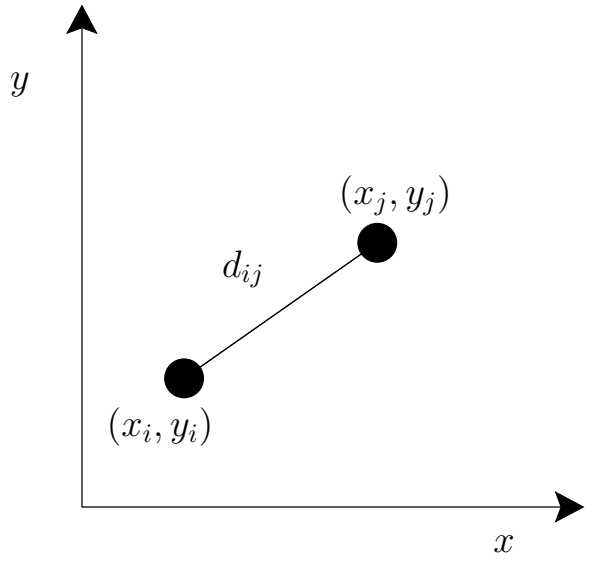


Figura 2.4: Ilustración del concepto de distancia euclídea.

Métodos Jerárquicos

Estos métodos construyen una estructura de tipo árbol jerárquico denominado *dendrograma* (véase Figura 2.6). Estos métodos pueden subdividirse básicamente en dos tipos: *divisivos* y *aglomerativos*.

Divisivos

En este tipo de algoritmos, todos los objetos son asignados en un inicio a un solo grupo. En los pasos subsecuentes a éste, los objetos más *distantes* o *disimilares* son separados del grupo para formar uno nuevo. Este proceso continúa hasta que cada objeto sea parte de un grupo distinto, es decir, que cada grupo solo contenga a un solo objeto.

Aglomerativos

En este tipo de algoritmos, cada objeto es asignado a un grupo diferente. En los pasos subsecuentes, los dos grupos más *cercanos* o *similares* se únen para formar un nuevo grupo conformado por los elementos de cada uno, reduciendo de esta manera

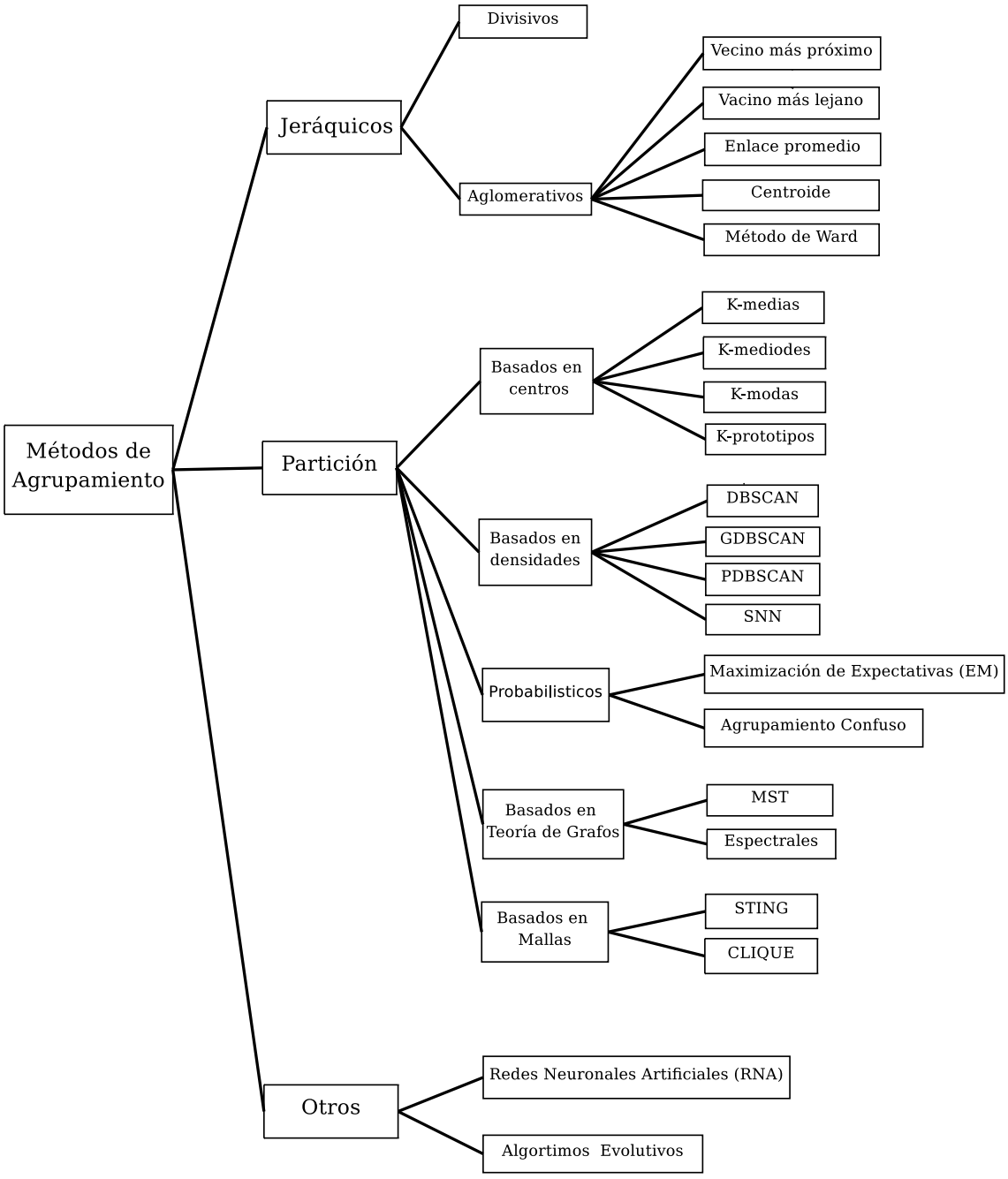


Figura 2.5: Clasificación de los métodos de agrupamiento [8, 43].

el número de grupos en cada paso hasta unirlos todos en uno solo. En la Figura 2.6 se muestra un ejemplo de dendrograma. En el caso de los métodos o algoritmos aglomerativos el dendrograma se va contruyendo de abajo hacia arriba, mientras que en los métodos divisivos se construye de arriba hacia abajo.

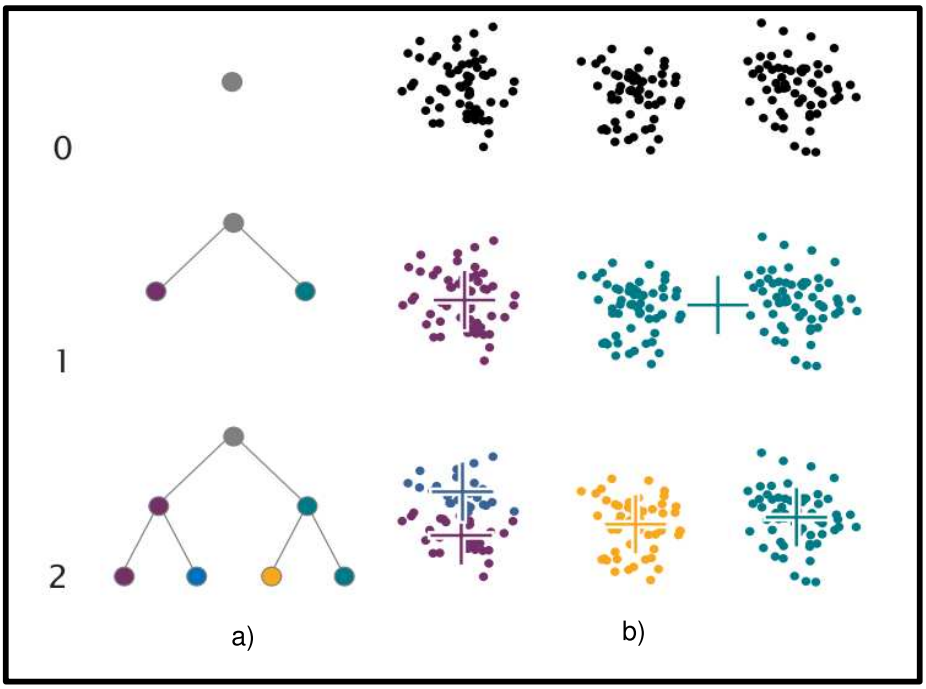


Figura 2.6: Formación de un dendrograma o árbol de jerarquías.

El nivel más alto del dendrograma representa la máxima disimilitud (menor similitud) que puede existir entre los elementos del grupo, esto es el caso en el que todos los elementos han sido ubicados en el mismo grupo. Los niveles intermedios representan grupos de elementos cuya disimilitud corresponde a la obtenida por los clientes que los conforman (el valor de la disimilitud depende del criterio del método que se use para agrupar). El nivel más bajo que puede tener un dendrograma se presenta cuando cada elemento es asignado a un grupo diferente (formado únicamente por dicho elemento) obteniendo de esta manera la menor disimilitud (máxima similitud) posible. Cada rama del dendrograma representa un grupo diferente.

Algunos de los métodos jerárquicos más conocidos son *vecino más próximo*

(single linkage), *el vecino más lejano* (complete linkage), *enlace promedio* (average linkage), *método de Ward* y el *método del centroide*. Una descripción breve se presenta a continuación. Se pueden encontrar en la literatura métodos jerárquicos más sofisticados, como por ejemplo el propuesto por Guha, Rastogi y Shim [32].

- **Vecino más próximo** [34, 54]: Este método consiste en unir los grupos considerando la menor de las distancias existentes entre los objetos más cercanos de distintos grupos. Es decir, mide la proximidad entre dos grupos calculando la distancia entre sus objetos más próximos o la similitud entre sus objetos más semejantes.
  Ventaja: Crea grupos más homogéneos.
  Desventaja: Permite cadenas de alineamientos entre objetos muy lejanos.

- **Vecino más lejano** [34]: A diferencia del método del vecino más próximo, en este método los grupos se unen considerando la menor de las distancias existentes entre los miembros más lejanos de distintos grupos .
  Ventaja: Resuelve el problema del método del vecino más próximo.
  Desventaja: Crea grupos menos homogéneos.

- **Enlace promedio** [34]: En cada paso de este método se unen los dos grupos cuya distancia promedio de todos los elementos de un grupo a todos los elementos del otro sea la mínima. A diferencia de los métodos anteriores, este método no depende de un par de elementos extremos.
  Ventaja: Tiende a combinar grupos con varianza pequeña.
  Desventaja: Tiende a generar grupos por lo general de tamaño similar.

- **Método de Ward** [34]: Este método agrupa elementos de modo que se minimice una determinada función objetivo que por lo general es la suma de las distancias cuadradas intra-grupo.
  Ventaja: No es sensible a objectos distantes (*outliers*).
  Desventaja: Tiende a generar grupos demasiados pequeños y demasiados equilibrados en tamaño.

- **Método del centroide** [34]: En este método, la distancia entre dos grupos está dada por la distancia (usualmente la distancia euclídea o euclídea cuadrada) entre sus centros o centroides. Los centroides de los grupos corresponden a la media o el promedio de cada grupo. En este método cada vez que los objetos son agrupados se procede a recalcular los centroides.

  Ventaja: Es menos afectado por objetos distantes.

  Desventaja: Pueden producir resultados confusos en grupos donde la distancia entre un par de centroides sea menor que la distancia de un par de grupos unidos anteriormente.

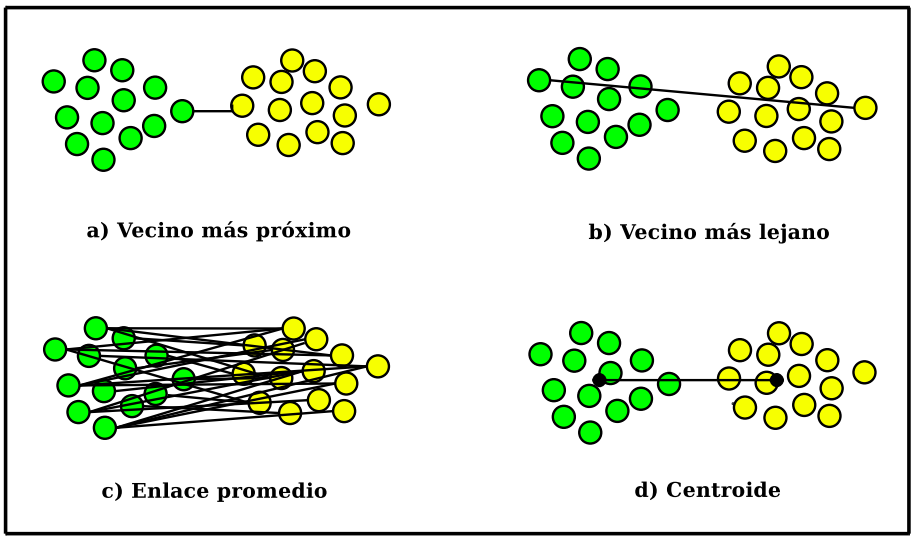


Figura 2.7: Comparación de las medidas de distancia de los métodos jerárquicos [34].

Algoritmos de Partición

Estos métodos, también llamados métodos de partición, no construyen un árbol de jerarquías o dendrograma, en lugar de ello, asignan objetos a grupos una vez que el número de grupos es dado. Dicho de otra manera, particiona el conjunto de elementos en un número de grupos predefinido. Éstos tienden a determinar K grupos que optimizan un cierto criterio. Además, tienen la ventaja de que pueden resolver aplicaciones que involucran conjuntos de datos de gran tamaño para los cuales la

construcción de dendrogramas es computacionalmente ineficiente. Se puede conocer el número de particiones factibles de un conjunto de n objetos en K grupos utilizando la siguiente fórmula [13]:

$$\text{Particiones Factibles} = \frac{1}{K!}\sum_{k=0}^{K}(-1)^k\binom{K}{k}(K-k)^n. \tag{2.2}$$

La Tabla 2.1 muestra algunos valores de la fórmula (2.2) para diferentes combinaciones de n y K. Como puede observarse en la tabla, el número de particiones factibles crece considerablemente al aumentar el número de elementos (n) y el número de grupos a formar (K). Para un conjunto de elementos de tamaño $n = 30$ y un número de grupos $K = 6$ la implementación de un método exacto como el método enumerativo puede resultar muy ineficiente para obtener la mejor partición. Una instancia de tamaño $n = 17000$ y un número de segmentos $K = 60$ resultaría prácticamente imposible enumerar todas las posibles combinaciones para resolverla de manera exacta.

Núm. de elementos (n)	Núm. de grupos (K)	Particiones factibles
10	2	511
20	2	524287
30	2	536870911
10	4	34105
20	4	45232115901
30	4	48004081105038304
10	6	22827
20	6	4306078895384
30	6	29931010274694863257 6

Tabla 2.1: Número de particiones factibles para diferentes valores de n y K.

Algunos de los algoritmos de partición, encontrados en la literatura, se describen brevemente a continuación.

Algoritmos basados en centros

- **Algoritmo K-medias**: Un algoritmo sumamente utilizado, por su sencilla implementación computacional y rapidez para particionar un conjunto de objetos, es el algoritmo de K-medias [36]. La utilización de este algoritmo es conveniente cuando los datos a agrupar son muchos o cuando se busca refinar un agrupamiento obtenido mediante un método jerárquico. Para este algoritmo se supone que el número de grupos es conocido con anticipación. La Figura 2.8 muestra un ejemplo de este algoritmo cuyos pasos son los que se muestran en el Pseudocódigo 2. El algoritmo comienza seleccionando aleatoriamente K

Pseudocódigo 2 K-medias(V)

Entrada: V {Conjunto de elementos}

Salida: X, K {Mejor partición encontrada, centroides de dicha partición}

1: Seleccionar K elementos de V;
2: **Para todo** $i \in V \setminus K$ **hacer**
3: Asignar i elemento a su centroide $k \in K$ más cercano.
4: **Fin Para**
5: Recalcular K (media aritmética de cada grupo de la partición)
6: Repetir pasos 2-4 hasta que se satisfaga un criterio de parada

elementos (llamados centros o centroides) de un conjunto V los cuales serán los elementos representativos de cada grupo a formar. Una vez seleccionados dichos elementos, el resto del conjunto V es asignado a su centroide más cercano según una medida de distancia o similitud (Paso 2-4). Cuando todos los elementos han sido asignados, se procede a recalcular los centroides de cada grupo formado, calculando la media aritmética de cada grupo (Paso 5). Los Pasos 2-4 se repiten hasta que se cumple un determinado criterio de parada, como por ejemplo, número máximo de iteraciones alcanzado, la partición no mejora o la ubicación de los centroides ya no cambió después del recálculo. El método suele ser muy sensible a la configuración inicial de centros. Una mala selección de éstos puede ocasionar que el algoritmo no pueda encontrar

una partición mejor después de ciertas iteraciones quedando atrapado en un óptimo local de menor calidad.

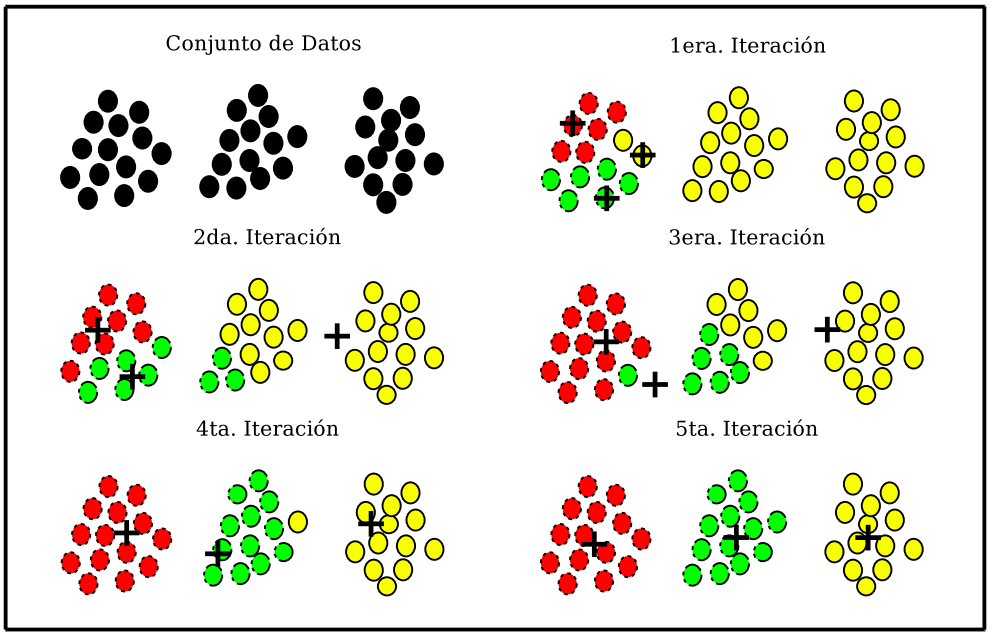


Figura 2.8: Ejemplo de agrupamiento usando el algoritmo K-medias.

En la Figura 2.9 se puede observar que los centroides finales (indicados por +) obtienen una partición de mala calidad ya que la selección inicial no pudo obtener la agrupación correcta (cada grupo representado por una nube de puntos).

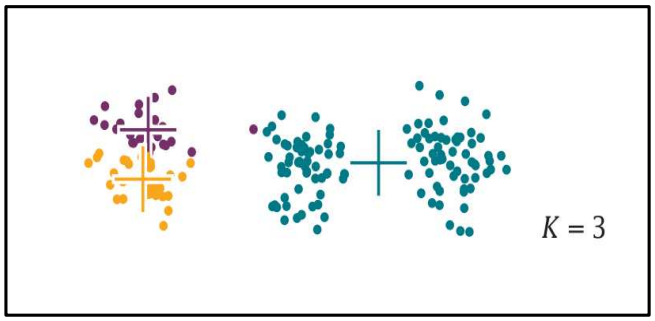


Figura 2.9: Sensibilidad K-medias con respecto a la selección de centros iniciales.

En la agrupación obtenida, una nube de puntos se divide en dos grupos y las dos nubes restantes se unieron para crear uno sólo, por lo que es conveniente utilizar algún procedimiento adicional para que dicha configuración sea buena inicialmente. Una forma de construirla puede ser mediante una clasificación obtenida por un algoritmo jerárquico o por alguna otra manera sistemática para obtener dicha configuración inicial de centroides.

- **Algoritmo K-medioides**: En el algoritmo K-medioides [8] un grupo es representado por uno de sus elementos (medioide). Un medioide puede ser definido como el objeto de un grupo cuya disimilitud con respecto a todos los demás objetos (del mismo grupo) es la mínima. A diferencia del algoritmo K-medias, éste tiene la ventaja de ser más flexible en cuanto al tipo de atributo (para el K-medias debe ser numérico) y menos sensible a puntos extremos. Cuando los medioides han sido seleccionados, los grupos se forman uniendo el resto de los elementos al medioide más cercano y la función objetivo que se desea minimizar es la distancia promedio o alguna otra medida de disimilitud entre un punto y su medioide.

  Algunas versiones de éste algoritmo son el llamado *partición en torno a medioides* (PAM) y el *agrupamiento para grandes aplicaciones* (CLARA, por Clustering Large Applications) [49].

  PAM [49] comienza seleccionando K objetos llamados medioides (inicialmente puede ser de manera aleatoria) del conjunto de objetos. Los grupos son formados asignando los objetos restantes a su medioide más cercano. Si la función objetivo puede ser mejorada por un intercambio entre un medioide y un objeto que no lo es, entonces el intercambio se lleva a cabo. Esto se repite hasta que no se encuentre mejora en la función objetivo. Una desventaja que presenta este algoritmo en comparación con el de K-medias, es su complejidad computacional, la cual es de orden cuadrático $O(n^2)$ mientras que la del algoritmo K-medias es lineal $O(n)$. Sin embargo, se han desarrollado diferentes versiones del mismo para eliminar este problema. Una de estas versiones se describe a

continuación.

CLARA (Clustering LARge Applications) [49] es una extensión del algoritmo K-medioides que soluciona el problema del manejo de grandes conjuntos de datos. La diferencia entre PAM y CLARA es que el segundo se basa en muestreos. Este método consiste en usar solamente una pequeña parte del conjunto de datos en lugar de utilizarlo todo. Dependiendo del tamaño de la muestra del conjunto de datos, la eficiencia es mejor en comparación a la de K-medioides. Una vez seleccionada la muestra a usar, PAM es usada en cada una de ellas. La calidad del agrupamiento depende de dicha selección. La intuición es que si la muestra es seleccionada de manera aleatoria, entonces es representativa del conjunto total de datos, y los medioides serán similares tal y como si hubieran sido escogidos del conjunto total de datos. Una extensión de CLARA que aplica lo último mencionado, es el método llamado CLARANS (Clustering Large Applications based upon Randomized Search) [60].

- **Algoritmo K-modas**: Este algoritmo [39, 41] es otra extensión del algoritmo K-medias aplicado a problemas con atributos categóricos. Por lo general usa una medida de asociación [67] para medir la relación entre los objetos. A diferencia del algoritmo K-medias los objetos representativos del grupo están dados por las modas (los más frecuentes) de cada uno de ellos y no por la media. Para recalcular las modas después de haber asignado todos los objetos se basa en algún método basado en frecuencias.

- **Algoritmo K-prototipos**: Este algoritmo [42] es una combinación de los algoritmos K-medias y K-modas utilizado en problemas que utilizan atributos tanto numéricos como categóricos con el fin de eliminar las limitaciones que el algoritmo de K-medias tiene con respecto a el tipo de atributo que debe ser usado. El algoritmo K-prototipos consiste en seleccionar inicialmente K objetos distintos aleatoriamente, a los que se les denomina *prototipos*, del conjunto de datos. Posteriormente para cada objeto que no es considerado prototipo, se obtiene la distancia a su prototipo más cercano (basada en una función objetivo

ponderada que depende de ambos tipos de atributos) y se recalcula el prototipo después de cada asignación. El prototipo para los atributos numéricos representa la media del grupo formado hasta el momento y el prototipo para los atributos categóricos es representado por el valor del atributo (categórico) más frecuente hasta el momento. Una vez asignados todos los objetos de la forma mencionada, estos son reubicados de manera similiar al proceso anterior excepto que después de cada reubicación, tanto el grupo anterior como el actual son actualizados.

Métodos basados en densidades

Los métodos basados en densidades [8] toman el principio de la densidad de puntos para agrupar los elementos. Consideran a los grupos como regiones densas de puntos que están separadas por regiones de baja densidad. Existen algunos algoritmos de este tipo como DBSCAN y algunas versiones distintas del mismo, SNN y DENCLUE, por mencionar algunos.

El algoritmo DBSCAN (Density Based Spatial Clustering of Applications with Noise) [23], define conceptos como punto central (puntos que tienen en su vecindad una cantidad de puntos mayor o igual que un umbral específico), borde y ruido. Este algoritmo comienza seleccionando un punto p arbitrario, si p es un punto central, se comienza a construir un grupo y se ubican en su grupo todos los objetos *denso-alcanzables* desde p. Si p no es un punto central, se visita otro objeto del conjunto de datos. El proceso continúa hasta que todos los objetos han sido procesados. Los puntos que quedan fuera de los grupos formados se llaman puntos ruido, los puntos que no son ni ruido ni centrales se llaman puntos borde. De esta forma DBSCAN construye grupos en los que sus puntos son o puntos centrales o puntos borde. Un grupo puede tener más de un punto central. Otras versiones del método DBSCAN son el algoritmo GDBSCAN (Generalized DBSCAN) [69] el cual puede agrupar elementos tanto numéricos como categóricos y el algoritmo PDBSCAN (Parallel DBSCAN) [85] el cual es una versión en paralelo del método DBSCAN creado para resolver problemas de gran tamaño en un tiempo menor y el cual hace uso de un

cierto número de ordenadores para realizar una partición del conjunto de datos distribuyendo dicha partición en cada uno de ellos para luego aplicar DBSCAN en cada uno.

Otro algoritmo basado en densidad es el algoritmo SNN (Shared Nearest Neighbors) [22] el cual consiste en encontrar a los vecinos más cercanos de cada punto del conjunto de datos y define la similitud entre cada par de puntos en términos de cuántos vecinos más cercanos comparten los dos puntos. Usando esta definición de similitud elimina ruido y puntos extremos, identifica puntos centrales y construye grupos alrededor de éstos.

Métodos basados en Teoría de Grafos

Los métodos basados en Teoría de Grafos [43] definen grupos a partir de un grafo (representación gráfica de una red que consiste en un conjunto de vértices y un conjunto de aristas que unen a dichos vértices) derivado de una medida de similitud. La definición del grupo se hace simplemente en función de la representación gráfica. Para cada par de objetos se calcula el valor numérico que indica su similitud de modo que se genera un grafo. Métodos de este tipo son los métodos de agrupamiento espectral [3] y los métodos basados en la construcción de un árbol de mínima expansión [43].

- **Agrupamientos Espectrales**: El agrupamiento espectral [3] se basa en algoritmos de partición de grafos a partir de su matriz de adyacencia. Se le llama espectral debido a que hace uso del *espectro* de dicha matriz (donde el espectro representa al conjunto formado por todos los valores propios de esa matriz) para poder agrupar los datos. Estos métodos tienen la ventaja de que las soluciones a los problemas de agrupamiento se obtienen utilizando métodos de álgebra lineal estándar y usualmente son más eficientes que algunos algoritmos de agrupamiento tradicionales. Algunas aplicaciones de estas técnicas pueden encontrarse en los trabajos de Kurucz et al. [52], Ng, Jordan y Weiss [59], Von Luxburg [55] y Verma y Meilă [79].

- **Construcción de un Árbol de Mínima Expansión (MST)**: Otros de los métodos basados en teoría de grafos son aquéllos que involucran la construcción de un *árbol de mínima expansión* (MST) de un determinado conjunto de datos [43]. Una vez obtenido dicho grafo, las aristas de mayor longitud son eliminadas para generar los grupos. La Figura 2.10 muestra un ejemplo de este método. Una aplicación de agrupamiento de este tipo puede encontrarse en el trabajo de Xu y Olman [86], quienes proponen tres métodos de agrupamiento basados en un MST cada uno de los cuales minimiza una determinada función objetivo (la distancia total de lo K sub-árboles, la distancia del centro los demás elementos del grupo y la distancia total del mejor elemento representativo a los demás). Dicho trabajo fué aplicado a dos conjuntos de datos de expresión de genes.

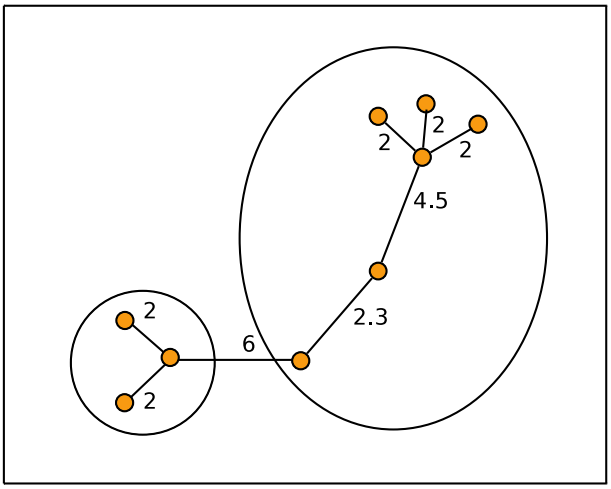


Figura 2.10: Ejemplo de agrupamiento usando el MST [43].

Métodos probabilísticos

- **Maximización de Expectativas (EM)**: El algoritmo EM (Expectation Maximization) es utilizado en los casos en los que no se conoce la distribución de cada dato y no conocemos los parámetros de las distribuciones. Este algoritmo comienza adivinando los parámetros de las distribuciones y los usa para obtener las probabilidades para las cuales cada objeto pertenezca a un grupo. Usa dichas probabilidades para re-estimar los parámetros hasta converger.

- **Agrupamiento Confuso**: En los métodos tradicionales de partición, cada elemento solo puede pertenecer a un solo grupo. Un agrupamiento confuso o borroso [43], en cambio, asocia a cada elemento un grado de pertenencia con cada grupo. Esta asociación la hace mediante el uso de una función de pertenencia. El resultado no necesariamente es una partición del conjunto de datos más bien representa un agrupamiento donde cada grupo está formado por un conjunto confuso o borroso de todos los elementos.

  La Figura 2.11 ilustra un ejemplo de agrupamiento confuso en donde un agrupamiento encontrado por métodos tradicionales de partición está representado por los círculos pequeños de diferente contorno, mientras que un agrupamiento confuso (grupos delimitados por los círculos grandes) algunos de los elementos pueden ser compartidos en dos o más grupos.

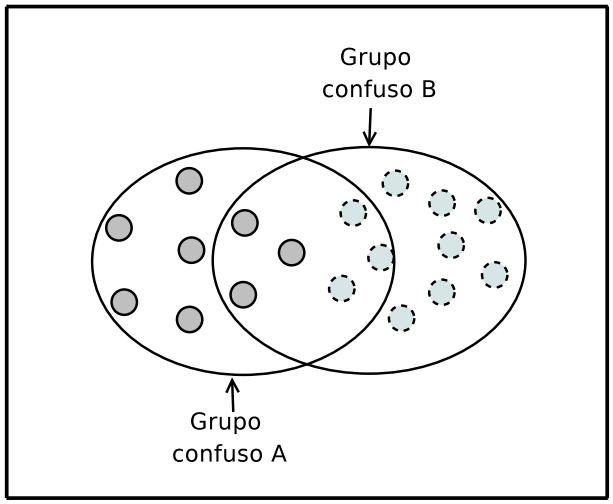


Figura 2.11: Ejemplo de agrupamiento confuso [43].

Métodos basados en celdas o mallas

Estos métodos dividen el espacio en un número finito de celdas que forman una estructura de rejilla en la que se lleva a cabo el agrupamiento. Las celdas que contienen un cierto número de puntos son tratadas como regiones densas y son conectadas para formar grupos. Algunos algoritmos de agrupamiento que pertenecen

a esta clase son: STING (STatistical INformation Grid-based method), CLIQUE (CLustering In QUEst) y Wavecluster.

El método STING [83] primero divide el área en diferentes celdas rectangulares de diferentes niveles de manera que se forme una estructura jerárquica la cual representa la información de agrupamiento para diferentes niveles. Por otro lado, CLIQUE [1] comienza buscando todas las regiones densas en un espacio unidimensional correspondiente a cada uno de los atributos. Una vez encontras dichas regiones, genera un conjunto de celdas bidimensionales, posiblemente densas, asociadas con un par de celdas unidemensionales. Este procedimiento es realizado hasta obtener un conjunto de dimensión k, posiblemente denso, combinando celdas de dimensión $k-1$. Wavecluster [72], mientras tanto, usa la transformación de wavelets para transformar el espacio original al dominio de las frecuencias mediante una transformación matemática de tipo convolución. Es un método muy potente, sin embargo, no es eficiente en espacios de alta dimensionalidad.

Otros métodos

Otras métodos de agrupamiento están basados en redes neuronales artificiales y métodos evolutivos, entre otros.

- **Redes Neuronales Artificiales (ANN por sus siglas en inglés)**: Una *red neuronal artificial* es un sistema basado en el cerebro humano [44, 88]. Es un esfuerzo por simular dentro de hardware especializado o software sofisticado, las múltiples capas de elementos de procesamiento más simples llamadas neuronas. Cada neurona se une a la otras vecinas con coeficientes variantes de conectividad que representan las fuerzas de estas conexiones. Las *redes neuronales artificiales* están inspiradas por la función del cerebro biológico y por consiguiente mucha de la terminología se encuentra relacionada.

  Existe una gran variedad de modelos de ANN, estos dependen del objetivo para el cual fueron diseñados y del problema práctico que solucionan. Uno de

las modelos conocidos en el área de agrupamiento que utiliza ANN es el de los mapas auto-organizados (Self-Organizing Map, SOM) [80].

- **Métodos Evolutivos**: Estos métodos son motivados por la evolución natural. Hacen uso de operadores evolutivos (selección, recombinación y mutación) y de una población de soluciones para la obtención de una partición, óptima globalmente, de los elementos [43]. Considera a las soluciones candidatas para el agrupamiento como cromosomas. Éstos son evaluados según una función que determina la probabilidad de sobrevivencia que dichos cromosomas tienen en la siguiente generación.

