

**PROTOTIPO COMPUTACIONAL PARA IDENTIFICAR PATRONES DE
COMPORTAMIENTO DE USUARIOS DE SERVICIO A DOMICILIO APLICANDO
MINERÍA DE DATOS.**



**PABLO ANDRÉS SANTAMARÍA PARRA – U00010983
ESTUDIANTE DE INGENIERÍA DE SISTEMAS**

**UNIVERSIDAD AUTÓNOMA DE BUCARAMANGA
FACULTAD DE INGENIERÍA DE SISTEMAS
SISTEMAS DE INFORMACIÓN E INGENIERÍA DEL SOFTWARE
BUCARAMANGA**

2008

**PROTOTIPO COMPUTACIONAL PARA IDENTIFICAR PATRONES DE
COMPORTAMIENTO DE USUARIOS DE SERVICIO A DOMICILIO APLICANDO
MINERÍA DE DATOS.**

**PABLO ANDRÉS SANTAMARÍA PARRA – U00010983
ESTUDIANTE DE INGENIERÍA DE SISTEMAS**

**Trabajo de Grado para Optar al Título de:
Ingeniero de Sistemas**

DIRECTOR:

MSc. JAVIER HERNANDEZ CÁCERES

ASESORES:

Mag. SANDRA CRISTINA SANGUINO GALVIS

MSc. CARLOS AUGUSTO QUINTERO DUQUE

**UNIVERSIDAD AUTÓNOMA DE BUCARAMANGA
FACULTAD DE INGENIERÍA DE SISTEMAS
SISTEMAS DE INFORMACIÓN E INGENIERÍA DEL SOFTWARE
BUCARAMANGA**

2008

Nota de Aceptación

Firma del Jurado

Firma del Jurado

Firma del Director

Bucaramanga, 30 de Mayo de 2008

LISTA DE TABLAS

	pág.
Tabla 1 Resultados Weka para el grupo de datos "Titanic".....	34
Tabla 2 Resultados Algoritmo Apriori.....	35
Tabla 3 Comparación de Varianza.....	38

LISTA DE FIGURAS

	pág.
Figura 1. Fases de la Minería de Datos.....	15
Figura 2. Taxonomía de Técnicas de Minería de Datos.....	20
Figura 3. Ejemplo de Dendograma.	24
Figura 4. Mercado de servidores a través de los dominios de Internet.....	29
Figura 5. Seudo Código Apriori.	31
Figura 6. Página de "Inicio" del Prototipo Computacional.	40
Figura 7. Página de "Quiénes Somos" del Prototipo Computacional.	41
Figura 8. Página de "Servicios" del Prototipo Computacional.....	42
Figura 9. Página de "Establecimientos" del Prototipo Computacional.	43
Figura 10. Página de "Contáctenos" del Prototipo Computacional.....	44
Figura 11. Página de "Mis Datos" del Prototipo Computacional.	46
Figura 12. Página de "Promociones" del Prototipo Computacional.	47
Figura 13. Página de "Resultado Promociones" del Prototipo Computacional.....	48
Figura 14. Página de "Clientes Frecuentes" del Prototipo Computacional.	49
Figura 15. Página de "Clientes-Productos" del Prototipo Computacional.....	50
Figura 16. Página de "Hábitos de Compra" del Prototipo Computacional.	51
Figura 17 Resultados de k-medias para dos grupos.....	54
Figura 18 Resultado de Clientes Frecuentes para la video tienda Chaplin.	56

AGRADECIMIENTOS

Quiero expresar mis más sinceros agradecimientos a todas las personas que me dieron su colaboración y orientación en el desarrollo de este trabajo, en especial al director y asesores del proyecto.

A Nathalia por su asesoría, comprensión y apoyo, así como a mis padres por haberme patrocinado y servido de soporte en todo momento, no solo durante esta etapa sino en todas las etapas de la vida.

TABLA DE CONTENIDO

	pág.
INTRODUCCIÓN	12
1. MARCO TEÓRICO	14
1.1 MINERÍA DE DATOS.....	14
1.1.1 Concepto.....	14
1.1.2 Fases de un Proyecto de Minería de Datos	15
1.1.3 Almacenes de Datos	16
1.1.4 Técnicas Multivariadas.....	19
1.1.5 Análisis de Datos	21
1.1.6 Análisis de Componentes Principales	21
1.1.7 Técnicas de Agrupación.....	22
1.1.7.1 Algoritmo K-medias.....	23
1.1.7.2 Algoritmos Jerárquicos.....	24
1.1.8 Reglas de Asociación.....	25
1.2 AMBIENTE DE DESARROLLO	26
1.2.1 PHP.....	26
1.2.2 MySQL	27
1.2.3 XHTML.....	28
1.2.4 Apache.....	28
2. DESARROLLO DEL PROTOTIPO	30
2.1 PROGRAMACIÓN DE ALGORITMOS	30
2.1.1 Pruebas de funcionamiento de los algoritmos.....	34
2.1.1.1 Algoritmo Apriori.....	34
2.1.1.2 Algoritmo de K-medias.....	35
2.1.1.3 Algoritmo de componentes principales (método de covarianza).....	37

2.2 INTERFAZ GRÁFICA	38
2.3 SÁBANA DE DATOS	51
3. RESULTADOS.....	53
3.1 RESULTADOS OBTENIDOS CON EL PROTOTIPO	53
3.2 ANÁLISIS DE RESULTADOS.....	57
4. TRABAJOS FUTUROS.....	59
5. CONCLUSIONES	61
BIBLIOGRAFÍA.....	63

LISTA DE ANEXOS

	pág.
Anexo A. Algoritmo Apriori.....	65
Anexo B. Código Componentes Principales, Método de Covarianza.....	76
Anexo C. Algoritmo de K-medias.	85
Anexo D. Código para Hallar “Clientes Frecuentes”.	90
Anexo E. Código para Hallar “Clientes Productos”.....	94
Anexo F. Resultados SPSS para el Algoritmo de K-medias.	98
Anexo G. Resultados Algoritmo de K-medias en PHP.....	104
Anexo H. Resultados Exactos Algoritmo de K-medias en PHP.....	109
Anexo I. Resultados en SPSS para Componentes Principales.....	113
Anexo J. Resultados Algoritmo de Componentes Principales en PHP.....	115
Anexo K. Mapa del Sitio Completo.....	117

RESUMEN

El prototipo computacional objeto de la presente investigación pretende analizar patrones de comportamiento de usuarios de servicio a domicilio con el objetivo de servir como herramienta a los establecimientos comerciales que deseen seleccionar los clientes a los que les envían promociones o descubrir los hábitos de compra de la población, para poder predecir su comportamiento.

Para lograr este fin es utilizada la Minería de Datos, donde se aplican diferentes técnicas multivariadas, analizando así los patrones de comportamiento de los clientes.

La primera técnica que se utiliza es el análisis de componentes principales, que permite reducir las variables a estudiar, generando nuevas variables conocidas como componentes principales. Estos componentes permiten analizar toda la información empleando menos variables. Luego se utilizan dos algoritmos de agrupación (cluster) para identificar la afinidad de los clientes con determinados productos, lo que permitirá dirigir las promociones hacia los clientes que realmente se interesarán.

Además se hace uso de la técnica de reglas de asociación para predecir patrones interesantes de compra, donde se determina la relación que existe entre la compra de un producto con la de otro.

Adicionalmente, la metodología de Data Warehouse utilizada para elaborar el almacén de datos, permite recopilar información histórica bajo un esquema que facilita la consulta de los datos para su análisis. Proporcionando así a los establecimientos una solución que les permite almacenar y analizar la información de sus clientes.

El prototipo esta desarrollado para trabajar en la web, convirtiéndose así en la base de una herramienta comercial cuyos beneficios se verán reflejados directamente en el desempeño comercial de las empresas.

Palabras clave: Algoritmos, Minería de Datos, Patrones de Comportamiento.

INTRODUCCIÓN

Debido a las dificultades de desplazamiento en la ciudad y al crecimiento de la demanda de productos acompañados de una logística de distribución y adquisición más cómoda para el consumidor de las ciudades, gran cantidad de establecimientos comerciales prestan servicio a domicilio, este, se ha convertido en una solución ideal para obtener un producto o servicio sin necesidad de desplazarse.

Considerando que en los negocios modernos lo más valioso es la información, se vuelve de vital importancia para los establecimientos comerciales almacenar todos los datos de sus clientes. Y aún más importante procesar todos estos datos conseguidos para obtener conocimiento sobre el comportamiento y las preferencias de sus clientes en el sector de domicilios.

En este sentido, los conocimientos en Ingeniería son fundamentos ya que ayudan a crear modelos con posibilidad de adaptación al sector empresarial y cuyo diseño debe adecuarse a las tendencias y necesidades latentes del sector.

Basados en investigaciones previas que se nombran como antecedentes e integrando documentos actuales, el enfoque de este proyecto se dirige al diseño de un prototipo computacional aplicando Minería de Datos, y utilizando PHP, MySQL y XHTML. Éste, finalmente permite identificar patrones de comportamiento de usuarios de servicio a domicilio dentro del área de influencia de los establecimientos vinculados.

David Hand define la minería de datos como: “El análisis de grandes cantidades de datos para encontrar relaciones inesperadas y resumir los datos en nuevo

conocimiento que es tanto entendible como útil para el dueño de los datos”.¹ Es decir, la minería de datos transforma datos en nuevo conocimiento útil.

Las técnicas de minería de datos empleadas permitirán a los establecimientos comerciales seleccionar los clientes según su afinidad con ciertas variables que representan a los productos disponibles. También permitirán seleccionar los clientes frecuentes de un establecimiento, los clientes que prefieren algún producto determinado, así como la relación que existe entre la compra de un producto con la de otro, es decir, patrones poblacionales de compra.

Las pruebas que se realizan con el prototipo utilizan datos ficticios. Esto se debe a la poca disponibilidad de datos, y teniendo en cuenta el alto valor que estos tienen, así como el costo que deben asumir los establecimientos para obtenerlos (Investigaciones de Mercado, Bases de Datos). Sin embargo, como los datos son totalmente necesarios para llevar a cabo la experimentación, se simulan dos meses de pedidos a domicilio de aproximadamente ciento cincuenta personas.

De esta manera, el proyecto en desarrollo genera un prototipo de una versión inteligente del servicio a domicilio con el análisis del comportamiento del cliente como mayor valor agregado.

¹ HAND, David, MANNILA, Heikki, SMYTH, Padhraic. Principles of Data Mining. Bradford. 2001. p.1

1. MARCO TEÓRICO

1.1 MINERÍA DE DATOS

1.1.1 Concepto “La tarea fundamental de la minería de datos es encontrar modelos inteligibles a partir de los datos”.² En otras palabras, la minería de datos se refiere a extraer conocimiento de grandes cantidades de datos.

La minería de datos enfrenta dos retos grandes; trabajar con grandes cantidades de datos, los cuales generalmente poseen problemas como ruido, datos ausentes, intratabilidad o volatilidad, ya que provienen de sistemas de información. Y emplear las técnicas multivariadas adecuadas para poder analizar los datos.

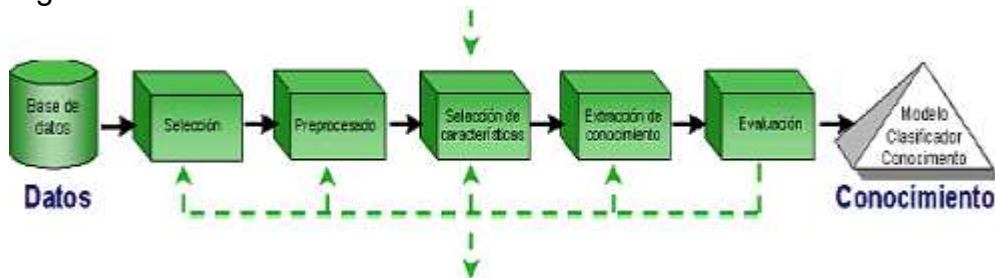
Es importante tener en cuenta que el término “Minería de Datos” es usualmente asumido como sinónimo del término KDD (Knowledge Discovery in Databases) que significa descubrimiento de conocimiento en bases de datos.³ Pero como veremos en el inciso 1.2, la Minería de Datos no es más que un simple e importante paso en el proceso de descubrimiento de conocimiento en bases de datos.

² HERNANDEZ ORALLO, José, RAMIREZ QUINTANA, Maria José, FERRI RAMÍREZ, César. Introducción a la Minería de Datos. Pearson Prentice Hall. 2004. p.3

³ HAN, Jiawei, KAMBER, Micheline. Data Mining Concepts and Techniques. Morgan Kaufmann Publishers. 2001. p.5

1.1.2 Fases de un Proyecto de Minería de Datos Sin importar la técnica específica de extracción de conocimiento elegida, el procedimiento a seguir para la realización de un proyecto de minería de datos es siempre invariable, empezando por la fase de filtrado de datos, hasta llegar a la fase de interpretación y evaluación.

Figura 1. Fases de la Minería de Datos.



Fuente: DAEDALUS - DATA, Decisions and Language, S. A. Fases de la Minería de Datos. Disponible en Internet: < <http://www.daedalus.es/AreasMDFases-E.php> > [Citada 14 de Agosto de 2007].

Fase de Filtrado de datos: Una vez recopilados los datos, éstos deben ser pasar por un proceso de filtrado, con el fin de eliminar los valores incorrectos, inválidos, desconocidos o que no sean adecuados para el algoritmo a utilizar. Esta fase es necesaria, ya que el formato de los datos contenidos en el almacén de datos nunca es ideal, y en la mayor parte de los casos no es posible aplicar ningún tipo de algoritmo de minería de datos sin antes filtrarlos.

Fase de Selección de variables: En muchas ocasiones la alta dimensionalidad (muchas variables) es un problema, y peor aún si se tienen pocas instancias. El éxito de esta fase está en seleccionar las características eligiendo las variables más influyentes en el problema, sin apenas sacrificar la calidad del modelo de conocimiento obtenido del proceso. Existen muchas técnicas para realizar este tipo de proyección: análisis de componentes principales, algunas técnicas del análisis factorial, uso de mapas auto-organizativos, etc.

Fase de Minería de Datos: Esta fase es la más diciente del proceso de KDD, aquí es donde se produce el nuevo conocimiento mediante una técnica de minería de datos. Según el problema a ser resuelto se aplica una técnica de minería de datos diferente con el objetivo de cumplir una tarea. Estas tareas pueden ser; clasificación, agrupamiento, identificación de correlaciones, reglas de asociación, etc.

Fase de Interpretación y evaluación: Una vez obtenido el modelo, se debe medir la calidad de los patrones descubiertos por el algoritmo de minería de datos elegido, se debe comprobar que las conclusiones que arroja son válidas y suficientemente satisfactorias. Cuando los patrones obtenidos corresponden a la aplicación de distintas técnicas, se deben comparar los modelos en busca de aquel que se ajuste mejor al problema. Si se da el caso donde ninguno de los modelos alcanza los resultados esperados, alguno de los pasos anteriores debe ser alterado para generar nuevos modelos.⁴

1.1.3 Almacenes de Datos “Un almacén de datos es un conjunto de datos históricos, internos o externos, y descriptivos de un contexto o área de estudio, que están integrados y organizados de tal forma que permiten aplicar eficientemente herramientas para resumir, describir y analizar los datos con el fin de ayudar en la toma de decisiones estratégicas.”⁵

⁴ DAEDALUS - DATA, Decisions and Language, S. A. Fases de la Minería de Datos. Available from Internet: <<http://www.daedalus.es/AreasMDFases-E.php>> [Cited 14 de Agosto de 2007].

⁵ HERNANDEZ ORALLO, José, RAMIREZ QUINTANA, Maria José, FERRI RAMÍREZ, César. Introducción a la Minería de Datos. Pearson Prentice Hall. 2004. p.47

Existen varios tipos de almacenes de datos, empezando por simples bases de datos relacionales, hasta llegar a almacenes más complejos como bases de datos orientadas a objetos.

Un almacén de datos tiene el propósito de recuperar información, informes, además de analizar y minar los datos. Utiliza datos históricos y descriptivos, útiles para el análisis y la sumariación. También es útil para diferentes tipos de usuarios, como directores, ejecutivos y analistas.