  En el área de agrupamiento, los algoritmos evolutivos escogen una población de soluciones aleatoriamente. Cada una de estas soluciones representa una partición del conjunto de elementos. Cada solución es evaluada y se aplican los operadores evolutivos a dichas soluciones para generar la siguiente población de soluciones. Este último paso se repite hasta que se cumpla un criterio de parada. Algunos métodos de este tipo son los algoritmos genéticos [30], estrategias de evolución [4] y programación evolutiva [70].

2.1.8 Validación de una Partición

Como se pudo observar en la sección anterior, existe una gran gama de métodos para agrupar objetos con características similares. Los métodos de agrupamiento generalmente producen particiones de un conjunto de datos sin tomar en cuenta si existen o no grupos reales. Cada uno de estos métodos puede identificar grupos en donde los elementos que los conforman son diferentes. Es entónces cuando surge la necesidad de saber cual de ellos es el más conveniente, es decir, cual proporciona la mejor partición de un conjunto de datos.

En apoyo a esta decisión, existen algunos índices que miden la calidad de una partición dada. Dichos índices incluso son aplicados para obtener el número óptimo de grupos a formar (desventaja que presentan muchos métodos de agrupamiento en

los que el número de grupos debe conocerse con anticipación).

Estos índices, conocidos como *índices de validación de agrupamientos*, evalúan la calidad de un agrupamiento o partición resultante. Ortega Lobo et al. [61] mencionan dos criterios usados comúnmente para validar un agrupamiento:

1. *Compacidad:* Los elementos que conforman un grupo deben estar lo más cercanos (similares) posible.

2. *Separación:* Los grupos deben estar lo más separados posible (la similitud entre los grupos de una partición debe ser mínima, o dicho de otra manera, que un grupo no sea muy parecido a otro dentro de una misma partición).

Se puede encontrar en la literatura una descripción más detallada sobre estos índices así como algunas de sus aplicaciones en problemas de agrupamiento [2, 10, 33, 48, 61, 64, 66, 82]. Algunos de los índices más representativos encontrados en la literatura se mencionan a continuación.

1. **Índice de Huberts (H)** [61]:

$$H = \frac{1}{M} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} X(i,j)Y(i,j) \tag{2.3}$$

donde M es el número total de parejas del conjunto de datos de tamaño N ($M = \frac{N(N-1)}{2}$), $X(i,j)$ y $Y(i,j)$ son las matrices asociadas a las particiones $\bar{X}$ (encontrada con un método de agrupamiento) y $\bar{Y}$ (partición real conocida del conjunto de datos) definidas como:

$X(i,j) = \{1$, si x_i y $x_j \in \bar{X}$ pertenecen a grupos diferentes, 0 en otro caso$\}$;
$Y(i,j) = \{1$, si y_i y $y_j \in \bar{Y}$ pertenecen a grupos diferentes, 0 en otro caso$\}$.

Donde $x = (x_1, \ldots, x_N)$ y $y = (y_1, \ldots, y_N)$ corresponden a elementos de las particiones $\bar{X}$ y $\bar{Y}$ respectivamente. Grandes valores de este índice indican fuerte similitud entre $\bar{X}$ e $\bar{Y}$.

2. **Índice de Davies-Bouldin (DB)** [48, 61]:

$$DB = \frac{1}{K}\sum_{i=1}^{N} \max_{j=1,\ldots,K;j\neq i}(d_{ij}), \tag{2.4}$$

donde $d_{ij} = \frac{S_i+S_j}{d(c_i,c_j)}$, K es el número de grupos, S_i es la distancia promedio de todos los elementos del grupo i a su respectivo centro c_i y $d(c_i, c_j)$ es la distancia entre los centros de los grupos c_i y c_j. De manera que el valor de d_{ij} es pequeño si los grupos i y j son compactos y sus respectivos centros se encuentran lejanos uno del otro. Por lo tanto, un valor pequeño de (2.4) significará que se cuenta con un buen agrupamiento. El cálculo de este índice tiene un orden de complejidad $O(n)$.

3. **Índice de Dunn (D)** [33, 38]:

$$D = \frac{d_{\text{mín}}}{d_{\text{máx}}}, \tag{2.5}$$

donde $d_{\text{mín}}$ representa la distancia más pequeña entre dos elementos de diferentes grupos, mientras que $d_{\text{máx}}$ es la distancia más grande entre dos elementos del mismo grupo. Grandes valores de D indican la presencia de mejores agrupamientos. Este índice es de fácil interpretación pero muy inestable con la presencia de puntos extremos ya que solamente dos distancias son consideradas. El orden de complejidad del cálculo de este índice es $O(n)$.

4. **Índice C** [33, 57] :

$$C = \frac{S + S_{\text{mín}}}{S_{\text{máx}} - S_{\text{mín}}}, \tag{2.6}$$

para este índice, S es la suma de todas las distancias de todos los pares de elementos que conforman un mismo grupo (l es el número de esos pares), $S_{\text{mín}}$ es la suma de todos las l distancias más pequeñas si se consideraran todos los pares de elementos, $S_{\text{máx}}$ es la suma de las l distancias más grandes si todos los pares de elementos fueran considerados. La ventaja de este índice es que la evaluación de calidad es muy acertada. Su desventaja es que requiere que los grupos sean del mismo tamaño lo cual no aplica en muchos casos. El orden de complejidad requerido para su cálculo es $O(n^2)$.

5. **Índice de Silhouette (S)** [64]:

$$S = \frac{1}{K}\sum_{k=1}^{K} S_k, \tag{2.7}$$

donde $S_k = \dfrac{\text{mín}(\bar{d}_b(k,j)) - \bar{d}_w(k)}{\text{máx}(\bar{d}_w(k), \bar{d}_b(k,j))}$, $\bar{d}_w(k)$ es al distancia promedio desde el elemento k a los demás elementos de su mismo grupo y $\bar{d}_b(i,j)$ es la distancia promedio desde el elemento i a todos los demás elementos de otro grupo j. Es decir, que para cada elemento k, S_k es la medida que representa que tan similar es dicho elemento con respecto a los demás que conforman el mismo grupo. La suma promedio de esos valores representa el valor total de este índice, el cual se encuentra dentro del rango [-1, 1]. Valores altos de este índice indican mayor calidad de la partición.

2.2 Aplicaciones de Segmentación

Existen muchas áreas de aplicación donde la segmentación es fundamental para poder manejar mejor la información [9, 11, 74, 76]. Algunas de las aplicaciones encontradas en la literatura se describen en esta sección.

Bruco y Stahl [13] desarrollan un método exacto basado en ramificación y acotamiento para el problema de agrupamiento enfocándose en la minimización de uno de los siguientes criterios: diámetro de la partición, suma de distancias intra-grupo y dos variaciones más de ésta última. El método es probado para instancias de seis elementos agrupados en dos segmentos (instancias pequeñas) y veintiuno agrupados de dos a ocho segmentos obteniendo particiones óptimas con respecto al diámetro de la partición y suma de las distancias intra-grupo, por mencionar algunos.

Martí et al. [14] desarrollan un procedimiento metaheurístico para problemas de agrupamiento multiobjetivo, basado en búsqueda tabú y búsqueda dispersa el cual fué aplicado a un problema de segmentación de mercados que consiste en encontrar un agrupamiento de clientes que maximiza dos criterios para mostrar la efectividad

del procedimiento. Este procedimiento, llamado SSPMO, consta de dos fases. La primera consiste en la generación de un conjunto inicial de puntos eficientes por medio de $p+1$ búsquedas tabú, donde p es el número de objetivos a considerar. La segunda consiste en combinar las soluciones de dicho conjunto y actualizarlo usando una búsqueda dispersa.

Golsefid, Ghazanfari y Alizadef [31] proponen una función basada en conceptos de reglas de asociación y en el valor de exportación de productos básicos, la cual se introdujo en el algoritmo K-medias y fué aplicada a un caso de estudio de segmentación de clientes de la Organización de Promoción de Comercio de Irán (TPO). Esta función mide la disimilitud entre las canastas de exportación de diferentes países. Los criterios que fueron usados para segmentar son el valor de los productos básicos, el tipo de producto y la correlación entre productos de exportación. Para determinar la calidad de la segmentación con respecto a esta función se comparó la calidad obtenida usando la distancia euclídea como métrica. También se calculó el número óptimo de segmentos obteniendo el máximo valor de los promedios de las mínimas similitudes encontrados para diferentes valores de K. Una vez encontrado el mejor número de segmentos, se aplicó el algoritmo K-medias usando la función propuesta. El beneficio de la partición obtenida es analizada utilizando el modelo RFM (Recency, Frecuency and Monetary) [45] el cual mide el intervalo entre la ocasión más *reciente* de exportación, la frecuencia de exportación y el valor total monetario dentro de un periodo específico. Esta metodología fué aplicada a una instancia de 210 países a los cuales Irán, según información de las bases de datos de la TPO, exporta bienes y servicios (se obtuvieron 222078 transacciones relacionadas con la exportación a estos países). Los bienes y servicios son alrededor de 16 mil tipos los cuales fueron categorizados en 99 grupos. Los resultados obtenidos mostraron que las partición encontrada con la la función propuesta superó a la encontrada con la distancia euclídea. Se evaluaron las particiones para encontrar el mejor número de segmentos (usando un rango de $K = \{2, 3, \ldots, 10\}$) resultando $K = 5$ el mejor de todos. Se aplicó RFM para obtener las características de cada segmento.

Scheuerer y Wendolsky [71] proponen un método heurístico basado en búsqueda dispersa y búsqueda tabú para un problema de agrupamiento con capacidad. El objetivo es particionar un conjunto de clientes en K segmentos de manera que se minimice la suma de las distancias euclídeas de todos los centros a todos los clientes de su respectivo segmento y que a la vez no exceda un límite de capacidad conocido para cada grupo formado.

Hartuv y Shamir [37] desarrollaron un algoritmo polinomial para el análisis de segmentos basado en técnicas sobre teoría de grafos. Ellos definen un grafo de similitud de cual los segmentos de dicho grafo corresponden a subgrafos altamente conexos. El objetivo es encontrar segmentos de manera que se satisfagan dos criterios: homogenidad entre los elementos del mismo segmento y separación entre elementos pertenecientes a diferentes segmentos. La distancia entre un par de elementos esta dada por el mínimo número de aristas que existen entre ambos. El algoritmo fué probado sobre datos simulados de expresión de genes y mostró buenos resultados aún en presencia de altos niveles de ruido.

Zhang Fern y Brodley [25], basados en dos técnicas ya existentes (propuestas por Strehl y Ghosh [77]), proponen una técnica de formulación de grafos que permite modelar un problema de ensamble de grupos (combinar múltiples agrupamientos o particiones para formar un agrupamiento superior) como un *grafo bipartito* el cual puede ser resuelto eficientemente. Esta técnica consiste en obtener un conjunto de agrupamientos o particiones (ensamble de grupos) $C = \{C^1, \ldots, C^R\}$ y a partir de éste construir un grafo $G = (V, W)$, donde V representa al conjunto de vértices formado tanto por los grupos como por el conjunto de n elemetos a agrupar y W es el peso de las aristas que unen dichos vértices donde $W = 1$ si dada una pareja de vértices i, j el vértice i corresponde a un vértice elemento y j a un vértice de grupo y además i pertenece a j, $W = 0$ en otro caso. El grafo resultante es un *grafo bipartito*. La Figura 2.12 ilustra un ejemplo del funcionamiento del método. El método fué aplicado a cinco instancias construidas aleatoriamente (agrupadas usando el algoritmo K-medias) y se comparó con los propuestos por Strehl y Ghosh para tres

tamaños de ensambles (20, 40 y 60). Para evaluar la calidad de los agrupamientos hicieron uso del criterio NMI (Normalized Mutual Information) [77]. Los resultados mostraron que la metodología propuesta obtuvo los mejores resultados.

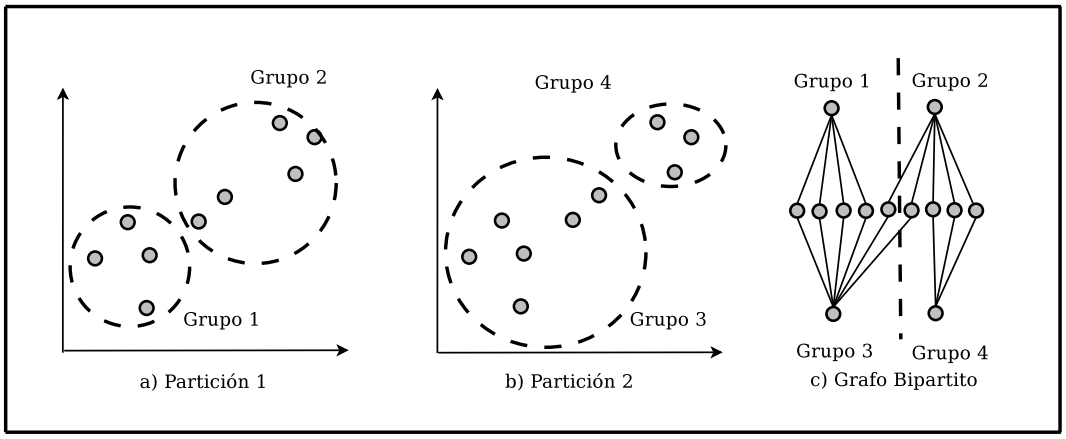


Figura 2.12: Ejemplo del método propuesto por Zhang Fern y Brodley [25].

Negreiros y Palhano [58] proponen dos modelos de agrupamiento llamados p-CCCP y g-CCCP para un problema de agrupamiento con capacidad limitada por cada grupo resultante. Para el primero, el número de segmentos es conocido y se desea minimizar la suma de las disimilitudes en cada segmento. Para el segundo, se desea encontrar el número de segmentos que agrupa a todos los elementos con la mínima disimilitud posible. Debido a que el problema de agrupamiento que no toma en cuenta algún límite de capacidad por cada segmento formado es NP-duro, proponen un algoritmo de dos fases. La primera fase construye una solución usando el algoritmo de Forgy [16], y la segunda fase consiste en una fase de mejora basada en una filosofía VNS la cual consiste de dos estrategias: centro más cercano y centro aleatorio. En la primera estrategia se escogen aleatoriamente elementos los cuales son insertados en el segmento más cercano. La segunda consiste en escoger elementos aleatoriamente e insertarlos en un segmento seleccionado de forma aleatoria.

Sheng y Liu [73] proponen una búsqueda local la cual es hibridizada con un algoritmo genético para un agrupamiento de K-medioides de un conjunto grande de datos, el cual es un problema de optimización conocido como NP-duro. La búsqueda

local propuesta selecciona K centros o medioides de un conjunto de datos y trata, eficientemente, de minimizar la disimilitud interna total para cada segmento. Dado que la solución obtenida es un óptimo localmente, la búsqueda local es hibridizada con un algoritmo genético para obtener de esta forma la búsqueda local llamada HKA (*Hybrid K-medioid Algorithm*). Este algoritmo se probó con dos conjuntos de datos de expresion de genes y fué comparado con algoritmos genéticos existentes para problemas de agrupamiento. Los resultados mostraron que HKA puede obtener mejores soluciones y de manera más eficiente.

Cano et al. [15] proponen un método para el problema de agrupamiento basado en un GRASP para tratar de eliminar el problema de alcanzar óptimos locales (problema frecuente del algoritmo K-medias). El método propuesto aplica el algoritmo de Kaufman como base en la fase de contrucción para obtener soluciones iniciales (en la cual una lista restringida de candidatos es creada con los objetos más prometedores a ser centros representativos de los segmentos) y el algortimo K-medias como un algortimo de búsqueda local. El método fué comparado con cuatros heurísticas de agrupamiento simples obteniendo mejores resultados.

Vicente, Rivera y Mauricio [81] adaptan la metaheurística GRASP para la resolución del problema de agrupamiento basado en los principios del algoritmo K-medias. El algoritmo llamado GRASPKM, aprovecha la rápida convergencia del K-medias evitando el incoveniente de alcanzar óptimos locales. El algoritmo consta de tres fases: *inicialización, construcción* y *búsqueda de la mejor solución*. En la primera fase se obtienen los K centros iniciales al igual que los segmentos de dichos centros. La segunda fase es una adaptación del algoritmo K-medias, la cual se realiza principalmente sobre la asignación de los objetos (asignar elementos a su centro más cercano) creando una lista restringida de candidatos (RCL) que contenga a los posibles segmentos a los cuales puede ser reasignado un objeto, el cual se selecciona de forma aleatoria. La tercera fase corresponde a la implementación de un algoritmo de mejora basado en el algoritmo propuesto por Fränti y Kivijärvi [26] aplicando algunos cambios. Se hicieron comparaciones contra un algoritmo genético

existente para un problema de agrupamiento llamado KGA-clustering y contra el algoritmo propuesto por Cano et al. [15]. Los resultados superaron en algunas instancias al KGA-clustering y en el resto fueron muy parecidos. Mientras que para la comparación con el algoritmo de Cano et al., GRASPKM obtuvo resultados de mejor calidad.

Xia et al. [84] proponen un algoritmo de agrupamiento basado en un modelo de regresión lineal. Dicho modelo mide la relación que existe entre la lealtad de los clientes (variable dependiente) y la satisfacción de los mismos (variable independiente). El objetivo consiste en encontrar que tanta relación existe entre ambas variables. Para ello es necesario conocer bien a sus clientes siendo necesaria la aplicación de la segmentación. Este modelo fué probado en un caso real de una industria de telefonía celular en China usándose una muestra real de 2708 individuos compradores de teléfonos celulares mayores a los 18 años de edad. Los resultados mostraron que para algunos grupos la relación entre la satisfacción y lealtad era significativa, mientras que para otros la lealtad era independiente de la satisfacción.

Decker, Scholz y Wagner [18] aplican un método basado en redes neuronales artificiales y uno basado en el algoritmo de K-medias a un problema de segmentación de consumidores de productos alimenticios. Para obtener el número óptimo de segmentos a formar utilizaron el criterio *JUMP* propuesto por Sugar y James [78]. Los datos utilizados correspondieron a una submuestra real de 4266 consumidores y 37 segmentos (encontrados usando el critero de Sugar y James). Los métodos fueron comparados obteniendo resultados muy similares. La conclusión de dichos autores es que las deficiencias que puede presentar el algoritmo elegido para agrupar pueden no ser determinantes si el procesamiento y elección de los datos se realiza correctamente.

Guha, Rastogi y Shim [32] proponen un algoritmo de agrupamiento, llamado ROCK (Robust Clustering using Links), para datos con atributos binarios y categóricos basado en el contexto de enlaces (links) para medir la similitud ó aproximación entre un par de puntos. Este algoritmo pertenece a la clase de los algoritmos de agrupamiento jerárquico. Dados un par de puntos o nodos, los enlaces de éstos re-

presentan los puntos o nodos vecinos que ambos comparten. Dada esta información, los puntos que pertenecen a un mismo grupo comparten una mayor cantidad de vecinos y por lo tanto más enlaces. El objetivo es maximizar el número de enlaces para cada par de puntos pertenecientes a un mismo segmento y al mismo tiempo minimizar el número de enlaces para cada par de puntos pertenecientes a diferentes segmentos. Una extensión de este algoritmo es propuesto por Hernández Valadez [40].

Capítulo 3

Planteamiento del Problema y Modelación

La segmentación de clientes consiste en dividir a un conjunto de clientes en grupos definidos, con diferentes necesidades, recursos, ubicación, comportamiento, etc., que podrían requerir de diferentes productos y/o servicios al igual que distintas estrategias de mercadotecnia. De esta manera, la empresa que requiere de dicha segmentación prepara perfiles de aquellos segmentos resultantes e identifica diversas maneras de atacar cada uno de ellos ya sea con determinados servicios, publicidad y precios, por mencionar algunos. En este capítulo se describe el problema de segmentación abordado y el modelo matemático propuesto.

3.1 Descripción del Problema

Como ya se ha mencionado anteriormente, en este trabajo se trata con un problema real. El objetivo principal es encontrar una partición de un conjunto de clientes cuya disimilitud con respecto a cuatro atributos principales sea la mínima posible. Se espera obtener segmentos lo más homogéneos posible conforme a estos atributos, es decir, que los clientes que conforman cada segmento tengan atributos similares. Dado que uno de los atributos considerados es la ubicación geográfica, se requiere que los segmentos resultantes estén formados por clientes que se encuentren a un cierto nivel de cercanía. La principal razón de segmentar a los clientes de esta

manera es debido a requerimientos de la empresa para la aplicación posterior de diferentes estrategias de mercadotecnia en cada segmento establecido.

3.2 Datos y Supuestos

Dado que se trabaja con un caso de estudio, los datos proporcionados forman parte de una muestra real de una instancia típica del problema. El problema abordado es un problema combinatorio con datos deterministas. Debido a que es un caso muy particular de la empresa, se desarrolló un modelo matemático que representa el problema planteado por la misma. Este modelo es mono-objetivo, en donde la función que mide el costo de la partición se representa por la suma ponderada de las disimilitudes entre cuatro atributos del cliente. Estos atributos forman una parte importante en el momento de la segmentación y tienen que ver con el *tipo de contrato* y el *tipo de establecimiento* del cliente, sus respectivas *coordenadas geográficas* y el *volumen de compra* (medido en número de cajas) que el cliente demanda de un determinado producto ofrecido por la empresa. Cada uno de esos productos, usualmente llamados por la empresa como SKUs (veáse sección 2.1.1), cuenta de igual manera con un determinado número de atributos: si es o no retornable, en que presentación se ofrecen y la marca del producto.

Los requerimientos de la empresa con respecto a los atributos asociados al cliente son: (a) la disimilitud en cuanto al tipo de contrato (establecimiento) de una pareja de clientes toma valor 0 si el tipo de contrato (establecimiento) de dicha pareja corresponde al mismo, de lo contrario toma valor 1; (b) para medir la disimilitud con respecto a la ubicación geográfica entre una pareja de clientes, se considera la distancia euclídea entre sus respectivas coordenadas geográficas; (c) para el caso del atributo volumen de compra, se considera una medida que considera la diferencia entre las proporciones de los volúmenes que los clientes i y j demandan del conjunto de SKUs.

La Tabla 3.1 muestra los atributos que se estudian en esta tesis y sus niveles correspondientes. El tipo de SKU cuenta a su vez con tres atributos (*retornabilidad, presentación* y *marca*) y sus correspondientes niveles.

Atributo	**Niveles**
SKU	**Retornabilidad :** Retornable y No retornable **Presentación:** 6.5 Oz, 8 Oz., 200 ml., 250 ml., 310 ml., 340 ml., 12 Oz., 458 ml., 400 ml., 500 ml., 600 ml., 750 ml., 1 l., 1.5 l., 2 l., 2.5 l., Bolsa en Caja (Bag In Box). **Marca:** Colas, Sangría, Toronja, Lima limón, Sabor, Naranja, Toronja 2, Manzana, Agua Gasificada, Agua Saborizada, Agua Purificada, Jugo BF, Jugo MM, Té Helado, Energetizante P, Energetizante B, Energetizante TE.
Contrato	B, C, CB, CBO, CO, CP, CPB, CPBO, CPO, P, PB, PO.
Establecimiento	Restaurante de servicio rápido, Restaurante de servicio completo, Farmacia, Abarrotes locales, Vendedor ambulante, Rutas al hogar, Escuela primaria/secundaria, Todos los demás, Comerciante en general, Licor/cerveza/vino/refresco, Tienda de alimentos especializada, Otros alimentos y bebidas, Tienda de conveniencia, Servicio automotriz, Gobierno, Venta al mayoreo, Industria/agric/servicio público, Diversión, Abarrotes de descuento, Servicios de venta al detalle, Tiempo Libre, Preparatoria/universidad, Recreación, Hospedaje, Bar/establecimiento bajo licencia, Servicios, Contrato al mayoreo, Escuela comercial/técnica, Salud/hospital, Transporte, Oficina, Club de precios, Minisúper.

Tabla 3.1: Atributos identificados en la muestra real.

3.3 Modelo Matemático

Un modelo matemático es la descripción, en forma matemática, de una situación real. En esta sección se propone una modelo matemático para representar el problema abordado en esta tesis.

Conjuntos e índices

V Conjunto de clientes; $V = \{1, 2, ..., n\}$.

K Conjunto de segmentos; $K = \{1, 2, ..., p\}$.

S Conjunto de tipos de SKUs; $S = \{1, 2, ..., l_1\}$.

C Conjunto de tipos de contrato; $C = \{1, 2, ..., l_2\}$.

E Conjunto de tipos de establecimiento; $E = \{1, 2, ..., l_3\}$.

n Número de clientes.

l_1 Número de SKUs.

l_2 Número de contratos.

l_3 Número de establecimientos.

i, j Índices de clientes; $i, j \in V$.

k Índice de segmentos; $k \in K$.

Parámetros

d_{ij} Distancia euclídea entre el cliente i y el cliente j; $i, j \in V$.

A Matriz (clientes/SKUs), $A = (a_{is})$, a_{is} representa el volumen en cajas que un cliente i demanda del SKU s; $i \in V$, $s \in S$.

c_i Tipo de contrato del cliente i, $i \in V$, $c_i \in C$.

e_i Tipo de establecimiento del cliente $i, i \in V, e_i \in E$.

Parámetros calculados

q_{ij}^{SKU} Disimilitud entre un par de clientes i y j con respecto al atributo SKU, $i, j \in V$.

h_{ij} Disimilitud entre un par de clientes i y j con respecto al tipo de contrato, $i, j \in V$ donde $h_{ij} = 0$ si $c_i = c_j$ y $h_{ij} = 1$ en otro caso.

g_{ij} Disimilitud entre un par de clientes i y j con respecto al tipo de establecimiento, $i, j \in V$ donde $g_{ij} = 0$ si $e_i = e_j$ y $g_{ij} = 1$ en otro caso.

Variables

X_k Conjunto de clientes asignados al segmento k, $k \in K$.

$m(k)$ Nodo centro del segmento k, $k \in K$.

Sea X una p-partición de V dada por $X = (X_1, ..., X_p)$. Por definición, X cumple las siguientes condiciones [13]:

1. $X_k \neq \emptyset, k \in K$.
2. $X_{k_1} \cap X_{k_2} = \emptyset$; $k_1, k_2 \in K$, $k_1 \neq k_2$.
3. $\cup_{k \in K} X_k = V$.

La primera condición establece que un segmento X_k no puede estar vacío, es decir, que debe asignársele al menos un cliente. La segunda condición se refiere a que dos segmentos X_{k_1} y X_{k_2} no pueden tener como miembro de éstos a un mismo cliente. Por último, la tercera condición asegura que todo cliente $i \in V$ debe ser asignado a un segmento.

Sea entonces Π la colección de todas las p-particiones factibles de V. El problema consiste en encontrar una p-partición $X = (X_1, ..., X_p)$ cuya disimilitud con respecto a los atributos del cliente (dispersión, volumen de compra, tipo de contrato y tipo de establecimiento) sea mínima:

$$\min_{X \in \Pi} f(X) = \alpha_1 f_{disp}(X) + \alpha_2 f_{sku}(X) + \alpha_3 f_{cont}(X) + \alpha_4 f_{est}(X), \tag{3.1}$$

donde

$$\alpha_1, \ldots, \alpha_4 \in [0, 1],$$
$$\alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 = 1$$

y $f_{disp}(X), f_{sku}(X), f_{cont}(X)$ y $f_{est}(X)$ son las medidas de disimilitud para los atributos de dispersión, volumen de compra, tipo de contrato y tipo de establecimiento, respectivamente. Éstas se describen con mayor detalle a continuación.

3.3.1 Dispersión

La disimilitud con respecto a la dispersión es de vital importancia para la empresa debido a que se requieren segmentos compactos para poder garantizar que el promotor que atenderá a un determinado segmento recorrerá una distancia razonable para atender a todos sus clientes. Además otra razón por la cual se requiere de segmentos compactos está basada en la reducción de inconformidades entre sus clientes en cuanto al tipo de estrategia implementada en su respectivo segmento.

Es decir, supongamos que se decide aplicar en un segmento A una promoción como *"en la compra de un refresco de cola llevate gratis una playera"* y otra promoción diferente en un segmento B como *"en la compra de un refresco de cola más veinte pesos llevate un llavero"*. Entonces, si los segmentos A y B están formados tal que no existe una buena compacidad, habrá un número de clientes fronterizos bastante grande, y por consecuencia, una mayor incidencia de inconformidad por parte de los clientes del segmento B al enterarse que la promoción del segmento A es más tentadora. En cambio, si ambos segmentos fueran formados tal que la compacidad fuera mejor, existe un menor número de clientes fronterizos y por ende existe menor probabilidad de que la cantidad de clientes que estuviesen inconformes sea mayor.

La Figura 3.1 muestra la diferencia entre dos segmentos compactos y no compactos. Aplicando el ejemplo anterior, podemos observar que el número de clientes que probablemente se muestren inconformes es mayor cuando los segmentos no son compactos (seis clientes) que cuando si lo son (dos clientes).

Para medir la disimilitud con respecto a este atributo se tomaron en cuenta cuatro formas distintas las cuales pueden clasificarse como las *basadas en centros* y las *no basadas en centros.*

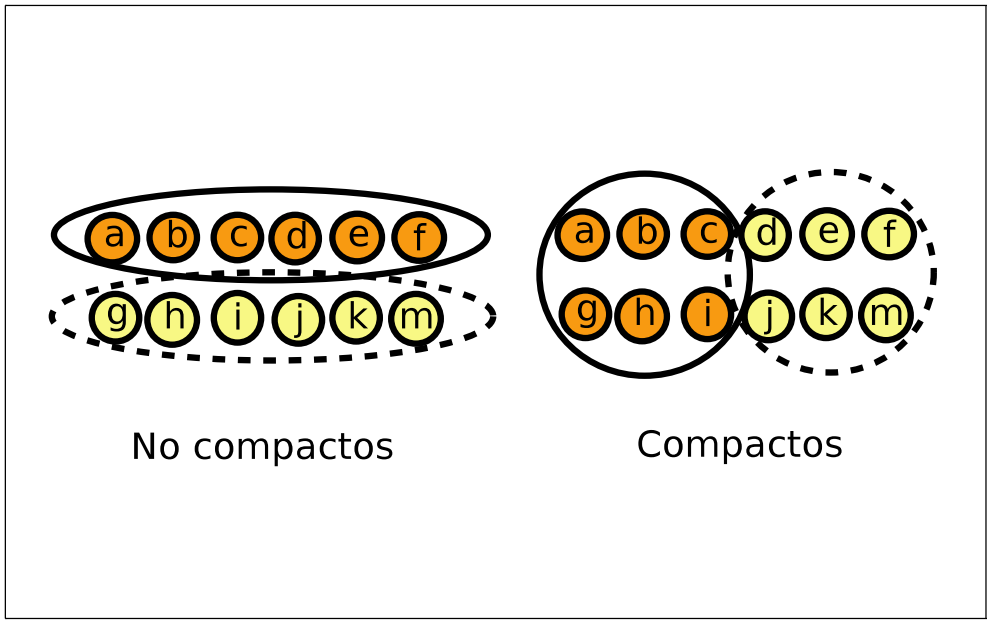


Figura 3.1: Ejemplo de segmentos compactos y no compactos.

Dispersión Basada en Centros

Para medir la dispersión con respecto a un centro utilizamos dos funciones distintas llamadas p-centro y p-mediana.

p-centro

El problema del p-centro es un problema de localización conocido como NP-duro [46]. Consiste en colocar p centros y asignar clientes a ellos (cada uno puede servir a diferente número de clientes) de manera que se minimice la máxima distancia entre un cliente y su centro. Las aplicaciones de este problema son usualmente la localización de estaciones de bomberos, policía, ambulancias o unidades de urgencias.

Esta definición aplicada al problema de segmentación de clientes abordado en esta tesis, podemos definirla de la siguiente manera. Dada una partición factible X, se requiere evaluar la mayor distancia que existe entre el cliente central $m_1(k)$ y un cliente $j \in X_k$ perteneciente al mismo segmento (esto para todos los segmentos de la partición). De dichas distancias se desea minimizar la mayor de todas. De

esta manera ninguna de las distancias restantes será mayor a ésta, por lo tanto, los segmentos serán a lo sumo tan dispersos como la distancia máxima que se ha minimizado. La siguiente ecuación muestra lo antes mencionado:

$$f_{disp1}(X) = \max_{k \in K} \left(\max_{j \in X_k} \left\{ d_{m_1(k),j} \right\} \right), \tag{3.2}$$

donde el cliente central $m_1(k)$ para cada segmento k, es aquél cuya máxima distancia entre él y todos los demás clientes pertenecientes al mismo segmento es la mínima,

$$m_1(k) = \arg\min_{i \in X_k} \left(\max_{j \in X_k} d_{ij} \right). \tag{3.3}$$

p-mediana

Por otra parte, el problema de la p-mediana considera una situación en la que se requiere particionar un conjunto de clientes en exactamente p grupos. Cada grupo estará definido no sólo por el conjunto de clientes que lo forman, sino también por la ubicación de su mediana. Las medianas a su vez determinan el costo del grupo. Cada cliente es asignado a la mediana más cercana. El objetivo consiste en encontrar una partición de costo mínimo del conjunto de clientes. El problema de la p-mediana es un problema NP-duro [47] cuyas aplicaciones están basadas en la ubicación de almacenes, fábricas, sucursales bancarias y estaciones de ferrocarril, por mencionar algunas.

Este tipo de medida representa la suma de las distancias euclídeas del cliente central $m_2(k)$ a todos los demás clientes del segmento. Se suma para todos los segmentos de la partición

$$f_{disp2}(X) = \sum_{k \in K} \sum_{j \in X_k} d_{m_2(k),j}, \tag{3.4}$$

donde $m_2(k)$ está dado por aquel cliente cuya suma de distancias hacia todos los demás es la mínima (mediana)

$$m_2(k) = \arg\min_{i \in X_k} \left(\sum_{j \in X_k} d_{ij} \right). \tag{3.5}$$

Disimilitud de Dispersión no Basada en Centros

Con el objetivo de evaluar el comportamiento de la solución sin considerar la dispersión en base a un centro, se utilizan dos criterios más para medir dicha dispersión conocidos como *diámetro de la partición* y *suma de las distancias intragrupo* [13].

Diámetro de la Partición

Para un segmento X_k, la máxima distancia entre una pareja de objetos de ese segmento es llamada *diámetro del segmento*. El máximo diámetro de entre todos los p segmentos es llamado *diámetro de la partición*. Minimizar esta métrica produce segmentos compactos en el sentido de que aquellos clientes con mayor dispersión son ubicados en diferentes segmentos. Al minimizar el diámetro de la partición los demás segmentos disminuyen sus diámetros resultando ser más compactos cada vez,

$$f_{disp3}(X) = \max_{k \in K} \left(\max_{i,j \in X_k} \{d_{ij}\} \right). \tag{3.6}$$

Suma de las Distancias Intragrupo

Este tipo de medida consiste en sumar todas las distancias de los elementos que conforman cada segmento de la partición. El minimizar esta suma, se indica que cada segmento está conformado por clientes relativamente cercanos,

$$f_{disp4}(X) = \sum_{k \in K} \sum_{i<j \in X_k} d_{ij}. \tag{3.7}$$

3.3.2 Volumen de Compra

Para medir la disimilitud entre los clientes que forman un partición X con respecto a este atributo, se ha tomado como referencia la manera de medir dicha disimilitud propuesta por la empresa,

$$f_{sku}(X) = \sum_{k \in K} \sum_{i<j \in X_k} q_{ij}^{SKU}, \tag{3.8}$$

donde

$$q_{ij}^{SKU} = \sqrt{\sum_{s \in S} \left(\frac{a_{is}}{a_i^T} - \frac{a_{js}}{a_j^T} \right)^2}, \tag{3.9}$$

$$a_i^T = \sum_{s \in S} a_{is}. \tag{3.10}$$

Es decir, la disimilitud q_{ij}^{SKU}, entre dos clientes i y j está dada por la suma de diferencias entre los porcentajes relativos de sus respectivos volúmenes de SKU dada por la ecuación (3.9). La ecuación (3.10) expresa el volumen total para el cliente i. Esto se realiza para todos los segmentos de una partición para obtener la disimilitud total de la misma (3.8). La Figura 3.2 ilustra un ejemplo de como medir la disimilitud entre dos clientes con respecto al volumen de compra.

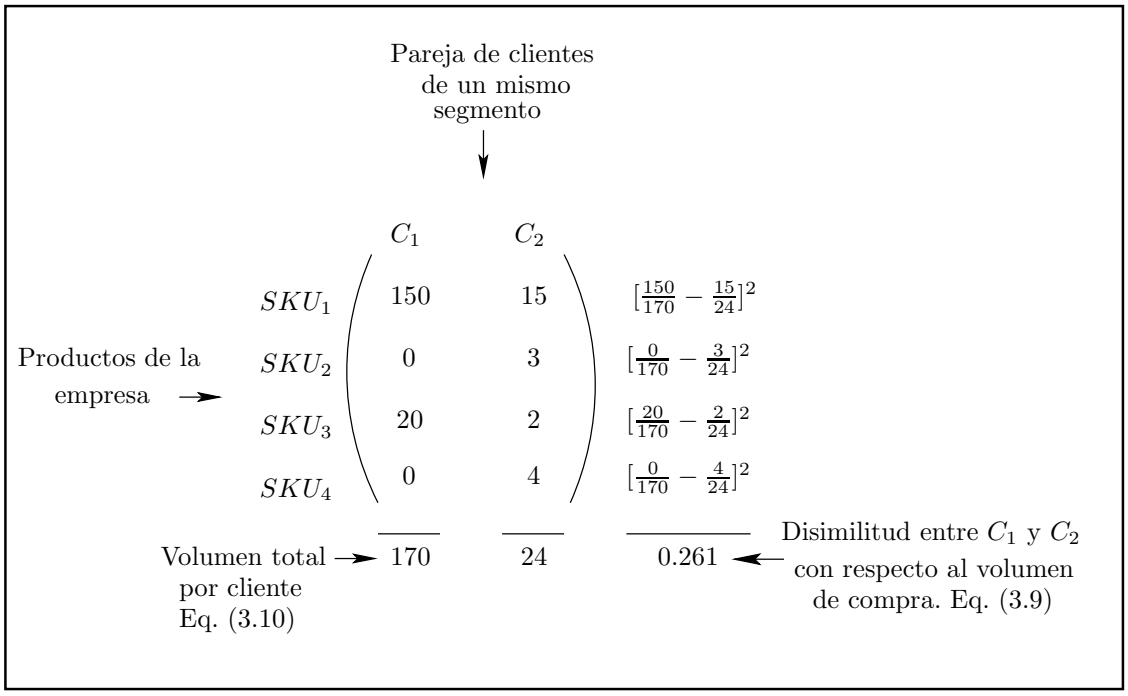


Figura 3.2: Ejemplo de disimilitud entre cliente C_1 y cliente C_2 con respecto al volumen de compra de cuatro diferentes SKUs.

3.3.3 Tipo de Contrato y Tipo de Establecimiento

Para estos tipos de atributos, por requerimiento de la empresa, los valores que pueden tomar solamente son igual a $h_{ij} = 0$ ($g_{ij} = 0$) si los contratos (establecimientos) son iguales o $h_{ij} = 1$ ($g_{ij} = 1$) en caso contrario. Entonces el cálculo de la disimilitud de una partición dada está dada por la suma las disimilitudes de todos los clientes pertenecientes a cada segmento. Las ecuaciones (3.11) y (3.12) muestran la medida de disimilitud para el tipo de contrato y tipo de establecimiento, respectivamente,

$$f_{cont}(X) = \sum_{k \in K} \sum_{i<j \in X_k} h_{ij}, \tag{3.11}$$

$$f_{estab}(X) = \sum_{k \in K} \sum_{i<j \in X_k} g_{ij}. \tag{3.12}$$

Capítulo 4

Metodología de Solución

La metodología de solución que se plantea en esta tesis está basada en el uso de heurísticas o métodos aproximados los cuales no garantizan que la solución encontrada sea la óptima o la mejor de todas, sino que será una solución aproximada que se pretende sea buena y que puede encontrarse en tiempo razonable. Este capítulo se divide en tres secciones principales: preprocesamiento de datos, construcción de particiones y mejora de la solución.

En la sección de preprocesamiento de datos se mencionan los pasos que se siguieron para reducir la instancia real y aprovechar mejor los recursos computacionales. Por otro lado, en la sección de construcción de particiones se describe el algortimo p-medias y la adaptación de éste al problema abordado. Por último en la sección de mejora de la solución se describe el método de búsqueda local desarrollado para mejorar la solución del algoritmo p-medias.

La Figura 4.1 muestra un diagrama de flujo de los principales pasos de la metodología propuesta.

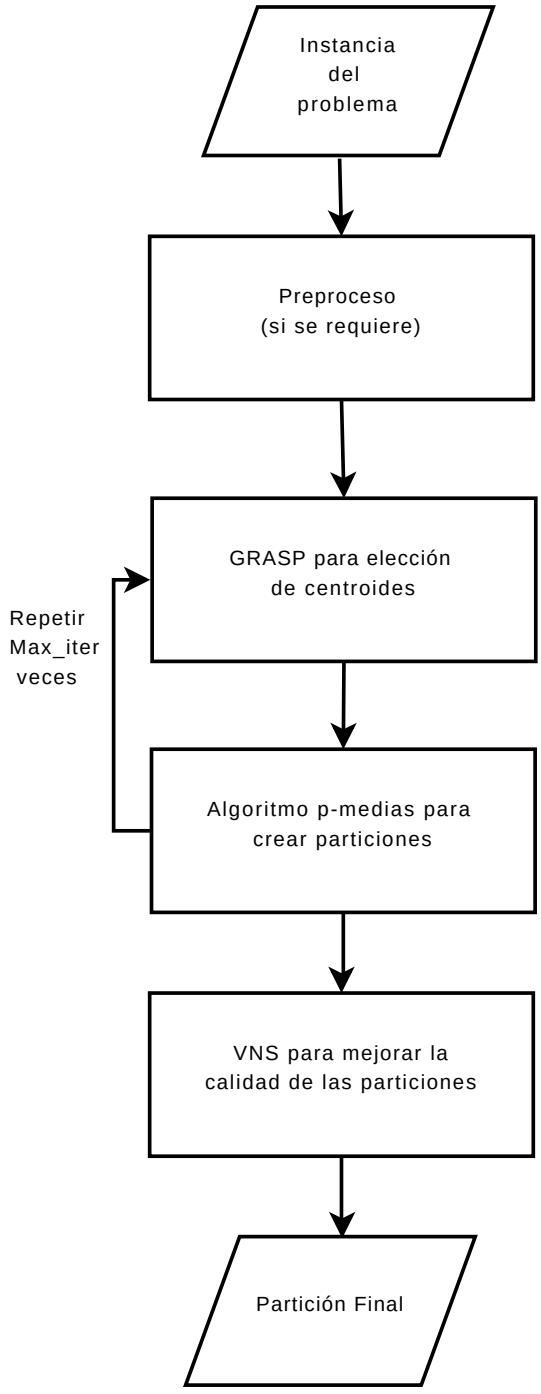


Figura 4.1: Esquema general de la metodología propuesta.

4.1 Preprocesamiento de Datos

En esta fase se pretende analizar la estructura de los datos de manera que se reduzca la dimensión de los mismos y trabajar solamente con aquéllos que verdaderamente requieran de operaciones adicionales para su manejo. Como se trata un caso de estudio, se requirieron datos reales los cuales fueron brindados por la empresa.

Esta fase puede ser opcional, debido a que puede no ser muy significativa para instancias de tamaño pequeño; sin embargo, para instancias demasiado grandes la aplicación de ésta puede reducir el tiempo de cómputo de las siguientes dos fases de la metodología.

Para el preprocesamiento de datos, primeramente, se trata de reducir el número de SKUs, agrupando a aquéllos que sean muy similares. Dada tal reducción, se procede a obtener una matriz (clientes-atributos) de la cual se obtiene la matriz de correlación entre clientes.

Para finalizar, dada la matriz de correlación de dimensión $n \times n$ (clientes-clientes), se agrupan aquellos clientes que tengan un determinado *umbral de tolerancia* γ con respecto al coeficiente de correlación, formándose de esta manera los *meta-clientes* (grupos conformados por dos o más de clientes altamente correlacionados entre sí).

4.1.1 Reducción de Número de SKUs

Para la empresa, cada SKU representa un tipo de producto distinto con determinadas características o atributos. En este caso se manejan tres atributos por SKU: retornabilidad (retornable o no retornable), presentación (si está dado en mililitros, litros, tetrapack, etc.) y marca (marca para refresco de cola, de sabor, jugos, etc).

Para la reducción de SKUs, se utiliza el método de agrupación de tipo jerárquico conocido como *el vecino más próximo* mencionado en el Capítulo 2. La métrica

que se utiliza para medir que tan similares son dos productos está dada por la *Distancia de Pearson* (4.1) la cual hace uso del coeficiente de correlación de Pearson (índice estadístico que mide la relación lineal entre dos variables cuantitativas) para obtener la distancia o dismilitud entre dos productos dados [53, 87]. La selección de dicha métrica se debe a que esta fase está enfocada principalmente a reducir el número de SKUs por medio del uso del coeficiente de correlación de Pearson entre los atributos de éstos. La distancia de Pearson está representada de la siguiente manera:

$$DP = 1 - CC, \tag{4.1}$$

donde CC es el coeficiente de correlación de Pearson. Valores altos de correlación entre dos productos dan como resultado una distancia menor.

Al utilizar un método de agrupamiento jerárquico puede obtenerse un dendrograma, en el cual cada rama de éste representa un agrupamiento distinto. Es entonces cuando se debe determinar cuantos grupos se deben formar. Una manera de hacerlo es recorriendo el dendrograma de arriba hacia abajo e ir obteniendo para cada rama el menor coeficiente de correlación (sea negativo o positivo) entre los productos que conforman el grupo para dicho nivel. Mientras ese coeficiente sea menor a un τ (valor mínimo del coeficiente de correlación permitido para un grupo de productos) se siguen explorando las ramas subsecuentes hasta que el menor para cada nivel sea mayor o igual a dicho valor. Entonces dicho nivel es seleccionado obteniendo grupos de productos con al menos τ de correlación.

La Figura 4.2 muestra un ejemplo de dendrograma en el cual se desean encontrar grupos de elementos cuyo nivel de correlación mínimo permitido es de $\tau = 0.90$, para cada grupo de elementos (ramas del dendrograma) se calcula el nivel de correlación entre los elementos obteniendo el mínimo para cada grupo (correspondiente al número en cada rama del dendrograma). Entonces se va examinando cada rama del árbol jerárquico hasta encontrarse con al menos τ de correlación. En ese entonces el grupo representado por dicha rama se elige como uno de los que formarán el agrupamiento final. En el caso de la Figura 4.2 el número de grupos para un $\tau = 0.90$ es de cinco grupos de productos.

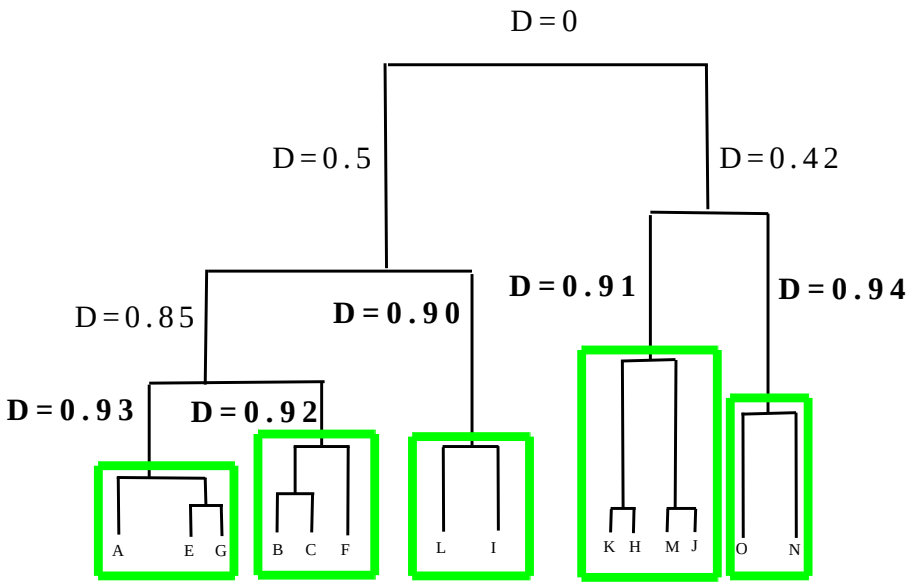


Figura 4.2: Ejemplo de selección de grupos de SKUs dado un dendrograma. Para un $\tau = 0.90$, se forman cinco grupos.

4.1.2 Correlación entre Clientes

Una vez reducido el número de SKUs, se procede a formar una matriz clientes-atributos (Figura 4.3) compuesta por los grupos formados después de la reducción de los SKUs, el valor númerico del tipo de contrato y de establecimiento, así como de las coordenadas geográficas. Ya que formado la matriz, el siguiente paso consiste en obtener submatrices de la misma para posteriormente proceder a obtener la matriz de correlación entre clientes de cada submatriz.

Como este problema se restringe a valores binarios para el tipo de contrato y de establecimiento se desarrollaron tres estrategias. La primera estrategia consiste en obtener submatrices con aquellos clientes cuyos tipos de contrato y establecimiento sean iguales. Las otras dos estrategias consisten en formar las submatrices con aquellos clientes cuyo tipo de contrato sea igual pero varíe el tipo de establecimiento y viceversa, esto con el objetivo de dar más flexibilidad a la empresa en caso de que su decisión sea que alguno de los atributos cuyos valores son binarios puedan no serlo.

Una vez obtenidas las submatrices, se obtiene la matriz de correlación para cada

una de ellas. La Figura 4.3 muestra la matriz clientes-atributos que se requiere para aplicar lo antes mencionado. Si se decide aplicar la primera estrategia, entonces una

$$\begin{array}{c} \\ C_1 \\ C_2 \\ C_3 \\ C_4 \\ C_5 \end{array} \begin{array}{c} \begin{array}{cccccc} S_1 & S_2 & TC & TE & x & y \end{array} \\ \left(\begin{array}{cccccc} 0.3 & 0.7 & 1 & 1 & 0.3 & 0.6 \\ 0.2 & 0.8 & 2 & 2 & 0.2 & 0.4 \\ 1 & 0 & 2 & 1 & 0.1 & 0.6 \\ 0 & 1 & 1 & 1 & 0 & 1 \\ 0.5 & 0.5 & 3 & 2 & 1 & 0 \end{array} \right) \end{array}$$

Figura 4.3: Matriz clientes-atributos.

submatriz estaría formada por los clientes (C_1, C_4), las otras por (C_2), (C_3) y (C_5), respectivamente. Si se aplicara la segunda estrategia sería por (C_1, C_4), (C_2, C_3) y (C_5). Por último si se aplicara la tercera se formarían tres submatrices conformadas por los clientes (C_1, C_3), (C_2, C_5) y (C_4).

4.1.3 Formación de Metaclientes

Un metacliente podemos definirlo como un conjunto de clientes con características muy similares que son agrupados para conformar uno solo. Para la creación de metaclientes se utilizaron las matrices de correlación resultantes de la etapa anterior. Dada una matriz de correlación, se crea una lista de todos aquellos clientes que tienen un coeficiente de correlación por encima de un umbral de tolerancia γ dado. Es decir, supongamos que tenemos cinco clientes C_1, C_2, C_3, C_4 y C_5 cuyos coeficientes de correlación se muestran en la Figura 4.4. Supongamos también, que el coeficiente de correlación mínimo que deben tener los clientes para poder formar un metacliente es de $\gamma = 0.90$, o bien, 90 % si queremos verlo como porcentaje.

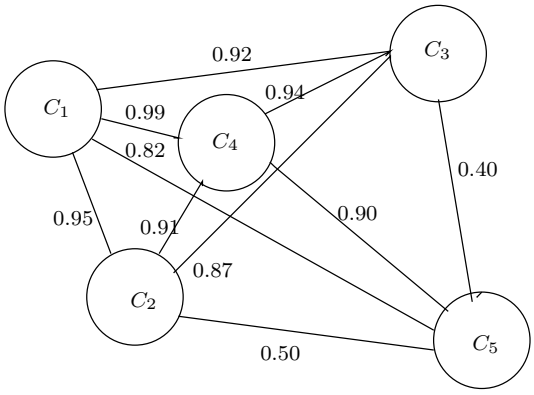
La Figura 4.5 muestra la respresentación de dicha matriz mediante un grafo en

	C_1	C_2	C_3	C_4	C_5
C_1	1	0.95	0.92	0.99	0.82
C_2	0.95	1	0.87	0.91	0.50
C_3	0.92	0.87	1	0.94	0.40
C_4	0.99	0.91	0.94	1	0.90
C_5	0.82	0.50	0.40	0.90	1

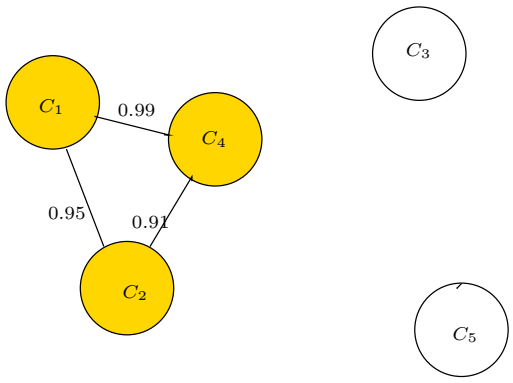
Figura 4.4: Matriz de correlación de clientes.

donde los nodos corresponden a los clientes y las aristas corresponden al coeficiente de correlación entre ellos.

Como podemos observar, los clientes C_1 y C_2 pueden unirse para formar un metacliente (ya que el coeficiente de correlación entre ambos es mayor a 0.90 y además, no han sido seleccionados para formar otro metacliente). El cliente C_3 no puede seleccionarse para formar parte del metacliente formado por C_1 y C_2 debido a que su coeficiente de correlación con C_2 esta por debajo de 0.90, aún y cuando el coeficiente de correlación con C_1 está por encima del mínimo permitido.



(a) Grafo que representa el nivel de correlación entre los clientes.



(b) Ejemplo de la selección final de los clientes que conforman el nuevo metacliente.

Figura 4.5: Ejemplo de creación de metaclientes.

El cliente C_4 se selecciona para unirse a C_1 y C_2 debido a que sus coeficientes de correlación con C_1 y con C_2 son mayores a 0.90. Por otro lado, C_5 no puede seleccionarse por la misma razón que C_3. Ahora, se puede decir que ya se ha formado un metacliente con la unión de C_1, C_2 y C_4; sin embargo, no se ha terminado de verificar si aún se puede crear otro metacliente. Para ello, comenzamos con el primer cliente de los que aún no forman parte de un metacliente, en este caso C_3, y verificamos su nivel de correlación con los demás clientes. Los clientes C_1, C_2 y C_4 ya no pueden seleccionarse, por lo tanto, solo se verifica el coeficiente de correlación con C_5, pero como dicho coeficiente se encuentra por debajo de 0.90, no se puede formar un metacliente con ambos y como ya no hay más elementos que verificar, se finaliza la creación de metaclientes. Como resultado para este ejemplo se ha reducido la cantidad de clientes de cinco a solamente tres, donde el cliente MC_1 (metacliente 1) está formado por C_1, C_2 y C_4, mientras que C_3 y C_5 se quedan igual.

El Pseudocódigo 3 describe la manera de crear los metaclientes donde $MC(i, M_x)$ representa el coeficiente de correlación mínimo entre el cliente i y los clientes que conforman el metacliente M_x.

La ubicación geográfica de cada uno de los metaclientes resultantes será el punto medio entre todos los que conformar el metacliente, el volumen de compra será la suma de todos sus volúmenes por SKU y el tipo de contrato y establecimiento serán aquéllos que fueron más frecuentes.

La creación de metaclientes pretende ser una forma de reducir el conjunto de datos aprovechando algunas de sus propiedades ó características. De esta manera se tratará de reducir el tiempo de cómputo en instancias de gran tamaño usualmente encontradas en problemas reales.

Pseudocódigo 3 *metaclientes*(V', MC, γ)

Entrada:

V' : Conjunto de elementos, $V' \in V$;

MC : Matriz de correlación correspondiente a los elementos de V';

γ : Nivel mínimo de correlación permitido para formar metaclientes;

Salida: $M = (M_1, \ldots, M_x)$: Metaclientes;

1: $\bar{V} \leftarrow V'$;
2: $x \leftarrow 0$;
3: **Mientras** $(\bar{V} \neq \emptyset)$ **hacer**
4: $x \leftarrow x + 1$; $M_x \leftarrow \emptyset$;
5: Tomar un elemento $i \in \bar{V}$;
6: $M_x \leftarrow M_x \cup \{i\}$;
7: $\bar{V} \leftarrow \bar{V} \setminus \{i\}$;
8: **Para** $j \in \bar{V}$ **hacer**
9: **Si** $(MC(j, M_x) >= \gamma)$ **entonces**
10: $M_x \leftarrow M_x \cup \{j\}$;
11: $\bar{V} \leftarrow \bar{V} \setminus \{j\}$;
12: **Fin Si**
13: **Fin Para**
14: **Fin Mientras**
15: **Regresar** $M = (M_1, \ldots, M_x)$.

4.2 Construcción de Particiones

En esta parte de la metodología se pretende crear un partición inicial del conjunto de clientes. Para ello, se utiliza el algoritmo de partición p-medias (también llamado K-medias). Se mencionan también las ventajas y desventajas del algoritmo. Para que los resultados de dicho algoritmo tuvieran significancia al problema tratado en esta tesis, se hicieron algunas modificaciones de las cuales se hace mención durante este capítulo.

4.2.1 Algoritmo p-medias

El algoritmo p-medias [36] es un algoritmo de partición que comienza con la selección de p centroides iniciales los cuales representarán a un segmento distinto. Posteriormente, asigna los elementos restantes a su centroide más cercano según una métrica dada y se evalúa la partición. Cuando todos los elementos han sido asignados se procede a la siguiente iteración que consiste en recalcular los centroides y reasignar el resto de los elementos a estos nuevos centroides. El algoritmo procede iterativamente y termina cuando un determinado criterio de parada se ha alcanzado.

Ventajas y Desventajas

Como ya se mencionó en el Capítulo 2, el algoritmo p-medias muestra algunas ventajas y desventajas. Para ajustar este algoritmo al problema que se trata en esta tesis, se realizaron varias modificaciones que no solo ayudan a resolver el problema con las características requeridas, sino que también ayudan a obtener mejores soluciones al problema tratado. Estas modicaciones se enumeran a continuación:

1. Para medir la similitud entre una pareja cualesquiera de clientes i y j de un segmento X_k se usará la función objetivo ponderada (3.1) que se ha formulado para este problema.

2. El centro de un segmento está dado por el cliente cuya distancia hacia todos los demás clientes de su mismo segmento, es la menor (no será la media aritmética como usualmente se hace con el p-medias).

3. El criterio de parada a utilizar para el algoritmo es una combinación de dos criterios: cuando la solución ya no mejore y cuando no exista movimiento en cuanto a los centroides.

4.2.2 GRASP para la Obtención de Centros Iniciales

Una de las desventajas que se conoce del algoritmo p-medias es que la mejor solución encontrada por el algoritmo depende ampliamente de la selección inicial de centroides. Por esta razón, se desarrolló un GRASP (Pseudocódigo 5) cuyas fases de contrucción y post-procesamiento están basadas en una heurística de construcción voraz propuesta por Erkut, Ürküsal y Yenyceriogˇlu [21], utilizada para problemas de p-dispersión [20] (pseudocódigo 4) y en el algoritmo p-medias, respectivamente.

Pseudocódigo 4 p-dispersión(V, p)

Entrada:

V : Conjunto de nodos;

p : Número de nodos dispersos deseados;

Salida: P : Conjunto de nodos dispersos;

1: Encontrar v_1 y v_2 en V tal que $d(v_1, v_2) = \text{máx}\{d_{ij} : 1 \leq i < j \leq n\}$;
2: $P \leftarrow \{v_1, v_2\}$;
3: **Mientras** $(|P| < p)$ **hacer**
4: Encontrar $j \in V \setminus P$ tal que $d(j, P) = \text{máx}\{d(i, P) : i \in V \setminus P\}$;
5: $P \leftarrow P \cup \{j\}$;
6: **Fin Mientras**
7: **Regresar** P.

En el Pseudocódigo 4 se ilustra una de las heurísticas de Erkut, Ürküsal y Yenyceriogˇlu [21] para el problema de p-dispersión. Éste toma como entrada un conjunto V de nodos y regresa un conjunto P de nodos dispersos. En el Paso 4, la distancia $d(i, P)$ está dada por $d(i, P) = \text{mín}\{d_{ij} : j \in P\}$. Es decir que $d(i, P)$ es la mínima distancia que existe entre el nodo i y el conjunto P (conjunto de centros seleccionados hasta el momento).

Ahora bien, regresando al problema de interés, la motivación de desarrollar un procedimiento de búsqueda aleatorizado, adaptativo y voraz radica en la necesidad de obtener soluciones de mejor calidad, las cuales se pretende se encuentren en

menor tiempo al seleccionar los centros iniciales de una manera más sistemática y no puramente aleatoria. Seleccionar los centros de una manera aleatoria puede tambien ocasionar como resultado la obtención de una mala solución dado que dicha selección queda atrapada en un óptimo local en el cual la calidad de la solución no es buena. Además, la convergencia a dicho óptimo local puede ser rápida si la ubicación de los centros iniciales es muy cercana a la de los centros finales, o bien, puede requerir más tiempo si no lo es.

Con un GRASP es más probable que se converja más rápido a una solución (para un β adecuado) que el procedimiento completamente aleatorio, sin dejar de mencionar que basta con ajustar el parámetro de calidad β para obtener soluciones de buena calidad para diferentes conjuntos de datos a manejar (si el conjunto es muy compacto, disperso, etc.), incluso usar la selección completamente aleatoria ó completamente voraz si es necesario.

El Pseudocódigo 5 muestra la implementación del GRASP para el problema de segmentación de clientes que se aborda en esta tesis. La fase constructiva del Paso 3 consiste en seleccionar p clientes dispersos de un total de n, los cuales representan a los centros iniciales requeridos por el p-medias. El procedimiento de obtención de centros (mostrado en el Pseudocódigo 6) es una extensión de la heurística propuesta por Erkut, Ürküsal y Yenyerioğlu [21] para el problema de p-dispersión que consiste en permitir una elección aleatoria voraz siguiendo la filosofía de GRASP. La diferencia es que ahora, en lugar de tomar una selección totalmente voraz, se selecciona aleatoriamente un elemento de una lista restringida de candidatos (LRC) la cual contiene aquellos candidatos cuya evaluación de su función voraz está dentro de un $\beta\,\%$ de la mejor evaluación ($\beta \in [0, 1]$ es el parámetro de umbral de calidad de GRASP explicado en el Capítulo 2). Inicialmente la LRC se construye con todas aquellas parejas de clientes (i, j) cuya dispersión se encuentre a un $\beta\,\%$ de $d^{\text{máx}}$ ($d^{\text{máx}}$ es la máxima distancia entre una pareja de clientes i, j del conjunto V). Una vez construída la LRC, se selecciona una pareja de clientes (i, j) de esa lista de forma aleatoria la cual es agregada al conjunto P (conjunto de centros seleccionados). En

Pseudocódigo 5 GRASP(V, p, β, Max_iter)

Entrada:

V : Conjunto de clientes;

p : Número de centros;

β : Parámetro de umbral de calidad;

Max_iter: Número máximo de iteraciones;

Salida: X^{best} : Mejor partición encontrada;

1: $X^{best} \leftarrow \emptyset$;
2: **Mientras** (Max_iter > 0) **hacer**
3: $P \leftarrow$ kdispersión(V, p, β); {Obtiene p centros dispersos}
4: $X \leftarrow$ kmedias(V, P); {Refina centros y obtiene su partición}
5: **Si** $(f(X) < f(X^{best}))$ **entonces**
6: $X^{best} \leftarrow X$;
7: **Fin Si**
8: Max_iter $\leftarrow$ Max_iter - 1;
9: **Fin Mientras**
10: **Regresar** X^{best}.

cada una de las siguientes $p - 2$ iteraciones (p es el número de clientes dispersos que se requieren encontrar), se agrega un nuevo cliente a la solución. Para ello se calcula, para todo cliente $i \in \mathrm{V}\backslash P$, la mínima distancia entre el cliente i y los clientes que forman parte de P. Se obtiene la máxima ($d^{\text{máx}}$) y la mínima ($d^{\text{mín}}$) de dichas distancias. Se construye nuevamente la LRC pero ahora con aquellos clientes cuya distancia mínima al conjunto P se encuentre a un $\beta\,\%$ del nuevo valor de $d^{\text{máx}}$. Una vez construída dicha lista, se escoge un elemento de ella al azar y se agrega al conjunto P. La fase de contrucción del GRASP finaliza cuando $|P| = p$.

El Paso 4 del Pseudocódigo 5 corresponde a la asignación de clientes a los clientes centrales encontrados en el Paso 3 por medio del algoritmo p-medias el cual fué adaptado al problema tratado en esta tesis. Esta fase, la cual se muestra de manera detallada en el Pseudocódigo 7, comienza asignando todo cliente $i \in V \setminus P$

Pseudocódigo 6 kdispersión(V, p, β)

Entrada:

V: Conjunto de clientes;

p: Número de centros;

β: Parámetro de calidad;

Salida: P: Conjunto de centros iniciales;

1: $P \leftarrow \emptyset$; $\bar{V} \leftarrow V$;
2: $d^{\text{máx}} = \underset{i,j \in \bar{V}}{\text{máx}}\{d_{ij}\}$; $d^{\text{mín}} = \underset{i,j \in \bar{V}}{\text{mín}}\{d_{ij}\}$;
3: LRC $= \{(i,j) \in \bar{V}\colon d_{ij} \geq d^{\text{máx}} - \beta(d^{\text{máx}} - d^{\text{mín}})\}$;
4: $(i,j) \leftarrow$ aleatorio(LRC);
5: $P \leftarrow P \cup \{i,j\}$;
6: $\bar{V} \leftarrow \bar{V} \setminus \{i,j\}$;
7: **Mientras** $(|P| < p)$ **hacer**
8: Calcular $d^{\text{máx}} = \underset{i \in \bar{V}}{\text{máx}}\{d(i,P)\}$; $d^{\text{mín}} = \underset{i \in \bar{V}}{\text{mín}}\{d(i,P)\}$;
9: LRC $= \{i \in \bar{V}\colon d(i,P) \geq d^{\text{máx}} - \beta(d^{\text{máx}} - d^{\text{mín}})\}$;
10: $i \leftarrow$ aleatorio(LRC);
11: $P \leftarrow P \cup \{i\}$;
12: $\bar{V} \leftarrow \bar{V} \setminus \{i\}$;
13: **Fin Mientras**
14: **Regresar** P.

a su cliente central $l \in P$ más cercano. Una vez asignados todos los clientes a su segmento X_k, los centros se reasignan calculando para cada cliente $i \in X_k$ la suma de las disimilitudes entre dicho cliente y los demás del mismo segmento utilizando la ecuación (3.1).

El nuevo centro de cada segmento será aquel cliente cuya suma de disimilitudes sea la menor. Se asignan los clientes a dichos centros los cuales, posteriormente, se recalculan nuevamente. El algoritmo termina cuando la calidad de la solución empeora ó cuando ya no hay cambio en los centros.