Una base de datos relacional es un almacén de datos, consiste en una colección de tablas, cada una con un nombre único. Cada tabla está formada por un conjunto de atributos, donde usualmente se guardan numerosas tuplas (registros). Cada tupla en la tabla representa un objeto identificado con un identificador único y descrito por un conjunto de atributos. Un modelo de Entidad-Relación (ER) es comúnmente utilizado para representar los modelos de datos por medio de entidades y relaciones. En estas bases de datos, se pueden hacer consultas por medio de un lenguaje como SQL o también a través de un asistente con interfaz gráfica de usuario.⁶

Otro ejemplo de almacén de datos es un data warehouse, que es un depósito de información recolectada de diferentes fuentes, almacenada bajo un único esquema. Para su construcción es necesario pasar por los procesos de limpieza, transformación, integración y carga de datos.

Un data warehouse es un almacén de datos que se mantiene de manera separada a una base de datos operacional en una organización. Los sistemas de data warehouse permiten la integración de varias aplicaciones y soportan el

⁶ HERNANDEZ ORALLO, José, RAMIREZ QUINTANA, Maria José, FERRI RAMÍREZ, César. Introducción a la Minería de Datos. Pearson Prentice Hall. 2004. p.9

procesamiento de información proporcionando una plataforma sólida que almacena datos históricos para su análisis.

La construcción de data warehouses incluye la limpieza e integración de datos, puesto que es un paso importante del proceso de minería de datos. Además, el data warehouse es un paso importante en el proceso de minería de datos, ya que proporciona herramientas OLAP (On-Line Analytical Processing, Procesamiento analítico en línea) para el análisis interactivo de datos, que facilitara la minería de datos.

Un data warehouse posee las siguientes cuatro características:

- Orientada a objetos: Esta organizada en torno a objetos como clientes, proveedores, productos y ventas. Concentrándose en el modelamiento y análisis de datos para la toma de decisiones. De aquí que los data warehouses proporcionen información concisa acerca de ciertos objetos, ignorando información que no sea útil para la toma de decisiones.
- Integrada: Es construida a partir de diversas fuentes, como bases de datos relacionales, archivos planos, y registros de transacciones en línea.
- Variante en el tiempo: En un data warehouse se almacena información histórica. Cada estructura clave del data warehouse contiene, ya sea explícita o implícitamente un elemento temporal. Por ejemplo se llevan registros de los últimos 10 años.
- No volátil: A pesar de ser siempre un almacén físicamente separado de datos transformados en una aplicación con un ambiente operacional, no se requieren mecanismos de procesamiento, recuperación o concurrencia.

Generalmente solo se necesitan dos operaciones para el acceso a los datos; carga inicial de datos y acceso a los datos.

En resumen, un data warehouse es un almacén de datos que sirve como soporte físico donde se almacena la información que se necesita en una empresa para la toma de decisiones estratégicas.

La principal razón por la cual es importante mantener un data warehouse separado de la base de datos operacional, es porque esto permite mantener el buen rendimiento y alto desempeño de ambos sistemas. Las consultas que se le hacen a un data warehouse son complejas, y esto podría afectar el rendimiento de las actividades operacionales.⁷

En la actualidad se tienen sistemas de información que cuentan con bases de datos operacionales que han sido adecuadas para soportar consultas complejas. Y esto ha permitido que la separación entre estos dos tipos de almacenamiento empiece a decrecer, por esta razón el almacén de datos del presente proyecto, consistirá en una base de datos operacional que permitirá hacer consultas complejas y llevar el registro de datos históricos, cumpliendo así con las características de un data warehouse.

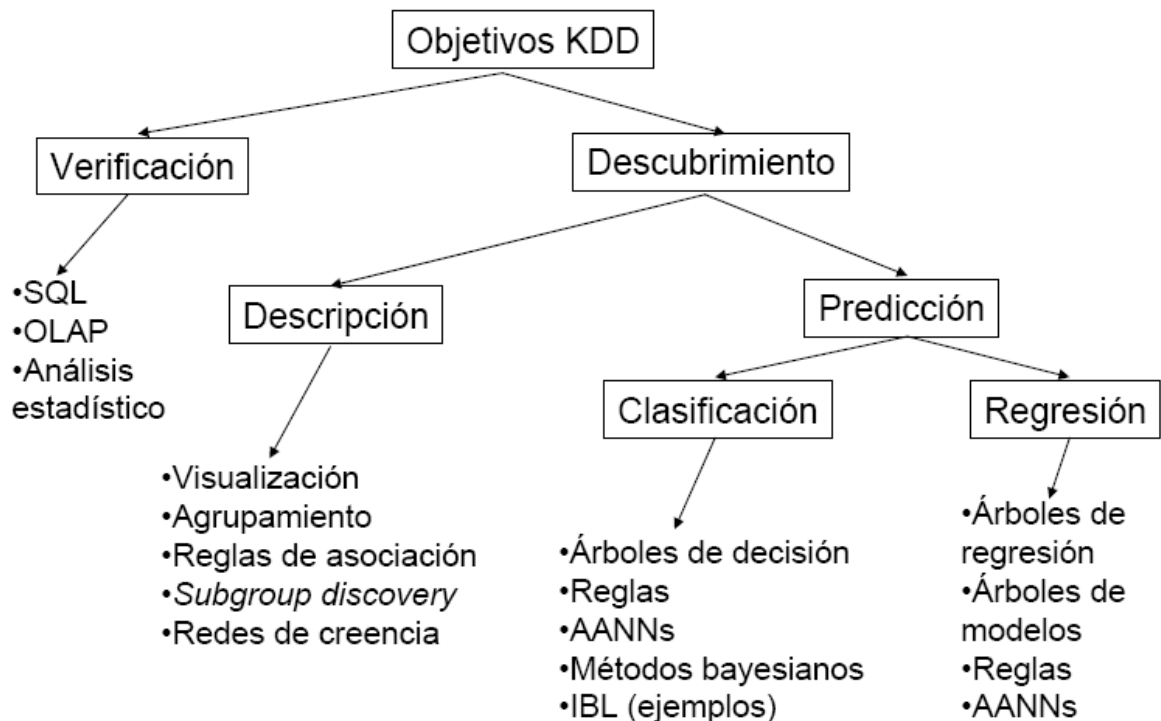
1.1.4 Técnicas Multivariadas “Existen diferentes paradigmas detrás de las técnicas utilizadas para esta fase: técnicas de inferencia estadística, árboles de decisión, redes neuronales, inducción de reglas, aprendizaje basado en instancias, algoritmos genéticos, aprendizaje bayesiano, programación lógica inductiva y varios tipos de métodos basados en núcleos, entre otros. Cada uno de estos

⁷ HAN, Jiawei, KAMBER, Micheline. Data Mining Concepts and Techniques. Morgan Kaufmann Publishers. 2001. p.39, 40 y 44

paradigmas incluye diferentes algoritmos y variaciones de los mismos, así como otro tipo de restricciones que hacen que la efectividad del algoritmo dependa del dominio de aplicación, no existiendo lo que podríamos llamar el método universal aplicable a todo tipo de aplicación”.⁸

Como se puede apreciar en la Figura 3, las técnicas multivariadas se pueden clasificar según el objetivo que cumplen. Por tal motivo las técnicas utilizadas en el presente proyecto son; reglas de asociación, análisis de componentes principales y técnicas de agrupamiento, ya que buscan la descripción y clasificación de datos.

Figura 2. Taxonomía de Técnicas de Minería de Datos.



Fuente: El Proceso de Descubrimiento de Conocimiento a Partir de Bases de Datos. Disponible en Internet: <<http://www.di.ujaen.es/asignaturas/dm/tema2.pdf>> [Citada 28 de Octubre de 2007].

⁸ HERNANDEZ ORALLO, José, RAMÍREZ QUINTANA, María José, FERRI RAMÍREZ, César. Introducción a la Minería de Datos. Pearson Prentice Hall. 2004. p.27

1.1.5 Análisis de Datos Para garantizar la aplicabilidad del análisis multivariable, es necesario realizar pruebas como mecanismos de comprobación de hipótesis.

La prueba de normalidad es una de las hipótesis más significativas. Consiste en mostrar la distribución de frecuencias de cada variable métrica individualmente. Para que cualquier prueba estadística sea válida, es necesario que este perfil no se desvíe de la distribución normal. Además, la mayor parte de las técnicas multivariadas requieren que las variables sean multivariadamente normales, es decir, que las combinaciones de las mismas también posean esta forma.

La segunda prueba que se hace sobre los datos para poder aplicar las técnicas multivariadas, denominada homoscedasticidad, se refiere a la igualdad de varianzas entre las variables independientes, sin importar si éstas son métricas o no, respecto a una variable o unas variables dependientes.

La última prueba que se lleva a cabo es la linealidad, especialmente en técnicas multivariadas basadas en medidas de asociación como las correlaciones, incluyendo la regresión múltiple, la logística, el análisis factorial, el análisis conjunto y los sistemas de ecuaciones estructurales. La prueba consiste en analizar que tan correlacionadas están dos variables entre sí. Mientras la significancia más se acerque a 0, más correlacionadas estarán.⁹

1.1.6 Análisis de Componentes Principales Consiste en una técnica muy tradicional, conocida y eficiente para reducir la dimensionalidad por transformación. Consiste en transformar los atributos o variables originales x_1, x_2, \dots, x_m en otro conjunto de atributos f_1, f_2, \dots, f_p , donde $p \leq m$ (condición necesaria para que exista reducción de variables).

⁹ Docente Javier Hernández Cáceres. Capacitación y Asesoría. Septiembre de 2007

Su eficiencia está representada en dos características importantes; la primera es que los atributos f_s generados, son totalmente independientes entre si, y la segunda es que los primeros f_s serán más significativos que los últimos. Por ejemplo, f_3 será más representativo que f_5 . Permitiendo representar los datos en la cantidad de variables de la preferencia de quien aplica la técnica.

En otras palabras, el objetivo es convertir de un vector original x_1, x_2, \dots, x_m en otro vector f_1, f_2, \dots, f_p . Siendo los primeros f_s más representativos. Y esto se logra asumiendo que la varianza deseada para los nuevos atributos es mayor para los primeros atributos que para los últimos.

El procedimiento para llevar a cabo el análisis de componentes principales empieza por normalizar los datos, para que cada atributo pertenezca al mismo rango. El siguiente paso consiste en computar los vectores ortogonales p como base de los datos normalizados. Luego los componentes principales son almacenados en orden de significancia, permitiendo que las variables más representativas sirvan como un conjunto de ejes de la nueva información, proporcionando información importante acerca de la varianza. Finalmente se eliminan los componentes más débiles, es decir, los que tienen menor varianza.¹⁰

1.1.7 Técnicas de Agrupación El proceso de agrupar un conjunto de objetos físicos o abstractos en clases, donde los objetos van a tener características similares se denomina agrupamiento (clustering), siendo ésta una de las tareas más comunes en minería de datos. Un grupo (cluster) es una colección de objetos que tienen alguna similitud, asimismo teniendo una disimilitud con los objetos de

¹⁰ HERNANDEZ ORALLO, José, RAMIREZ QUINTANA, Maria José, FERRI RAMÍREZ, César. Introducción a la Minería de Datos. Pearson Prentice Hall. 2004. p.79-81

otro grupo. De esta forma el concepto de distancia entra a jugar un rol importante ya que se tendrá en cuenta para el agrupamiento de los objetos.

Éstas técnicas han sido aplicadas en numerosas aplicaciones, donde se destacan aplicaciones de reconocimiento de patrones, análisis de datos, procesamiento de imágenes e investigación de mercados. El agrupamiento permite identificar regiones densas y dispersas, descubriendo así la distribución de patrones y relaciones interesantes entre los atributos de los objetos.¹¹

1.1.7.1 Algoritmo K-medias Es uno de los métodos de agrupamiento por partición más conocidos, junto con el método de K-medoids. El algoritmo K-medias (K-means) toma un parámetro de entrada, k , y se particionan conjuntos de n objetos en k grupos, agrupando los objetos según las distancias de separación que tengan entre sí. Lo que se busca con el algoritmo es situar a los k grupos o centros en el espacio, ubicándolos de forma que los datos pertenecientes al mismo grupo tengan características similares.

El algoritmo trabaja de manera iterativa, empezando por una selección al azar de los k objetos que inicialmente representarán los centros o medias de cada uno de los k grupos. Cada uno de los objetos restantes, son asignados al grupo cuyo centro se encuentre más cerca, es decir, con el que se tengan mayores similitudes. Una vez asignados todos los objetos a un grupo, se computan los nuevos centros para cada grupo, y se repite el proceso iterativamente hasta que la función converja.

Este algoritmo funciona bien cuando los objetos forman nubes compactas que se encuentran separadas unas de otras. Aunque hay que tener en cuenta la gran

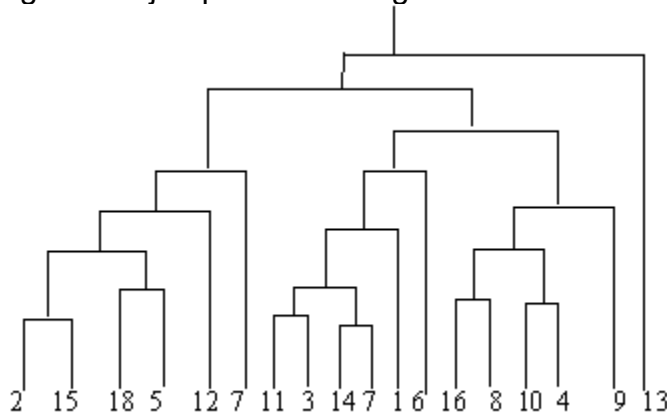
¹¹ HAN, Jiawei, KAMBER, Micheline. Data Mining Concepts and Techniques. Morgan Kaufmann Publishers. 2001. p.335

importancia que tiene la escogencia del número de grupos que se desea obtener, así como de la ubicación inicial de los centros.¹²

1.1.7.2 Algoritmos Jerárquicos Este tipo de algoritmos crean una descomposición jerárquica de los objetos almacenados. Teniendo una ventaja sobre el algoritmo de k-medias, ya que no es necesario estimar, cuántos grupos puede haber en los datos.

Los métodos jerárquicos se basan en la construcción de un árbol en el que las hojas son los elementos del conjunto de ejemplos, y el resto de los nodos son subconjuntos de ejemplos que pueden ser utilizados como particionamiento del espacio. Este gráfico se denomina dendrograma y se muestra en la Figura 4.

Figura 3. Ejemplo de Dendrograma.



Fuente: QOF Group. The Web Page Of Chemometrics - Reconhecimento de Padrões. Disponible en Internet: <<http://www.dq.fct.unl.pt/QOF/chem9.html>> [Citada 27 de Septiembre de 2007].

Los algoritmos jerárquicos pueden ser clasificados como aglomerativos y divisivos, según la forma como se crea la descomposición:

¹² HAN, Jiawei, KAMBER, Micheline. Data Mining Concepts and Techniques. Morgan Kaufmann Publishers. 2001. p.349

- **Aglomerativos.** La construcción del árbol parte desde las hojas, hasta llegar a la raíz, partiendo de tantos grupos como elementos existan, hasta finalmente converger todos los elementos en un mismo grupo llegando a la raíz.
- **Desaglomerativos o divisivos.** La construcción del árbol parte desde la raíz, partiendo de un único grupo, pero se van haciendo divisiones gradualmente hasta conseguir que cada hoja sea un grupo.¹³

1.1.8 Reglas de Asociación Las reglas de asociación son muy comunes para expresar patrones de un almacén de datos. Dichos patrones pueden ser utilizados para un mejor conocimiento del problema en cuestión, apoyando la toma de decisiones.

Como muchas reglas de asociación diferentes pueden obtenerse a partir de un pequeño conjunto de datos, se deben utilizar únicamente aquellas reglas que apliquen para un número razonablemente largo de instancias y que dichas instancias tengan una precisión suficientemente alta.

La *cobertura* de una regla de asociación es el número de instancias para las cuales la predicción ha sido correcta, esto es generalmente llamado *soporte*. Y su *precisión* llamada también *confianza* es el número de instancias en donde la predicción fue correcta, expresada como una fracción de todas las instancias para las cuales la regla de asociación aplicó.

Por ejemplo para la regla: “*Si Compra cerveza = Sí entonces Compra hielo = Si*”. La cobertura es el número de clientes que compran cerveza y hielo, mientras que la confianza es la fracción de los compradores de cerveza que compran hielo

¹³ HAND, David, MANNILA, Heikki, SMYTH, Padhraic. Principles of Data Mining. Bradford. 2001. p.308-315

también. Lo que permite a un establecimiento predecir cuantos de los clientes que compran cerveza, comprarán hielo también.