Pseudocódigo 7 kmedias(V, P)

Entrada:

V: Conjunto de clientes;

$P = \{i_1, \ldots, i_p\}$: Conjunto de centros iniciales;

Salida:

X: Partición final;

1: $X_k \leftarrow \{i_k\}$, $k \in P$;
2: **Mientras** (solución no empeore y los centros cambien) **hacer**
3: **Para** $i \in V \setminus P$ **hacer**
4: $\hat{k} = \arg\min_{k \in P}\{f(i, i_k)\}$; {Asigna cliente i a su centro más cercano (similar)}
5: $X_{\hat{k}} \leftarrow X_{\hat{k}} \cup \{i\}$;
6: **Fin Para**
7: **Para todo** $k \in P$ **hacer**
8: $i^* = \arg\min_{i \in X_k} \left\{ \sum_{j \in X_k} f(i, j) \right\}$ {Recalcula centros}
9: $P \leftarrow \{i^*\}$;
10: **Fin Para**
11: **Fin Mientras**
12: **Regresar** $X = \{X_1, \ldots, X_p\}$.

4.2.3 Análisis del Número Ideal de Segmentos

Para el uso del algoritmo p-medias se necesita conocer a priori el número de segmentos a formar. Es entonces cuando surge la necesidad de saber cual número de segmentos será el indicado para obtener la mejor partición posible de un determinado conjunto clientes. Para ello ya han sido desarrollado índices de validación de particiones que tienen como principal objetivo evaluar la calidad de una partición dada según ciertos criterios (véase Sección 2.1.7).

En esta tesis emplearemos el índice de Davies-Bouldin (2.4) para obtener el número de segmentos para el cual la calidad de la partición obtenida por el p-medias es mejor [48, 61].

4.3 Mejora de la Solución

Una desventaja del algoritmo p-medias y de cualquier método heurístico es que no existe garantía de que la solución encontrada sea óptima globalmente (la mejor de todas las posibles soluciones) sino más bien es óptima de manera local (la mejor de un cierto conjunto de posibles soluciones). La mejor partición encontrada por el GRASP propuesto puede ser mejorada aún más aplicando a dicha partición un procedimiento basado en una búsqueda de entornos variables o VNS (véase Sección 2.1.6) en el cual se aplican movimientos de inserción simple e intercambios. El entorno de inserción simple consiste en reasignar un cliente a un segmento diferente. El entorno de intercambio consiste en intercambiar dos clientes de segmentos distintos. El Pseudocódigo 8 muestra el esquema general del procedimiento de mejora basado en una VNS.

Pseudocódigo 8 VNS(X)

Entrada: X : Mejor partición encontrada por GRASP;

Salida: X: Partición mejorada;

1: ahorro_total $\leftarrow 1$;
2: $\epsilon \leftarrow 1.0e$-04; iter $\leftarrow 0$;
3: **Mientras** ((ahorro_total > 0) y (iter < 3)) **hacer**
4: $(X) \leftarrow$ inserción(X); ahorro_ins $\leftarrow f(X)$;{Primer entorno}
5: $(X) \leftarrow$intercambio(X); ahorro_inter $\leftarrow f(X)$;{Segundo entorno}
6: ahorro_total $\leftarrow$ ahorro_ins + ahorro_inter;
7: **Si** (ahorro_total $< \epsilon$) **entonces**
8: iter $\leftarrow$ iter + 1;
9: **Fin Si**
10: **Fin Mientras**
11: **Regresar** $X = (X_1, \ldots, X_p)$.

La búsqueda local de inserción simple (mostrada en el Pseudocódigo 9), se efectúa seleccionando un segmento al azar y aleatoriamente tomar un cliente de dicho segmento e insertarlo en uno de los segmentos restantes (aquél en el que se haya encontrado la primera mejora). Si al haber tratado de insertarlo en todos los segmentos restantes no se encontró mejora alguna, la inserción no se lleva a cabo y se trata con otro cliente del mismo segmento. Esto se repite hasta que se hayan terminado de explorar todos los posibles movimientos de inserción para todos los segmentos a evaluar.

De la misma manera, la búsqueda de intercambio (Pseudocódigo 10), se lleva a cabo seleccionando un segmento k al azar, $k \in K$, y aleatoriamente tomar un cliente $i \in X_k$ de dicho segmento e intercambiarlo por un cliente $j \in X \setminus X_k$, de otro de los segmentos restantes (aquél con el que se encuentre la primera mejora). Si después de haber evaluado los posibles movimientos de intercambio no se encuentra mejora, el intercambio no se efectúa. Esto se repite hasta que se hayan evaluado todos los movimientos posibles.

Cabe recalcar que no todos los clientes, ni todos los segmentos pueden llegar a ser evaluados. Es decir que solo se tomarán en consideración determinados segmentos y determinada cantidad de clientes por cada uno de ellos. Esto con el fin de disminuir el tiempo de cómputo para resolver instancias de gran tamaño.

Pseudocódigo 9 inserción(X)

Entrada: $X = \{X_1, \ldots, X_p\}$;

Salida: $X = \{X_1, \ldots, X_p\}$;

$m \leftarrow 0$; mejora $\leftarrow$ Sí; {Contador de segmentos e indicador de mejora}
Mientras ($m \leq \frac{p}{2}$) **hacer**
  Si (mejora = Sí) **entonces**
    $\bar{P} \leftarrow P$;
  Fin Si
  mejora $\leftarrow$ No; $m \leftarrow m + 1$;
  Seleccionar un segmento $k_1 \in \bar{P}$ al azar;
  $\bar{P} \leftarrow \bar{P} \setminus \{k_1\}$; $\bar{Y} \leftarrow X_{k_1}$;
  $z \leftarrow \text{mín}\{|\bar{X}_{k_1}|, |\frac{n}{2p} - \bar{X}_{k_1}|\}$;
  Mientras ($z > 0$) **hacer**
    Seleccionar un cliente $i \in \bar{Y}$ al azar; $\bar{Y} \leftarrow \bar{Y} \setminus \{i\}$;
    $z \leftarrow z - 1$; $r \leftarrow 1$;
    Mientras (($r \leq k_1$) y (mejora = No)) **hacer**
      Si ($r \neq k_1$) **entonces**
        $\phi(i, r) = \sum_{j \in X_{k_1}} f_{ij} - \sum_{j \in X_r} f_{ij}$;
        Si ($\phi(i, r) > 0$) **entonces**
          $X_{k_1} \leftarrow X_{k_1} \setminus \{i\}$; $X_r \leftarrow X_r \cup \{i\}$;
          mejora $\leftarrow$ Sí;
        Fin Si
      Fin Si
      $r \leftarrow r + 1$;
    Fin Mientras
  Fin Mientras
Fin Mientras
Regresar $X = \{X_1, \ldots, X_p\}$;

Pseudocódigo 10 intercambio(X)

Entrada: $X = \{X_1, \ldots, X_p\}$;

Salida: $X = \{X_1, \ldots, X_p\}$;

$m \leftarrow 0$; mejora $\leftarrow$ Sí; {Contador de segmentos e indicador de mejora}
Mientras $(m \leq \frac{p}{3})$ **hacer**
  Si (mejora = Sí) **entonces**
    $\bar{P} \leftarrow P$;
  Fin Si
  mejora $\leftarrow$ No; $t \leftarrow 0$; $m \leftarrow m + 1$;
  Seleccionar un segmento $k_1 \in \bar{P}$ al azar;
  $\bar{P} \leftarrow \bar{P} \setminus \{k_1\}$; $\bar{Y} \leftarrow X_{k_1}$;
  $z \leftarrow$ mín$\{|\bar{X}_{k_1}|, |\frac{n}{4p} - \bar{X}_{k_1}|\}$;
  Mientras $(z > 0)$ **hacer**
    Seleccionar un cliente $i \in \bar{Y}$ al azar; $\bar{Y} \leftarrow \bar{Y} \setminus \{i\}$;
    $z \leftarrow z - 1$; $r \leftarrow 1$;
    Mientras $((r \leq k_1)$ y (mejora = No)) **hacer**
      Si $(r \neq k_1)$ **entonces**
        Para cada cliente $j \in X_r$ **hacer**
          $$\phi(i,j) = \left(\sum_{q \in X_{k_1}} f_{iq} - \sum_{q \in X_r} f_{iq}\right) + \left(\sum_{q \in X_r} f_{jq} - \sum_{q \in X_{k_1}} f_{jq}\right);$$
          Si $(\phi(i,j) > 0)$ **entonces**
            $X_{k_1} \leftarrow X_{k_1} \setminus \{i\}$; $X_r \leftarrow X_r \cup \{i\}$;
            $X_r \leftarrow X_r \setminus \{j\}$; $X_{k_1} \leftarrow X_{k_1} \cup \{j\}$;
            mejora $\leftarrow$ Sí;
          Fin Si
        Fin Para
      Fin Si
      $r \leftarrow r + 1$;
    Fin Mientras
  Fin Mientras
Fin Mientras
Regresar $X = \{X_1, \ldots, X_p\}$;

Capítulo 5

Resultados Computacionales

En este capítulo se pretende analizar la metodología, mediante determinados experimentos computacionales, para mostrar el funcionamiento de la misma en la creación de particiones. Primeramente se pretenden estudiar los parámetros para los cuales el algoritmo obtiene mejores resultados sobre la muestra real. Posteriormente, se usan dichos parámetros de manera fija para mostrar las particiones encontradas en un ambiente gráfico y analizar las mismas para obtener conclusiones.

Un tercer experimento es realizado para mostrar las mejoras proporcionadas por un procedimiento de búsqueda local sobre la partición resultante y discutir las ventajas y desventajas de utilizar dicho procedimiento para mejorar la calidad de la solución. Por último, se aplica la metodología a la muestra preprocesada (pre-agrupando mediante metaclientes) mostrando las ventajas y desventajas del uso del preproceso.

Algunos aspectos a considerar con respecto a la aplicación de la metodología son los siguientes:

- **Exportación de la base de datos a un formato estandar:** para poder transformar la base de datos de la muestra brindada por la empresa se utilizó la aplicación de Excel 2007 ya que cuenta con la ventaja de no tener tan limitado el rango de filas y columnas permitidas como las versiones anteriores. Esto resulta de gran ayuda cuando la base de datos a exportar es de gran tamaño. A partir de esta aplicación se pudo transformar la base de datos (con extensión

.dbf) a un formato estándar para poder obtener la información útil para la aplicación de la metodología.

- **Preproceso:** para esta fase se utilizó software estadístico como MINITAB para reducir el número de SKUs mediante el uso de métodos de agrupamiento incluídos en el software como lo son los métodos de agrupamiento jerárquico. Una vez que se conoce que SKUs pertenecerán a un determinado grupo se hace uso de un método desarrollado para crear la instancia en el formato que se necesitará para aplicarla a las siguientes fases. Para obtener las correlaciones entre los clientes que forman la instancia (una vez reducido el número de SKUs) se desarrolló un programa en el lenguaje C++ que calcula dicho coeficiente reportando la matriz de correlaciones en un archivo .xls el cual es utilizado en la siguiente fase del preproceso. Para esta fase siguiente (creación de metaclientes) también fue desarrollado un método en C++ que hace uso de la matriz de correlaciones de la fase anterior para agrupar a aquellos clientes que tengan un $\gamma\,\%$ de correlación entre ellos, como resultado se obtiene el nuevo conjunto de clientes (metaclientes).

- **Contrucción y Mejora de Particiones:** una vez obtenida la instancia a tratar (ya sea la real o preprocesada) se aplica el método de solución propuesto desarrollado en el lenguaje C++ el cual hace uso de la instancia y algunos datos relacionados con ella. Como salida se obtienen archivos con información detellada sobre los resultados finales.

- **Visualización de Particiones:** Para la visualización de las soluciones obtenidas se utilizó Gnuplot, una herramienta gratuita para la creación de gráficos, el cual está disponible para varios sistemas operativos populares como UNIX, Linux y Windows.

5.1 Descripción de la Instancia Real

Dado a que se trabaja un caso de estudio se obtuvo una muestra de datos real para probar la metodología. Dicha muestra fue proporcionada por la empresa y llevada a un formato estándar. Esta muestra contiene la información de 17332 clientes en cuanto al tipo de contrato, establecimiento, volumen de compra por cada SKU así como sus respectivas coordenadas geográficas. El número de SKUs identificado fue de 201 tipos diferentes, 32 tipos de establecimiento, 12 tipos de contrato distintos y la respectiva ubicación geográfica de cada uno de los clientes. Los experimentos Todos los experimentos (a excepción del preprocesamiento de los datos) se realizaron en un servidor Dell con procesador Intel Core(TM) 2 Quad CPU a 2.4 GHz. con 3.2 Gb. de memoria RAM bajo la distribución de Ubuntu versión 9.04 de Linux. La Figura 5.1 muestra como se encuentran distribuidos los clientes de la muestra real. De dicha muestra se extrajeron cuatro submuestras cuyos tamaños corresponden a 1000, 3000, 5000 y 7000 respectivamente, para la realización de los experimentos.

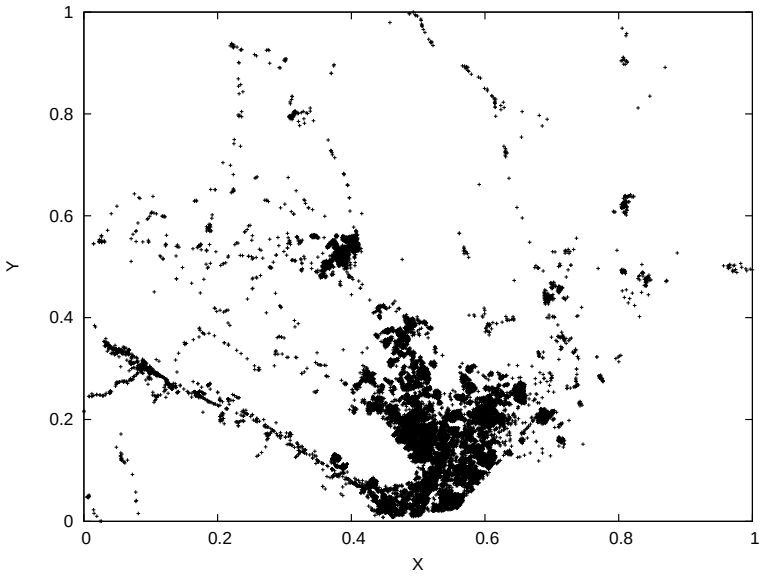


Figura 5.1: Representación gráfica de la ubicación geográfica de 17332 clientes.

5.2 Experimento A: Selección de Parámetros

5.2.1 Número de Ejecuciones del algoritmo p-medias

Objetivo: Mientras más repeticiones del algoritmo sean permitidas se incrementa la probabilidad de encontrar una mejor solución. Sin embargo, realizar esto requiere de mayor tiempo de cómputo. Es por ello que el objetivo de este experimento es determinar el número de ejecuciones para el algoritmo p-medias mediante la visualización de las mejoras obtenidas durante 100 repeticiones del algoritmo y de esta manera poder controlar el tiempo de cómputo requerido y al mismo tiempo obtener una partición de mejor calidad.

Características: Para este experimento se ejecutó 100 veces el algoritmo p-medias (usando selección de centros totalmente aleatoria, es decir GRASP con $\beta = 1$), se fijó el número de segmentos de $p \in \{10, 20, 30, 40, 50\}$ para cuatro instancias de tamaño $n \in \{1000, 3000, 5000, 7000\}$. La ponderación para cada uno de los atributos se fijó en $\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = 0.25$. Se evalúan los cuatro tipos de dispersión.

Discusión: En las Figuras 5.9 a la 5.5 se muestran las mejoras en la función objetivo encontradas durante las 100 ejecuciones del algoritimo. Cada punto representa la mejor solución obtenida en cada ejecución (solo se muestran las que fueron mejor que la reportada anteriormente). Como puede observarse, la mayor cantidad de mejoras obtenidas por el algoritmo (que ocurre con más frecuencia) se encuentra dentro de las 50 primeras repeticiones, solo en algunos casos y con menor frecuencia superan este número. Estas mismas observaciones se encontraron en los experimentos realizados para los valores extremos de los parámetros de ponderación de la función objetivo $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)$ mostrados en las Figuras A.1 a la A.4 del Apéndice A.

Conclusiones: Como conclusión se decide repetir el algoritmo p-medias a lo sumo 50 veces dado que en esta fase de construcción de particiones lo único que se

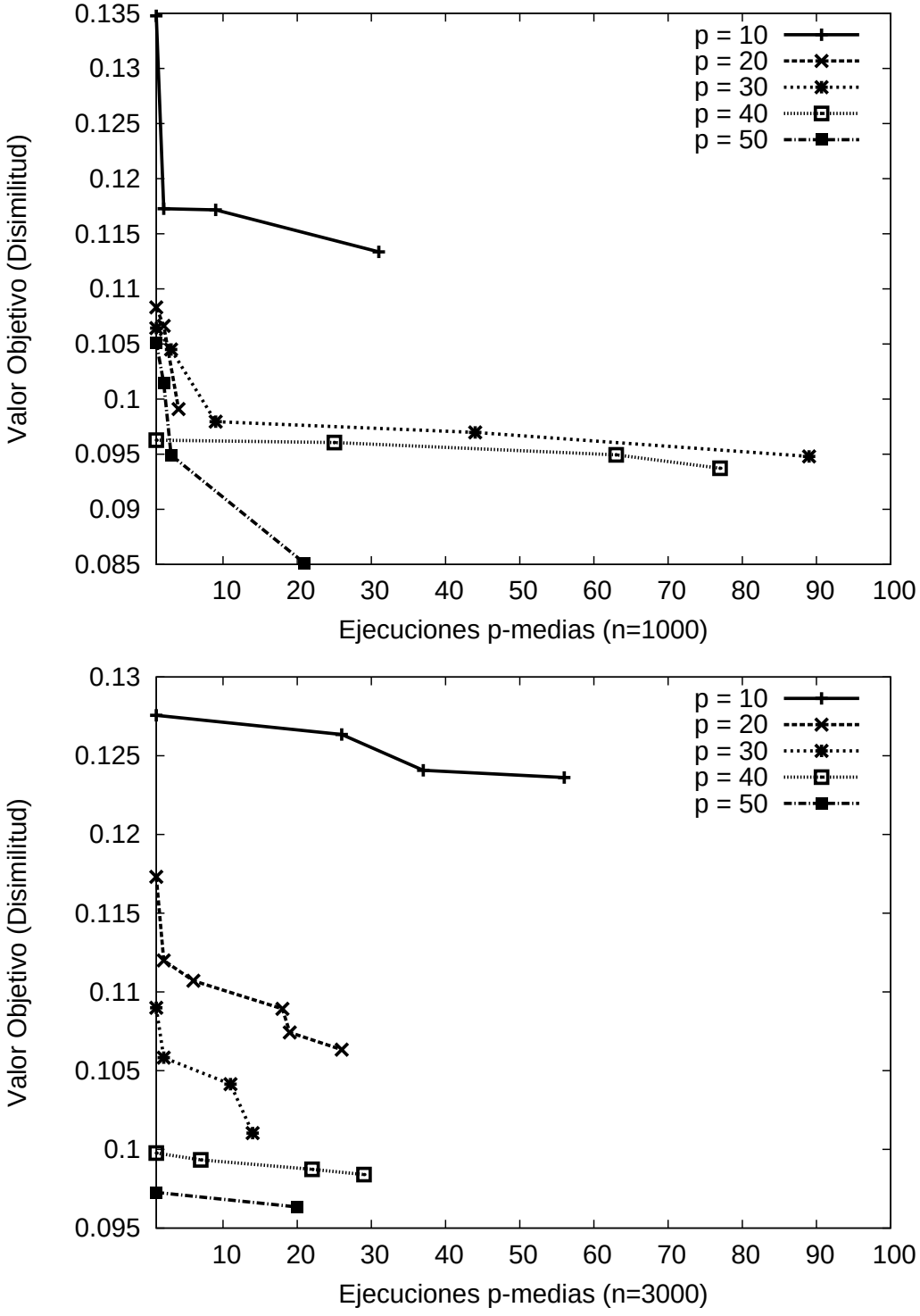


Figura 5.2: Evaluación de convergencia del algoritmo p-medias usando $f_{disp1}(X)$ (p-centro) como medida de dispersión.

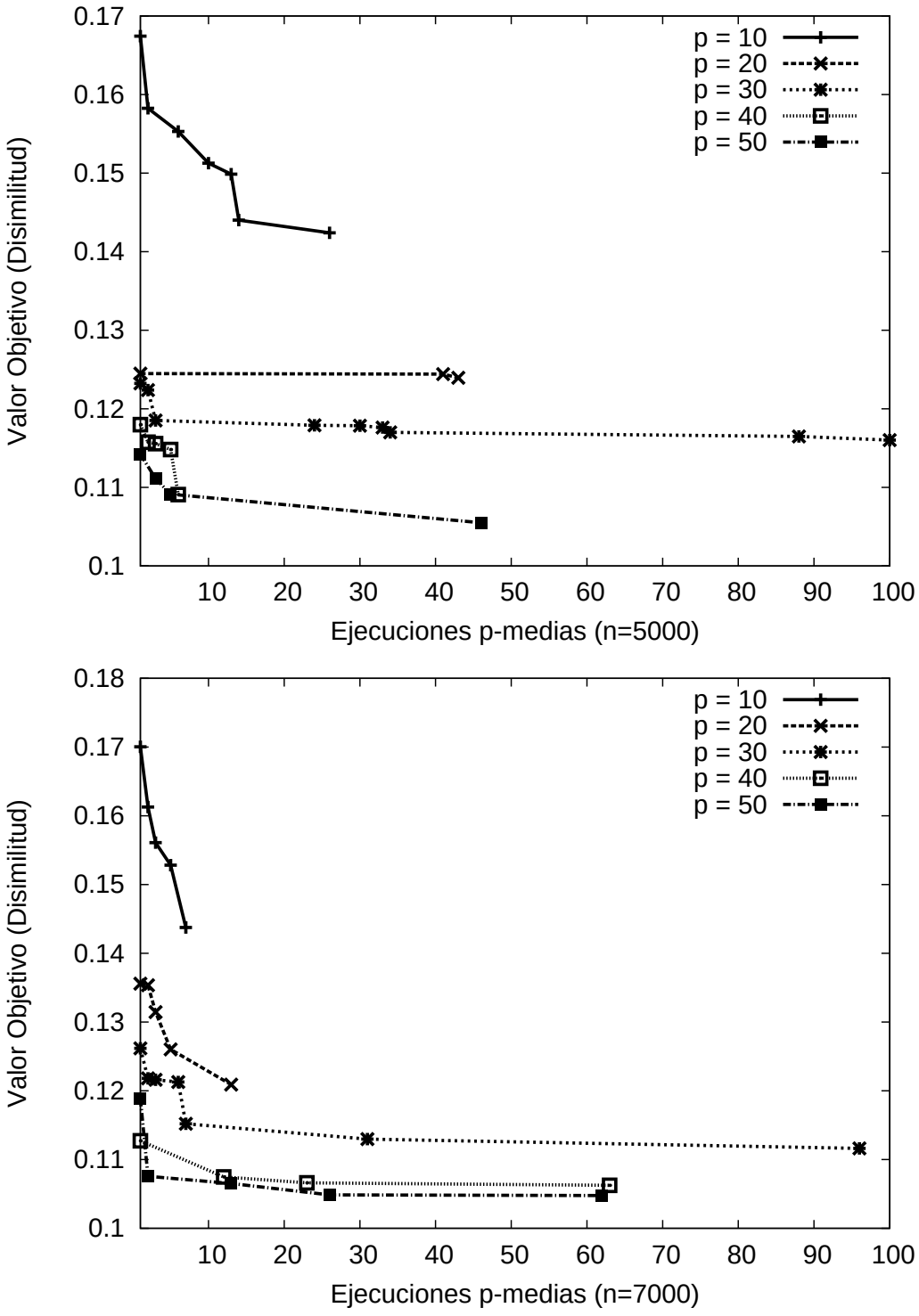


Figura 5.3: Evaluación de convergencia del algoritmo p-medias usando $f_{disp1}(X)$ (p-centro) como medida de dispersión.

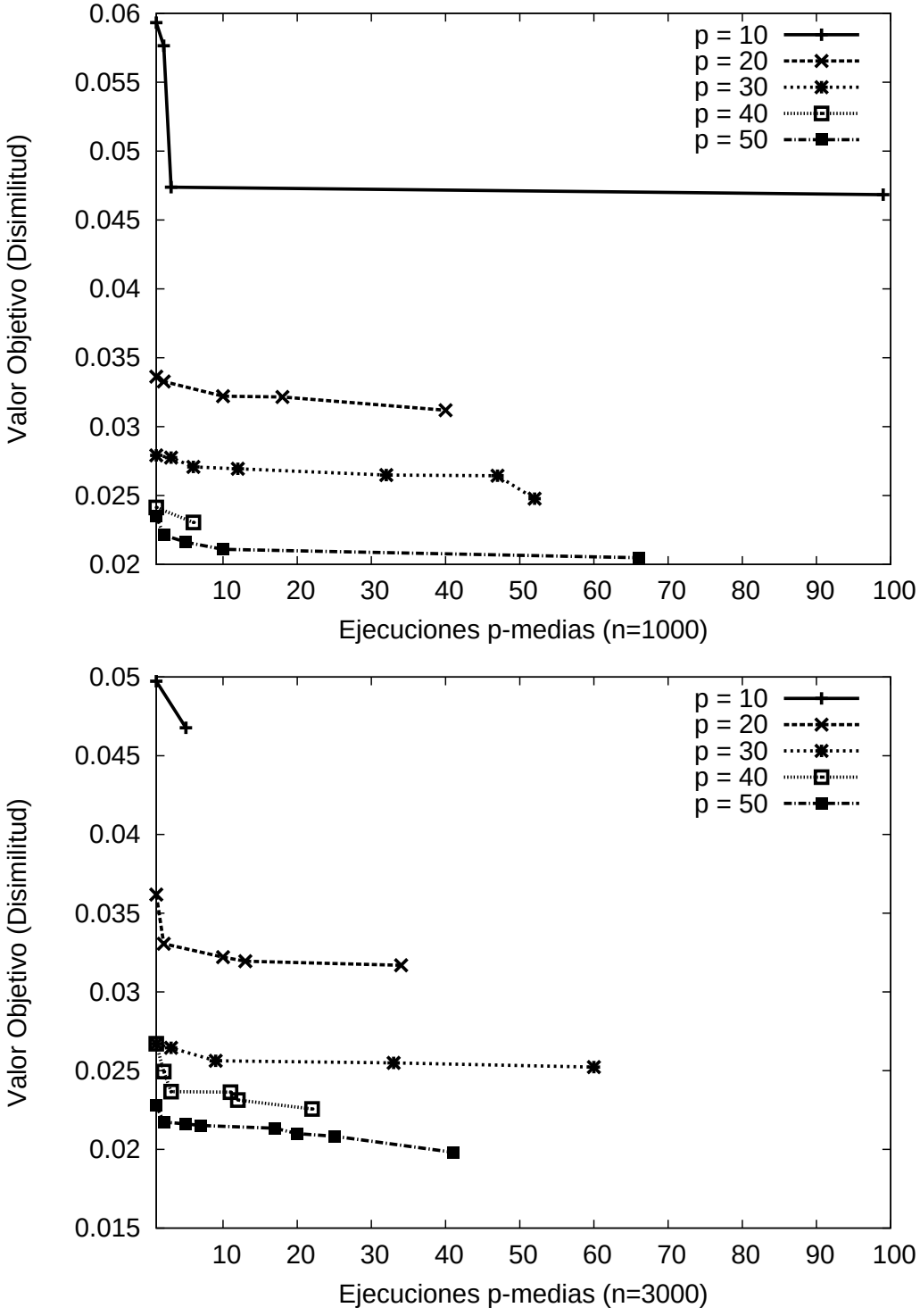


Figura 5.4: Evaluación de convergencia del algoritmo p-medias usando $f_{disp2}(X)$ (p-mediana) como medida de dispersión.

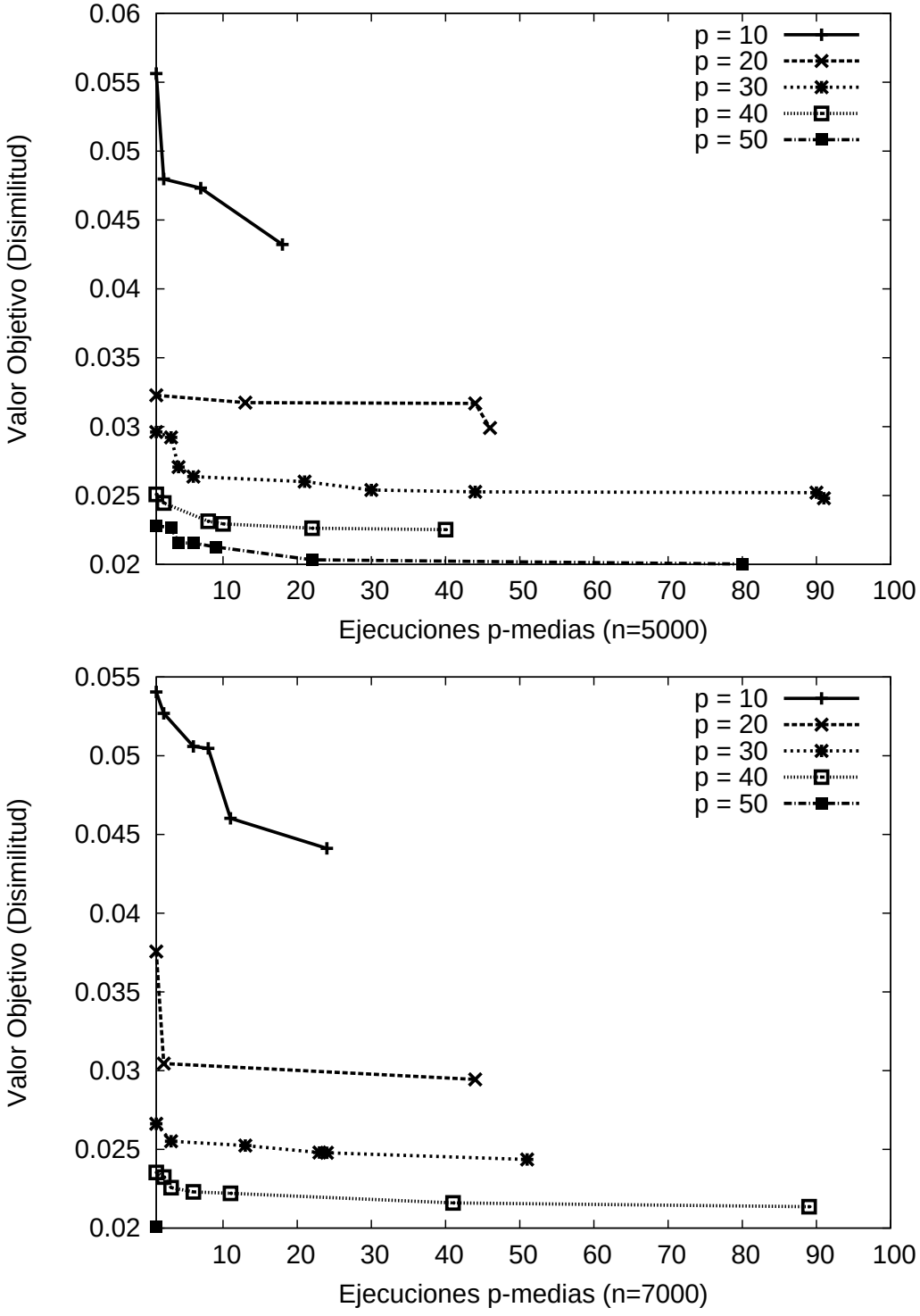


Figura 5.5: Evaluación de convergencia del algoritmo p-medias usando $f_{disp2}(X)$ (p-mediana) como medida de dispersión.

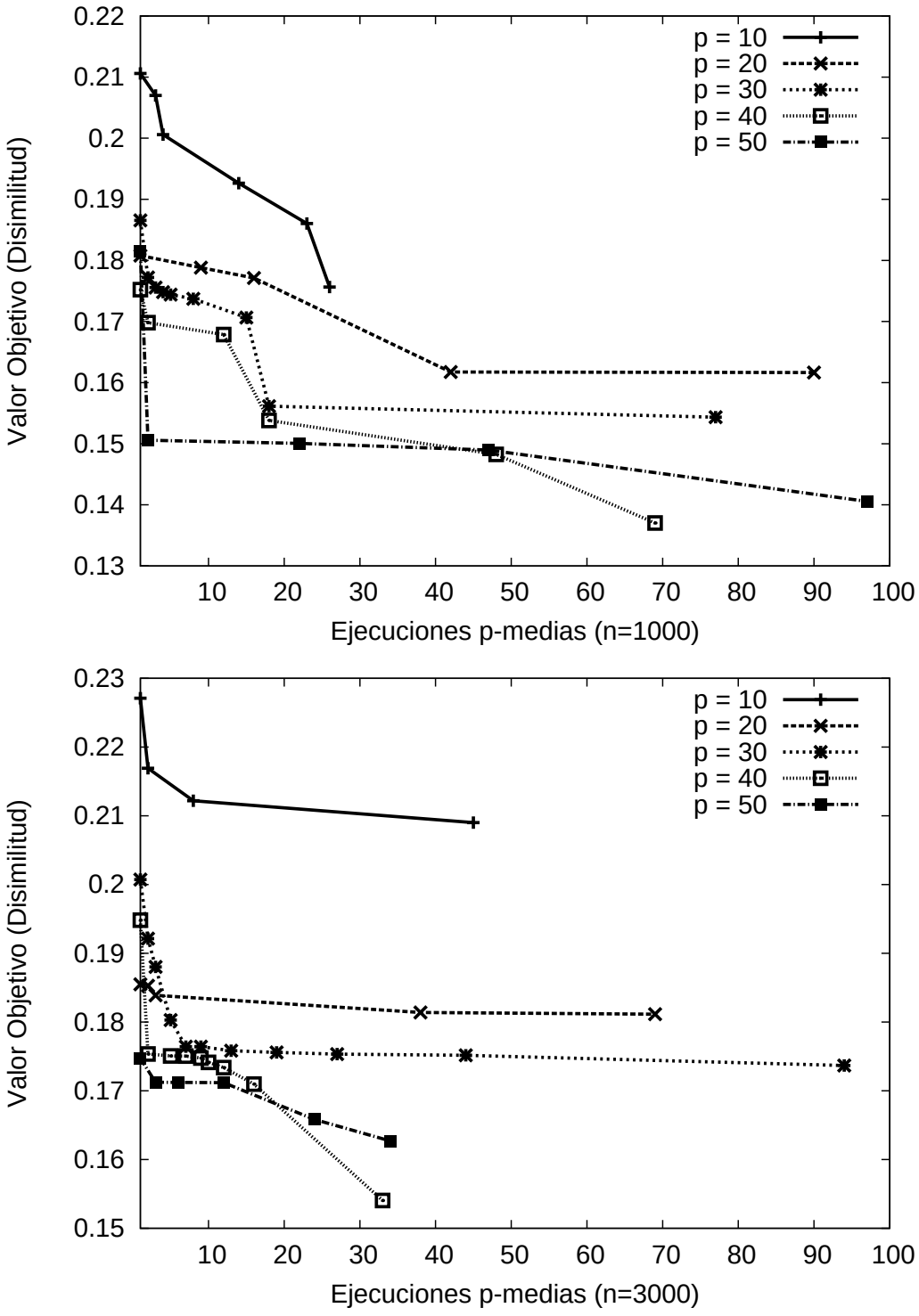


Figura 5.6: Evaluación de convergencia del algoritmo p-medias usando $f_{disp3}(X)$ (diámetro de la partición) como medida de dispersión.

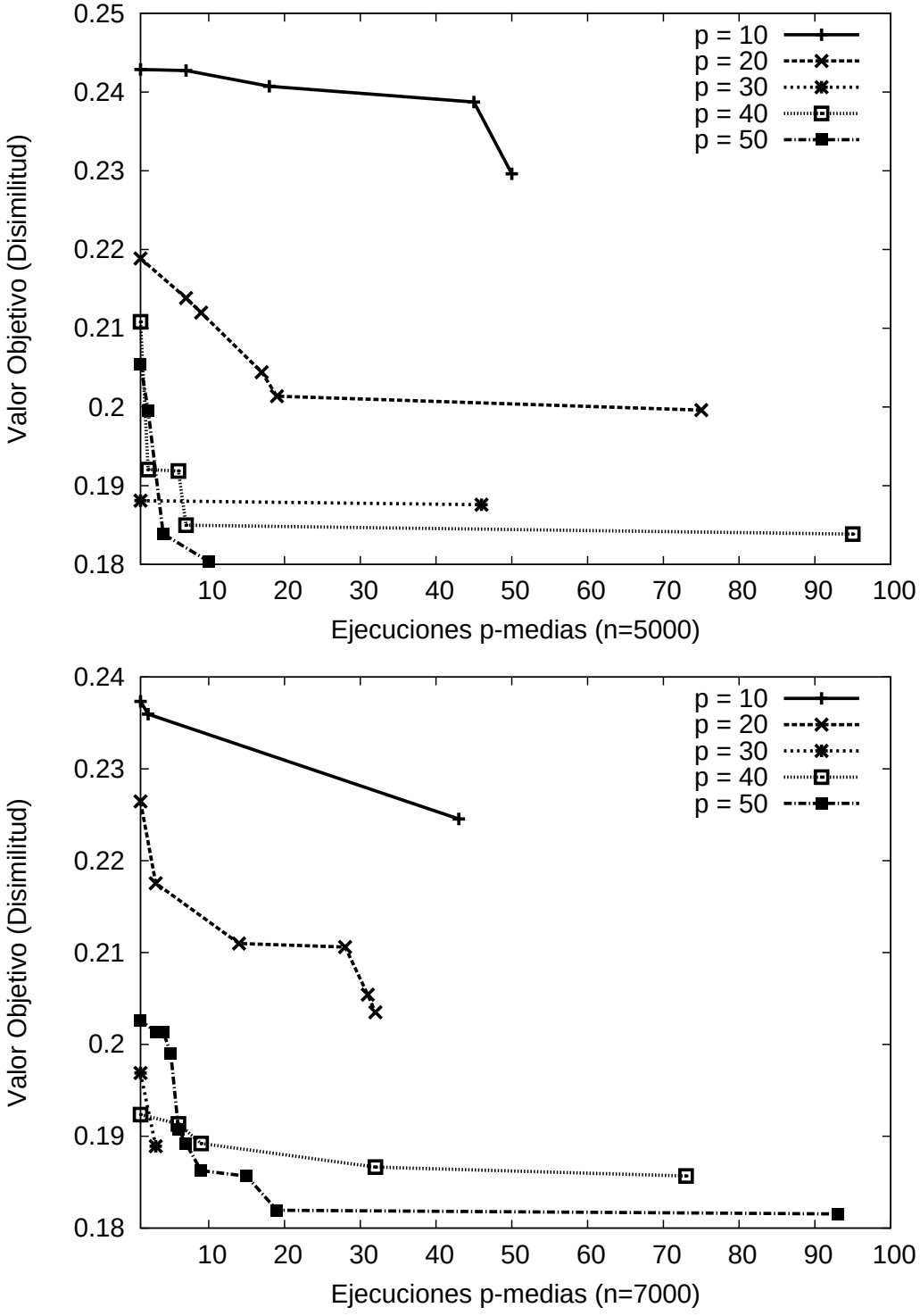


Figura 5.7: Evaluación de convergencia del algoritmo p-medias usando $f_{disp3}(X)$ (diámetro de la partición) como medida de dispersión.

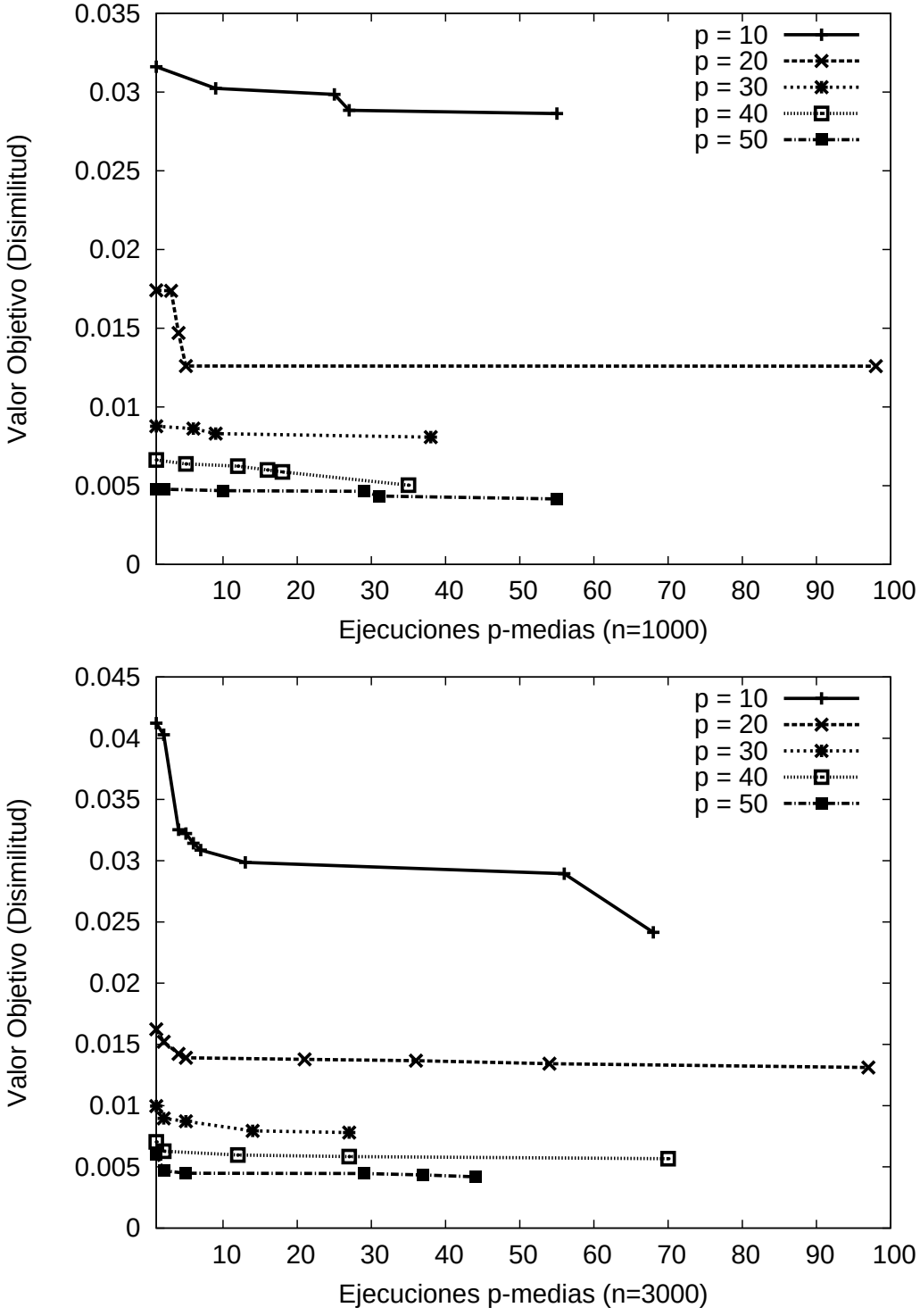


Figura 5.8: Evaluación de convergencia del algoritmo p-medias usando $f_{disp4}(X)$ (suma de las distancias intragrupales) como medida de dispersión.

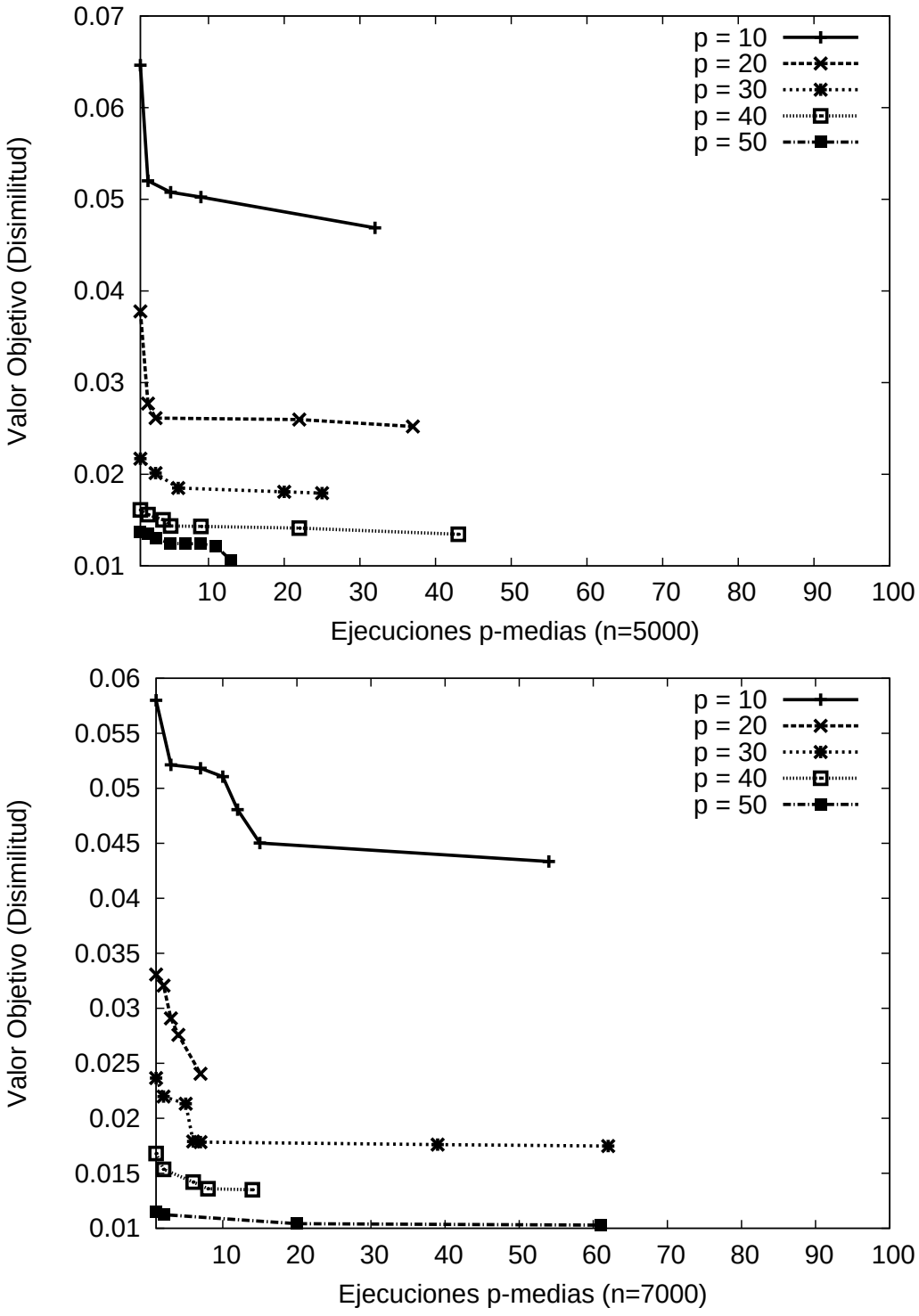


Figura 5.9: Evaluación de convergencia del algoritmo p-medias usando $f_{disp4}(X)$ (suma de las distancias intragrupales) como medida de dispersión.

requiere es obtener una partición de manera rápida cuya calidad no sea tan mala para posteriormente partir de ella y tratar de mejorar la calidad de la partición mediante el uso de un método de búsqueda local. Con este número de repeticiones se puede obtener una mejor solución que si lo ejecutaramos cinco o diez veces.

5.2.2 Elección del Número de Segmentos (p)

Objetivo: Obtener el mejor número de segmentos p para un determinado conjunto de clientes utilizando para ello el índice de validación de Davies-Bouldin (veáse Sección 2.1.7).

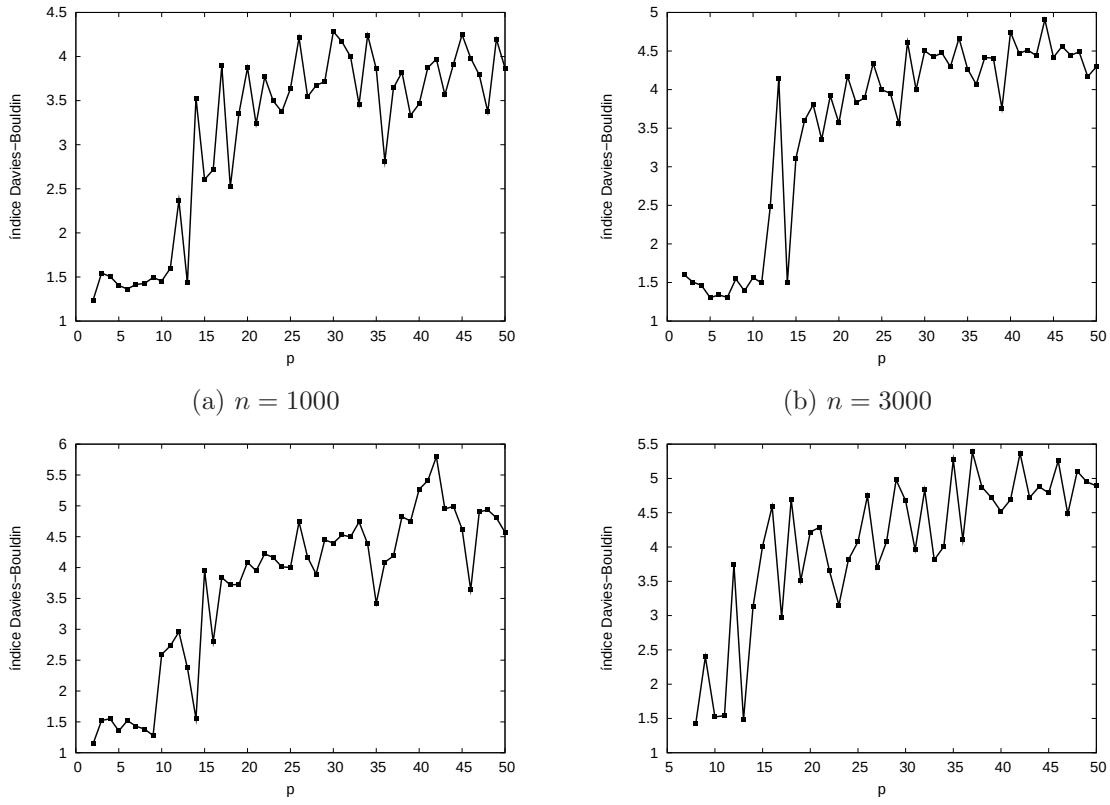
Características: Para este experimento se aplicaron 50 repeticiones del algortimo p-medias (número de repeticiones seleccionado en el experimento anterior), se fijó el parámetro $\beta = 1$ (selección de centros totalmente aleatoria), número de clientes $n = \{1000, 30000, 5000, 7000\}$ y $\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = 0.25$. El número de segmentos evaluados fue de $p \in \{2, 3, \ldots, 50\}$ para instancias de $n \in \{1000, 3000, 5000\}$ y $p \in \{8, \ldots, 50\}$ para $n = 7000$ ya que para la instancia real de $n = 17332$ el número de segmentos usados por la empresa se encuentra entre 60 y 80 segmentos generalmente, lo que corresponde entre 216 y 288 clientes por segmento (si fueran asignados de manera que existiera la misma cantidad de ellos por segmento).

Para instancias de 1000, 3000 y 5000 se han evaluado 49 diferentes valores de p puesto que el tiempo empleado es menor que para la instancia de 7000 clientes, además que el rango del número de segmentos (tomando como referencia el usado en la instancia real) sigue estando dentro del rango [2-50]. Para la instancia de tamaño 7000, el número de segmentos se encuentra en el intervalo [8-50] entre los cuales se encuentra el rango que podría suponerse en la realidad.

Discusión: Una vez aplicado el índice de Davies-Bouldin se obtuvieron las gráficas correspondientes a los diferentes tamaños de instancias y tipos de dispersión empleados. Como se puede observar, el número de segmentos no excede de $p = 13$; esto

puede ser debido a que los clientes tienen características muy similares utilizando la combinación de parámetros de ponderación $\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = 0.25$ en la función objetivo. Sin embargo tambien puede ser causa de la selección de centros iniciales para los cuales no se encontró alguna partición con un mejor índice de Davies-Bouldin. Las Figuras 5.13 a 5.11 muestran el comportamiento del índice de Davies-Bouldin encontrado para los valores de p establecidos. Mientras que la Tabla 5.1 muestra los tres mejores valores encontrados por el índice. Los números remarcados representan el mejor valor encontrado de los tres.

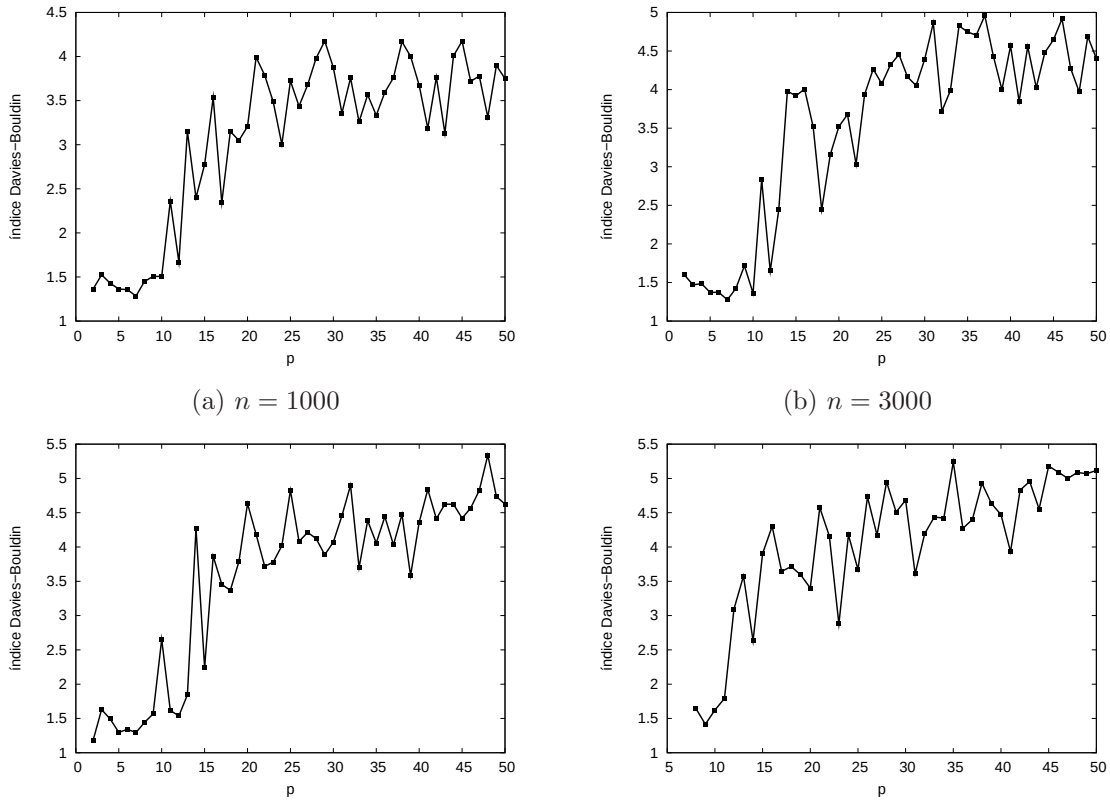
Conclusión: Se seleccionaron los tres mejores valores de p para los cuales el índice de Davies-Bouldin fue mejor (para cada tamaño de instancia y para cada tipo de dispersión). La Tabla 5.1 muestra los mejores tres valores de p encontrados por el índice de validación para cada tipo de instancia. El tipo de dispersión 1 corresponde a la suma de distancias intra-grupo, el tipo 2 al diámetro, el tipo 3 al de p-centro y el tipo 4 al de p-mediana



(a) $n = 1000$ (b) $n = 3000$

(c) $n = 5000$ (d) $n = 7000$

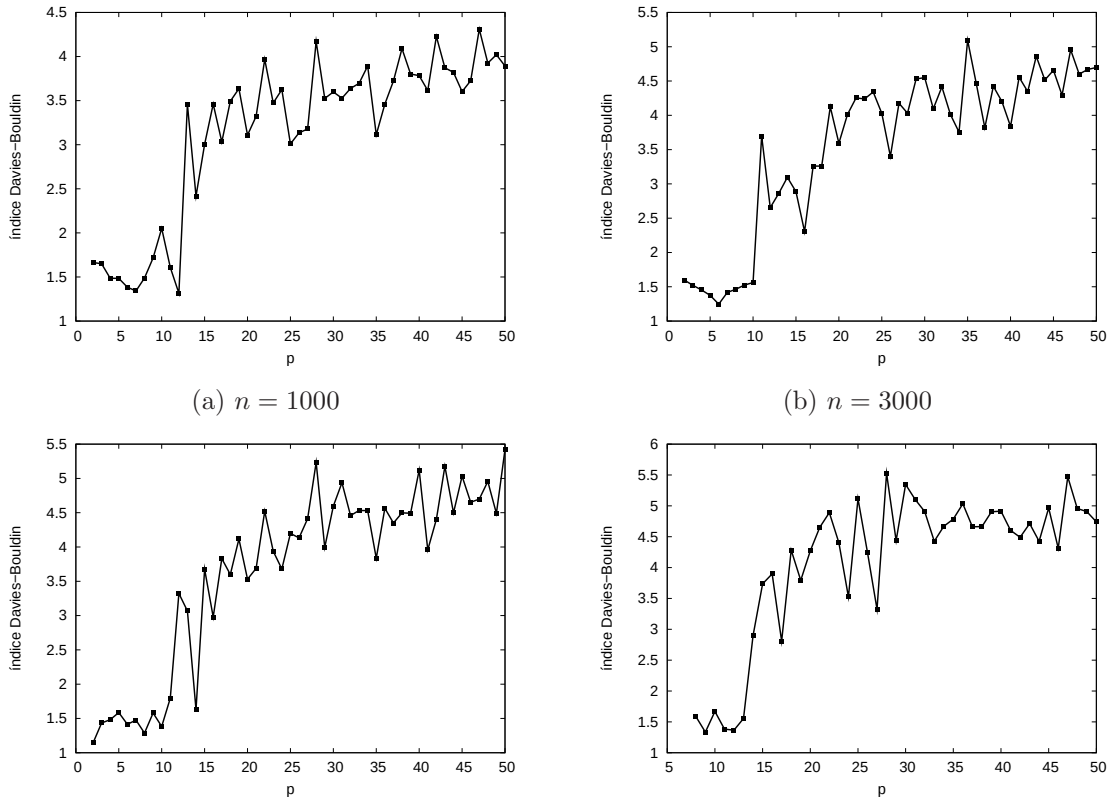
Figura 5.10: Evaluación del número de segmentos mediante el índice de Davies-Bouldin usando $f_{disp1}(X)$ (p-centro).



(a) $n = 1000$ (b) $n = 3000$

(c) $n = 5000$ (d) $n = 7000$

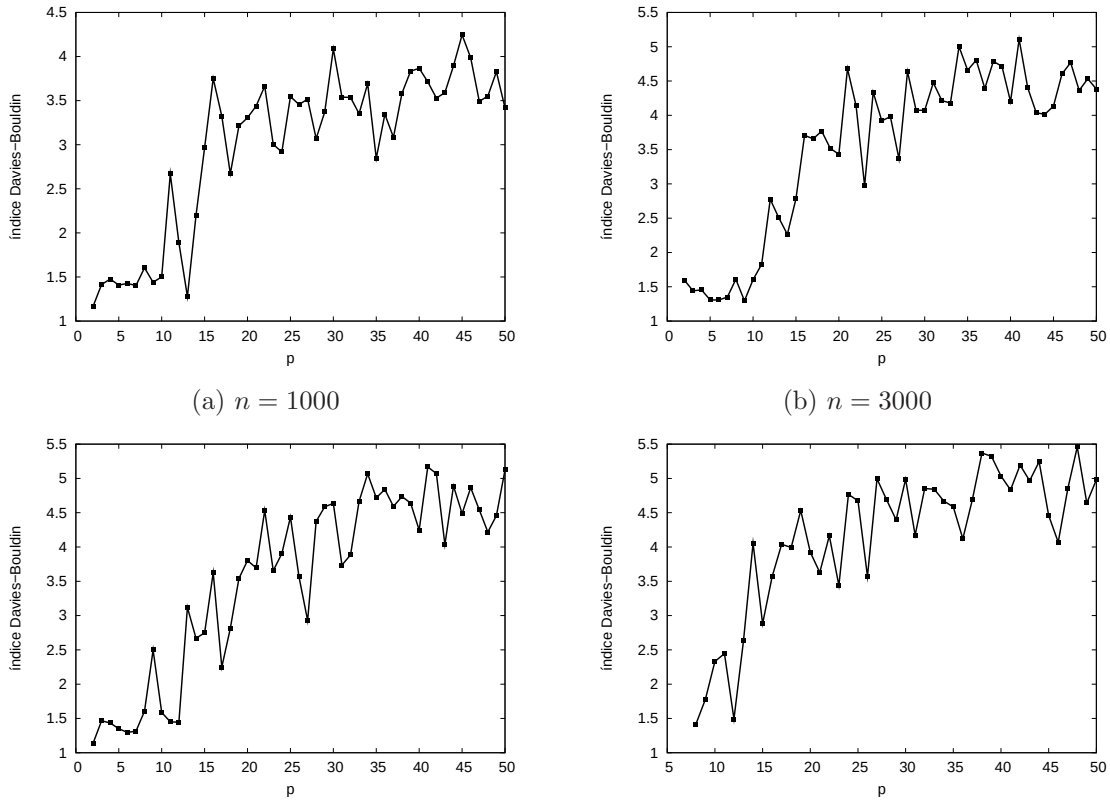
Figura 5.11: Evalución del número de segmentos mediante el índice de Davies-Bouldin usando $f_{disp2}(X)$ (p-mediana).



(a) $n = 1000$ (b) $n = 3000$

(c) $n = 5000$ (d) $n = 7000$

Figura 5.12: Evalución del número de segmentos mediante el índice de Davies-Bouldin usando $f_{disp3}(X)$ (diámetro de la partición).



(a) $n = 1000$ (b) $n = 3000$

(c) $n = 5000$ (d) $n = 7000$

Figura 5.13: Evalución del número de segmentos mediante el índice de Davies-Bouldin usando $f_{disp4}(X)$ (suma de las distancias intragrupales).

Dispersión	n	Real (p)	Davies-Bouldin (p)
1	1000	3-5	**2**, 7, 13
2	1000	3-5	6, 7, **12**
3	1000	3-5	**2**, 5, 6
4	1000	3-5	5, 6, **7**
1	3000	10-15	5, 6, **9**
2	3000	10-15	5, **6**, 7
3	3000	10-15	**5**, 6, 7
4	3000	10-15	5, **7**, 10
1	5000	16-25	**2**, 6, 7
2	5000	16-25	**2**, 8, 10
3	5000	16-25	**2**, 5, 9
4	5000	16-25	**2**, 5, 7
1	7000	23-35	**8**, 9, 12
2	7000	23-35	**9**, 11, 12
3	7000	23-35	**8**, 10, 13
4	7000	23-35	8, **9**, 10

Tabla 5.1: Número de segmentos encontrados por el índice de Davies-Bouldin para distintos tamaños de instancias y tipos de dispersión.

5.2.3 Parámetro de Calidad (β)

Objetivo: Determinar el valor del parámetro de calidad del GRASP para el cual la elección de centros iniciales proporciona mejores soluciones durante el algoritmo p-medias para las cuatro instancias en general.

Características: Para el siguiente experimento se ha fijado el número de segmentos cuyos valores se muestran en la Tabla 5.1 identificados para las cuatro instancias de tamaño $n \in \{1000, 3000, 5000, 7000\}$. Los valores del parámetro de calidad a evaluar son $\beta \in \{0, 0.2, 0.4, 0.6, 0.8, 1\}$. El algoritmo es repetido 50 veces para cada valor de β y n.

Discusión: Las Tablas 5.5 a 5.4 muestran el valor objetivo encontrado para diferentes valores de β aplicando las diferentes medidas de dispersión a una instancia de cada tamaño considerando los tres mejores valores de p encontrados con el índice de Davies-Bouldin. Entre los mejores valores de β encontrados de manera general fueron $\beta \in \{0.4, 0.6\}$ para la mayoria y $\beta \in \{0.8, 1\}$ para algunos casos. Los valores de $\beta \in \{0, 0.2\}$ reportaron los peores resultados.

Conclusión: Se decide seleccionar $\beta = 0.6$ dado que fue el valor para el cual se aproximaban las mejores evaluaciones de la función objetivo con mayor frecuencia. El valor de $\beta = 0.4$ pudo tambien ser seleccionado, sin embargo dado que las instancias contienen puntos extremos (clientes) lejanos a la mayoría de los clientes esto puede disminuir la cantidad de centros iniciales potenciales.

n	p	$\beta = 0$	$\beta = 0.2$	$\beta = 0.4$	$\beta = 0.6$	$\beta = 0.8$	$\beta = 1$
1000	2	0.2967	0.2967	0.2629	0.2629	0.26534	0.2701
1000	5	0.1845	0.1636	0.1595	0.1574	0.1523	0.1555
1000	6	0.1574	0.1446	0.1394	0.1472	0.1434	0.1452
3000	5	0.1773	0.1579	0.1603	0.1636	0.1590	0.1568
3000	6	0.1778	0.1463	0.1490	0.1494	0.1484	0.1490
3000	7	0.1578	0.1449	0.1413	0.1398	0.1366	0.1411
5000	2	0.3859	0.2739	0.2705	0.2701	0.2724	0.2718
5000	5	0.2307	0.1673	0.1647	0.1639	0.1599	0.1657
5000	9	0.2050	0.1303	0.1261	0.1327	0.1267	0.1258
7000	8	0.1563	0.1320	0.1325	0.1377	0.1333	0.1310
7000	10	0.1458	0.1265	0.1254	0.1268	0.1266	0.1255
7000	13	0.1303	0.1181	0.1167	0.1192	0.1192	0.1157

Tabla 5.2: Valor objetivo obtenido al variar el parámetro de calidad β usando como dispersión $f_{disp1}(X)$ (p-centro).

n	p	$\beta = 0$	$\beta = 0.2$	$\beta = 0.4$	$\beta = 0.6$	$\beta = 0.8$	$\beta = 1$
1000	5	0.1115	0.0874	0.0854	0.0870	0.0845	0.0821
1000	6	0.0872	0.0751	0.0721	0.0732	0.0728	0.0724
1000	7	0.0879	0.0639	0.0650	0.0682	0.0638	0.0625
3000	5	0.0935	0.0781	0.0823	0.0799	0.0833	0.0862
3000	7	0.0726	0.0599	0.0611	0.0611	0.0601	0.0639
3000	10	0.0577	0.0443	0.0485	0.0459	0.0468	0.0453
5000	2	0.3052	0.2421	0.1871	0.1868	0.1872	0.1884
5000	5	0.1717	0.0856	0.0838	0.0804	0.0820	0.0805
5000	7	0.1258	0.0587	0.0601	0.0585	0.0608	0.0611
7000	8	0.0697	0.0535	0.0523	0.0548	0.0500	0.0515
7000	9	0.0639	0.0459	0.0488	0.0452	0.0454	0.0501
7000	10	0.0599	0.0442	0.0446	0.0438	0.0455	0.0464

Tabla 5.3: Valor objetivo obtenido al variar el parámetro de calidad β usando como dispersión $f_{disp2}(X)$ (p-mediana).

n	p	$\beta = 0$	$\beta = 0.2$	$\beta = 0.4$	$\beta = 0.6$	$\beta = 0.8$	$\beta = 1$
1000	6	0.2401	0.2271	0.2112	0.2019	0.2053	0.2244
1000	7	0.2420	0.2140	0.1995	0.2059	0.2081	0.2069
1000	12	0.2030	0.1928	0.1779	0.1709	0.1803	0.1688
3000	5	0.2701	0.2561	0.2555	0.2536	0.2551	0.2491
3000	6	0.2620	0.2283	0.2373	0.2330	0.2344	0.2365
3000	7	0.2414	0.2255	0.2239	0.2280	0.2219	0.2199
5000	2	0.4794	0.3821	0.3806	0.3806	0.3812	0.3807
5000	6	0.2861	0.2198	0.2191	0.2218	0.2176	0.2221
5000	7	0.2559	0.2064	0.2109	0.2045	0.2073	0.2071
7000	9	0.2388	0.2159	0.2160	0.2112	0.2128	0.2166
7000	11	0.2344	0.2105	0.2043	0.2046	0.2077	0.2031
7000	12	0.2289	0.2052	0.1972	0.2037	0.2050	0.2026

Tabla 5.4: Valor objetivo obtenido al variar el parámetro de calidad β usando como $f_{disp3}(X)$ (diámetro de la partición).

n	p	$\beta = 0$	$\beta = 0.2$	$\beta = 0.4$	$\beta = 0.6$	$\beta = 0.8$	$\beta = 1$
1000	2	0.2183	0.1930	0.1877	0.18848	0.1885	0.1870
1000	7	0.0721	0.0461	0.0433	0.0448	0.04632	0.0466
1000	13	0.0252	0.0218	0.0206	0.0207	0.0220	0.0201
3000	5	0.0794	0.06634	0.06561	0.06070	0.06652	0.06411
3000	6	0.0804	0.0493	0.0488	0.04930	0.0546	0.0535
3000	9	0.0418	0.0336	0.0305	0.0286	0.0304	0.0346
5000	2	0.3029	0.1820	0.1804	0.1811	0.1836	0.18797
5000	6	0.1558	0.0533	0.0499	0.0520	0.0494	0.04977
5000	7	0.1162	0.0433	0.0387	0.0399	0.0401	0.0433
7000	8	0.0565	0.0359	0.0363	0.0365	0.0321	0.0370
7000	9	0.0499	0.0285	0.0298	0.0321	0.0315	0.0291
7000	12	0.0403	0.0229	0.0234	0.0221	0.0229	0.0223

Tabla 5.5: Valor objetivo obtenido al variar el parámetro de calidad β, usando como dispersión $f_{disp4}(X)$ (suma de las distancias intragrupales).