Para lograr un aprendizaje a partir de reglas de asociación, es necesario basarse en su confianza y su cobertura. Estos algoritmos buscan reglas que cumplan exigencias mínimas para estas medidas. La dificultad radica en el hallazgo las reglas de asociación que cumplan los requerimientos de confianza mínima y máxima propuestos.¹⁴

1.2 AMBIENTE DE DESARROLLO

La combinación de PHP, XHTML y MySQL permite el desarrollo de páginas web dinámicas, utilizando MySQL para la base de datos, XHTML para la parte gráfica, embebiendo el código PHP dentro de éste, para manejar la inteligencia de la aplicación. Utilizando además el servidor Apache, para mostrar los datos en función del código en el navegador.

1.2.1 PHP “PHP (acrónimo de Hypertext Preprocessor) es un lenguaje de código abierto interpretado, de alto nivel, embebido en páginas XHTML y ejecutado en el servidor”.¹⁵

PHP surge como el heredero de un producto anterior, llamado PHP/FI que fue creado por Rasmus Lerdorf en 1995, inicialmente como un simple conjunto de scripts de Perl para controlar los accesos a su trabajo online, pero PHP/FI 2.0 no se liberó oficialmente hasta Noviembre de 1997. Más adelante PHP 3.0 se convirtió en la primera versión que se parecía fielmente al PHP tal y como se

¹⁴ WITTEN, Ian H, FRANK, Eibe. Data Mining Practical Machine Learning Tools and Techniques with Java Implementations. Morgan Kaufmann Publishers, 2000. p.63

¹⁵ PHP: Introducción. Available from Internet: <URL: <http://www.php.net/manual/es/introduction.php#intro-what-is> >. [cited 5 de Septiembre de 2007].

conoce hoy en día. Después de pasar por la versión 4.0 de PHP, PHP 5.0 es liberada en Julio de 2004. La última versión estable que se puede encontrar en la página oficial de PHP (<http://www.php.net>) es la versión 5.2.4.

PHP permite la creación de páginas web dinámicas y a pesar de que existe una gran cantidad de lenguajes y ambientes de desarrollo diseñados para Internet, PHP es uno de los lenguajes de lado de servidor más ampliamente utilizados. Esto se debe a que PHP combina ciertas características, siendo un software de libre distribución y multiplataforma que sigue la filosofía de código abierto, además de su alta compatibilidad con MySQL.

1.2.2 MySQL “Otro producto de código abierto, MySQL es el generador de bases de datos que permite a PHP y a Apache trabajar de forma conjunta para acceder y mostrar datos en un formato legible en un navegador. Se trata de un servidor SQL diseñado para grandes cargas y que puede procesar consultas muy complejas”.¹⁶

MySQL permite el diseño y administración de bases de datos relacionales, donde se almacenan los datos en tablas separadas en lugar de poner todos los datos en un solo lugar, lo que genera una mayor velocidad y flexibilidad. Para permitir combinar las tablas al momento de hacer consultas, las tablas son enlazadas al definir relaciones.

MySQL posee infinidad de ventajas y cualidades, es un servidor de bases de datos muy rápido, seguro, y fácil de usar. Además adecuado para ofrecer datos a través de Internet, puesto que tiene la capacidad de procesar grandes cantidades de datos, contando con medidas de seguridad avanzadas.

¹⁶ GLASS, Michale, SCOUARNEC, Yann Le, NARAMORE, Elizabeth, MAILER, Gary, STOLZ, Jeremy, GERNER, Jason. Desarrollo Web con PHP, Apache y MySQL. Anaya Multimedia. 2004.

Si se mira a MySQL desde un punto de vista técnico se le podría definir como un sistema cliente/servidor que se compone de un servidor SQL multihilo, herramientas administrativas, varios programas clientes, y un amplio número de interfaces de programación (APIs).

1.2.3 XHTML XHTML 2 es un lenguaje diseñado para la representación de documentos de Internet. Este lenguaje surge como la evolución de HTML, pero en su construcción se tuvo en cuenta las normas de la Especificación XML 1.0, por esta razón, los documentos XHTML son realmente documentos XML, aunque nace como una reforma de HTML 4.01. A pesar de ser documentos XML, no se necesita saber especificaciones acerca de la sintaxis de XML para poder utilizarlo.

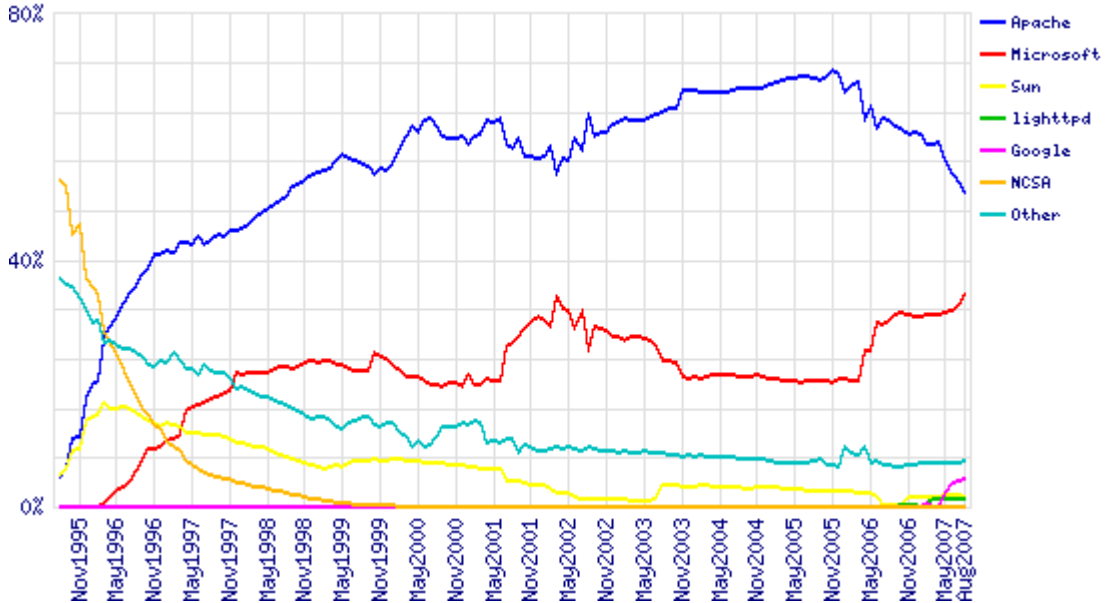
“XHTML se ha diseñado de tal forma que HTML regrese a sus raíces, convirtiéndose en un grupo de marcación que describe una clase particular de contenido independiente de la forma como dicho contenido va a ser visualizado”.¹⁷

1.2.4 Apache “Apache actúa como servidor Web. Su labor principal consiste en analizar cualquier archivo solicitado por un navegador y mostrar los resultados correctos en función del código del archivo. Apache es muy potente y puede realizar prácticamente cualquier tarea que se le pida”.¹⁸ En general, tiene tres aplicaciones principales; se puede utilizar para alojar un sitio Web para el público general, una intranet de una empresa o simplemente para probar páginas personales antes de cargarlas a un servidor seguro en otro equipo.

¹⁷ VALENTINE, Chelsea, MINNICK, Chris. XHTML. Prentice Hall. 2001.

¹⁸ GLASS, Michale, SCOUARNEC, Yann Le, NARAMORE, Elizabeth, MAILER, Gary, STOLZ, Jeremy, GERNER, Jason. Desarrollo Web con PHP, Apache y MySQL. Anaya Multimedia. 2004.

Figura 4. Mercado de servidores a través de los dominios de Internet.



Fuente: Netcraft. Disponible en Internet: <[http:// http://news.netcraft.com/](http://http://news.netcraft.com/)> [Citada 17 de Septiembre de 2007].

Apache es el servidor más usado en Internet, La Figura 5 muestra el uso de los diferentes tipos de servidores a nivel mundial, Apache posee 65,153,417 usuarios que representan el 50,92% del total de usuarios de servidores en Internet.

2. DESARROLLO DEL PROTOTIPO

2.1 PROGRAMACIÓN DE ALGORITMOS

Como primera medida se buscaron algoritmos desarrollados en PHP, herramienta utilizada en este proyecto, pero los algoritmos desarrollados para minería de datos en PHP son escasos, y no están disponibles fácilmente en la web.

Entonces se optó por una segunda opción que consistió en tomar algoritmos ya programados en otros lenguajes, y traducirlos o implementarlos en PHP. Con esta segunda alternativa también se generaron inconvenientes, debido a la diferencia entre lenguajes, principalmente porque lenguajes como Java o C++ utilizan librerías que son propias del lenguaje, y que son difícilmente descifrables al momento de traducirlas a otro lenguaje de programación.

En una era con tendencia al código abierto y a la reutilización de código, sería inadmisibles programar desde cero algoritmos que ya han sido programados, no tiene sentido reinventar la rueda. Por lo tanto la búsqueda de un punto de partida, llevó a una tercera opción que consistió en tomar algoritmos realizados en pseudo código, lo que permite adaptarlos a cualquier lenguaje de programación.

El algoritmo utilizado para encontrar Reglas de Asociación partió de un pseudo código muy sencillo, que se puede apreciar en la figura 5.

Figura 5. Seudo Código Apriori.

```
 $C_k$ : Candidate itemset of size k  
 $L_k$ : frequent itemset of size k  
  
 $L_1 = \{\text{frequent items}\};$   
for ( $k = 1; L_k \neq \emptyset; k++$ ) do begin  
   $C_{k+1}$  = candidates generated from  $L_k$ ;  
  for each transaction  $t$  in database do  
    increment the count of all candidates in  $C_{k+1}$   
    that are contained in  $t$   
   $L_{k+1}$  = candidates in  $C_{k+1}$  with min_support  
end  
return  $\cup_k L_k$ ;
```

Fuente: Apriori Algorithm. Disponible en Internet:

< http://www.cs.sunysb.edu/~cse634/lecture_notes/07apriori.pdf > [Citada 25 de Febrero de 2008].

Teniendo este sencillo seudo código como soporte, se implementó el algoritmo Apriori, para hallar reglas de asociación. El algoritmo Apriori basa su comportamiento en la búsqueda de conjuntos de ítems que cumplan con una cobertura mínima. Para lograr esto, primero se construyen los conjuntos formados por sólo un ítem que superan la cobertura mínima. Este conjunto de conjuntos se utiliza para construir el conjunto de conjuntos de dos ítems, y así sucesivamente hasta que se llegue a un tamaño en el cual no existan conjuntos de ítems con la cobertura requerida.

Una vez hallados los conjuntos más numerosos que cumplan con la cobertura, se generan todas las posibles combinaciones con los ítems del grupo que cumplan con la forma “**Si condición entonces resultado**”, y de estas se seleccionan las que cumplan con la confianza mínima.

Por ejemplo para el conjunto {Productos refrescantes, Comida típica, Películas de acción}, se pueden generar 6 combinaciones:

- ***Si** Productos refrescantes **entonces** Comida típica y Películas de acción.*
- ***Si** Comida típica **entonces** Productos refrescantes y Películas de acción.*
- ***Si** Películas de acción **entonces** Productos refrescantes y Comida típica.*
- ***Si** Productos refrescantes y Comida típica **entonces** Películas de acción.*
- ***Si** Productos refrescantes y Películas de acción **entonces** Comida típica.*
- ***Si** Comida típica y Películas de acción **entonces** Productos refrescantes.*

Seleccionando de éstas únicamente las que cumplen con el requisito mínimo de confianza. Para ver el algoritmo completo, ver el anexo A.

Otro algoritmo de Minería de datos que se implementó fue el algoritmo de componentes principales, para ellos se utilizó el método de covarianza, que para lograr determinar los componentes empieza por hallar la matriz de covarianza a partir de la matriz de datos, obteniendo así una matriz de covarianza $n \times n$, donde n es el número de variables. El siguiente paso consiste en hallar los valores y vectores propios, para esto se implementó el método iterativo de Jacobi. Luego la matriz de vectores propios se multiplica por la matriz de datos (la matriz de datos debe estar centrada), obteniendo así la matriz de componentes principales, donde se relaciona cada caso con cada componente principal. Posteriormente se debe hallar el porcentaje de varianza asociado a cada componente, lo que permitirá identificar la cantidad de información que representa cada componente, y utilizar ésto como medida para seleccionar únicamente los componentes más relevantes. Por último se puede determinar la matriz que asocia los componentes principales con las variables iniciales para conocer que tan representadas están las variables originales en cada componente. Para ver el algoritmo completo, ver el anexo B.

El algoritmo de k-medias también fue programado y su funcionamiento depende del número de grupos a identificar, este número está definido como un parámetro de entrada que es seleccionado por el usuario. Es un método iterativo donde cada grupo se ve representado por un centroide, y cada caso es clasificado en el grupo

cuyo centroide se encuentre más cerca de él, hablando en términos de distancia euclídea. A medida que ocurren las iteraciones, el algoritmo va recalculando los centroides como el centro de masas de los miembros de su grupo, y a su vez se recalcula la distancia euclídea de cada punto con los centroides, generando una nueva clasificación, repitiendo el proceso hasta el momento en que los centroides no cambian más. Para ver el algoritmo completo, ver el anexo C.

El prototipo también utiliza dos algoritmos simples que con sencillas consultas a la base de datos permiten la identificación de clientes frecuentes. El primero genera un listado de clientes que han hecho pedidos a determinado establecimiento, esto se logra utilizando un ciclo para recorrer los pedidos realizados y contabilizando el número de pedidos de cada cliente al establecimiento. Los resultados son entregados al usuario final en forma de tabla, donde se organizan los clientes según la frecuencia con la que realizan los pedidos. Por último se muestra una gráfica que permite ver los porcentajes de frecuencia de pedidos en el establecimiento, tomando como cien por ciento el total de clientes analizados. Para ver el algoritmo completo, ver el anexo D.

El otro algoritmo de identificación de clientes frecuentes, relaciona directamente a los clientes con los productos. Esto se logra utilizando un ciclo para recorrer los pedidos realizados y contabilizando el número de pedidos que cada cliente ha hecho del producto seleccionado. Los resultados son entregados al usuario final en forma de tabla, donde se organizan los clientes según la frecuencia con la que realizan los pedidos. Por último se muestra una gráfica que permite ver los porcentajes de frecuencia de pedidos del producto seleccionado, tomando como cien por ciento el total de clientes analizados. Para ver el algoritmo completo, ver el anexo E.

2.1.1 Pruebas de funcionamiento de los algoritmos Para comprobar el adecuado funcionamiento de los algoritmos, se utilizaron conjuntos de datos que permitieron aplicar los algoritmos programados en PHP, y comparar los resultados obtenidos con otros resultados generados por software reconocido de Minería de Datos.

2.1.1.1 Algoritmo Apriori. Para evaluar el comportamiento del algoritmo Apriori se utilizó un set de datos de los sobrevivientes del Titanic, donde se tienen en cuenta 4 variables, para 2201 registros de personas.

Los resultados obtenidos con Weka para una cobertura mínima de 0.5 y una confianza mínima de 0.9 se pueden apreciar en la tabla 1.

Tabla 1 Resultados Weka para el grupo de datos "Titanic".

```
Minimum support: 0.5
Minimum metric <confidence>: 0.9
Number of cycles performed: 10

Generated sets of large itemsets:

Size of set of large itemsets L(1): 4

Size of set of large itemsets L(2): 4

Size of set of large itemsets L(3): 1

Best rules found:

1. Clase=0 1210 ==> Edad=1 1204    conf:(1)
2. Sexo=1 Sobrevivió?=0 1364 ==> Edad=1 1329    conf:(0.97)
3. Sobrevivió?=0 1490 ==> Edad=1 1438    conf:(0.97)
4. Sexo=1 1731 ==> Edad=1 1667    conf:(0.96)
5. Edad=1 Sobrevivió?=0 1438 ==> Sexo=1 1329    conf:(0.92)
6. Sobrevivió?=0 1490 ==> Sexo=1 1364    conf:(0.92)
```

Fuente: Resultados Software WEKA.

Mientras que los resultados obtenidos con el algoritmo programado en PHP se pueden observar en la tabla 2, donde se etiquetaron las variables de la misma manera que en Weka para poder comparar resultados con mayor facilidad.

Tabla 2 Resultados Algoritmo Apriori

edad=1 - sobrevivio=0 - >> sexo=1 - Cob=1438 Conf=1329/1438=92%
sexo=1 - sobrevivio=0 - >> edad=1 - Cob=1364 Conf=1329/1364=97%

Fuente: Resultados Pruebas en Prototipo Programado.

El algoritmo determinó correctamente las reglas número 2 y 5. La diferencia radica en que el algoritmo programado en PHP no trata simplemente de hallar todas las reglas que cumplan la condición de confianza y cobertura, sino de seleccionar las reglas más interesantes. En esta prueba solo se obtienen las reglas que involucran 3 variables porque el algoritmo selecciona primero los conjuntos del mayor número de variables posible que cumplan con la cobertura mínima (en este caso de 3 variables), luego determina todas las posibles combinaciones de ese o esos conjuntos y selecciona solo las combinaciones que cumplan con la confianza mínima.

2.1.1.2 Algoritmo de K-medias. Para evaluar el comportamiento del algoritmo de k-medias se utilizó un grupo de datos titulado “housing” que contiene 506 casos, involucrando 13 variables con valores numéricos apropiados para trabajar técnicas de agrupamiento y componentes principales.

Los resultados obtenidos con SPSS para k=5 (es decir, 5 grupos) muestran diferentes tablas de resultados, entre las cuales se destacan:

- Tabla que indica a que cluster pertenece cada caso.
- Tabla de centroides iniciales.
- Tabla de centroides finales.
- Tabla con el número de casos correspondientes a cada cluster.

Estos resultados se pueden observar en el anexo F.

Mientras que los resultados obtenidos mediante el algoritmo de k-medias programado en PHP se pueden apreciar en el anexo G.

Al comparar los resultados generados con SPSS y el software programado, se encontró una diferencia importante, que consiste en la determinación de los centroides iniciales. La cuestión de los centroides iniciales ha sido tema de discusión siempre, ya que estos influyen directamente en la conformación de los grupos.

El algoritmo programado en PHP halla sus centroides basado en las distancias máxima y mínima de cada variable. Por ejemplo si una variable tiene como valor mínimo 30 y como valor máximo 90 y se ha seleccionado $k = 5$, (es decir, 5 grupos) los valores iniciales de los 5 centroides en esa variable serían 40, 50, 70, 80 y 90. Mientras que SPSS, halla los centroides iniciales de manera diferente.

Pero aún así los resultados obtenidos fueron muy similares, obteniendo una diferencia del 3% aproximadamente (mas o menos 15 casos en grupos diferentes).

Luego de hacer esta prueba y no encontrar resultados idénticos se realizó una segunda prueba que consistió en copiar los centroides iniciales obtenidos con SPSS y aplicar el algoritmo programado en PHP. Los resultados coincidieron en

un 100%, incluso los valores para los centroides finales y los números que identifican cada uno de los grupos. Los resultados de esta segunda prueba se pueden observar en el anexo H.

2.1.1.3 Algoritmo de componentes principales (método de covarianza).

Para evaluar el algoritmo de componentes principales también se utilizó el grupo de datos titulado “housing”, utilizado para evaluar el desempeño del algoritmo de k-medias.

Los resultados obtenidos con SPSS se encuentran en el Anexo I y se destacan las siguientes tablas:

- Varianza total explicada.
- Matriz de covarianza.

Al comparar los resultados con los que se obtienen con el algoritmo programado se encuentra que la matriz de covarianza es muy similar, lo cual quiere decir que el algoritmo ha iniciado correctamente, ya que la matriz de covarianza es uno de los primeros pasos para hallar componentes principales.

Al continuar con el análisis se encuentra que la varianza total explicada es exactamente igual en SPSS que con el algoritmo programado, lo que permite determinar que el algoritmo está correctamente programado, puesto que la varianza nos indica el porcentaje de información que representa cada componente. Por ejemplo si un componente tiene un 70% de varianza total, quiere decir que al utilizar solo ese componente se estaría perdiendo el 30% de información.

El algoritmo programado en PHP selecciona los mejores componentes, de tal forma que se pierda menos del 25% de información seleccionando la menor cantidad de componentes, es por eso que en la prueba se selecciona únicamente el componente principal 1, ya que su varianza supera el 75%.

Los resultados obtenidos con el algoritmo programado en PHP se pueden observar en anexo J, donde se podrán constatar la comparación que se hace en la tabla 3.

Tabla 3 Comparación de Varianza.

Componente	SPSS	Algoritmo Programado
1	80.457	80.46%
2	16.271	16.27%
3	2.141	2.14%

Fuente: Pruebas Software SPSS y en Prototipo Programado.

2.2 INTERFAZ GRÁFICA

Parte importante de un prototipo computacional es la interfaz gráfica. Ésta permite la interacción de los usuarios finales con el prototipo, por tal motivo, se encuentra dividida en dos partes, según el público al que va dirigida. La primera constituye la parte correspondiente al público en general, y la segunda corresponde a la parte que únicamente los establecimientos podrán observar.

Público General:

La primera página que se presenta es la página de inicio, que permite observar las diferentes categorías para hacer un pedido a domicilio. Ver figura 6.

En el menú (tanto inferior como superior) se observan los enlaces a las diferentes páginas que conforman la aplicación. En el banner que encabeza la página se provee un espacio para posicionar un logo y un nombre de empresa, en este caso

se ha escogido “Web Delivery” como ejemplo. También se puede encontrar un espacio para poner el número telefónico al cual se harían todos los pedidos ha domicilio, así como un espacio para el slogan y una frase que represente a la empresa.

En resumen, la interfaz gráfica se ha manejado de acuerdo a las proyecciones futuras del prototipo computacional, es decir, la interfaz muestra como quedaría el prototipo computacional si evolucionara hasta el punto que permitiera hacer pedidos tanto telefónicamente como a través de internet.

También se puede observar que existe un formulario de ingreso que solicita un usuario y una contraseña, este formulario es utilizado por los establecimientos comerciales para acceder a la información generada por los algoritmos de minería de datos.

Figura 6. Página de "Inicio" del Prototipo Computacional.



Fuente: Pantalla capturada directamente del explorador Web. Autor del proyecto.

La página correspondiente a “Quienes Somos” muestra simplemente el perfil empresarial, en este caso continuamos con “Web Delivery” como ejemplo. Ver figura 7.

Figura 7. Página de "Quienes Somos" del Prototipo Computacional.



Fuente: Pantalla capturada directamente del explorador Web. Autor del proyecto.

La página correspondiente a servicios muestra una descripción completa de los dos tipos de servicios que en un futuro estará en capacidad de prestar la empresa ejemplo “Web Delivery”. Ver figura 8.

Figura 8. Página de "Servicios" del Prototipo Computacional.



Fuente: Pantalla capturada directamente del explorador Web. Autor del proyecto.

En la página de establecimientos se muestra de nuevo el formulario de acceso, así como un formulario que será utilizado en un futuro para el registro de nuevos establecimientos en el sistema. Ver figura 9.

Figura 9. Página de "Establecimientos" del Prototipo Computacional.

Pedir domicilio Quienes somos Servicios Establecimientos Contáctenos

WebDelivery

Pedidos telefónicos: 6767676

Los domicilios han evolucionado
Todo lo que desea ordenar a domicilio, ahora de una forma novedosa.

Inscriba su establecimiento.

Nombre: Ciudad:
Tipo: Representante:
Usuario: Contraseña:

Ingreso

Nombre:
Contraseña:

Pedir domicilio Quienes somos Servicios Establecimientos Contáctenos

© WebDelivery Todos los derechos reservados.
[Suscribirse](#)

Designed By: Template World

XHTML CSS

WebDelivery

Fuente: Pantalla capturada directamente del explorador Web. Autor del proyecto.

La página de contáctenos presenta un formulario para el envío de inquietudes, además de la información de contacto de las diferentes sucursales de la empresa. Para ilustrar mejor esta sección se continúa con el ejemplo de "Web Delivery". Ver figura 10.

Figura 10. Página de "Contáctenos" del Prototipo Computacional.

Pedir domicilio Quiénes somos Servicios Establecimientos **Contáctenos**

WebDelivery

Pedidos telefónicos: 6767676

Los domicilios han evolucionado
Todo lo que desea ordenar a domicilio, ahora de una forma novedosa.

Oficina Principal.
Bucaramanga
Calle 52 No. 52 -87, Cabecera
Teléfono: 6454545
Fax: 6565656
Correo: ventas@domiciliosweb.com

Sucursal Barranquilla.
Calle 52 No. 52 -87, Cabecera
Teléfono: 6454545
Fax: 6565656
Correo: contacto@domiciliosweb.com

Sucursal Medellín.
Calle 52 No. 52 -87, El Santuario
Teléfono: 6454545
Fax: 6565656
Correo: domicilios@domiciliosweb.com

Sucursal Bogotá.
Calle 52 No. 52 -87, Centro
Teléfono: 6454545
Fax: 6565656
Correo: gerencia@domiciliosweb.com

Nombre:

Correo electrónico:

Mensaje:

IR

Pedir domicilio Quiénes somos Servicios Establecimientos Contáctenos

©WebDelivery. Todos los derechos reservados.
Suscribirse

Designed By: Template World

XHTML CSS

WebDelivery

Fuente: Pantalla capturada directamente del explorador Web. Autor del proyecto.

Establecimientos:

Para ingresar a la zona de establecimientos es necesario tener el usuario y la contraseña de acceso. Una vez autenticado el establecimiento tendrá acceso al panel de control.

En la página inicial de la zona de establecimientos se podrán observar todos los datos pertinentes al establecimiento. Para ilustrar con un ejemplo, se ha escogido

el establecimiento “Donde esta Javier?” un restaurante de comidas rápidas. Ver figura 11.

En el menú principal se pueden observar los enlaces de navegación de la sesión de establecimientos, mientras que en el banner que encabeza la página se muestra el nombre del establecimiento, además de un botón que permite salir y volver a la zona para el público en general.

Además de los datos del establecimiento, esta página tiene un formulario de inquietudes, donde en un futuro se dará soporte técnico a los establecimientos en la utilización de la herramienta.

Figura 11. Página de "Mis Datos" del Prototipo Computacional.



Fuente: Pantalla capturada directamente del explorador Web. Autor del proyecto.

En la página de promociones se muestra un formulario para escoger el número de grupos a determinar, además del resultado de aplicar la primera técnica de minería de datos seleccionada para este prototipo; Análisis de componentes principales, prerequisite para aplicar el algoritmo de k-medias para determinar grupos de clientes. (Ver figura 12) Los componentes principales desempeñan un papel vital con respecto a la eficiencia del algoritmo de k-medias, ya que reduce la dimensionalidad de los datos, es decir, se representa la misma información con menos variables, logrando así mayor velocidad con el algoritmo de k-medias.

Figura 12. Página de "Promociones" del Prototipo Computacional.



Fuente: Pantalla capturada directamente del explorador Web. Autor del proyecto.

El número de clusters que se selecciona en el formulario se utiliza como parámetro de entrada para mostrar los resultados de las promociones. Dichos resultados se pueden apreciar en la figura 13, donde para cada grupo se genera un listado de los clientes que pertenecen al grupo. Además de eso se ha etiquetado cada grupo con las 3 preferencias principales de los clientes del grupo (se obtienen consultando la tabla de preferencias para los clientes del grupo), así como el número de clientes que hay en cada grupo.

Adicionalmente se han incluido dos distancias con respecto a los centroides; Máxima distancia a centroide interno y Mínima distancia a centroide externo. Estas distancias son datos importantes para estimar la cantidad ideal de clusters, lo que hay que buscar en este caso es que los datos estén compactos en su grupo y distantes de otros grupos, es decir, será mejor mientras la "Máxima distancia a centroide interno" sea pequeña y la "Mínima distancia a centroide externo" sea grande.

Figura 13. Página de "Resultado Promociones" del Prototipo Computacional.

GRUPO 2:

Id	Nombre	Cuenta
5	Raul Buenahora Angarita	3
8	Nicole Maria Gil Briceño	4
9	Veronica Maria Herrera	5
10	Lorena Ochoa	5
15	Jose Esteban Contreras	8
18	Ana Virginia Quintanilla	11
21	Lina Maria Castro Castañeda	12
27	Efren Leonel Bermudes Gil	16
29	Oneida Lucia Riaño Orduz	17
1001	Lola Jimenez	1001
1008	Cesar Antonio Baquero	1004
1020	Joaquin Fernando Medina	1011
1027	Silvio Aramado Chávez	1015
1032	CIRO ALBERTO IBARRA NAVAS	1019
1033	Gloria Juya Vargas	1019
1034	Rafael Tobias Quintana Moreno	1020
1039	Ricardo Gómez Murcia	1023
1092	eduardo valencia lopez	1055
1100	ALBERTO AMARIS NIETO	1059

Máxima distancia a centroide interno:**0.59884**
Mínima distancia a centroide externo:**0.31179**
Número de casos:**19**

Preferencias principales del grupo:
- Productos Económicos
- Películas en Estreno
- El Hielo

Fuente: Pantalla capturada directamente del explorador Web. Autor del proyecto.

La página de "Clientes Frecuentes" corresponde a un listado de clientes similar al mostrado en los resultados de la promoción, la única diferencia es que el listado de clientes es escogido según la frecuencia con la que pidan domicilios al establecimiento. Ver figura 14.

Figura 14. Página de “Clientes Frecuentes” del Prototipo Computacional.



Id	Nombre	Frecuencia	Veces
3	marisol briceño orduz	Baja	1
9	veronica maria herrera	Baja	1
14	humberto león nuñez	Baja	1
21	lina maria castro castañeda	Baja	1
1002	eduardo hernandez leon	Baja	1
1004	manuel gilardo guzmán muñoz	Baja	1
1008	cesar antonio baquero	Baja	1
1013	maría emilse ramírez pulido	Baja	1
1022	david mosquera silva	Baja	1
1026	lbo herley bolafios ordoñez	Baja	1
1029	isail antonio giraldo vanegas	Baja	1
1032	ciro alberto ibarra navas	Baja	1
1048	oscar enrique caro garcia	Baja	1
1068	edgar rodriguez	Baja	1
1076	luis fernando pinzon manrique	Baja	1
1081	carlos eduardo arroyo cubillos	Baja	1
1087	jose luis martinez sandoval	Baja	1
1100	alberto amaris nieto	Baja	1



Fuente: Pantalla capturada directamente del explorador Web. Autor del proyecto.

Mientras que la página de “Clientes-Productos” corresponde a diferentes listados de clientes según la frecuencia con la que soliciten determinado producto. El producto debe ser seleccionado en el formulario que se encuentra en la parte derecha de la pantalla. Ver figura 15.

Figura 15. Página de "Clientes-Productos" del Prototipo Computacional.



Fuente: Pantalla capturada directamente del explorador Web. Autor del proyecto.

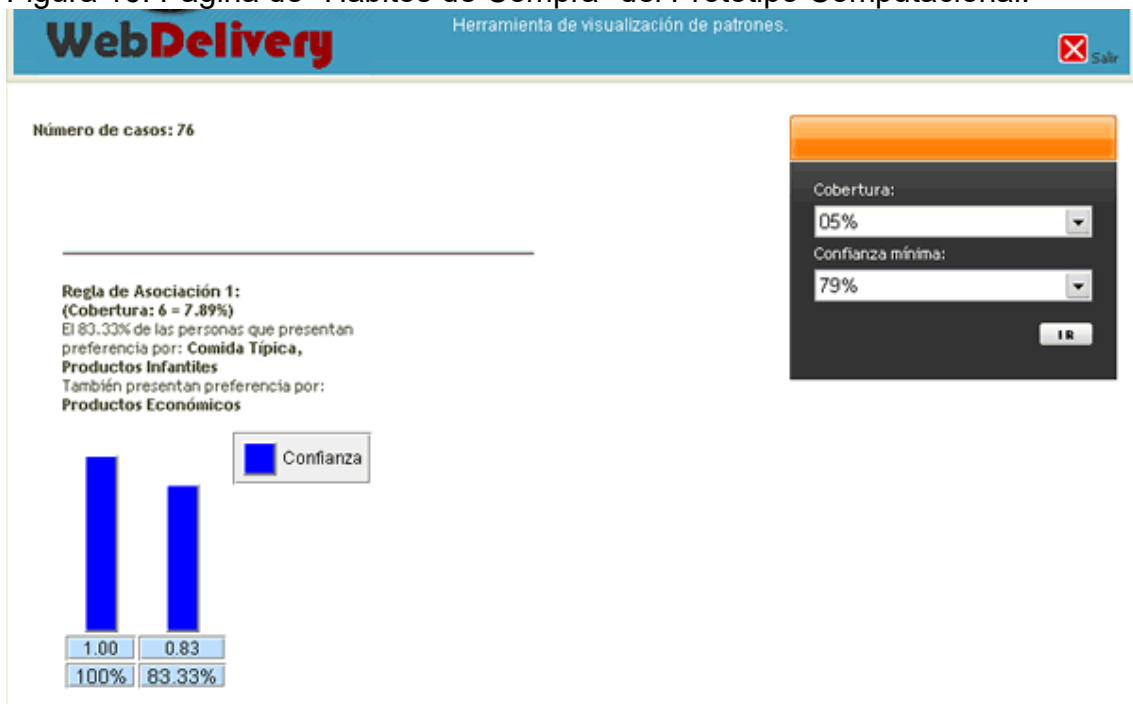
La página de hábitos de compra muestra las reglas de asociación encontradas. Estas reglas de asociación siempre relacionan un grupo de preferencia con otro, y se refuerza la visualización de cada regla de asociación con una gráfica que representa la confianza.