5.3 Experimento B: Contrucción y Mejora de Particiones

Objetivo: Construir particiones utilizando los parámetros establecidos en los experimentos anteriores y tratar de mejorarlas mediante el método de búsqueda local propuesto (VNS) para determinar las ventajas y desventajas de su uso.

Características: Para este experimento se aplicó, a una instancia de cada tamaño (1000, 3000, 5000, 7000), el algoritmo de GRASP propuesto utilizando los parámetros $\beta = 0.6$ y el tres mejores valores de p (veáse la Tabla 5.5) encontrados mediante el índice de Davies-Bouldin. La mejor solución obtenida es introducida a un método de búsqueda local basado en entornos variables con la finalidad de mejorar la solución encontrada en la construcción. La ponderación de la función objetivo es de α_1=α_2=α_3=$\alpha_4 = 0.25$.

Discusión: La Tabla 5.6 muestra los resultados obtenidos al aplicar el método propuesto. Las columnas de dicha tabla muestran el valor objetivo encontrado para la fase de construcción y la de mejora, el porcentaje relativo de mejora obtenido una vez aplicada la VNS así como los tiempos de cómputo (medidos en segundos) tanto para cada fase como total. Como puede observarse en dicha tabla, la fase de VNS encontró mejoras de hasta el 41 % como lo fue en el caso de la instancia de $n = 1000$ y $p = 13$ o $n = 7000$ y $p = 12$. El tiempo de cómputo empleado no excede de los 500 segundos para los casos estudiados, sin embargo, este tiempo puede variar dependiendo los movimientos realizados durante la VNS. El tiempo de cómputo requerido de GRASP depende tambien de la convergencia hacia el óptimo local.

Conclusión: Para los casos estudiados, el método de búsqueda local propuesto mejoró notablemente la solución obtenida por la fase contructiva en un tiempo de cómputo razonable.

n	p	f(GRASP).	f(VNS)	Mejora(%)	t_{GRASP}	t_{VNS}	t_{Total}
1000	2	0.18874	0.18473	2.167	3.4	2.13	5.53
1000	7	0.04602	0.03521	30.69	3.62	3.53	7.15
1000	13	0.02091	0.01483	41.04	3.67	4.98	8.65
3000	5	0.06407	0.05427	18.06	36.81	33.07	69.88
3000	6	0.05227	0.04139	26.30	35.71	66.86	102.57
3000	9	0.03098	0.02453	26.29	38.87	47.82	86.69
5000	2	0.17968	0.17876	0.51	104.15	10.45	114.6
5000	6	0.05200	0.04163	24.92	111.56	116.33	227.89
5000	7	0.04462	0.03313	34.68	111.73	81.19	192.92
7000	8	0.03178	0.02705	17.47	223.88	136.76	360.64
7000	9	0.03193	0.02284	39.80	222.54	157.05	379.59
7000	12	0.02151	0.01535	40.08	221.21	198.35	419.56

Tabla 5.6: Resultados obtenidos al aplicar el método propuesto usando la función de dispersión $f_{disp4}(X)$.

5.3.1 Sensibilidad de la Solución: Variación de los Parámetros de Ponderación (α_r)

Objetivo: Visualizar las particiones finales para diferentes valores de ponderación (α_1,α_2,α_3,α_4) en la función objetivo y mostrar como varía la solución usando diferentes valores de estos parámetros.

Características: Para este experimento se han fijado el número de segmentos a $p = 5$ y el tamaño de instancia a $n = 1000$, esto con el fin de poder visualizar mejor la variación de las particiones al cambiar los parámetros de ponderación. El parámetro de calidad usado es de $\beta = 0.6$. Se emplea el método completo usando la función de dispersión de la suma de distancias intra-grupo (3.7). Los valores a variar corresponden al parámetro de ponderación de la función objetivo (estos se muestran en la Tabla 5.7).

Caso	α_1	α_2	α_3	α_4
A	0.10	0.40	0.10	0.40
B	0.70	0.10	0.10	0.10
C	0.25	0.25	0.25	0.25
D	1	0	0	0
E	0	1	0	0
F	0	0	1	0
G	0	0	0	1

Tabla 5.7: Valores a variar de los parámetros de ponderación de la función objetivo.

Discusión: Las Figuras 5.14 a 5.19 muestran las particiones finales para cada caso. Los primeros tres casos corresponden a una evaluación similiar a la que hace la empresa. Primero se observa la partición final dando más peso a los atributos comerciales *volumen de compra* y *tipo de establecimiento* (Caso A) los cuales son

considerados primeramente como los de mayor importancia para la empresa (siempre y cuando la dispersión no sea muy afectada).

Como se puede observar en la Figura 5.14 los segmentos de la partición resultante, percibida desde la ubiciación geográfica, se empalman unos con otros. Además presenta gran dispersión entre los clientes de cada segmento formado. Entonces en estos casos la empresa le va dando más peso a la dispersión para evitar tener segmentos donde la dispersión se ve muy afectada. Por ejemplo si se asignan los pesos del Caso B, la dispersión mejora mucho pero eso deteriora la asignación con respecto a los demás atributos puesto que tienen importancia más baja al momento de segmentar (Figura 5.15). Las Figuras 5.16 a 5.19 muestran la partición final cuando se asignan pesos iguales y los casos donde cada atributo tienen el máximo peso de importancia, respectivamente.

Conclusión: En las particiones obtenidas se pudo observar que no se pueden apreciar grupos bien definidos de clientes cuando los cuatro atributos son considerados igual de importantes. Aunque en la práctica lo ideal para la empresa es obtener segmentos de clientes cuyo tipo de establecimiento y volumen de compra sean lo más similares posible, esto no siempre ocurre puesto que en la realidad la ubicación de los clientes es un factor que afecta a los segmentos con estas características sin dejar de mencionar que existe el atributo de tipo de contrato el cual tambien influye. Cabe recalcar que no se pretende evaluar los valores de α_r para los cuales la solución es la mejor ya que el valor de este parámetro lo determina la empresa según sus necesidades.

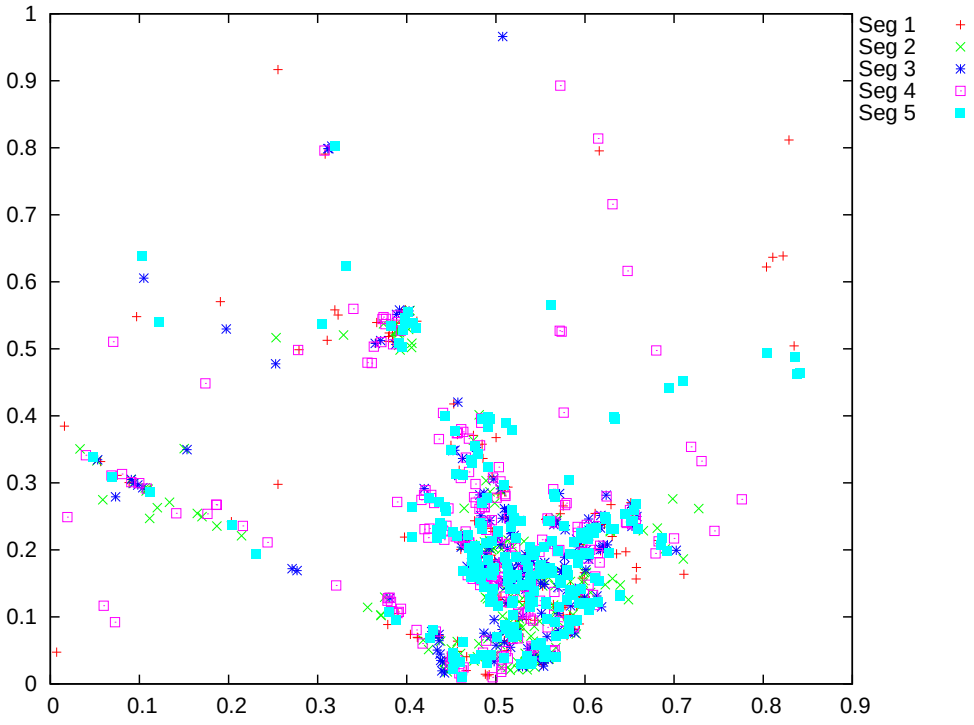


Figura 5.14: Segmentación final al aplicar el Caso A.

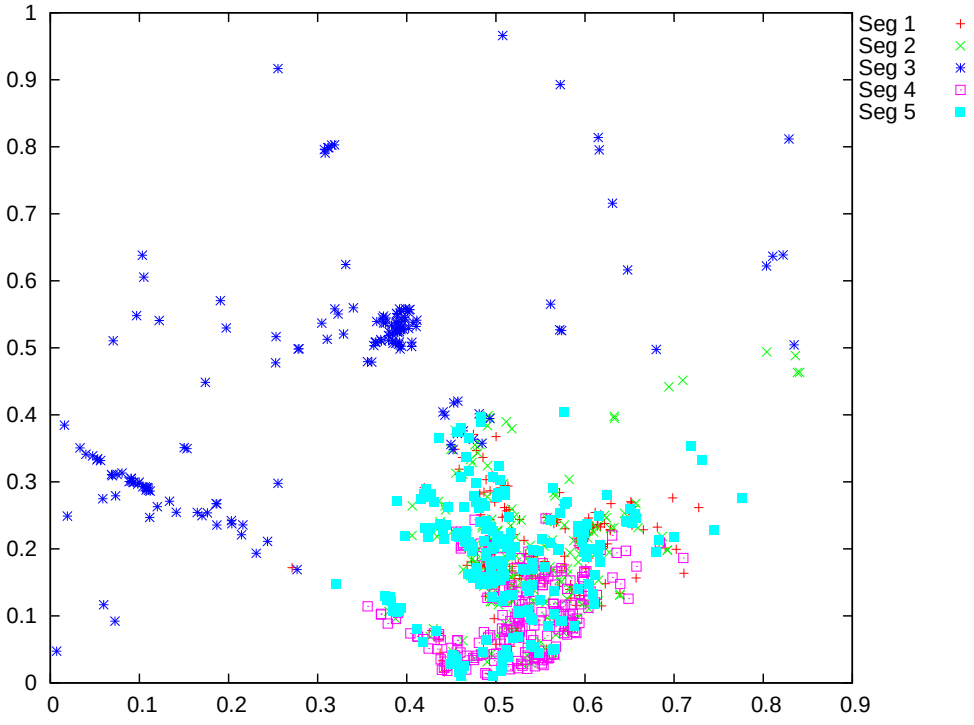


Figura 5.15: Segmentación final al aplicar el Caso B

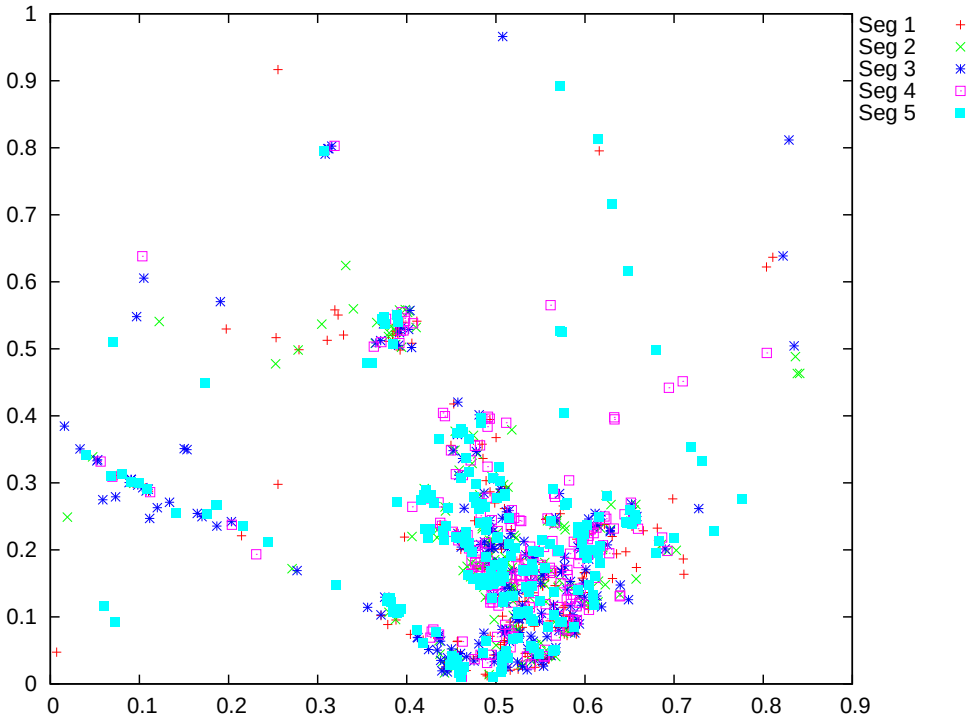


Figura 5.16: Segmentación final al aplicar el Caso C.

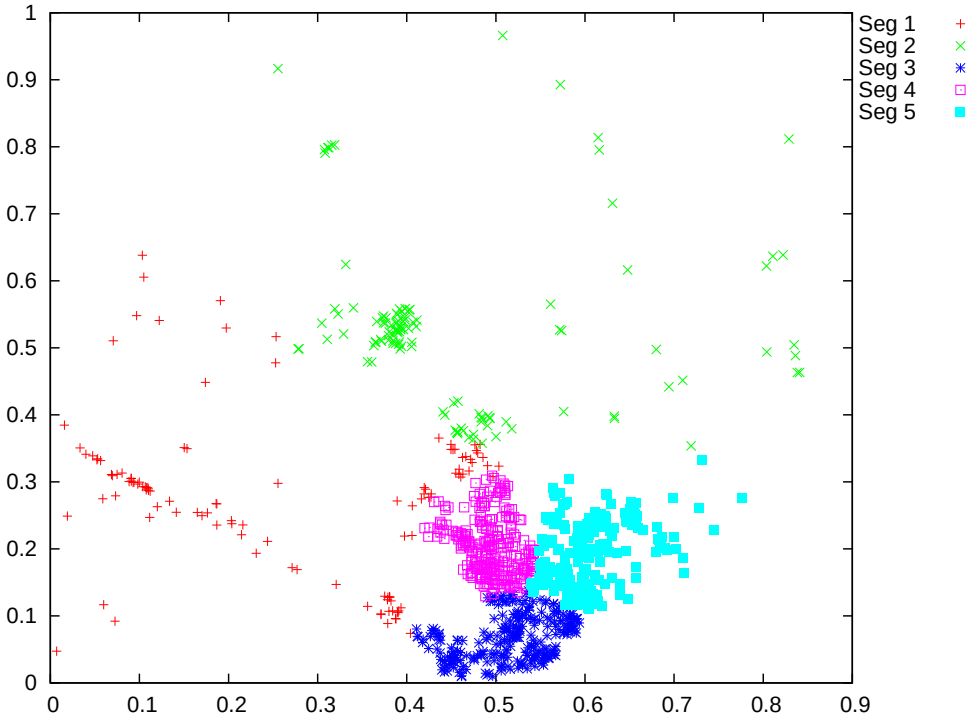


Figura 5.17: Segmentación final al aplicar el Caso D.

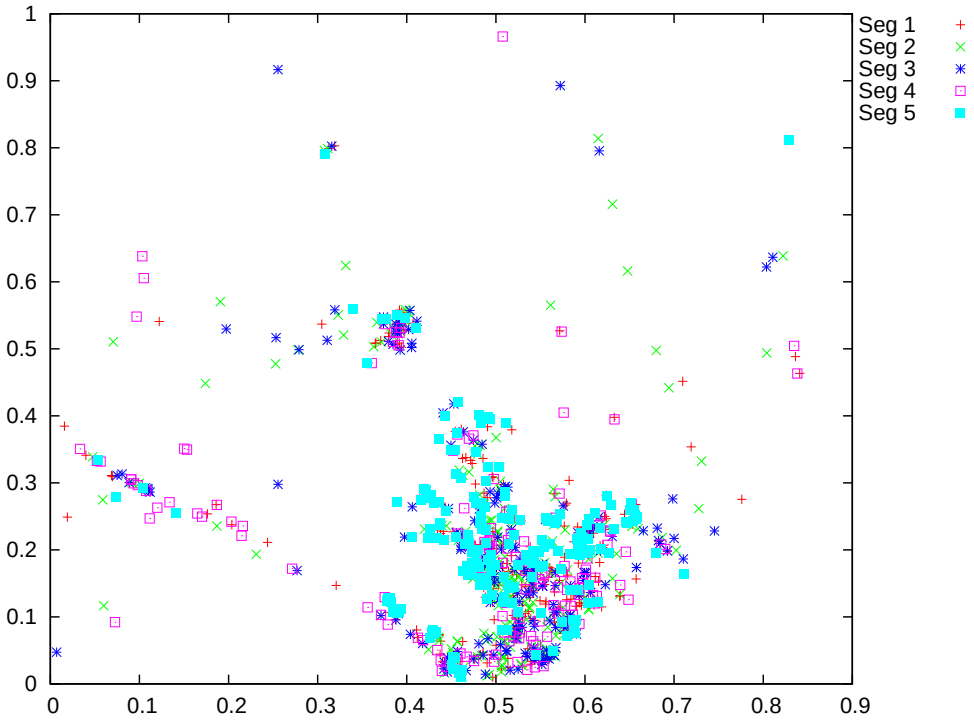


Figura 5.18: Segmentación final al aplicar el Caso E.

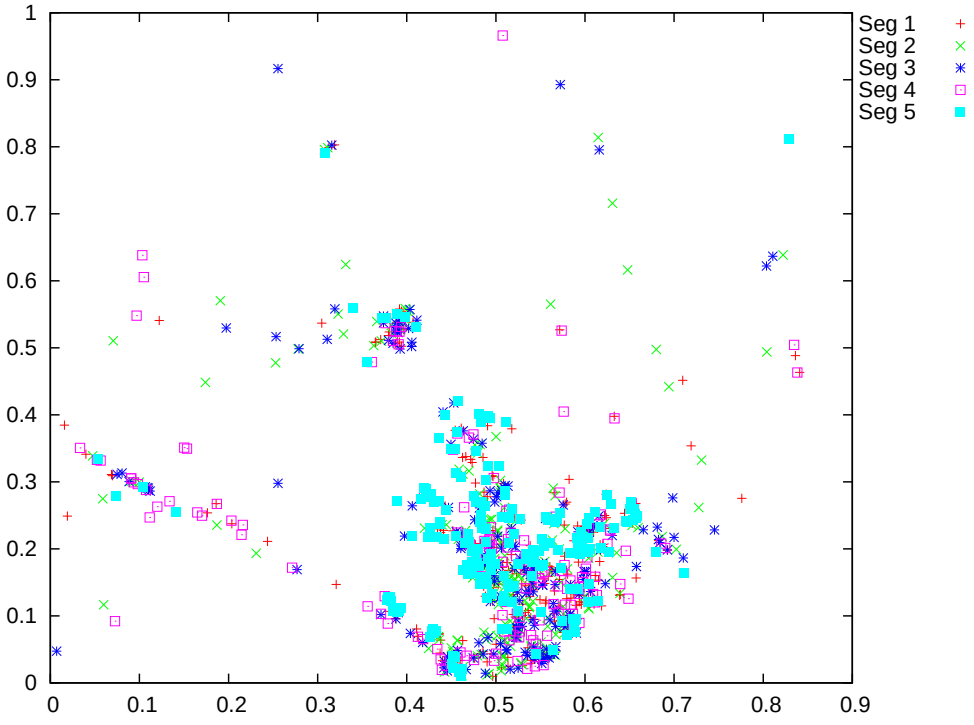


Figura 5.19: Segmentación final al aplicar el Caso F.

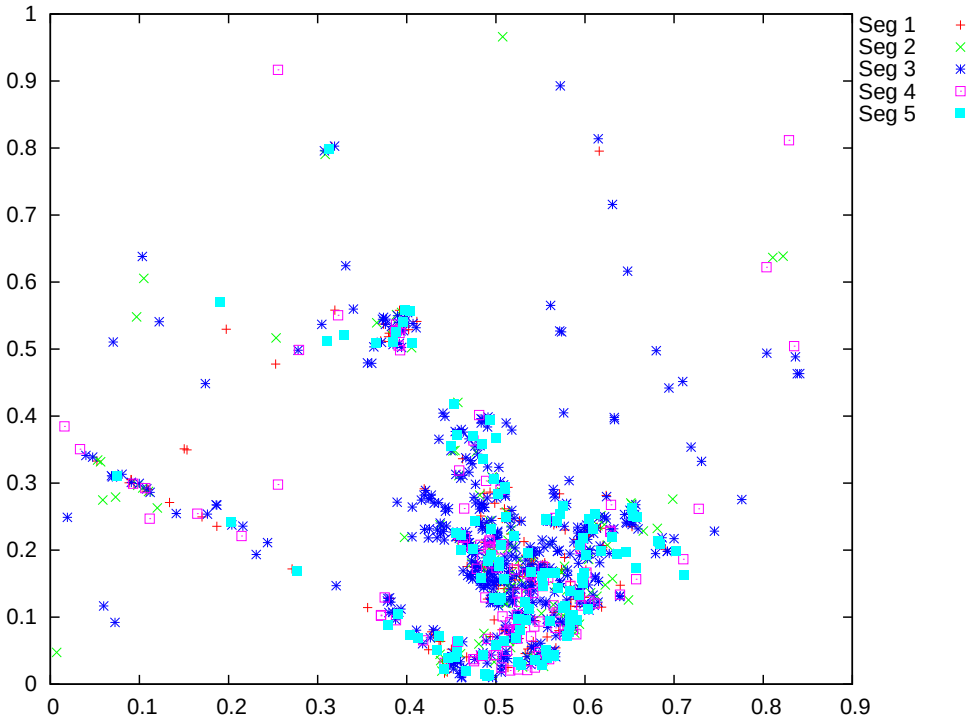


Figura 5.20: Segmentación final al aplicar el Caso G.

5.4 Experimento C: Metodología Aplicada a Instancia Preprocesada

5.4.1 Reducción del Número de SKUs

Objetivo: Reducir el número de productos mediante la agrupación de éstos utilizando un algoritmo de agrupamiento jerárquico. La razón del agrupamiento de productos es debido a que se desea reducir el tiempo ó costo computacional para instancias de gran tamaño. Además de aprovechar la estructura que tienen los datos para para hacer mejor uso de los recursos computacionales. Cabe recalcar que el agrupamiento de los productos identificados no afecta el comportamiento del cliente en cuanto el volumen de compra ya que dicho volumen estará representado por la suma de los volúmenes de todos los productos que el cliente compra para cada grupo resultante.

Características: La cantidad de productos identificados en la muestra real es de 210 de productos. Se utilizan tres métodos de agrupamiento jerárquico : vecino más próximo, vecino más lejando y enlace promedio en conjunto con la distancia de Pearson como métrica para agrupar los productos según su coeficiente de correlación. De los tres métodos utilizados para agrupar el conjunto de SKUs, uno es seleccionado para obtener por medio de su dendrograma los grupos que han de formarse. Para ello es utilizado un parámetro de tolerancia τ, el cual indica el mínimo coeficiente de correlación que debe existir en cada grupo representado por cada rama del dendrograma formado, el párametro de tolerancia seleccionado proporciona un número fijo de grupos de SKUs a formar. Los niveles de tolerancia analizados fueron para $\tau \in \{0.95, 0.90, 0.85, 0.80, 0.75, 0.70, 0.65, 0.60\}$ de los cuales se seleccionó aquel que fue más conveniente para la reducción.

Discusión: El resultado principal obtenido de este experimento es la reducción del número de SKUs identificados en la instancia real proporcionada el cual inicialmente fue de 201 y fue reducido a solamente 32 grupos donde para cada uno de ellos el coeficiente de correlación mínimo permitido (τ) es de 0.90. Para la obtención de dicho resultado, primeramente se observó en forma detallada los diferentes SKUs identificados, de dicha observación se pudieron identificar SKUs cuyos valores de atributos eran idénticos. La razón de que existan productos con características completamente iguales se debe a que éstos verdaderamente pertenecen a productos diferentes no es posible distinguir la diferencia entre ellos puesto que es necesario incluír más atributos que puedan mostrar dicha diferencia. Como se consideran solamente tres atributos por producto en este caso, dichos SKUs son pre-agrupados de manera que dos o más productos con valores idénticos formen uno solo con dichas características. Una vez aplicada esta primera reducción el número de SKUs que se obtuvo fue de 114 productos. La Tabla 5.8 muestra un ejemplo de como se realizó dicho pre-agrupamiento. Se realizó una prueba de hipótesis para mostrar

Un vez reducido el número de productos por medio del agrupamiento de áquellos con características idénticas, se procede a la aplicación de métodos jerárquicos

SKU	Retornabilidad	Presentación	Marca
1	Retornable	1 Lto.	Sabor
2	Retornable	355 ml.	Colas
3	Retornable	355 ml.	Colas
4	Retornable	500 ml.	Jugo
5	Retornable	600 ml.	Agua
6	Retornable	2 Ltos.	Colas
7	No Retornable	2 Ltos.	Colas
8	No Retornable	1 Lto.	Sabor
9	No Retornable	600 ml.	Sabor
10	No Retornable	600 ml.	Sabor

Tabla 5.8: Reducción del número de SKUs mediante la pre-agrupación de productos con características idénticas.

como el vecino más proximo, vecino más lejano y enlace promedio para agrupar el resto de los productos utilizando la distancia de Pearson $(1 - D)$ para medir la similitud entre un par de productos mediante su coeficiente de correlación (D). Para cada uno de ellos se obtuvo su correspondiente dendrograma los cuales se muestran en las Figuras 5.21, 5.22 y 5.23, respectivamente.

Para cada uno de los dendrogramas se obtuvo el coeficiente de correlación Pearson mínimo que se encontraba para cada grupo formado (identificado por cada rama del árbol de jerarquías). Una vez recorrido todo el dendrograma, se analizó el número de grupos formados según un nivel de tolerencia τ permitido. Dicho nivel especifica el mínimo coeficiente de correlación permitido en cada grupo de SKUs formado para ser seleccionado como tal.

Se evaluaron diferentes valores de τ para los tres métodos de agrupamiento jerárquico y los resultados obtenidos se muestran en la Tabla 5.9. Observando la información de dicha tabla, los métodos más convenientes para agrupar los SKUs fueron los métodos *vecino más lejano* y *enlace promedio* dado que requieren formar

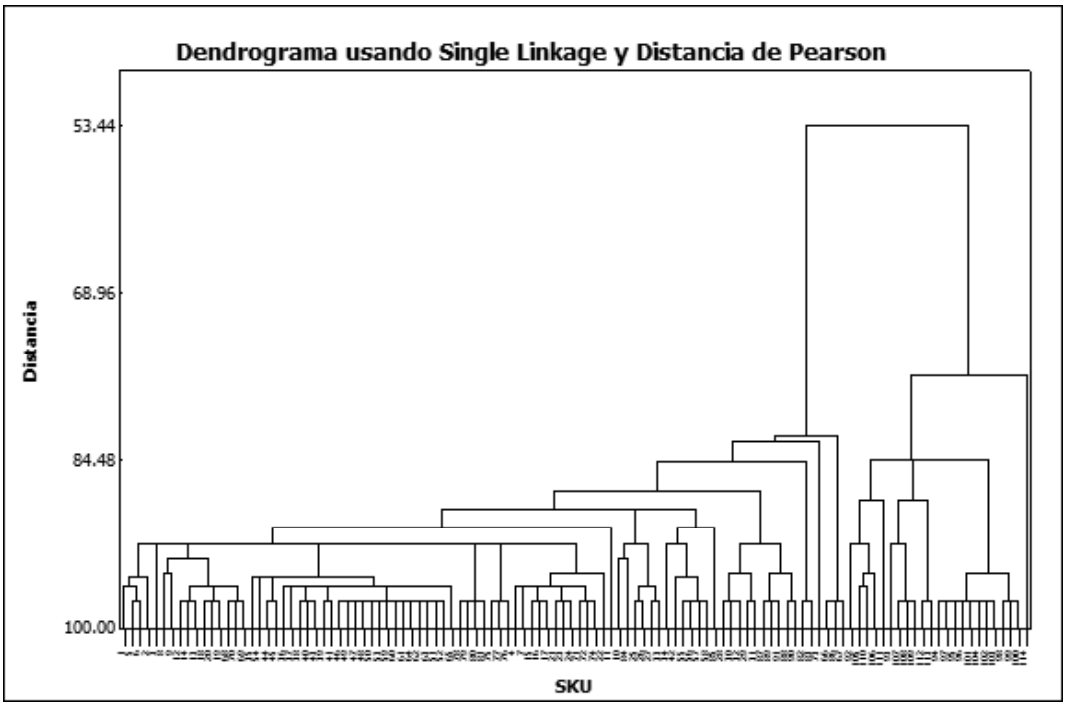


Figura 5.21: Dendrograma obtenido por MINITAB al aplicar el vecino más cercano.

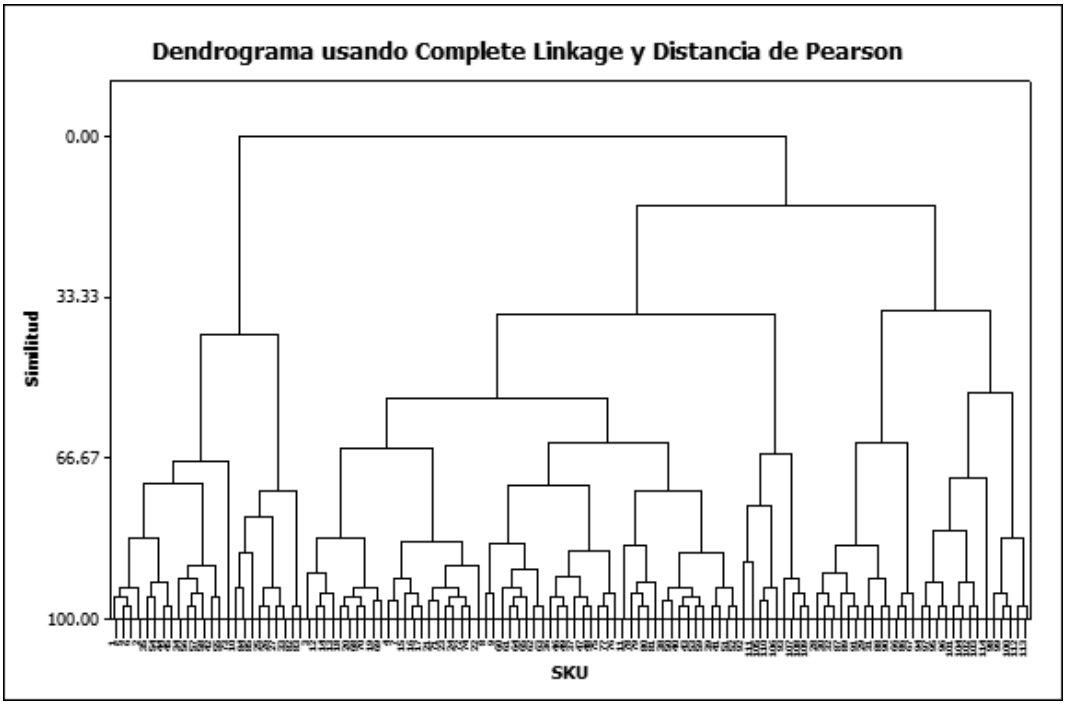


Figura 5.22: Dendrograma obtenido por MINITAB al aplicar el vecino más lejano.