En la figura 16 se puede observar que al principio de los resultados se muestra el número total de casos analizados, además cada regla de asociación permite

apreciar su cobertura de dos formas; en porcentaje y con el número de casos que satisface la regla.

El formulario que se encuentra en la derecha permite seleccionar la cobertura y confianza mínimas de las reglas de asociación. Con esto los establecimientos pueden filtrar las reglas de asociación que mejor se ajusten a sus requerimientos.

Figura 16. Página de "Hábitos de Compra" del Prototipo Computacional.



Fuente: Pantalla capturada directamente del explorador Web. Autor del proyecto.

2.3 SÁBANA DE DATOS

La recolección de datos constituye una parte importante de la Minería de Datos fundamental para poder realizar pruebas. Por tal motivo se generó una sábana de datos con todo lo necesario para experimentar con el prototipo. Para esto fue necesario simular datos para 13 tablas:

- Países
- Ciudades
- Cuentas
- Personas
- Tipos de establecimientos
- Establecimientos
- Productos
- Tipos de características para productos
- Características para productos
- Productos en establecimientos
- Pedidos
- Detalles de pedido
- Preferencias¹⁹

Donde se simularon pedidos realizados por cien personas aproximadamente, durante 2 meses. Todo con el propósito de generar una tabla de preferencias, que se utiliza como la principal tabla para la extracción de información, permitiendo así aplicar las diferentes técnicas de minería de datos.

¹⁹ Nota de pie: Tabla de preferencias disponible en el anexo D.

3. RESULTADOS

3.1 RESULTADOS OBTENIDOS CON EL PROTOTIPO

El prototipo computacional elaborado presenta dos tipos de resultados; el primero de ellos corresponde a las secciones que involucran algoritmos de Minería de Datos para su obtención, mientras el segundo tipo de resultados no involucra este tipo de algoritmos, sin embargo muestra información importante para conocer el comportamiento de los clientes.

Resultados de Algoritmos de Minería de Datos:

Se pueden observar dos secciones del prototipo computacional que utilizan algoritmos de Minería de Datos para la obtención de resultados; una de estas secciones es “Promociones”, que utiliza el análisis de componentes principales como primera medida para la reducción de dimensionalidad, obteniendo un nuevo grupo de variables representativas de la muestra.

A partir de estas variables trabaja el algoritmo de k-medias. El usuario podrá seleccionar la cantidad de grupos en los que desea clasificar los clientes. Cuando el usuario ya ha seleccionado la cantidad de grupos, el sistema clasifica los clientes como se aprecia en la figura 17, donde se han capturado los resultados generados al dividir los clientes en 2 grupos.

Figura 17 Resultados de k-medias para dos grupos.

GRUPO 1:

Id	Nombre	Cuenta
3	Marisol Briceño Orduz	2
6	Alcira Briceño Orozco	4
11	Olga María Jaramillo Torres	6
16	Roberto de Jesus Brochero	9
17	Judith Ester Duque Lopez	10
19	Laura Sofía Quintanilla	11
20	Nancy Elena Castañeda	12
22	Andrea María Castro Castañeda	12
23	Ana Dolores Reyna Vanegas	13
24	Clemente Pinzon Bernal	14
26	Efren Ramón Bermudes Gil	16
28	Lina Sofía Bermudes Gil	16
30	Adrian Ricardo Gomez Lipez	18
1003	ELEIDA ROSA GOMEZ CAMPUZANO	1002
1007	Ignacio Bejarano Rodriguez	1004
1013	María Emilse Ramirez Pulido	1007
1015	Jesus Duqueiro Rojas Alvarado	1008
1018	Luis Alberto Cardona Bedoya	1010
1019	CARLOS ALBERTO TREJOS ORTIZ	1011
1022	DAVID MOSQUERA SILVA	1012
1023	JAIME PAVAJEALI BOLAÑO	1013
1025	MILLER BOLAÑOS GOMEZ	1014
1031	Jhovany Moreno Trujillo	1018
1040	Victor Armando Cubides	1023
1094	Juan Carlos Durán Barrero	1056
1096	Diego Fernando Acevedo Holguin	1058
1098	NELSON ENRIQUE CARRILLO VIZCAINO	1059

Máxima distancia a centroide interno: **1.2334** **Preferencias principales del grupo:**
Mínima distancia a centroide externo: **0.84217** - Productos Refrescantes
Número de casos: **27** - Productos Económicos
 - Productos Infantiles

GRUPO 2:

Id	Nombre	Cuenta
4	Jaime Alberto Rey Briceño	2
5	Raul Buenahora Angarita	3
7	Fernando Jose Gil Briceño	4
8	Nicole María Gil Briceño	4
9	Veronica María Herrera	5
10	Lorena Ochoa	5
12	Leonor Solano Nuñez	7
13	Pedro Leonardo León Nuñez	7
14	Humberto León Nuñez	7
15	Jose Esteban Contreras	8
18	Ana Virginia Quintanilla	11
21	Lina María Castro Castañeda	12
25	Luis Miguel Torres Pabon	15
27	Efren Leonel Bermudes Gil	16
29	Oneida Lucia Riaño Orduz	17
1001	Lola Jimenez	1001
1002	EDUARDO Hernandez leon	1001
1004	Manuel Gildardo Guzmán Muñoz	1002
1005	Hector vargas	1003
1006	ROBINSON ERAZO	1003
1008	Cesar Antonio Baquero	1004
1009	José Arturo León Hurtado	1005
1010	BERNARDO MADURO SILVERA	1005
1011	YOMAN CESAR MONTERROSA GUERRERO	1006
1012	RAMIRO JOSE MENDOZA ACOSTA	1006
1014	Raul Eduardo Muñoz	1007
1016	Felix María Duarte	1009
1017	Rafael Rubio Acevedo	1010
1020	Joaquín Fernando Medina	1011
1021	Joel Moreno Caceres	1012
1024	Juan Carlos Arteaga Córdoba	1013
1026	Ibo Herley Bolaños Ordoñez	1014
1027	Silvio Aramado Chávez	1015
1028	Hernan Arturo Estrella	1016
1029	Isaí Antonio Giraldo Vanegas	1016
1030	Jhon Jarvi Arenas	1017
1032	CIRO ALBERTO IBARRA NAVAS	1019
1033	Gloria Juya Vargas	1019
1034	Rafael Tobias Quintana Moreno	1020
1035	Jorge Enrique Espitia Hernandez	1021
1036	HENRY VARGAS CANTOR	1021
1039	Ricardo Gómez Murcia	1023
1042	UBER ROBLES SAPUG	1024
1092	eduardo valencia lopez	1055
1093	Leonardo Alvarez	1056
1095	Juan Pablo Guerrero A	1057
1097	Jaime Arturo Suarez	1058
1099	ASMED DARIO MERCADO S.	1059
1100	ALBERTO AMARIS NIETO	1059

Máxima distancia a centroide interno: **0.77829** **Preferencias principales del grupo:**
Mínima distancia a centroide externo: **0.77257** - Productos Económicos
Número de casos: **49** - Películas en Estreno
 - Salsa Tártara

Fuente: Pantalla capturada directamente del explorador Web. Autor del proyecto.

La otra sección del prototipo que utiliza algoritmos de Minería de Datos es “Hábitos de Compra”, donde se comparan todas las características que poseen los productos. Y según las exigencias de los clientes, se hallan las reglas de asociación.

No tiene importancia que establecimiento observe los “Hábitos de Compra”, los resultados siempre serán iguales. Lo que marca la diferencia es el formulario de la parte derecha, que permite filtrar las reglas encontradas, ya que es allí donde el usuario debe seleccionar la confianza y la cobertura mínimas que aceptaría para considerar la regla de asociación útil. La cobertura corresponde al porcentaje de clientes para los que aplica la regla de asociación, mientras que la confianza corresponde al porcentaje de clientes que prefieren las dos características asociadas en la regla, tomando como 100% el número de clientes que prefieren la primera de las características asociadas en la regla.

Para ilustrar mejor esta sección en la figura 16 se presentan los resultados obtenidos al seleccionar 05% como cobertura mínima y 79% como confianza mínima. Donde solo se obtiene una regla de asociación, debido a la irregularidad de la muestra.

Otros Resultados:

El prototipo computacional proporciona otro tipo de indicadores de comportamiento del consumidor, estos indicadores corresponden a “Clientes Frecuentes” y “Clientes-Productos”. Los Clientes Frecuentes varían según el establecimiento, es decir, los resultados para esta sección serán diferentes según el establecimiento que haya ingresado al área segura. Como cada establecimiento tiene su propio usuario y contraseña, ninguno podrá observar los clientes frecuentes de otro establecimiento.

Para obtener una idea más clara de los resultados obtenidos en clientes frecuentes, en la figura 17 se capturaron los resultados obtenidos al ingresar como el establecimiento de películas Chaplin, que tiene una frecuencia baja de clientes.

Figura 18 Resultado de Clientes Frecuentes para la video tienda Chaplin.

Id	Nombre	Frecuencia	Veces
3	marisol briceño orduz	Baja	1
9	veronica maria herrera	Baja	1
14	humberto león nuñez	Baja	1
21	lina maria castro castañeda	Baja	1
1002	eduardo hernandez leon	Baja	1
1004	manuel gildardo guzmán Muñoz	Baja	1
1008	cesar antonio baquero	Baja	1
1013	maría emilse ramírez pulido	Baja	1
1022	david mosquera silva	Baja	1
1026	ibo herley bolaños ordoñez	Baja	1
1029	israil antonio giraldo vanegas	Baja	1
1032	ciro alberto ibarra navas	Baja	1
1048	oscar enrique caro garcia	Baja	1
1068	edgar rodriguez	Baja	1
1076	luis fernando pinzon manrique	Baja	1
1081	carlos eduardo arroyo cubillos	Baja	1
1087	jose luis martinez sandoval	Baja	1
1100	alberto amaris nieto	Baja	1



Fuente: Pantalla capturada directamente del explorador Web. Autor del proyecto.

Por el contrario, los resultados que se obtienen en “Clientes-Productos”, no tienen relación alguna con los establecimientos, es decir, que se obtendrá el mismo resultado si se ingresa con un establecimiento o con otro. Aquí se debe escoger un producto, para obtener la lista de clientes con su respectiva frecuencia de compra. Para entender mejor los resultados obtenidos en esta sección se puede observar la figura 15, donde se ha seleccionado el producto “Naranja”.

3.2 ANÁLISIS DE RESULTADOS

La información generada por el prototipo computacional resulta útil en gran medida para los establecimientos comerciales porque les permite conocer los gustos y preferencias de los clientes de maneras diferentes.

La sección de promociones, al clasificar los clientes permite a los establecimientos dirigir su publicidad únicamente a los clientes que realmente puedan estar interesados en ella, además de aportar criterios de decisión para saber a que grupos de personas aplicar determinadas estrategias comerciales. Mientras que las reglas de asociación permitirán incrementar las ventas de determinados productos, ofreciéndolos en el momento adecuado a las personas adecuadas. Estos fueron dos sencillos ejemplos de la utilidad que tiene el prototipo computacional desarrollado.

Aunque no es necesario ser experto en mercadeo para encontrar muchas otras formas de utilizar la información generada por el prototipo, se ha consultado a Clara Patricia Cárdenas²⁰ quien afirma; “La información es una herramienta fundamental para cualquier empresa, quien tiene la información, tiene el poder... pienso que este software es un gran aporte para gestionar los datos de los usuarios de los servicios a domicilio y de ello sacar tendencias útiles sobre las necesidades de los clientes sin necesidad de invertir grandes cantidades de dinero en Investigaciones de Mercado que en pocos meses puede estar obsoleta. Por el contrario, este software es una herramienta de Investigación de Mercados que proporciona información que se actualiza constantemente”. Así mismo Carlos Augusto Quintero Duque²¹ afirma que el prototipo computacional “Establece las diferencias entre categorías de productos y define las necesidades del cliente”.

²⁰ Clara Patricia Cárdenas: Directora Comercial y de Mercadeo C.I. AMEROIL S.A., Contadora UNAB, Diplomado en Marketing UNAB.

²¹ Carlos Augusto Quintero Duque: Director de la oficina de egresados y emprendimiento UNAB.

Es posible que alguna persona llegue a pensar que las diferentes secciones del prototipo hacen lo mismo, y en cierta medida tendría razón. En realidad todas las secciones del prototipo computacional están enfocadas en el análisis del comportamiento de los clientes y en la identificación de sus patrones de compra.

Pero ahí es donde está el verdadero valor agregado del trabajo, se busca el mismo objetivo de maneras diferentes y obteniendo resultados que se muestran de formas diferentes, en uno de los casos, con listados de clientes etiquetados con sus 3 mayores preferencias y en otro con reglas claras que explican las tendencias de comportamiento de los clientes.

4. TRABAJOS FUTUROS

La funcionalidad del prototipo computacional elaborado en este trabajo, se enfoca hacia la integración y el desarrollo de tecnología aplicada a procesos del entorno empresarial real.

La idea es diseñar un software que permita hacer domicilios por internet y analizar el comportamiento de los clientes de cada establecimiento según su perfil de consumo.

A pesar de que el prototipo que resulta de la elaboración de este trabajo permite analizar el comportamiento de los clientes a través de las reglas de asociación y además permite la selección de un grupo de clientes con determinada preferencia en común, dicho prototipo no está en capacidad de recibir pedidos de los clientes, ni de registrar nuevos establecimientos.

Para lograr hacer el presente proyecto se utilizó una sábana de datos ficticia, que incluyó 6 establecimientos y 344 pedidos realizados por 148 personas.

El trabajo proyectado a partir de este proyecto incluye fases que van desde el rediseño, hasta la implementación de la herramienta computacional en las empresas, permitiendo que un gran porcentaje de establecimientos que prestan servicio a domicilio en una ciudad, estén inscritos; convirtiéndose así en una fuente de información valiosa, que almacenará gran cantidad de datos, lo cual permitirá identificar patrones de comportamiento de usuarios con mayor precisión.

Estos patrones detectados después del análisis de millones de datos, son finalmente la información clave que le permitirá al nivel estratégico de una

organización tomar decisiones y diseñar objetivos acorde a las necesidades de su público objetivo.

Los resultados de la conversión de los datos en información a través de un software, se verá reflejada directamente en el desempeño comercial de las empresas, mediante aumento en las ventas, reducción de costos logísticos y de transporte, mejoras en el despacho de pedidos, disminución de quejas y reclamos, etc.

El siguiente paso consiste en aumentar la funcionalidad del prototipo computacional, adecuándolo para el registro de clientes y de establecimientos, y permitiendo que cada establecimiento incluya su portafolio de productos y servicios, con el que los clientes podrán hacer pedidos.

Los avances que son posibles de realizar siguiendo con la idea que propone este proyecto, se podrán observar en el mapa del sitio que fue diseñado (Ver Anexo K) y cuyo resultado será la base para futuras investigaciones de Ingeniería y otras relacionadas con herramientas de Marketing Directo en el sector de servicios a domicilios y en general con todos los aspectos relacionados con el comercio electrónico.

5. CONCLUSIONES

- La primera técnica que se aplicada en el prototipo computacional es el Análisis de Componentes Principales ya que permite reducir la dimensionalidad de los datos, eliminando variables no relevantes, agrupándolas en factores o componentes que facilitan a través de otra técnica (Agrupación) concluir cuáles son los clientes que se pueden asociar con los componentes hallados.
- La interfaz de usuario es el puente que permite la interpretación de las técnicas de minería de datos aplicadas con el usuario final. Por lo general el usuario final desconoce el tema de Minería de Datos, por lo tanto no entendería interfaces especializadas como las mostradas en programas del tipo SPSS o Weka, por lo que es necesario hacer interfaces entendibles por el público al que va dirigido, en este caso, gerentes o administradores de establecimientos comerciales que no poseen conocimiento de Minería de Datos.
- La Minería de Datos es una tecnología que permite recortar considerablemente la brecha que existe entre la capacidad de almacenar datos y la capacidad para interpretarlos. La capacidad para almacenar datos ha crecido en los últimos años a velocidades exponenciales. Mientras que la capacidad para analizarlos no ha crecido tanto. Explicándose así la gran importancia que ha adquirido la Minería de Datos en los últimos años, convirtiéndose en campo de investigación trascendental para la toma de decisiones.

- El hecho de involucrar disciplinas importantes de la ingeniería de sistemas como el desarrollo de Software, Bases de datos y Estadística, hace de este proyecto una experiencia sumamente enriquecedora para un ingeniero de sistemas.
- Para lograr resultados más satisfactorios es necesario una cantidad de datos considerablemente grande. La baja cantidad de datos que se incluyó en la sabana de datos utilizada fue un factor que dificultó el proceso de identificación de patrones, sin embargo, esta dificultad no constituyó un impedimento para lograr los objetivos.

BIBLIOGRAFÍA

Algoritmos de Minería de Datos programados en Java y en C, proporcionados por el docente Javier Hernández Cáceres. **Consultados:** 25 de Febrero de 2008.

Apriori Algorithm. **Disponible en:**

http://www.cs.sunysb.edu/~cse634/lecture_notes/07apriori.pdf **Visitada:** 25 de Febrero de 2008 **Autor:** WASILEWSKA, Anita.

El Proceso de Descubrimiento de Conocimiento a Partir de Bases de Datos. Available from Internet: <<http://wwwdi.ujaen.es/asignaturas/dm/tema2.pdf>> [Cited 28 de Octubre de 2007].

Fases de la Minería de Datos. **Disponible en:**

<http://www.daedalus.es/AreasMDFases-E.php> **Visitada:** 14 de Agosto de 2007

GLASS, Michale, SCOUARNEC, Yann Le, NARAMORE, Elizabeth, MAILER, Gary, STOLZ, Jeremy, GERNER, Jason. Desarrollo Web con PHP, Apache y MySQL. Anaya Multimedia. 2004.

HAND, David, MANNILA, Heikki, SMYTH, Padhraic. Principles of Data Mining. Bradford. 2001.

HAN, Jiawei, KAMBER, Micheline. Data Mining Concepts and Techniques. Morgan Kaufmann Publishers. 2001.

HERNANDEZ ORALLO, José, RAMIREZ QUINTANA, Maria José, FERRI RAMÍREZ, César. Introducción a la Minería de Datos. Pearson Prentice Hall. 2004.

Netcraft. **Disponible en:** <http://news.netcraft.com/> **Visitada:** 27 de Septiembre de 2007 **Autor:** Netcraft LTD.

PHP: Introducción. **Disponible en:**
<http://www.php.net/manual/es/introduction.php#intro-what> **Visitada:** 5 de Septiembre de 2007 **Autor:** The PHP Group.

The Web Page Of Chemometrics - Reconhecimento de Padrões. **Disponible en:**
<http://www.dq.fct.unl.pt/QOF/chem9.html> **Visitada:** 27 de Septiembre de 2007.
Autor: QOF Group.

VALENTINE, Chelsea, MINNICK, Chris. XHTML. Prentice Hall. 2001.

WITTEN, Ian H, FRANK, Eibe. Data Mining Practical Machine Learning Tools and Techniques with Java Implementations. Morgan Kaufmann Publishers, 2000.

ANEXOS

Anexo A. Algoritmo Apriori.

```
<?php
echo "hola";
//if para comprobar el envío de los parámetros de confianza y cobertura.
if (isset($_POST["cobertura"]) && isset($_POST["confianza"]) &&
$_POST["confianza"] != 'none' && $_POST["cobertura"] != 'none'){

    //función para determinar todas las posibles funciones de un grupo de datos
    function combinacion($total, $grupo){
        $nivel = 1;
        $nc = 0;
        $candidato = array();

        $numcomb = $total;
        for($i=0; $i<$total; ++$i){
            $candidato[$i][0] = $i;
        }

        while($nivel<$grupo){

            for($j=0; $j<($numcomb-1); ++$j){
                for($h=($j+1); $h<$numcomb; ++$h){
                    $aprueba = 1;
                    for($m=1; $m<$nivel; ++$m){
                        if($candidato[$j][$m] != $candidato[$h][$m-
1]){
```

```

        $aprueba = 0;
    }
}
if($aprueba == 1){

    for($m=0; $m<$nivel; ++$m){
        $candtemp[$nc][$m] =
$candidato[$j][$m];
    }
    $candtemp[$nc][$nivel] =
$candidato[$h][$nivel-1];
    ++$nc;
}
}
}

$candidato = $candtemp;
$numcomb = $nc;
$nc = 0;
++$nivel;
}

return $candidato;

} //end function

```

//se incluye la clase de características de productos, que viene representando las preferencias.

```
include("../t_caract_produ.php");  
$t_caract_produ1 = new t_caract_produ();
```

//se incluye la clase de preferencias, que asocia a los clientes con la característica de producto que prefieren.

```
include("../preferencias.php");  
$preferencia1 = new preferencias();
```

//se incluye la clase para graficar resultados.

```
include('../graphs_php/graphs.inc.php');  
$graph = new BAR_GRAPH("vBar");
```

//se genera un listado de las características de productos.

```
$registro_t_c_p = $t_caract_produ1->getT_caract_produ(1);  
$cantidad_t_c_p = $numvariables = mysql_num_rows($registro_t_c_p);
```

//se inicializa el vector para almacenar el número de personas que comparten determinada preferencia y además comparten otra preferencia determinada.

```
$mat = array();  
$numregla = 1;
```

//se genera un listado de los clientes.

```
$registro_per = $preferencia1->getPreferencias3(1);  
$numpersonas = $numcasos = mysql_num_rows($registro_per);
```

?>

```
<div class="service1">
```

```
<b>Número de casos: <?php echo $numcasos; ?></b>
```

```
</div>
```

```
<?php
```

```
//se inicializan las variables nohay se usa para saber si existen reglas o no.
```

```
$nohay = 0;
```

```
$confianza = $_POST["confianza"];
```

```
$cobertura = $_POST["cobertura"];
```

```
$soporte = $cobertura*$numcasos/100;
```

```
//se pasan los datos de la base de datos a la matriz mat, convirtiéndose de  
numéricos a booleanos
```

```
for($i=0; $i < $numcasos; ++$i){
```

```
    for($j=0; $j < $numvariables; ++$j){
```

```
        $registro_pre = $preferencia1->getPreferencias2(1,
```

```
mysql_result($registro_t_c_p,$j,"idt_caract_produ"),
```

```
mysql_result($registro_per,$i,"idpersona"));
```

```
        if(mysql_num_rows($registro_pre)>0 &&  
mysql_result($registro_pre,0,"rating")>4){
```

```
            $mat[$i][$j]=1;
```

```
        } else {
```

```
            $mat[$i][$j]=0;
```

```
        }
```

```
    }
```

```
}
```

```
//una vez se tiene la matriz, se empiezan a hallar las reglas.
```

```
$cov = array();
```

```
$nc = 0;
```

```
$nivel=0;
```

```
//se determina la cobertura para los primeros conjuntos de un item.
```

```
for($j=0; $j<$numvariables; ++$j){
```

```
    $conf[$j]=0;
```

```
    for($i=0; $i<$numcasos; ++$i){
```

```
        if($mat[$i][$j]==1){
```

```
            ++$conf[$j];
```

```
        }
```

```
    }
```

```
    if($conf[$j]>=$soporte){
```

```
        $candtemp[$nc][0] = $j;
```

```
        ++$nc;
```

```
    }
```

```
}
```

```
//La variable nivel representa el máximo de items en un grupo
```

```
$nivel = 1;
```

```
$numcandidatos = $nc;
```

```
$nc = 0;
```

```
$mantiene = 1;
```

```
$candidato = $candtemp;
```

```

//se realizan iteraciones hasta que no se cumpla con la cobertura mínima
requerida
while($numcandidatos>$nivel && $mantiene == 1){

    $mantiene = 0;

    //se comparan todos los grupos de candidatos (grupos) entre sí,
    determinando cuales de ellos comparten items, si no comparten items: aprueba
    sera = 0

    for($j=0; $j<($numcandidatos-1); ++$j){
        for($h=($j+1); $h<$numcandidatos; ++$h){
            $aprueba = 1;
            for($m=1; $m<$nivel; ++$m){
                if($candidato[$j][$m] != $candidato[$h][$m-1]){
                    $aprueba = 0;
                }
            }
            //Luego se determina que los nuevos grupos generados
            cumplan con los requisitos de cobertura.
            if($aprueba == 1){
                $conf=0;
                for($i=0; $i<$numcasos; ++$i){
                    $bandera=1;
                    for($m=0; $m<$nivel; ++$m){
                        if($mat[$i][$candidato[$j][$m]]!=1){
                            $bandera = 0;
                        }
                    }
                }
                if($mat[$i][$candidato[$h][$nivel-1]]==1 &&
                $bandera==1){

```

```

        ++$conf;
    }
}
//Luego se utiliza un candidato temporal para ir
almacenando los datos hasta el final de la iteración.
if($conf>=$soporte){
    $cov[$nc] = $conf;
    for($m=0; $m<$nivel; ++$m){
        $candtemp[$nc][$m] =
$candidato[$j][$m];
    }
    $candtemp[$nc][$nivel] =
$candidato[$h][$nivel-1];
    ++$nc;
    $mantiene = 1;
}
}
}
}
//Los datos pasan del candidato temporal a candidato, para ser
utilizados en la siguiente iteración
$candidato = $candtemp;
$numcandidatos = $nc;
$nc = 0;
++$nivel;
}
--$nivel; //se normaliza la variable nivel, porque se aumenta en 1 por estar
al final del ciclo anterior

```

//Se hallan todas las posibles combinaciones utilizando la función programada combina()

```
for($j=0; $j<$numcandidatos; ++$j){
    for($m=1; $m<=$nivel; ++$m){
        $total = $nivel+1;
        $grupo = $m;
        $matri = combinacion($total, $grupo);
        $numcomb = count($matri);
        for($h=0; $h<$numcomb; ++$h){
            if(count($matri[$h]) == $grupo){
                $covertura = 0;
                for($i=0; $i<$numcasos; ++$i){
                    $bander = 1;
                    for($k=0; $k<$grupo; ++$k){
                        if($mat[$i][$candidato[$j][$matri[$h][$k]]]!=1){
                            $bander = 0;
                        }
                    }
                }
                //se determina que la confianza sea
                aceptada
                if($bander == 1){
                    ++$covertura;
                }
            }
        }
        if(($cov[$j]*100/$covertura) >= $confianza){
            $nohay = 1;
        }
    }
}
```


//si la confianza es aceptada se imprimen

los resultados

?>

```
<div class="service1">
<div class="spacerright"></div>
<div class="contacto1"><div class="contacto">
<p><b>Regla de Asociaci&oacute;n <?php echo $numregla; ++$numregla; ?>: <br
/>(Cobertura: <?php echo $covertura; ?> = <?php echo
round($covertura*10000/$numcasos)/100; ?>%)</b><br />
    El <?php echo round($cov[$j]*10000/$covertura)/100; ?>% de las personas
que presentan preferencia por: <b><?php
    for($k=0; $k<$grupo; ++$k){
        if($k > 0){ echo ", "; }
        echo
mysql_result($registro_t_c_p,($candidato[$j][$matri[$h][$k]]),"nombre");
    }
    ?><br />
</b> Tambi&eacute;n presentan preferencia por: <b><?php
    $paso = 0;
    for($p=0; $p<=$nivel ; ++$p){
        $band = 1;
```

```

        for($k=0; $k<$grupo; ++$k){
            if($matri[$h][$k]==$p){
                $band = 0;
            }
        }
        if($band==1){
            if($paso == 1){ echo ", "; }
            echo
mysql_result($registro_t_c_p,($candidato[$j][$p]),"nombre");
            $paso = 1;
        }
    }
    ?><br />

```

</p>

</div>

<div class="contacto">

<?php

 //se grafica la confianza de la regla, tomando como primera barra los que prefieren solo i, y como segunda barra los que prefieren i y k.

```

    $numped = array();

```

```

    $numped[0]= "100%";//mysql_result($registro_t_c_p,5,"nombre");

```

```

    $numped[1]= round($cov[$j]*10000/$covertura)/100 .

```

```

"%";//mysql_result($registro_t_c_p,4,"nombre");

```

```

    $graph->labels = $numped;

```

```

    $graph->labelSize = 14;

```

```

    $datos = array();

```

```

    $datos[0] = 1;

```

```

    $datos[1] = round($cov[$j]*100/$covertura)/100;//round($porcentaje)/100;

```

```

    $graph->values = $datos;

```

```

        $graph->showValues = 2;
        $graph->legend = "Confianza";
        $graph->barLength = 1.2;
        echo $graph->create();
    ?>
<br />
    <br />
</div>
</div>
</div>
<?php
    }
    }
    }
}
} //cierre de if para comprobar el envío de los parámetros de confianza y cobertura.
?>
<?php
if($nohay==0){?>
<div class="service1">
<div class="spacerright"></div>
<div class="contacto1"><div class="contacto">
<b>No hay reglas de asociaci&ocute;n que cumplan con los par&aacute;metros
de confianza y cobertura seleccionados.</b>
</div></div></div>
<?php } ?>

```

Anexo B. Código Componentes Principales, Método de Covarianza.

```
<?php
```

```
//se incluyen las clases creadas para interactuar con la base de datos MySQL
```

```
include("../t_caract_produ.php");
```

```
$t_caract_produ1 = new t_caract_produ();
```

```
include("../preferencias.php");
```

```
$preferencia1 = new preferencias();
```

```
$registro_t_c_p = $preferencia1->getPreferencias5(1);
```

```
$numvariables = mysql_num_rows($registro_t_c_p);
```

```
$registro_per = $preferencia1->getPreferencias3(1);
```

```
$numcasos = mysql_num_rows($registro_per);
```

```
//se pasan los datos desde la base de datos a la matriz regla, y se normalizan con  
la funcion log
```

```
for($k=0; $k < $numcasos; ++$k){
```

```
    for($i=0; $i < $numvariables; ++$i){
```

```
        $registro_pre = $preferencia1->getPreferencias2(1,  
mysql_result($registro_t_c_p,$i,"t_caract_produ_idt_caract_produ"),  
mysql_result($registro_per,$k,"idpersona"));
```

```
        if(mysql_num_rows($registro_pre)<1){
```

```
            $regla[$k][$i] = log(1);
```

```
        } else {
```

```
            $regla[$k][$i] = log(mysql_result($registro_pre,0,"rating"));
```

```

        }
    }
}

$medias = array();
$covarianzas = array();
//se halla la media de cada variable
for($i=0; $i < $numvariables; ++$i){
    $medias[$i] = 0;
    for($k=0; $k < $numcasos; ++$k){
        $medias[$i] += $regla[$k][$i];
    }
    $medias[$i] = $medias[$i]/$numcasos;
}
//todo esta de la forma array[$caso][$variable]
//y pasa a ser array[$fila][$columna]
//Se halla la Matriz de covarianza aplicando la formula para ello

    for($i=0; $i<$numvariables; ++$i){
        for($j=0; $j<$numvariables; ++$j){
            $covarianzas[$i][$j] = 0;
            for($k=0; $k < $numcasos; ++$k){
                $covarianzas[$i][$j]+=(( $regla[$k][$i]-
$medias[$i])*( $regla[$k][$j]-$medias[$j]))/($numcasos-1);
            }
        }
    }
}

```

//se genera una matriz q inicial con todos sus valores en cero, menos la diagonal principal que los tendra en uno.

```
$ss = array();  
$ss = $covarianzas;  
$q = array();  
for($i=0; $i<$numvariables; ++$i){  
    for($j=0; $j<$numvariables; ++$j){  
        $q[$i][$j]=0;  
        if($i==$j){  
            $q[$i][$j]=1;  
        }  
    }  
}
```

```
$producto = array();  
$iteraciones = 100;
```

//se aplica el método de Jacobi para determinar la matriz de vectores propios y la diagonal principal de valores propios

```
for($h=0; $h<$iteraciones; ++$h){  
  
    $k=0; $l=1;  
    $maximo = abs($covarianzas[$k][1]);  
    for($i=0; $i<$numvariables-1; ++$i){  
        for($j=$i+1; $j<$numvariables; ++$j){  
            if(abs($covarianzas[$i][$j])>$maximo){  
                $k=$i;  
                $l=$j;  
                $maximo=abs($covarianzas[$i][$j]);  
            }  
        }  
    }  
}
```

```

        }
    }
}

$sumsq=0;
for($i=0; $i<$numvariables; ++$i){
    $sumsq+=$covarianzas[$i][$i]*$covarianzas[$i][$i];
}
$tolerancia=0.00000001*sqrt($sumsq)/$numvariables;
if($maximo<$tolerancia){
    break;
}

$p = array();
for($i=0; $i<$numvariables; ++$i){
    for($j=0; $j<$numvariables; ++$j){
        $p[$i][$j]=0;
        if($i==$j){
            $p[$i][$j]=1;
        }
    }
}

$y=$covarianzas[$k][$k]-$covarianzas[$l][$l];
if($y==0){
    $c=$s=sin(pi()/4);
} else {
    $x=2*$covarianzas[$k][$l];
    $z=sqrt($x*$x+$y*$y);