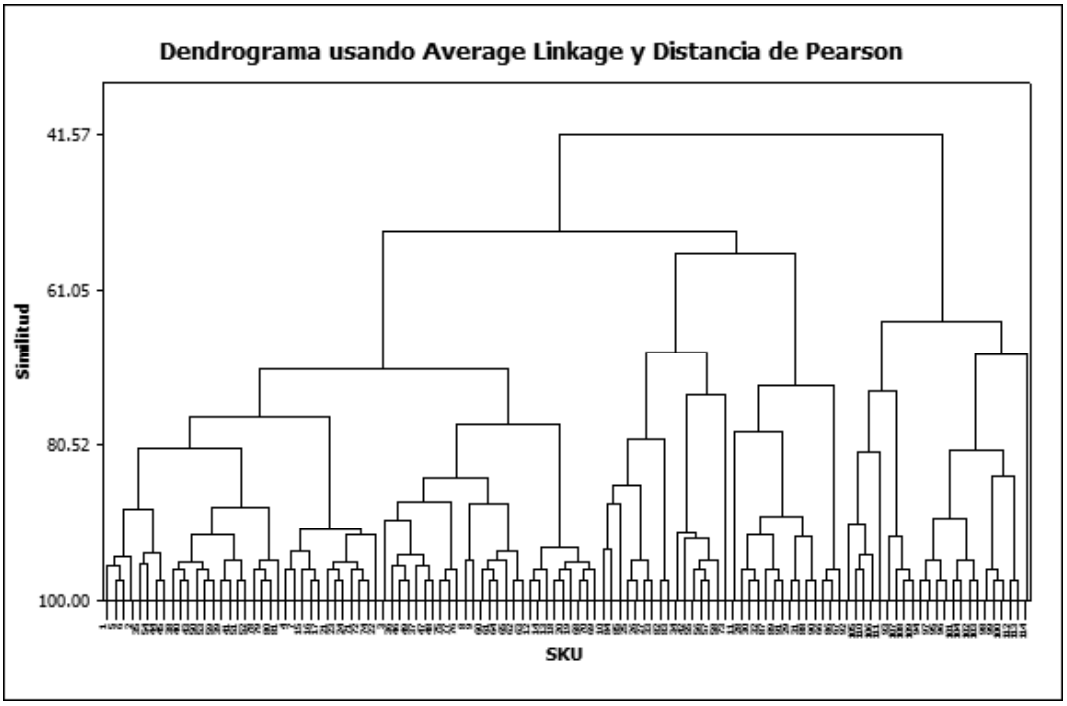


Figura 5.23: Dendrograma obtenido por MINITAB al aplicar el enlace promedio.

una menor cantidad de grupos para un mismo nivel τ. Para ambos métodos se analizaron las características de los grupos formados con un $\tau = 0.90$. Este nivel de tolerancia fue seleccionado ya que con éste se puede reducir una mayor cantidad de grupos sin afectar mucho la similitud intragrupal ya que el mínimo coeficiente permitido entre SKUs del mismo grupo es de $\tau = 0.90$.

Método	**Nivel de Tolerancia τ**								
Jerárquico	**1**	**0.95**	**0.90**	**0.85**	**0.80**	**0.75**	**0.70**	**0.65**	**0.60**
Más cercano	114	45	38	35	31	29	29	27	27
Más lejano	114	39	32	29	26	24	22	22	17
Promedio	114	40	32	29	26	23	21	21	19

Tabla 5.9: Grupos de SKUs obtenidos utilizando diferentes niveles de tolerencia τ

Para ambas agrupaciones encontradas utilizando un $\tau = 0.90$, las características de los SKUs que las conformaban fueron muy similares. Ambos agrupamientos presentaban muy pocas diferencias en cuanto a la asignación de los productos a ca-

da grupo. Decidir cual método agrupó mejor los 114 SKUs para dicho τ no puede determinarse fácilmente mediante las características de cada grupo formado puesto que son muy similares para ambos métodos. Por lo tanto se decidió seleccionar el método del vecino más lejano ya que presentó levemente un mejor agrupamiento que el método de enlace promedio y además para diferentes valores de τ el número de grupos formados es menor en la mayoría de los casos.

Conclusión: Como resultado se obtuvieron 32 grupos de SKUs de 201 identificados inicialmente. Esta reducción permite que una matriz de datos de entrada en la metodología propuesta que inicialmente era de 465884 filas (registros de clientes) y 205 columnas (números de SKUs, tipo de contrato, tipo de establecimiento y coordenadas geográficas) sea ahora de tamaño 173669 filas y 36 columnas. En dicha matriz el volumen de compra por cliente ya no es representado por el volumen de compra para cada SKU sino por la suma de los volúmenes de compra realizados por el cliente sobre los productos que conforman dicho grupo dividido entre el volumen de compra total del cliente sobre todos los productos.

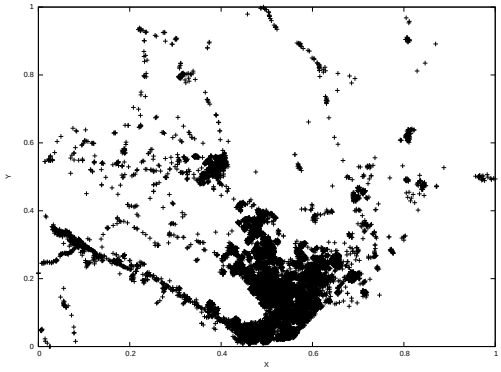
5.4.2 Creación de Metaclientes

Objetivo: Reducir el número de clientes a agrupar por medio de la creación de metaclientes (grupos de clientes con características similares) para resolver instancias de gran tamaño en menor tiempo y con el menor uso de los recursos computacionales posible.

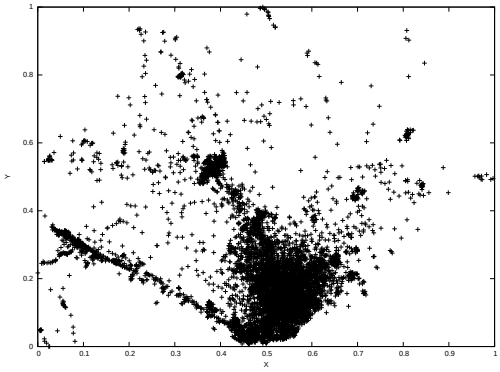
Características: Para la reducción se hace uso de la matriz de datos obtenida después de la reducción del número de SKUs. De dicha matriz se obtuvieron los grupos de clientes cuyo tipo de contrato y establecimiento fueran iguales. Se obtuvo la matriz de coeficientes de correlación de Pearson para cada uno de los grupos y se crearon metaclientes conformados por aquellos clientes del mismo grupo cuyo coeficiente de correlación fuera de $\gamma \in \{0.95, 0.90, 0.85, 0.80, 0.75\}$ entre todos ellos y poder deducir la reducción a ese nivel.

Discusión: La Figura 5.24 muestra, desde una perspectiva geográfica, el cambio sufrido en el conjunto de clientes real una vez creados los metaclientes. Podemos observar como es que la instancia real (Figura 5.24a) va cambiando y disminuyendo su tamaño (valor entre paréntesis) una vez que se crean los metaclientes para diferentes valores de tolerancia γ (Figuras 5.24b a 5.24f). El conjunto de clientes se torna cada vez más disperso cuando γ se decrementa.

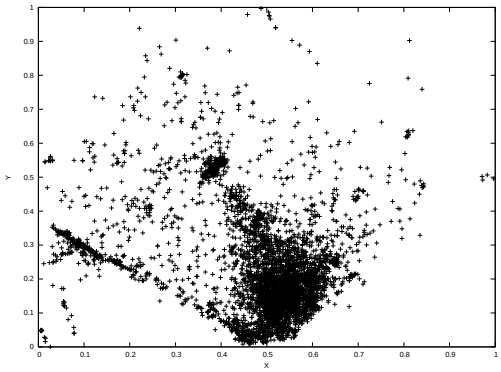
Conclusión: Se selecciona el conjunto de clientes obtenido con $\gamma = 0.95$. La razón es que se pudo reducir la instancia real de 17332 a 7203 clientes con tan solo dar 5 % de flexibilidad a la hora de crear metaclientes. Es decir, el conjunto de clientes resultante no es muy afectado y es más acorde a la instancia real. La reducción obtenida representa más del 58 %. Se realizó una prueba de hipótesis para probar la significancia del coeficiente de correlación seleccionado (véase Apéndice B).



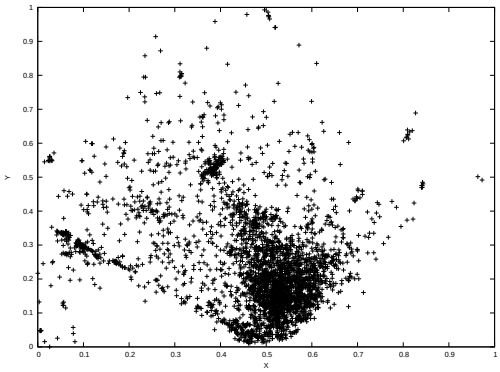
(a) Instancia no preprocesada - correlación 100 % (17332 clientes).



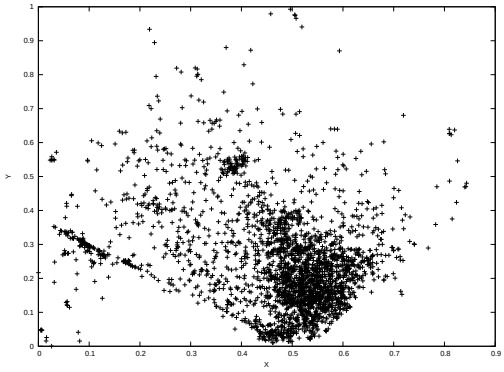
(b) Instancia preprocesada - correlación 95 % (7203 clientes).



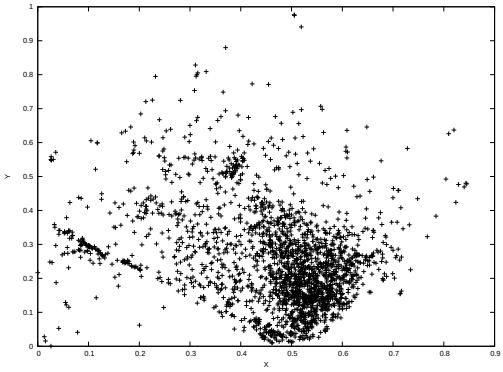
(c) Instancia preprocesada - correlación 90 % (4604 clientes).



(d) Instancia preprocesada - correlación 85 % (3434 clientes).



(e) Instancia preprocesada - correlación 80 % (2739 clientes).



(f) Instancia preprocesada - correlación 75 % (2256 clientes)

Figura 5.24: Instancia real antes y después de la creación de metaclientes.

5.4.3 Aplicación del Método Propuesto

Objetivo: Evaluar el beneficio de la aplicación del método propuesto.

Características: Los parámetros a utilizar en este experimento son $p = 9$ (obtenido mediante el índice de Davies-Bouldin), $\beta = 0.6$, se consideraron varios casos en el peso para los atributos: (a) $\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = 0.25$, (b) $\alpha_1 = 1$, (c) $\alpha_2 = 1$; (d) $\alpha_3 = 1$ y (e) $\alpha_4 = 1$. Se aplica el método utilizando la función de dispersión de suma de distancias intra-grupo (3.7).

Discusión: La Tabla 5.10 muestra los resultados obtenidos al aplicar el método tanto a la instancia preprocesada como a la real. En la tercera columna de la tabla se muestra el valor de la función objetivo obtenido en la fase de contrucción de la partición (p-medias/GRASP). En la cuarta se muestra el valor objetivo una vez que fue aplicada la fase de mejora (VNS). La siguiente columna muestra el porcentaje de mejora una vez que se aplicó la VNS así como los correspondientes tiempos de cómputo (medido en segundos) tanto para cada fase como para el método completo. Como puede observarse el tiempo de cómputo (en todos los casos) empleado para resolver la instancia preprocesada es mucho menor a resolver la instancia real bajo las mismas condiciones. Por ejemplo resolver la instancia real para el caso (e) requiere de 1523.2 segundos (25 minutos) mientras que para la instancia preprocesada solo requiere de 169.17 segundos (3 minutos).

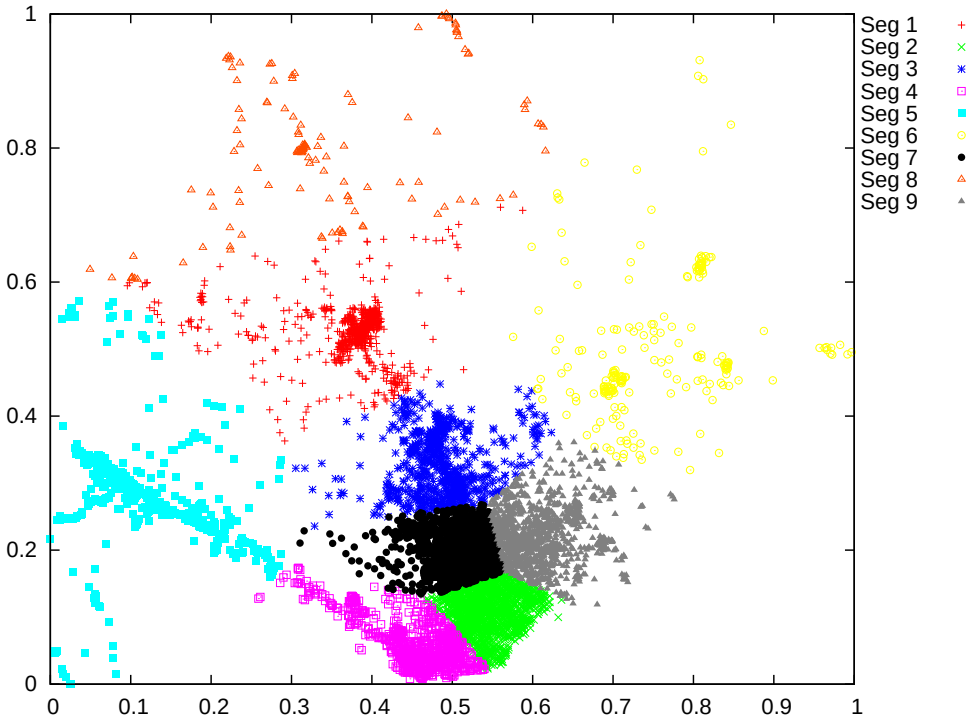
Por otro lado, las Figuras 5.25 a 5.26 muestran las particiones obtenidas para el caso (b). Se analizaron los casos en donde los parámetros de ponderacíon de los atributos de contrato y establecimiento tienen su peso máximo ($\alpha_3 = 1$ y $\alpha_4 = 1$). El algoritmo p-medias agrupa la mayor cantidad de clientes en determinados grupos obteniendo una mala calidad de la solución. Sin embargo, al aplicar la VNS la solución mejora notablemente e incluso los grupos formados ya no son grupos con un solo elemento. La Figura 5.27 muestra la partición del conjunto de clientes de la muestra real usando la asignación obtenida por la VNS en la instancia preprocesada.

En la figura se pueden distinguir algunos grupos no tan dispersos pero tambien existen algunos grupos en los que la asignación no resultó buena.

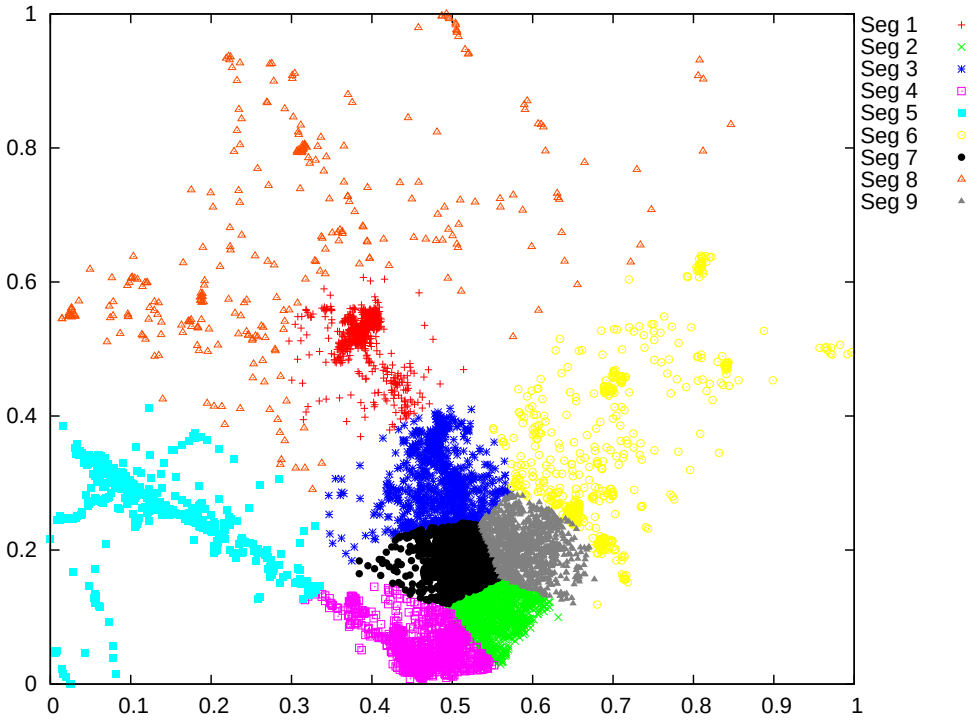
Conclusión: Primeramente se pudo observar que el algoritmo p-medias obtiene soluciones de muy mala calidad cuando solo se consideran atributos comerciales como tipo de contrato y establecimiento. Sin embargo, la aplicación de la VNS mejora mucho dicha solución. En cuanto las ventajas y desventajas del preproceso, se observó que las principales ventajas del preproceso es la gran reducción del conjunto real de clientes y por consecuencia el tiempo de cómputo para resolverlo es menor. El inconveniente es que el agrupamiento que parece ser de buena calidad para la instancia preprocesada no lo es demasiado para la real.

α_r	n	p	f(GRASP)	f(VNS)	Mejora(%)	t_{GRASP}	t_{VNS}	t_{Total}
a	7203	9	0.03895	0.03211	21.30	178.92	184.94	363.86
	17332	9	0.03247	0.02557	27.00	1377.17	930.20	2307.37
b	7203	9	0.00693	0.00606	14.33	627.88	105.11	732.99
	17332	9	0.00955	0.00559	70.85	1485.47	856.9	2342.37
c	7203	9	0.02795	0.02245	24.55	269.43	163.76	433.19
	17332	9	0.03084	0.02418	27.53	1660.16	845.12	2505.28
d	7203	9	0.00852	0.00023	3686.10	125.38	65.88	191.26
	17332	9	0.01240	0.00185	568.80	1471.23	637.98	2109.21
e	7203	9	0.08564	0.02973	188.05	110.38	58.79	169.17
	17332	9	0.07820	0.07820	0	1270.28	252.92	1523.2

Tabla 5.10: Resultados obtenidos al aplicar el método propuesto a la instancia real y preprocesada.

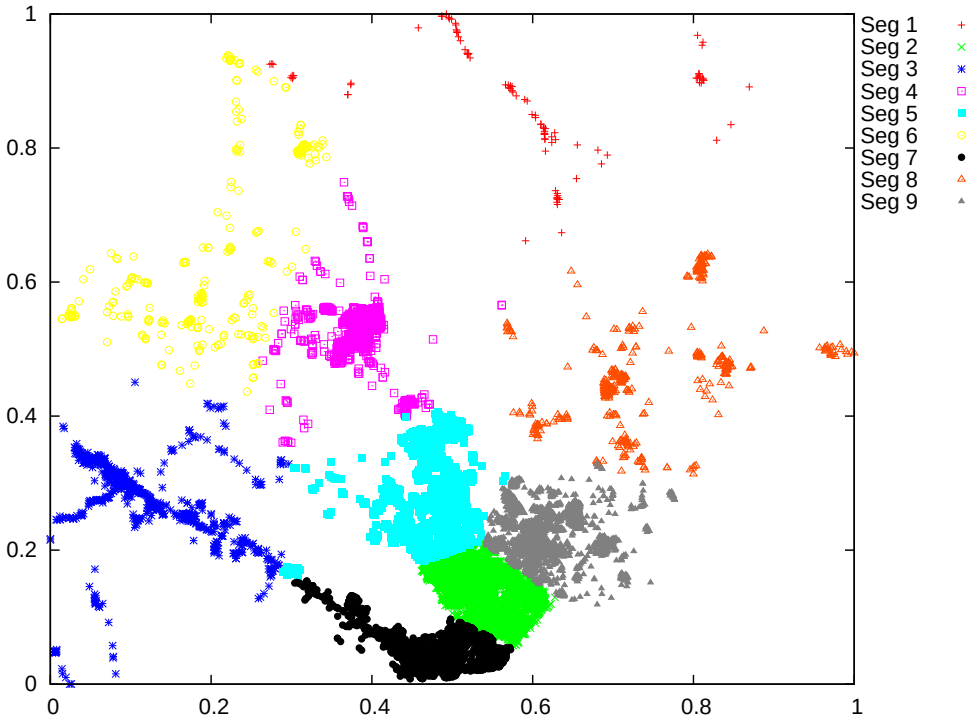


(a) Fase de construcción.

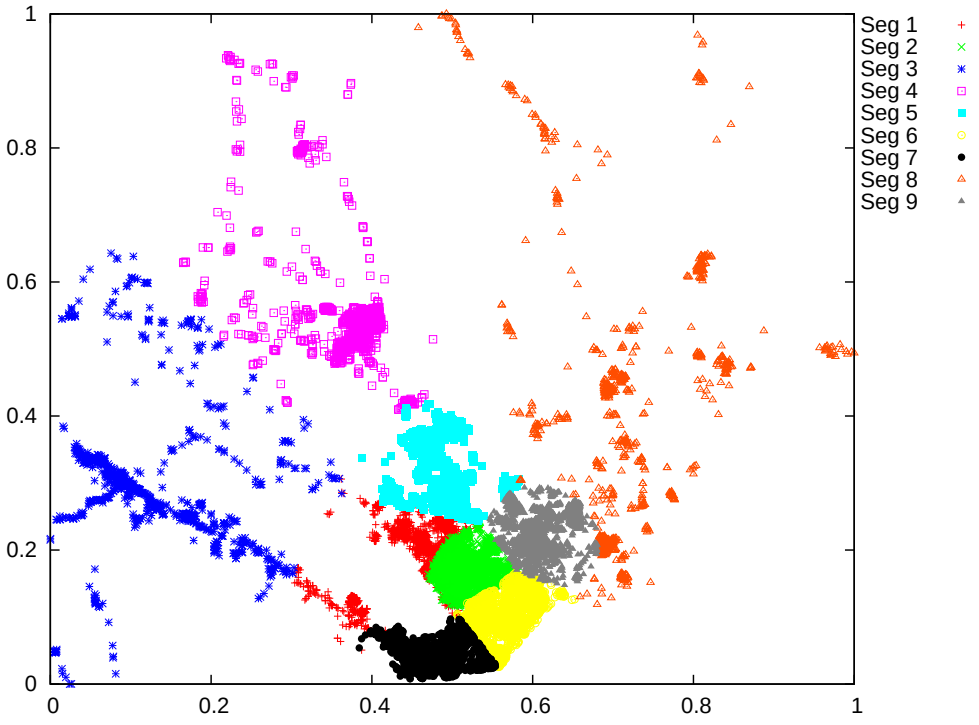


(b) Fase de mejora.

Figura 5.25: Particiones finales encontradas, en sus respectivas fases, al aplicar el método propuesto a la instancia preprocesada. Caso $\alpha_1 = 1$.



(a) Fase de construcción.



(b) Fase de mejora.

Figura 5.26: Particiones finales encontradas, en sus respectivas fases, al aplicar el método propuesto a la instancia real. Caso $\alpha_1 = 1$.

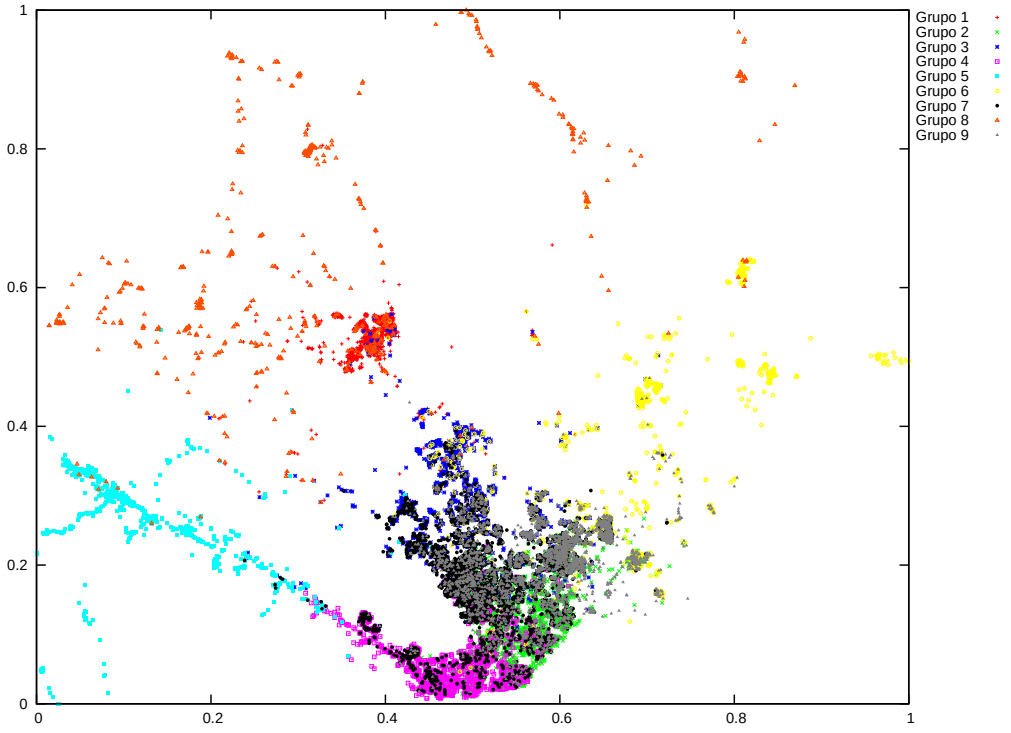


Figura 5.27: Partición obtenida de la muestra real tomando como referencia la asignación final al aplicar el método a la instancia preprocesada. Caso $\alpha_1 = 1$.

Capítulo 6

Conclusiones y Consideraciones

6.1 Conclusiones

En este trabajo de tesis se trató un caso real de segmentación de clientes de una empresa distribuidora de productos de la ciudad de Monterrey, N.L., México. Este problema de segmentación se basó en cuatro atributos principales considerados por la empresa como importantes a la hora de segmentar un determinado conjunto de clientes. Dichos atributos son la ubicación geográfica de los clientes, sus volúmenes de compra y el tipo de contrato y establecimiento de cada uno de ellos.

Dado que éste es un problema que no había sido estudiado formalmente previamente por la empresa, se comenzó analizándolo para posteriormente representarlo con un modelo matemático. Este modelo fue creado como un modelo de tipo combinatorio y cuyos datos son de tipo determinista (su valor se conoce con precisión). Una vez obtenido el modelo se estudió la instancia real obtenida de la empresa para buscar la manera de aprovechar su estructura y crear un procedimiento de preprocesamiento para poder reducir la dimensión del tamaño de la instancia con el objetivo de aprovechar mejor los recursos computacionales con los que se contaba.

Durante el análisis de la literatura se observó que el problema de segmentación, también llamado de agrupamiento, es un problema de gran interés en una gran variedad de áreas debido a la gran cantidad de estudios que han sido realizados para tratar de resolver un problema de esta índole. Como consecuencia de ello se han

creado numerosos métodos con la finalidad de resolver un determinado problema con el objetivo de obtener resultados de buena calidad y cuyo tiempo de cómputo para su resolución sea lo más eficientemente posible.

Para este trabajo se hizo uso del algoritmo K-medias el cual ha sido ampliamente utilizado en el área de agrupamiento (*clustering*) por su simple implementación computacional y rapidez para encontrar una solución al problema. Éste fué adaptado al problema tratado en esta tesis de manera que se pudieran obtener soluciones relacionadas al mismo considerando como la distancia entre un cliente y otro la suma ponderada de las disimilitudes de cada uno de los atributos considerados para el problema y considerando como centros a elementos del conjunto de clientes los cuales están representados por aquellos p clientes cuya distancia hacia los demás clientes de su segmento sea la mínima. Dado que el algoritmo K-medias tiene algunas desventajas que influyen en la calidad de la solución final se desarrolló un método basado en una heuristica constructiva utilizada en problemas de p-dispersión a la cual se adaptó a la filosofía de la metaheurística GRASP para poder dar diversidad de soluciones las cuales se pretenden sean de mejor calidad. Como un forma de poder mejorar aún más la solución se desarrolló también un método de búsqueda local basado en una búsqueda de entornos variables (VNS) compuesto por dos búsquedas locales simples, la primera de ellas consiste en insertar un cliente de un segmento a otro y la segunda consiste en intercambiar dos clientes pertenecientes a segmentos distintos cada uno.

Durante la experimentación se observó que el algoritmo K-medias efectivamente es un algoritmo que requiere poco tiempo para obtener soluciones. Dado que los centros no se seleccionan de forma determinista, la solución en cada nueva ejecución del algoritmo es diferente pudiendo ser buena o mala con respecto a otras. Por ello, se realizó un primer experimento para visualizar la calidad de la solución al repetir el algoritmo un determinado número de veces y determinar un número de repeticiones para el cual se puede garantizar soluciones de mejor calidad que si se ejecutara una sola vez así como evitar hacer uso excesivo de los recursos computacionales y del

tiempo de cómputo al repetir el algoritmo un número muy elevado de veces.

Una vez determinado el número de repeticiones necesarias para mejorar la solución obtenida por el K-medias se evaluó el número de segmentos para el cual la partición obtenida fué de mejor calidad mediante el uso del índice de Davies-Bouldin. Cabe recalcar que este experimento se realizó con el objetivo de determinar que número de segmentos utilizar y evitar tratar con valores escogidos totalmente al azar en los experimentos posteriores. En la práctica esto no puede ser totalmente válido puesto que puede darse el caso en el que la empresa cuente con una determinada forma de hacerlo (aunque en el caso tratado en esta tesis se conoce hasta el momento que no cuentan con algo parecido). Así que el cálculo de este índice puede ser una herramienta útil para los casos en los que determinar un número fijo de segmentos no pueda ser una decisión subjetiva.

Para aprovechar aún más la eficiencia del algoritmo K-medias se introdujo una heurística voraz el cual se adaptó a una filosofía GRASP para la selección de los centroides iniciales los cuales repercuten en gran medida a la solución final obtenida por el algortimo. Los resultados del experimento mostraron que una selección más sistemática obtiene mejores soluciones con más frecuencia que una selección completamente aleatoria.

Se desarrolló una búsqueda de entornos variables compuesta por dos heurísticas simples. La primera de ellas consiste en insertar un cliente de un segmento a otro. La segunda está basada en el intercambio de dos clientes pertenecientes a segmentos distintos. Esta VNS se aplicó a la solución obtenida por el K-medias mejorando dicha solución hasta en un 41 %.

El método propuesto se aplicó a distintos casos en los cuales la importancia de los atributos variaba. Esto con el fin de observar el comportamiento de la solución por dicha variación. Se realizó un experimento similiar a lo que comúnmente se hace en la empresa para determinar que peso darle a los atributos.

Se realizó un último experimento para ver las ventajas y desventajas de la

fase de preprocesamiento. Para ello, se preprocesó la instancia real obteniéndose una reducción de más del 58 %, es decir, de 17332 clientes de la muestra real se obtuvo una reducción a 7203 clientes. El método propuesto se aplicó a ambas instancias para comprar las soluciones resultantes y el tiempo de cómputo empleado para su obtención.

6.2 Contribuciones

Dado que es un problema nuevo que enfrenta la empresa, ésta no cuenta con un método adaptado a su problema en particular. El problema fué analizado y entendendio obteniéndose, como una primer contribución, la formalización del problema mediante la representación de un modelo matemático que considera los cuatro atributos contemplados por la empresa al momento de segmentar un determinado conjunto de sus clientes.

Partiendo de este modelo se desarrolló una metodología que resuelve el problema según las características requeridas. Dicha metodología está compuesta por algunos métodos conocidos por su fácil desarrollo e implementación computacional y algunos de de optimización metaheurística como VNS. La integración de éstos resulta en un método muy eficaz a el problema en cuestión. Por otro lado, se desarrolló un método de preprocesamiento para poder tratar instancias de gran tamaño de manera eficiente.

6.3 Trabajo a Futuro

A continuación se muestra una lista sobre algunas recomendaciones sobre el trabajo desarrollado en esta tesis:

Validez estadística: Dado que uno de los alcances de este trabajo fué aplicar la métodología a un caso estudio, algo recomendable es hacer un estudio empírico

más profundo considerando más instancias del problema para dar una mayor validez estadística a la métodología propuesta.

Mejora de la fase de preproceso: Como se pudo observar en el último experimento del Capítulo 5, esta fase redujo en gran cantidad el tamaño de la instancia real y los tiempos de cómputo al aplicar el método propuesto fué menor. Sin embargo al comparar las soluciones, usando una misma asignación, algunos grupos resultaron dispersos. Para la creación de metaclientes se selecciona el primer cliente cuyo coeficiente de correlación entre él y todos los demás que ya han sido seleccionados sea al menos γ. Debido a esto puede darse el caso que dicha selección no sea tan buena una vez que se hayan creado los metaclientes. Una recomendación es hacer una selecciíon más detallada de los clientes a ser agrupados.

Mejora del GRASP propuesto: El GRASP que se propone en este trabajo de tesis se basa solamente en la dispersión geográfica para obtener los centroides iniciales en su función voraz. Así que se puede cambiar dicho criterio usando la función objetivo ponderada para evaluar la distancia (en este caso disimilitud) entre los clientes tomando en cuenta los cuatros atributos y no solamente uno. Además con ello si se fuera a agrupar con respecto a un solo atributo este tambien sería el criterio para seleccionar los centroides iniciales. Otra mejora sería desarrollar un criterio de distancia para los atributos numéricos y otra para los categóricos y aplicar uno u otro, o bien, ambos cuando sea necesario.

Comparación con otros métodos: Ya que la metodología no ha sido comparada con otros métodos sería bueno considerar la adapatación de otras heurísticas y comparar las soluciones obtenidas por ambos métodos para determinar las ventajas y desventajas que resulten de dicha comparación. Incluso la misma metodología desarrollada para esta tesis puede ser tratada usando otros algortimos como, por ejemplo, aquéllos que se basan en la contrucción de un árbol de mínima expansión para construir particiones iniciales en lugar del K-medias.