```

```

        $c=sqrt(($z+$y)/(2*$z));
        $s=((($x/$y)/abs($x/$y))*sqrt(($z-$y)/(2*$z)));
    }

    $p[$k][$k]=$c;
    $p[$l][$l]=$c;
    $p[$k][$l]=$s;
    $p[$l][$k]=-s;

    $ptrans = array();
    //se halla la matriz transpuesta de p y se llama ptrans
    for($i=0; $i<$numvariables; ++$i){
        for($j=0; $j<$numvariables; ++$j){
            $ptrans[$j][$i] = $p[$i][$j];
        }
    }

    //se multiplica la matriz inicial a la matriz transpuesta
    for($i=0; $i<$numvariables; ++$i){
        for($j=0; $j<$numvariables; ++$j){
            $sum=0;
            for ($k=0; $k<$numvariables; $k++){
                $sum+=$covarianzas[$i][$k]*$ptrans[$k][$j];
                $producto[$i][$j]=$sum;
            }
        }
    }
}

```


//se multiplica la matriz p a la matriz resultado de la anterior
multiplicacion

```
for($i=0; $i<$numvariables; ++$i){
    for($j=0; $j<$numvariables; ++$j){
        $sum=0;
        for ($k=0; $k<$numvariables; ++$k){
            $sum+=$p[$i][$k]*$producto[$k][$j];
            $covarianzas[$i][$j]=$sum;
        }
    }
}
//se multiplica q a la matriz transpuesta
if($h==0){
    $q = $ptrans;
} else {
    for($i=0; $i<$numvariables; ++$i){
        for($j=0; $j<$numvariables; ++$j){
            $sum=0;
            for ($k=0; $k<$numvariables; ++$k){
                $sum+=$q[$i][$k]*$ptrans[$k][$j];
            }
            $q[$i][$j]=$sum;
        }
    }
}
//fin del método de Jacobi

$sumavalpropios = 0;
```

```

for($i=0; $i<$numvariables; ++$i){
    $sumavalpropios += $covarianzas[$i][$i];
}

$asos = array();

//se multiplica la matriz inicial centrada por la matriz de vectores propios.
for($i=0; $i<$numcasos; ++$i){
    for($j=0; $j<$numvariables; ++$j){
        $sum=0;
        for ($k=0; $k<$numvariables; ++$k){
            $sum+=$(regla[$i][$k]-$medias[$k])*$q[$k][$j];
        }
        $asos[$i][$j]=$sum;
    }
}

//se obtiene la matriz asos que es la matriz de componentes principales

//se inicializa en ceros la matriz de los componentes que se seleccionan
$seleccionados = array();
for($j=0; $j<$numvariables; ++$j){
    $seleccionados[$j] = 0;
}

$suma = 0;

//se seleccionan los componentes principales que cumplan con al menos el
75 por ciento de representatividad de la muestra
for($i=0; $suma<75; ++$i){
    $mayor = 0;

```

```

for($j=0; $j<$numvariables; ++$j){
    if(abs($covarianzas[$j][$j]*100/$sumavalpropios)>$mayor &&
$seleccionados[$j] == 0){
        $mayor = $covarianzas[$j][$j]*100/$sumavalpropios;
        $elegido = $j;
    }
}
$suma += $mayor;
$seleccionados[$elegido] = 1;
}

$numelegidos = $i;

```

//se halla la matriz de correlacion entre variables y componentes, las columnas son componentes

```

$varcomp = array();
for($i=0; $i<$numvariables; ++$i){?>
    <?php
for($j=0; $j<$numvariables; ++$j){ ?>
    <?php

if(abs(round(($q[$i][$j]*sqrt($covarianzas[$j][$j])/sqrt($ss[$i][$i]))*100000)/100000)
> 0){

        $varcomp[$i][$j] =
($q[$i][$j]*sqrt($covarianzas[$j][$j])/sqrt($ss[$i][$i])); }
        else{ $varcomp[$i][$j] = 0; }?>
    <?php } ?>
} <?php } ?>

```

```

<?php
    $selected = array();
    for($h=0; $h<$numvariables; ++$h){
        for($j=0; $j<$numvariables; ++$j){//j es la variable que representa
            $selected[$j][$h] = 0;
        }
    }
    for($h=0; $h<$numvariables; ++$h){//h es el componente principal

        for($i=0; $i<3; ++$i){
            $mayor = 0;
            for($j=0; $j<$numvariables; ++$j){
                if(abs($varcomp[$j][$h])>$mayor && $selected[$j][$h]
== 0){
                    $mayor = $varcomp[$j][$h];
                    $elegido = $j;
                }
            }
            $selected[$elegido][$h] = 1;
        }
    }
?>

```

Anexo C. Algoritmo de K-medias.

```
<?php
session_start();
if (isset($_SESSION["idestabl"])){

//se incluye los componentes principales en la matriz de componentes principales
generada con anterioridad
include_once("promo.php");
?>
<?php
//desde aca van el de k medias
$casos = $numcasos;
$variables=$numelegidos;

//se toma como parámetro de número de grupos el que seleccione el cliente en el
formulario de selección, si no selecciona ninguno, se toman 2 grupos.
if(isset($_POST["numgrupos"])){
    $klusters = $_POST["numgrupos"];
} else {
    $klusters = 2;
}
$iteraciones = 50;

$centroides = array();
$disteuc = array();
$pertenencia = array();
```

//se seleccionan solo los componentes principales que se seleccionaron, (es decir, los que cumplen con al menos el 75%)

```
for($i=0; $i<$casos; ++$i){
    $jcomp = 0;
    for($j=0; $j<$numvariables; ++$j){
        if($seleccionados[$j]==1){
            $regla[$i][$jcomp] = $asos[$i][$j];
            ++$jcomp;
        }
    }
}
```

//se determinan los centroides iniciales

```
for($j=0; $j<$variables; ++$j){
    for($h=0; $h<$klusters; ++$h){
        $centroides[$h][$j] = $regla[$h][$j];
    }
}
```

// los centroides son como casos

// i seran los casos, j las variables y h los clusters

```
for($m=0; $m<$iteraciones; ++$m){
```

//se determina cual es la distancia de cada caso a los diferentes centroides, seleccionando la distancia mínima

```
for($i=0; $i<$casos; ++$i){
    $distmin = 999999999999;
    for($h=0; $h<$klusters; ++$h){
        $disteuc[$i][$h]=0;
```

```

for($j=0; $j<$variables; ++$j){
    $disteuc[$i][$h]+=($regla[$i][$j]-
$centroides[$h][$j])*( $regla[$i][$j]-$centroides[$h][$j]);
}
$disteuc[$i][$h] = sqrt($disteuc[$i][$h]);
if($disteuc[$i][$h]<$distmin){
    $distmin = $disteuc[$i][$h];
}

```

//en la matriz de pertenencia se almacena a que grupo pertenece cada caso.

```

    $pertenencia[$i] = $h;
}
}
}

```

```

$centroidesviejos = $centroides;
$salir = 1;

```

//se hallan los nuevos centroides como el centro de masas de los casos en su grupo

```

for($h=0; $h<$klusters; ++$h){
    for($j=0; $j<$variables; ++$j){
        $centro = 0;
        $totalcasos = 0;
        for($i=0; $i<$casos; ++$i){
            if($pertenencia[$i] == $h){
                $centro += $regla[$i][$j];
                $totalcasos += 1;
            }
        }
    }
}

```

```

    }
    if($totalcasos>0){
        $centroides[$h][$j] = $centro/$totalcasos;
    }
    //si los centroides no cambian se deja de iterar
    if((round($centroidesviejos[$h][$j]*100000000)/100000000) !=
(round($centroides[$h][$j]*100000000)/100000000)){
        $salir = 0;
    }
}
}

if($salir == 1){
    break;
}

$compcluster = array();
//se determina una matriz que indica la relación entre cada variable con
cada cluster, para mas adelante seleccionar las variables de mayor incidencia en
el cluster
for($h=0; $h<$klusters; ++$h){
    for($j=0; $j<$numvariables; ++$j){
        $compcluster[$j][$h] = 0;
        for($i=0; $i<$numcasos; ++$i){
            if($pertenencia[$i]==$h){
                $compcluster[$j][$h] += abs($casos[$i][$j]);
            }
        }
    }
}
}

```



```
}  
  
$grupo = array();  
  
for($j=0; $j<$numvariables; ++$j){  
    $max=0;  
    for($h=0; $h<$clusters; ++$h){  
        if($compcluster[$j][$h]>$mayor){  
            $mayor = $compcluster[$j][$h];  
            $sele = $h;  
        }  
    }  
    $grupo[$j] = $sele;  
}  
}  
  
?>
```

Anexo D. Código para Hallar “Clientes Frecuentes”.

```
<?php
//se incluye la clase de pedidos
include("../pedidos.php");
$pedido1 = new pedidos();
//se incluye la clase de clientes
include("../personas.php");
$persona1 = new personas();

//se genera un listado de los clientes.
$registro_per = $persona1->getPersonas(1);
$cantidad_pedidos = array();

//se inicializa la variable que almacena el número máximo de pedidos que hace un
cliente de el establecimiento.
$maxpedidos = 0;

//se genera un listado de pedidos en el establecimiento.
$registro_ped = $pedido1->getPedidosEstabl($_SESSION["idestabl"]);

//se inicializa el vector que almacena la cantidad de pedidos que hace cada
persona en el establecimiento.
for($j=0; $j<mysql_num_rows($registro_per) ; ++$j){
    $cantidad_pedidos[mysql_result($registro_per, $j, "idpersonas")] = 0;
}

//se recorren los pedidos que se hicieron en el establecimiento.
```

```

for($j=0; $j<mysql_num_rows($registro_ped) ; ++$j){
    ++$cantidad_pedidos[mysql_result($registro_ped, $j,
"personas_idpersonas")]; //se le suma 1 a la cantidad de pedidos que ha realizado
la persona.
    if($cantidad_pedidos[mysql_result($registro_ped, $j,
"personas_idpersonas")]>$maxpedidos){
        //se actualiza el número máximo de pedidos que hace un cliente de el
producto seleccionado
        $maxpedidos = $cantidad_pedidos[mysql_result($registro_ped, $j,
"personas_idpersonas")];
    }
}

```

//se utiliza html para hacer una tabla organizada, definiendo los nombres de cada columna

```
?>
```

```

<div class="creditos">Id</div>
<div class="creditos0">Nombre</div>
<div class="creditos">Frecuencia</div>
<div class="creditos">Veces</div>
<div class="borderwidth"></div>
<?php

```

//se inicializan las variables de frecuencia así como el número total de clientes.

```

$numclientes = mysql_num_rows($registro_per);
$numalta = 0;
$nummedia = 0;
$numbaja = 0;
//ahora se imprimen los resultados obtenidos

```

```
for($i=$maxpedidos; $i>0; --$i){ //permite organizar los
clientes desde los que más han hecho pedidos hasta los que menos.
```

```
for($j=0; $j<mysql_num_rows($registro_per) ;
++$j){//se recorre el listado de clientes.
```

```
if($cantidad_pedidos[mysql_result($registro_per, $j, "idpersonas")]==$i){
//se seleccionan los que han hecho i pedidos. ?>
```

```
<div class="clearall">
```

```
<div class="creditos1"><?php echo mysql_result($registro_per,
$j, "idpersonas"); ?></div>
```

```
<div class="creditos2"><?php echo
strtolower(mysql_result($registro_per, $j, "nombres")."
".mysql_result($registro_per, $j, "apellidos")); ?></div>
```

```
<div class="creditos1"><?php //se le asignan los valores para
frecuencia: 1=baja, 2=media, 3 o más = alta.
```

```
if ($i==1) { ++$numbaja;
```

```
?>Baja<?php }elseif($i==2){ ++$nummedia; ?>Media<?php }else{ ++$numalta;
?>Alta<?php } ?></div>
```

```
<div class="creditos1"><?php echo $i; ?></div>
```

```
<div class="borderwidth1"></div>
```

```
</div>
```

```
<?php } } }?>
```

```
<b></b><br /><br /><br /><br />
```

```
<?php //se grafican los resultados de frecuencia.
```

```
include('../graphs_php/graphs.inc.php');
```

```
$graph = new BAR_GRAPH("vBar");
```

```
$numped = array();
```

```
$numped[0]= "Total";
```

```
$numped[1]= "Baja";
```

```
$numped[2]= "Media";  
$numped[3]= "Alta";  
$graph->labels = $numped;  
$graph->labelSize = 14;  
$datos = array();  
$datos[0] = 1;  
$datos[1] = round(($numbaja/$numclientes) * 100000) / 100000;  
$datos[2] = round(($nummedia/$numclientes) * 100000) / 100000;  
$datos[3] = round(($numalta/$numclientes) * 100000) / 100000;  
$graph->values = $datos;  
$graph->showValues = 2;  
$graph->legend = "Clientes x Frecuencia";  
$graph->barLength = 1.5;  
echo $graph->create();
```

?>

Anexo E. Código para Hallar “Clientes Productos”.

```
<?php
//if else para comprobar el envío de el producto seleccionado.
//$_POST["numvar"] corresponde a el producto seleccionado.
if(!isset($_POST["numvar"]) || (isset($_POST["numvar"]) &&
$_POST["numvar"]=="none")){ ?>
<b>Seleccione un producto para ver su listado de clientes.</b>
<?php } else { // se ponen los encabezados de la tabla?>
<div class="creditos">Id</div>
    <div class="creditos0">Nombre</div>
    <div class="creditos">Frecuencia</div>
    <div class="creditos">Veces</div>
    <div class="borderwidth"></div>
<?php
//se incluye la clase de pedidos
include("../pedidos.php");
$pedido1 = new pedidos();

//se incluye la clase de pedidos
include("../personas.php");
$persona1 = new personas();

//se genera un listado de los clientes.
$registro_per = $persona1->getPersonas(1);
$cantidad_pedidos = array();
```

//se inicializa la variable que almacena el número máximo de pedidos que hace un cliente de el producto seleccionado.

```
$maxpedidos = 0;
```

//se genera un listado de pedidos del producto seleccionado.

```
$registro_ped = $pedido1->getPedidosProd($_POST["numvar"]);
```

//se inicializa el vector que almacena la cantidad de pedidos que hace cada persona del producto seleccionado.

```
for($j=0; $j<mysql_num_rows($registro_per) ; ++$j){  
    $cantidad_pedidos[mysql_result($registro_per, $j, "idpersonas")] = 0;  
}
```

//se recorren los pedidos que se hicieron de el producto seleccionado.

```
for($j=0; $j<mysql_num_rows($registro_ped) ; ++$j){  
    ++$cantidad_pedidos[mysql_result($registro_ped, $j,  
"personas_idpersonas")]; //se le suma 1 a la cantidad de pedidos que ha realizado  
la persona.
```

```
    if($cantidad_pedidos[mysql_result($registro_ped, $j,  
"personas_idpersonas")]>$maxpedidos){
```

//se actualiza el número máximo de pedidos que hace un cliente de el producto seleccionado

```
        $maxpedidos = $cantidad_pedidos[mysql_result($registro_ped, $j,  
"personas_idpersonas")];
```

```
    }  
}
```

//se inicializan las variables de frecuencia así como el número total de clientes.

```
$numclientes = mysql_num_rows($registro_per);
```

```

$numalta = 0;
$nummedia = 0;
$numbaja = 0;

```

//ahora se imprimen los resultados obtenidos

```

for($i=$maxpedidos; $i>0; --$i){ //permite organizar los clientes desde los que más
han hecho pedidos hasta los que menos.

```

```

                for($j=0; $j<mysql_num_rows($registro_per) ;
++$j){//se recorre el listado de clientes.

```

```

                if($cantidad_pedidos[mysql_result($registro_per, $j, "idpersonas")]==$i){//se
seleccionan los que han hecho i pedidos. ?>

```

```

                <div class="clearall">

```

```

                <div class="creditos1"><?php echo mysql_result($registro_per,
$j, "idpersonas"); ?></div>

```

```

                <div class="creditos2"><?php echo
strtolower(mysql_result($registro_per, $j, "nombres")."
".mysql_result($registro_per, $j, "apellidos")); ?></div>

```

```

                <div class="creditos1"><?php //se le asignan los valores para
frecuencia: 1=baja, 2=media, 3 o más = alta.