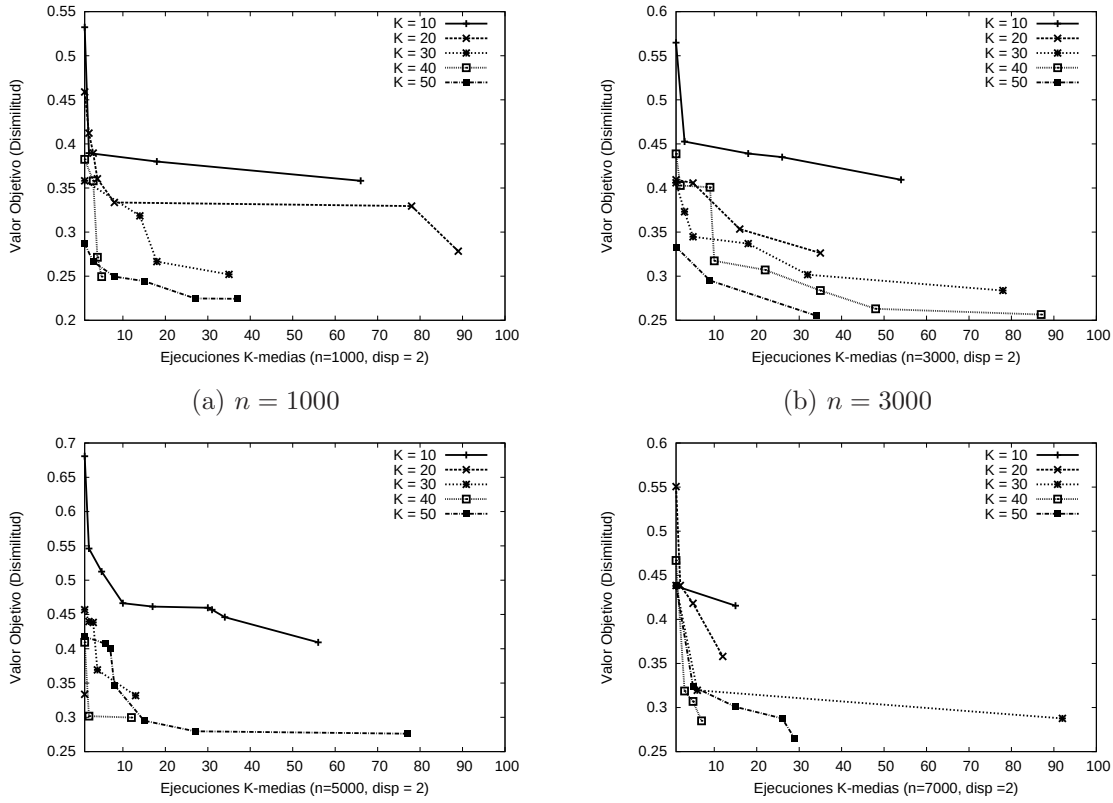
Mejora del método propuesto: Aunque el método cuenta con una fase de mejora de soluciones, basada en una búsqueda de entornos variables, se pueden encontrar aún más mejoras introduciendo el método propuesto a un esquema de Búsqueda Local Iterativa (ILS, por sus siglas en inglés). En este punto se está trabajando actualmente para el cual se ha implementado un algoritmo propuesto por Ruiz y Stützle [ref] el cual ha obtenido muy buenos resultados para problemas de secuenciación de tareas en una línea de flujo.

Mejora del método propuesto: En esta tesis, se aborda el problema como un modelo mono objetivo. Sin embargo, el problema es en realidad multi objetivo con cuatro criterios a optimizar simultáneamente. Por lo tanto, una extensión natural del trabajo es la de estudiarlo desde la perspectiva de la optimización multi objetivo, lo cual implica desde luego una panorámica de solución diferente.

Apéndice A

Experimento A: Casos Extremos

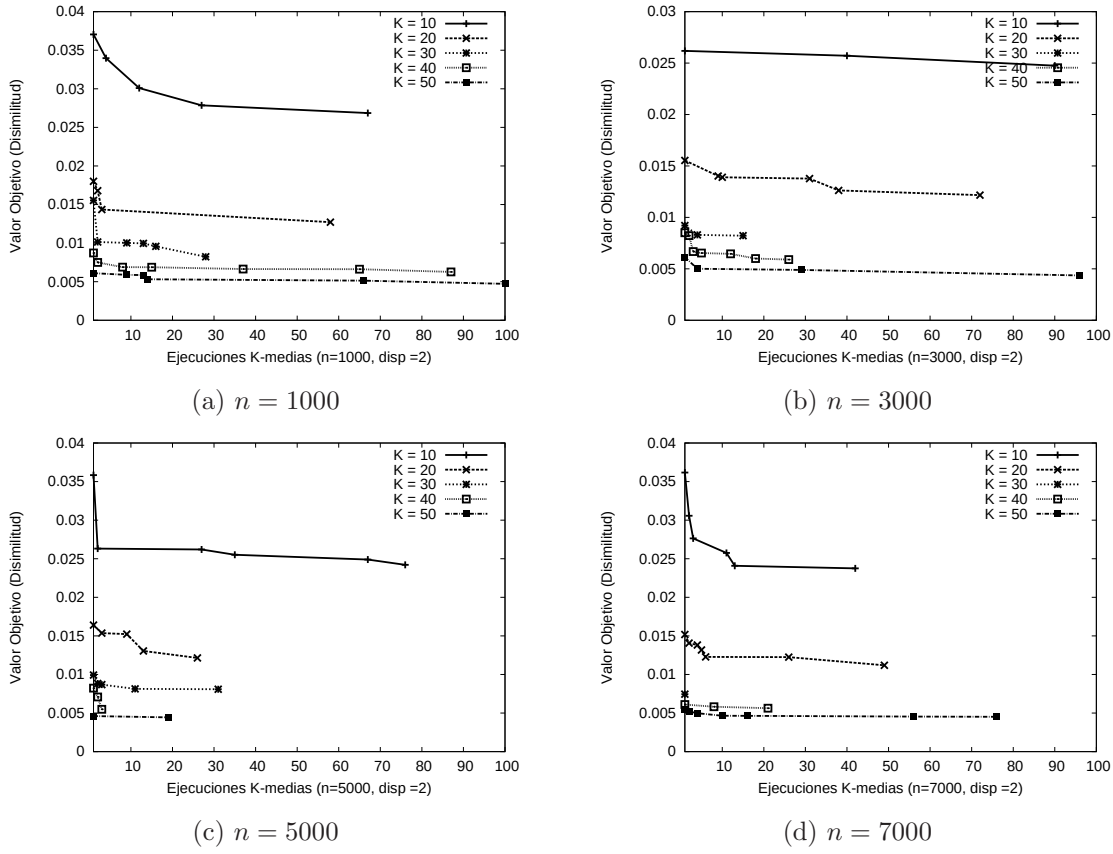
En este apéndice se muestran las figuras obtenidas al aplicar el Experimento A a los casos en los cuales el valor de los parámetros de ponderación de la función objetivo es el máximo. Esto con el fin de mostrar la convergencia hacia una determinada solución después de aplicar 100 repeticiones del algoritmo p-medias.



(a) $n = 1000$ (b) $n = 3000$

(c) $n = 5000$ (d) $n = 7000$

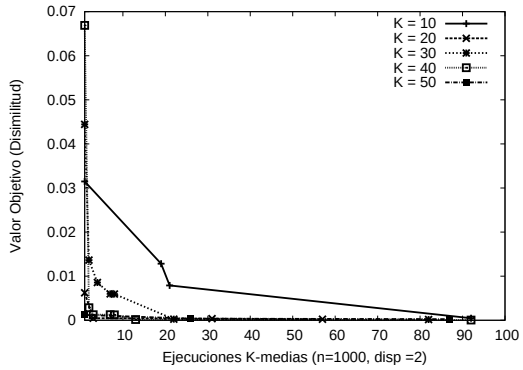
Figura A.1: Evaluación de la mejora de la partición (menor disimilitud) al aplicar 100 repeticiones del algoritmo p-medias utilizando la función (3.6) para medir la dispersión de la partición. Caso $\alpha_1 = 1$.



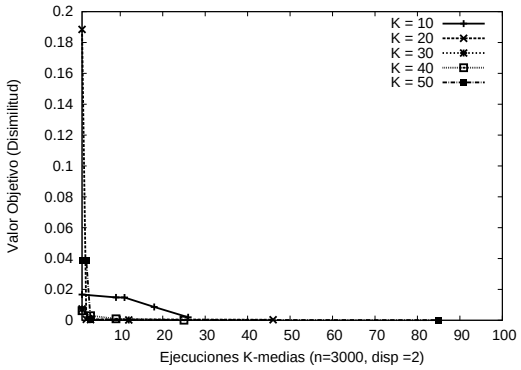
(a) $n = 1000$ (b) $n = 3000$

(c) $n = 5000$ (d) $n = 7000$

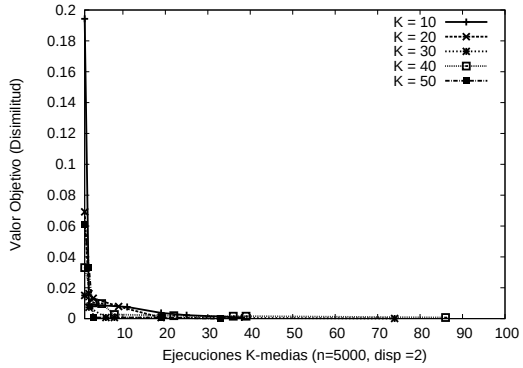
Figura A.2: Evaluación de la mejora de la partición (menor disimilitud) al aplicar 100 repeticiones del algoritmo p-medias utilizando la función (3.6) para medir la dispersión de la partición. Caso $\alpha_2 = 1$.



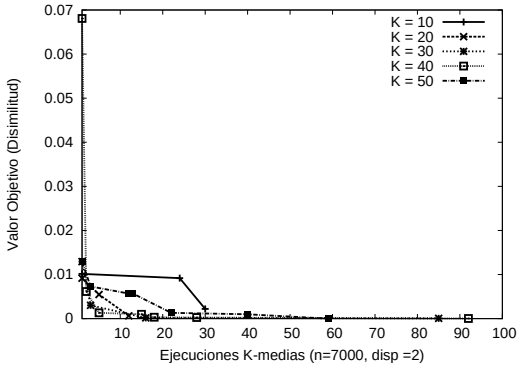
(a) $n = 1000$



(b) $n = 3000$

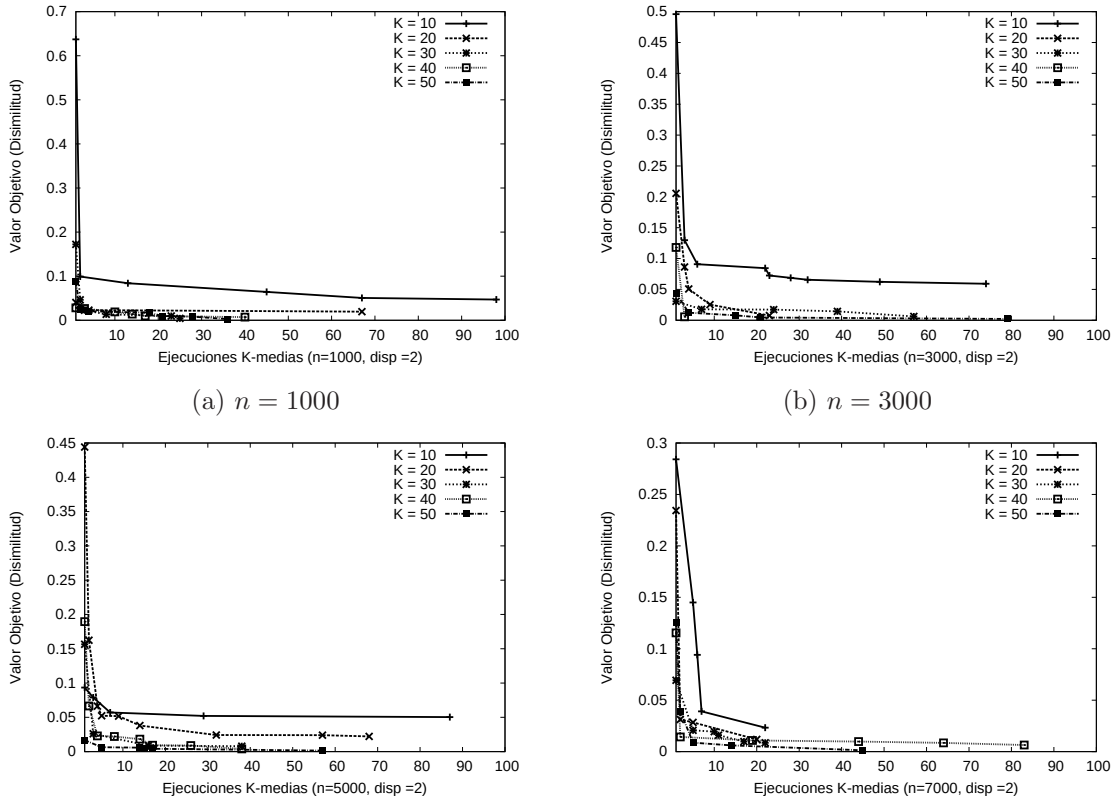


(c) $n = 5000$



(d) $n = 7000$

Figura A.3: Evaluación de la mejora de la partición (menor disimilitud) al aplicar 100 repeticiones del algoritmo p-medias utilizando la función (3.6) para medir la dispersión de la partición. Caso $\alpha_3 = 1$.



(a) $n = 1000$ (b) $n = 3000$

(c) $n = 5000$ (d) $n = 7000$

Figura A.4: Evaluación de la mejora de la partición (menor disimilitud) al aplicar 100 repeticiones del algoritmo p-medias utilizando la función (3.6) para medir la dispersión de la partición. Caso $\alpha_4 = 1$.

Apéndice B

Significancia del Coeficiente de Correlación

Coeficiente de correlación lineal de Pearson

El coeficiente de correlación de Pearson es un índice que mide el grado de covariación entre distintas variables relacionadas linealmente. Decimos *variables relacionadas linealmente* ya que pueden haber variables fuertemente relacionadas, pero no de forma lineal, en cuyo caso no se procede a aplicarse la correlación de Pearson. El coeficiente de correlación de Pearson toma valores entre 1 y -1. Un valor de 1 indica relación lineal perfecta positiva; un valor de 1 indica relación lineal perfecta negativa (en ambos casos los puntos se encuentran dispuestos en una línea recta); un valor de 0 indica relación lineal nula.

Significancia del coeficiente de correlación

Una vez calculado el valor del coeficiente de correlación interesa determinar si tal valor muestra que los clientes (variables) están relacionados en realidad o solo presentan dicha relación como consecuencia del azar. Es decir, si el valor del coeficiente de correlación obtenido tiene significancia estadística.

Un coeficiente de correlación se dice que es significativo si se puede afirmar, con una cierta probabilidad, que es diferente de cero (que no existe independencia). Para ello planteamos dos hipótesis posibles:

Hipótesis Nula: Los vectores correspondientes a los clientes son independientes, es decir no existe correlación entre ellos.

Hipótesis Alternativa: Los vectores correspondientes a los clientes no son independientes, es decir, existe correlación entre ellos.

Donde la hiótesis nula se rechaza con un nivel de confianza del 95 % si el coeficiente de correlación obtenido r cumple el siguiente criterio (para ello se hace uso del estadístico t el cual se distribuye según el modelo de probabilidad t *de Student* con $n-2$ grados de libertad):

$$r > 1.96/\sqrt{N},$$

donde N es le número de datos de cada cliente (N = 36) y r el valor del coeficiente de correlación seleccionado para la creación de metaclientes ($r = 0.95$). Entonces, sustituyendo en la expresión anterior, obtenemos el siguiente resultado:

$$0.95 > 1.96/\sqrt{36};$$
$$0.95 > 1.96/6;$$
$$0.95 > 0.32.$$

Por lo tanto, se rechaza la hipótesis nula con un nivel de confianza del 95 %. Es decir, se rechaza que no hay correlación entre los clientes a es nivel de confianza.

Bibliografía

[1] R. Agrawal, J. Gehrke, D. Gunopulos y P. Raghavan. Automatic subspace clustering of high dimensional data for data mining applications. *SIGMOD Record*, 27(2):94–105, 1998.

[2] M. F. Amasyali y O. Ersoy. Clusline: A new clustering algorithm. En *Proceedings of the International Symposium on Innovations in Intelligent Systems and Applications*, Estambul, Turquía, 2005.

[3] N. Archip, R. Rohling, P. Cooperberg, H. Tahmasebpour y S. K. Warfield. Spectral clustering algorithms for ultrasound image segmentation. En *Medical Image Computing and Computer-Assisted Intervention – MICCAI*, páginas 862–869. Springer, Berlín, Alemania, 2005.

[4] T. Bäck, F. Hoffmeister y H. P. Schwefel. A survey of evolution strategies. En R. K. Belew y L. B. Booker, editores, *Proceedings of the Fourth International Conference on Genetic Algorithms*, páginas 2–9. San Francisco, E.U.A., 1991.

[5] E. Ballestero. *Estudios de mercado: Una introducción a la mercadotecnia*. Alianza, Madrid, España, 1990.

[6] M. J. Barroso González y F. J. Alonso Sánchez. *Diccionario del marketing*. Paraninfo, Madrid, España, 1993.

[7] N. Belacel, M. Čuperlović-Culf, M. Laflamme y R. Ouellette. Fuzzy j-means and VNS methods for clustering genes from microarray data. *Bioinformatics*, 20(11):1690–1701, 2004.

[8] P. Berkhin. Survey of clustering data mining techniques. En J. Kogan, C. Nicholas y M. Teboulle, editores, *Grouping Multidimensional Data: Recent Advances in Clustering*, páginas 25–71. Springer, Berlín, Alemania, 2006.

[9] E. A. Blackstone, A. J. Buck, S. Hakim y U. Spiegel. Market segmentation in child adoption. *International Review of Law and Economics*, 28(3):220–225, 2008.

[10] N. Bolshakova, F. Azuaje y P. Cunningham. Incorporating biological domain knowledge into cluster validity assessment. En *Applications of Evolutionary Computing*, volumen 3907 de *Lecture Notes in Computer Science*, páginas 13–22. Springer, Berlín, Alemania, 2006.

[11] K. Boryczko y M. Kurdziel. Approximate clustering of noisy biomedical data. En *Computational Science – ICCS 2008*, volumen 5101 de *Lecture Notes in Computer Science*, páginas 630–640. Springer, Berlín, Alemania, 2008.

[12] J. T. Bowen. Market segmentation in hospitality research: No longer a sequential process. *International Journal of Contemporary Hospitality Management*, 10(7):289–296, 1998.

[13] M. J. Brusco y S. Stahl. *Branch-and-Bound Applications in Combinatorial Analysis*. Springer, Nueva York, E.U.A., 2005.

[14] R. Caballero, M. Laguna, R. Martí y J. Molina. Multiobjective clustering with metaheuristic optmization technology. Reporte técnico, Departamento de Estadística e Investigación Operativa, Universidad de Valencia, Valencia, España, 2006.

[15] J. R. Cano, O. Cordón, F. Herrera y L. Sánchez. A GRASP algorithm for clustering. En F. J. Garijo, J. C. Riquelme y M. Toro, editores, *Advances in Artificial Intelligence – IBERAMIA 2002*, volumen 2527 de *Lecture Notes in Computer Science*, páginas 214–223. Springer, Berlín, Alemania, 2002.

[16] H. Chernoff. Cluster analysis for applications. *SIAM Review*, 17(3):580–582, 1975.

[17] S. A. Cook. The complexity of theorem-proving procedures. En *Proceedings of the Third Annual ACM Symposium on Theory of Computing*, páginas 151–158. ACM, Nueva York, E.U.A, 1971.

[18] R. Decker, S. W. Scholz y R. Wagner. Growing clustering algorithms in market segmentation: Defining target groups and related marketing communication. En S. Zani, A. Cerioli, M. Riani y M. Vichi, editores, *Data Analysis, Classification and the Forward Search*, páginas 23–30. Springer, Berlín, Alemania, 2006.

[19] A. Duarte Muñoz, J. J. Pantrigo Fernández y M. Gallego Carrillo. *Metaheurísticas*. Dykinson, Madrid, España, 2007.

[20] E. Erkut. The discrete p-dispersion problem. *European Journal of Operational Research*, 46(1):48–60, 1990.

[21] E. Erkut, Y. Ürküsal y O. Yenycerioğlu. A comparison of p-dispersion heuristics. *Computers and Operation Research*, 21(10):1103–1113, 1994.

[22] L. Ertöz, M. Steinbach y V. Kumar. Finding clusters of different sizes, shapes, and densities in noisy, high dimensional data. En *Proceedings of the 2nd SIAM International Conference on Data Mining*, páginas 47–58, San Francisco, E.U.A., 2003.

[23] M. Ester, H. P. Kriegel, J. Sander y X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. En *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining*, páginas 226–231, Portland, E.U.A., 1996.

[24] T. A. Feo y M. G. C. Resende. Greedy randomized adaptive search. *Journal of Global Optimization*, 6(2):109–133, 1995.

[25] X. Z. Fern y C. E. Brodley. Solving cluster ensemble problems by bipartite graph partitioning. En *Proceedings of the 21st International Conference on Machine Learning*. ACM, Nueva York, E.U.A., 2004.

[26] P. Fränti y J. Kivijärvi. Randomized local search algorithm for the clustering problem. *Pattern Analysis and Aplications*, 3(4):358–369, 2000.

[27] M. García Torres, B. Melián Batista, J. A. Moreno Pérez, J. M. Moreno Vega y R. Rivero Martín. Búsquedas dispersa y de entorno variable en minería de datos. En *Actas del 3er Taller Nacional de Minería de Datos y Aprendizaje, TAMIDA*, páginas 309–316, Granada, España, 2005.

[28] M. R. Garey y D. S. Johnson. *Computers and Intractibility: A Guide of the Theory of NP-completeness*. W. H. Freeman & Company, Nueva York, E.U.A., 1979.

[29] F. Glover. Future paths for integer programming and links to artificial intelligence. *Computers and Operations Research*, 13(5):533–549, 1986.

[30] D. E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, Boston, E.U.A., 1989.

[31] S. M. Golsefid, M. Ghazanfari y S. Alizadeh. Customer segmentation in foreign trade based on clustering algorithms case study: Trade promotion organization of iran. *International Journal of Computer, Information, and Systems Science, and Engineering*, 1(3):175–181, 2007.

[32] S. Guha, R. Rastogi y K. Shim. ROCK: A robust clustering algorithm for categorical attributes. *Information Systems*, 25(5):345–366, 2000.

[33] S. Günter y H. Bunke. Validation indices for graph clustering. *Pattern Recognition Letters*, 24(8):1107–1113, 2003.

[34] G. K. Gupta. *Introduction to Data Mining with Case Studies*. Prentice-Hall of India, India, 2006.

[35] P. Hansen, N. Mladenovic y J. A. Moreno Pérez. Variable neighborhood search. *Revista Iberoamericana de Inteligencia Artificial*, 19:77–92, 2003.

[36] J. A. Hartigan y A. Wong. A k-means clustering algorithm. *Journal of the Royal Statistical Society, Series C: Applied Statistics*, 28(1):100–108, 1979.

[37] E. Hartuv y R. Shamir. A clustering algorithm based on graph connectivity. *Information Processing Letters*, 76(4–6):175–181, 1999.

[38] T. C. Havens, J. C. Bezdek, J. M. Keller y M. Popescu. Dunn's cluster validity index as a contrast measure of VAT images. En *Proceedings of the19th International Conference on Pattern Recognition (ICPR)*, páginas 1–4, Tampa, E.U.A., 2008.

[39] Z. He, S. Deng y X. Xu. Improving k-modes algorithm considering frecuencies of attribute values in mode. En *Computational Intelligence and Security*, volumen 3801 de *Lecture Notes in Computer Science*, páginas 157–162. Springer, Berlín, Alemania, 2005.

[40] E. Hernández Valadez. *Algoritmo de Clustering Basado en Entropía para Descubrir Grupos en Atributos de Tipo Mixto*. Tesis de Maestría, Centro de Investigación y de Estudios Avanzados, Instituto Politécnico Nacional, México, D.F., 2006.

[41] Z. Huang. A fast clustering algorithm to cluster very large categorical data sets in data mining. En *Proceedings of SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery*, páginas 1–8, Tucson, E.U.A., 1997.

[42] Z. Huang. Extensions to the k-means algorithm for clustering large data sets with mixed numeric and categorical values. *Data Mining and Knowledge Discovery*, 2(3):283–304, 1998.

[43] A. Jain, M. Murty y P. Flynn. Data clustering: A review. *ACM Computing Surveys*, 31(3):264–323, 1999.

[44] A. K. Jain, J. Mao y K. Mohiuddin. Artificial neural networks: A tutorial. *Computer*, 29(3):31–44, 1996.

[45] J. Jonker, N. Piersma y D. V. Den Poel. Joint optimization of customer segmentation and marketing policy to maximize long-term profitability. *Expert Systems with Applications*, 27(2):159–168, 2004.

[46] O. Kariv y S. L. Hakimi. An algorithmic approach to network location problems i: The p-centers. *SIAM Journal on Applied Mathematics*, 37(3):513–538, 1979.

[47] O. Kariv y S. L. Hakimi. An algorithmic approach to network location problems ii: The p-medians. *SIAM Journal on Applied Mathematics*, 37(3):539–560, 1979.

[48] T. Käster, V. Wendt y G. Sagerer. Comparing clustering methods for database categorization in image retrieval. En *Pattern Recognition*, volumen 2781 de *Lecture Notes in Computer Science*, páginas 228–235. Springer, Magdeburg, Germany, 2003.

[49] L. Kaufman y P. J. Rousseeuw. *Finding Groups in Data: An Introduction to Cluster Analysis*. Wiley-Interscience, Nueva York, E.U.A., 2005.

[50] P. Kotler. *Las preguntas más frecuentes sobre marketing (verticales de bolsillo)*. Editorial Norma, Bogotá, Colombia, 2008.

[51] P. Kotler, G. Armstrong, J. Saunders y V. Wong. *Introducción al marketing*. Prentice Hall, Madrid, España, 2a edición, 2000.

[52] M. Kurucz, A. Benczúr, K. Csalogány y L. Lukács. Spectral clustering in telephone call graphs. En *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 Workshop on Web Mining and Social Network Analysis*, páginas 82–91. ACM, Nueva York, E.U.A., 2007.

[53] R. Loganantharaj, S. Cheepala y J. Clifford. Metric for measuring the effectiveness of clustering of DNA microarray expression. *BMC Bioinformatics*, 7(2):1–15, 2006.

[54] D. A. Lupsa. Unsupervised single-link hierarchical clustering. *Studia Universitatis Babes Bolyai, Informática*, 50(2):11–22, 2005.

[55] U. Von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007.

[56] R. Martí. Procedimientos metaheurísticos en optimización combinatoria. *Matemàtiques*, 1(1):3–62, 2003.

[57] G. Milligan y M. Cooper. An examination of procedures for determining the number of clusters in a data set. *Psychometrika*, 50(2):159–179, 1985.

[58] M. Negreiros y A. Palhano. The capacitated centred clustering problem. *Computers and Operations Research*, 33(6):1639–1663, 2006.

[59] A. Y. Ng, M. I. Jordan y Y. Weiss. On spectral clustering: Analysis and an algorithm. En T. G. Dietterich, S. Becker y Z. Ghahramani, editores, *Advances in Neural Information Processing Systems 14*, páginas 849–856. The MIT Press, Cambridge, E.U.A, 2002.

[60] R. T. Ng y J. Han. CLARANS: A method for clustering objects for spatial data mining. *IEEE Transactions on Knowledge and Data Engineering*, 14(5):1003–1016, 2002.

[61] O. Ortega Lobo, E. J. Salazar Girón, C. M. Parra y A. C. Velez. A cluster validity index for comparing non-hierarquical clustering methods. En *Memorias del Encuentro de Investigación sobre Tecnologías de Información Aplicadas a la Solución de Problemas (EITI2002)*, volumen 1, páginas 320–324, Medellín, Colombia, 2002.

[62] I. H. Osman y J. P. Kelly, editores. *Meta-Heuristics: Theory and Applications*. Kluwer, Boston, E.U.A., 1996.

[63] C. Papadimitriou y K. Steiglitz. *Combinatorial Optimization: Algorithms and Complexity*. Prentice-Hall, Nueva York, E.U.A., 1998.

[64] S. Petrović y M. Milosavljević. A comparison between the silhouette index and the Davies-Bouldin index in labelling IDS clusters. En *Proceedings of the 11th Nordic Workshop on Secure IT Systems*, páginas 53–64, Linköping, Suecia, 2006.

[65] G. Polya. *How to Solve It: A New Aspect of Mathematical Method*. Princeton University Press, Princeton, E.U.A., 1971.

[66] S. Ray y R. H. Turi. Determination of number of clusters in k-means clustering and application in colour image segmentation. En *Proceedings of the 4th International Conference on Advances in Pattern Recognition and Digital Techniques*, páginas 137–143, Calcuta, India, 1999.

[67] M. E. Rodríguez-Salazar, S. Álvarez-Hernández y E. Bravo-Nuñez. *Coeficientes de asociación*. Plaza y Valdés, Madrid, España, 2001.

[68] S. Rudich y A. Wigderson, editores. *Computational Complexity Theory*, volumen 10 de *IAS/Park City Mathematics Series*. AMS, Providence, E.U.A., 2004.

[69] J. Sander, M. Ester, H. P Kriegel y X. Xu. Density-based clustering in spatial databases: The algorithm GDBSCAN and its applications. *Data Mining and Knowledge Discovery*, 2(2):169–194, 1998.

[70] M. Sarkar, B. Yegnanarayana y D. Khemani. A clustering algorithm using an evolutionary programming-based approach. *Pattern Recognition Letters*, 18(10):975–986, 1997.

[71] S. Scheuerer y R. Wendolsky. A scatter search heuristic for the capacitated clustering problem. *European Journal of Operational Research*, 169(2):533–547, 2006.

[72] G. Sheikholeslami, S. Chatterjee y A. Zhang. WaveCluster: A wavelet-based clustering approach for spatial data in very large databases. *The VLDB Journal*, 8(3-4):289–304, 2000.

[73] W. Sheng y X. Liu. A hybrid algorithm for k-medoid clustering of large data set. En *Proceedings of the 2004 Congress on Evolutionary Computation – CEC2004*, volumen 1, páginas 77–82, Portland, E.U.A., 2004.

[74] J. Shi y J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[75] W. R. Smith. Product differentiation and market segmentation as alternative marketing strategies. *The Journal of Marketing*, 21(1):3–8, 1956.

[76] N. Speer, H. Fröhlich, C. Spieth y A. Zell. Functional grouping of genes using spectral clustering and gene ontology. En *Proceedings of the International Joint Conference on Neural Networks (IJCNN 2005)*, volumen 1, páginas 298–303, Montreal, Canadá, 2005.

[77] A. Strehl y J. Ghosh. Cluster ensembles: A knowledge reuse framework for combining multiple partitions. *The Journal of Machine Learning Research*, 3:583–617, 2002.

[78] C. A. Sugar y G. M. James. Finding the number of clusters in a data set: An information theoretic approach. *Journal of the American Statistical Association*, 98(463):750–763, 2003.

[79] D. Verma y M. Meilă. A comparison of spectral clustering algorithms. Reporte Técnico TR UW-CSE-03-05-01, Department of Computer Science and Engineering, Universidad de Washington, Seattle, E.U.A., 2005.

[80] J. Vesanto y E. Alhoniemi. Clustering of the self-organizing map. *IEEE Transactions on Neural Networks*, 11(3):586–600, 2000.

[81] E. Vicente, L. Rivera y D. Mauricio. Grasp en la resolución del problema de clustering. *Revista de Investigación de Sistemas e Informática*, 2(2):16–25, 2005.

[82] G. Wang, Z. Wang, W. Chen y J. Zhuang. Classification of surface EMG signals using optimal wavelet packet method based on Davies-Bouldin criterion. *Medical and Biological Engineering and Computing*, 44(10):865–872, 2006.

[83] W. Wang, J. Yang y R. R. Muntz. STING: A statistical information grid approach to spatial data mining. En M. Jarke, M. J. Carey, K. R. Dittrich, F. H. Lochovsky, P. Loucopoulos y M. A. Jeusfeld, editores, *Proceedings of the 23rd International Conference on Very Large Data Bases (VLDB 97)*, páginas 186–195. Morgan Kaufmann, San Francisco, E.U.A., 1997.

[84] W. Xia, Z. Ping, W. Gao y L. Jia. Market segmentation based on customer satisfaction-loyalty links. *Frontiers of Business Research in China*, 1(2):211–221, 2007.

[85] X. Xu, J. Jochen y H. P. Kriegel. A fast parallel clustering algorithm for large spatial databases. *Data Mining and Knowledge Discovery*, 3(3):263–290, 1999.

[86] Y. Xu, V. Olman y D. Xu. Clustering gene expression data using a graph-theoretic approach: An application of minimum spanning trees. *Bioinformatics*, 18(4):536–545, 2002.

[87] M. Zhang, T. M. Therneau, M. A. McKenzie, P. Li y P. Yang. A fuzzy *c*-means algorithm using a correlation metrics and gene ontology. En *Proceedings of the 19th International Conference on Pattern Recognition (ICPR)*, páginas 1–4, Tampa, E.U.A., 2008.

[88] P. G. Zhang. Neural networks. En *Data Mining and Knowledge Discovery Handbook*, páginas 487–516. Springer, Nueva York, E.U.A., 2005.

Ficha autobiográfica

Diana Lucia Huerta Muñoz

Candidato para el grado de Maestro en
Ciencias en Ingeniera de Sistemas

Universidad Autónoma de Nuevo León

Facultad de Ingeniería Mecánica y Eléctrica

Tesis:

Diseño de Planes Eficientes para la Segmentación de Clientes con Múltiples Atributos

Nací el 8 de febrero de 1985 en Monterrey, Nuevo León. Soy la quinta hija del Sr. Santiago Huerta García y Ma. Ignacia Muñoz Armendáriz. Egresada de la Universidad Autónoma de Nuevo León (2002 - 2006) obteniendo el título de Ingeniero Administrador de Sistemas en mayo de 2007. Comencé mis estudios de maestría en el Programa de Posgrado en Ingeniería de Sistemas (PISIS) en enero de 2007.