```

```

                if ($i==1) { ++$numbaja;

```

```

?>Baja<?php }elseif($i==2){ ++$nummedia; ?>Media<?php }else{ ++$numalta;
?>Alta<?php } ?></div>

```

```

                <div class="creditos1"><?php echo $i; ?></div>

```

```

                <div class="borderwidth1"></div>

```

```

                </div>

```

```

                <?php }    }    }?>

```

```

<b></b><br /><br /><br /><br />

```



```
<?php //se grafican los resultados de frecuencia.
include('../graphs_php/graphs.inc.php');
$graph = new BAR_GRAPH("vBar");
$numped = array();
$numped[0]= "Total";
$numped[1]= "Baja";
$numped[2]= "Media";
$numped[3]= "Alta";
$graph->labels = $numped;
$graph->labelSize = 14;
$datos = array();
$datos[0] = 1;
$datos[1] = round(($numbaja/$numclientes) * 100000) / 100000;
$datos[2] = round(($nummedia/$numclientes) * 100000) / 100000;
$datos[3] = round(($numalta/$numclientes) * 100000) / 100000;
$graph->values = $datos;
$graph->showValues = 2;
$graph->legend = "Clientes x Frecuencia";
$graph->barLength = 1.5;
echo $graph->create();
} ?>
```

Anexo F. Resultados SPSS para el Algoritmo de K-medias.

Initial Cluster Centers

	Cluster				
	1	2	3	4	5
crim	.01	28.66	88.98	6.72	.15
zn	75.00	.00	.00	.00	.00
indus	4.00	18.10	18.10	18.10	25.65
nox	.41	.60	.67	.71	.58
rm	5.89	5.16	6.97	6.75	5.86
age	47.60	100.00	91.90	92.60	97.00
dis	7.32	1.59	1.42	2.32	1.94
rad	3.00	24.00	24.00	24.00	2.00
tax	469.00	666.00	666.00	666.00	188.00
ptratio	21.10	20.20	20.20	20.20	19.10
b	396.90	210.97	396.90	.32	370.31
lstat	14.80	20.08	17.21	17.44	25.41
class	18.90	16.30	10.40	13.40	17.30

Cluster Membership

Case Number	Cluster	Distance
1	5	24.751
2	5	46.428
3	5	40.985
4	5	59.684
5	5	59.387
6	5	58.245
7	5	37.702
8	5	55.079
9	5	60.773
10	5	47.393
11	5	54.546
12	5	45.944
13	5	39.581
14	5	37.529
15	5	46.973
16	5	36.821
17	5	44.412
18	5	45.065
19	5	106.898

Case Number	Cluster	Distance
20	5	38.996
21	5	58.871
22	5	49.598
23	5	53.244
24	5	59.176
25	5	54.086
26	5	97.324
27	5	51.840
28	5	95.743
29	5	53.112
30	5	48.407
31	5	61.975
32	5	59.034
33	5	162.888
34	5	62.351
35	5	150.625
36	5	24.820
37	5	22.457
38	5	25.221

Case Number	Cluster	Distance
39	5	31.800
40	5	71.769
41	5	75.372
42	5	70.946
43	5	68.277
44	5	68.466
45	5	49.875
46	5	53.126
47	5	53.446
48	5	57.447
49	5	66.724
50	5	49.257
51	5	36.816
52	5	35.522
53	5	49.403
54	5	49.078
55	1	98.061
56	5	95.566
57	5	80.000

Case Number	Cluster	Distance
58	5	86.869
59	5	29.320
60	5	18.034
61	5	18.980
62	5	42.041
63	5	19.548
64	5	18.672
65	5	61.748
66	5	95.326
67	5	90.934
68	5	77.348
69	5	72.409
70	5	73.087
71	5	59.874
72	5	52.362
73	5	58.961
74	5	60.994
75	1	72.619
76	1	33.414
77	1	13.343
78	1	34.125
79	1	27.769
80	1	46.108
81	5	25.974
82	5	19.043
83	5	26.843
84	5	13.358
85	5	36.651
86	5	35.036
87	5	37.411
88	5	35.616
89	5	37.667
90	5	22.897
91	5	22.756
92	5	27.572
93	5	16.890
94	5	31.958
95	5	28.218
96	5	35.335
97	5	23.912
98	5	32.085

Case Number	Cluster	Distance
99	5	32.935
100	5	23.108
101	1	35.097
102	1	35.156
103	4	261.821
104	1	36.695
105	1	36.668
106	1	41.112
107	1	39.807
108	1	31.984
109	1	41.804
110	1	36.579
111	1	38.093
112	1	37.820
113	1	41.644
114	1	44.049
115	1	34.833
116	1	41.787
117	1	35.388
118	1	37.500
119	1	43.410
120	1	35.370
121	5	93.641
122	5	97.444
123	5	100.552
124	5	104.266
125	5	101.624
126	5	98.278
127	5	106.576
128	1	46.285
129	1	49.638
130	1	48.392
131	1	48.596
132	1	48.882
133	1	44.225
134	1	44.141
135	1	117.074
136	1	48.323
137	1	40.542
138	1	48.107
139	1	48.402

Case Number	Cluster	Distance
140	1	49.465
141	1	45.098
142	1	55.116
143	1	42.451
144	1	41.958
145	1	42.367
146	1	201.706
147	1	204.707
148	1	37.630
149	1	32.684
150	1	33.958
151	1	27.298
152	1	42.314
153	1	35.184
154	1	113.390
155	1	57.264
156	4	244.634
157	4	244.697
158	1	35.170
159	1	35.161
160	1	30.995
161	1	41.012
162	1	36.916
163	1	44.591
164	1	41.537
165	1	32.521
166	1	133.656
167	1	39.266
168	1	144.747
169	1	79.064
170	1	48.781
171	1	83.331
172	1	36.137
173	5	43.623
174	5	39.755
175	5	30.311
176	5	35.481
177	5	28.563
178	5	33.062
179	5	33.066
180	5	86.950

Case Number	Cluster	Distance
181	5	91.340
182	5	86.914
183	5	94.134
184	5	95.155
185	5	92.272
186	5	86.711
187	5	89.207
188	1	55.230
189	1	59.936
190	1	58.514
191	1	66.273
192	1	60.523
193	1	65.256
194	5	64.199
195	5	58.884
196	5	74.921
197	5	85.549
198	5	90.510
199	5	84.209
200	1	107.839
201	1	106.172
202	5	98.340
203	5	106.294
204	5	99.111
205	5	99.741
206	5	39.316
207	5	19.955
208	5	26.642
209	5	20.404
210	5	50.976
211	5	42.026
212	5	41.535
213	5	20.078
214	5	29.856
215	5	66.037
216	5	23.371
217	5	20.575
218	5	35.658
219	5	44.720
220	5	42.019
221	5	48.522

Case Number	Cluster	Distance
222	5	52.382
223	5	41.891
224	5	44.671
225	5	46.602
226	5	51.510
227	5	48.901
228	5	46.238
229	5	57.727
230	5	50.586
231	5	38.867
232	5	43.579
233	5	43.329
234	5	45.744
235	5	46.384
236	5	37.705
237	5	41.144
238	5	39.658
239	5	46.661
240	5	30.292
241	5	27.209
242	5	29.735
243	5	31.010
244	5	56.576
245	5	60.320
246	5	56.920
247	5	57.752
248	5	60.096
249	5	55.766
250	5	66.306
251	5	69.290
252	5	72.339
253	5	73.111
254	5	74.475
255	5	77.452
256	5	82.497
257	5	84.391
258	5	42.273
259	5	47.873
260	5	46.969
261	5	31.157
262	5	40.501

Case Number	Cluster	Distance
263	5	45.007
264	5	42.052
265	5	40.083
266	5	16.875
267	5	33.193
268	5	30.470
269	5	23.798
270	5	54.455
271	5	55.638
272	5	67.225
273	5	54.230
274	5	54.737
275	5	41.024
276	5	36.483
277	5	33.383
278	5	43.633
279	5	40.707
280	5	66.591
281	5	64.698
282	5	64.569
283	5	65.516
284	5	114.368
285	5	81.416
286	5	51.056
287	5	89.162
288	5	46.658
289	5	41.073
290	5	53.124
291	5	75.949
292	5	76.816
293	5	77.593
294	5	44.813
295	5	28.455
296	5	35.158
297	5	24.157
298	5	26.263
299	1	94.674
300	1	100.876
301	1	84.567
302	5	57.761
303	5	66.592

Case Number	Cluster	Distance
304	5	67.453
305	5	59.953
306	5	57.265
307	5	60.405
308	5	59.513
309	5	43.809
310	5	40.351
311	5	54.009
312	5	34.714
313	5	49.077
314	5	43.250
315	5	46.348
316	5	41.694
317	5	44.605
318	5	38.719
319	5	35.761
320	5	34.599
321	5	23.122
322	5	23.023
323	5	24.091
324	5	29.105
325	5	27.734
326	5	46.565
327	5	35.380
328	5	26.111
329	1	56.211
330	1	62.919
331	1	49.919
332	5	43.979
333	5	53.036
334	5	58.577
335	5	58.502
336	5	60.379
337	5	57.437
338	5	56.632
339	5	59.363
340	5	57.646
341	5	56.824
342	5	23.480
343	1	32.756
344	1	66.165

Case Number	Cluster	Distance
345	1	76.671
346	1	62.953
347	1	60.348
348	5	104.402
349	5	67.925
350	5	66.600
351	5	64.804
352	1	64.551
353	1	78.615
354	5	117.234
355	5	92.029
356	5	93.451
357	3	11.504
358	3	11.082
359	3	15.273
360	3	13.919
361	3	17.836
362	3	36.115
363	3	13.650
364	3	33.622
365	3	35.375
366	3	34.882
367	2	23.510
368	4	85.974
369	3	38.492
370	3	37.519
371	3	36.929
372	3	39.844
373	3	50.469
374	3	23.636
375	3	27.152
376	3	17.160
377	3	24.551
378	3	15.863
379	3	19.827
380	3	17.318
381	3	79.079
382	3	17.439
383	3	17.963
384	3	18.153
385	2	18.482

Case Number	Cluster	Distance
386	3	23.023
387	3	24.260
388	3	24.280
389	3	22.852
390	3	16.641
391	3	12.617
392	3	13.288
393	3	17.931
394	3	12.906
395	3	13.719
396	3	12.478
397	3	15.196
398	3	15.845
399	3	36.331
400	2	46.699
401	3	25.887
402	3	19.212
403	3	15.747
404	3	21.342
405	2	47.058
406	3	59.461
407	3	22.784
408	2	40.236
409	2	21.400
410	2	117.311
411	4	65.652
412	4	29.507
413	4	36.261
414	2	86.991
415	4	57.731
416	4	35.470
417	4	37.376
418	4	81.446
419	4	72.197
420	4	26.595
421	2	25.306
422	2	25.529
423	2	10.198
424	4	53.507
425	4	55.402
426	4	48.808

Case Number	Cluster	Distance
427	4	45.513
428	4	46.472
429	4	53.499
430	4	26.658
431	4	40.960
432	4	39.106
433	4	56.186
434	4	55.852
435	4	55.823
436	4	64.188
437	4	32.422
438	4	48.360
439	4	32.628
440	3	14.424
441	3	15.581
442	3	8.274
443	3	15.500
444	3	11.020
445	2	55.825
446	4	24.843
447	2	23.647
448	3	9.521
449	3	15.127
450	2	12.137
451	4	55.472
452	3	32.294
453	3	7.078

Case Number	Cluster	Distance
454	3	14.283
455	4	49.564
456	4	25.008
457	4	46.787
458	4	53.133
459	2	25.465
460	3	13.784
461	2	41.258
462	3	10.152
463	3	14.519
464	3	12.447
465	3	27.671
466	2	59.140
467	4	38.670
468	2	36.585
469	3	25.868
470	3	35.224
471	3	14.697
472	3	13.150
473	3	19.425
474	3	28.911
475	3	34.103
476	2	11.544
477	3	13.157
478	3	38.520
479	3	10.108
480	3	7.405

Case Number	Cluster	Distance
481	3	29.306
482	3	20.925
483	3	21.188
484	3	51.397
485	3	51.236
486	3	39.554
487	3	13.868
488	3	37.848
489	3	50.090
490	3	66.154
491	2	55.642
492	3	50.252
493	3	50.894
494	1	36.772
495	1	43.887
496	1	53.040
497	1	32.516
498	1	31.689
499	1	32.315
500	1	30.874
501	1	32.460
502	5	24.355
503	5	30.604
504	5	41.635
505	5	39.616
506	5	35.978

Final Cluster Centers

	Cluster				
	1	2	3	4	5
crim	.74	11.06	11.13	14.87	.24
zn	9.95	.00	.00	.00	17.82
indus	12.98	18.58	18.56	17.92	6.67
nox	.58	.67	.67	.68	.48
rm	6.19	5.83	6.02	6.07	6.47
age	73.29	91.49	89.78	89.34	55.71
dis	3.33	1.95	2.09	2.02	4.87
rad	4.83	23.00	23.05	22.42	4.31
tax	406.08	668.25	668.14	643.56	276.55
ptratio	17.67	20.19	20.20	19.91	17.87

b	371.66	295.69	385.72	50.15	387.81
lstat	12.71	20.40	17.32	20.48	9.54
class	22.38	15.83	17.92	12.64	25.87

Number of Cases in each Cluster

Cluster	1	98.000
	2	20.000
	3	84.000
	4	36.000
	5	268.000
Valid		506.000
Missing		.000

Anexo G. Resultados Algoritmo de K-medias en PHP.

Matriz de Centroides Iniciales

Vble\Cent	Cent 1	Cent 2	Cent 3	Cent 4	Cent 5
crim	17.8	35.594	53.388	71.182	88.976
zn	20	40	60	80	100
indus	5.916	11.372	16.828	22.284	27.74
nox	0.482	0.579	0.677	0.774	0.871
rm	4.605	5.649	6.692	7.736	8.78
age	22.32	41.74	61.16	80.58	100
dis	3.329	5.528	7.728	9.927	12.127
rad	5.6	10.2	14.8	19.4	24
tax	291.8	396.6	501.4	606.2	711
ptratio	14.48	16.36	18.24	20.12	22
b	79.636	158.952	238.268	317.584	396.9
lstat	8.978	16.226	23.474	30.722	37.97
class	14	23	32	41	50

Tabla de Relación Caso–Grupo.

Id	Grupo
1	2
2	2
3	2
4	2
5	2
6	2
7	2
8	2
9	2
10	2
11	2
12	2
13	2
14	2
15	2
16	2
17	2
18	2
19	2
20	2

Id	Grupo
21	2
22	2
23	2
24	2
25	2
26	2
27	2
28	2
29	2
30	2
31	2
32	2
33	1
34	2
35	1
36	2
37	2
38	2
39	2
40	2

Id	Grupo
41	2
42	2
43	2
44	2
45	2
46	2
47	2
48	2
49	2
50	2
51	2
52	2
53	2
54	2
55	3
56	2
57	2
58	2
59	2
60	2

Id	Grupo
61	2
62	2
63	2
64	2
65	2
66	2
67	2
68	3
69	3
70	3
71	2
72	2
73	2
74	2
75	3
76	3
77	3
78	3
79	3
80	3

Id	Grupo
81	2
82	2
83	2
84	2
85	2
86	2
87	2
88	2
89	2
90	2
91	2
92	2
93	2
94	2
95	2
96	2
97	2
98	2
99	2
100	2
101	3
102	3
103	1
104	3
105	3
106	3
107	3
108	3
109	3
110	3
111	3
112	3
113	3
114	3
115	3
116	3
117	3
118	3
119	3
120	3
121	2
122	2
123	2

Id	Grupo
124	2
125	2
126	2
127	2
128	3
129	3
130	3
131	3
132	3
133	3
134	3
135	1
136	3
137	3
138	3
139	3
140	3
141	3
142	3
143	3
144	3
145	3
146	1
147	1
148	3
149	3
150	3
151	3
152	3
153	3
154	1
155	3
156	1
157	1
158	3
159	3
160	3
161	3
162	3
163	3
164	3
165	3
166	1

Id	Grupo
167	3
168	1
169	3
170	3
171	3
172	3
173	2
174	2
175	2
176	2
177	2
178	2
179	2
180	2
181	2
182	2
183	2
184	2
185	2
186	2
187	2
188	3
189	3
190	3
191	3
192	3
193	3
194	2
195	2
196	2
197	2
198	2
199	2
200	3
201	3
202	3
203	2
204	2
205	2
206	2
207	2
208	2
209	2

Id	Grupo
210	2
211	2
212	2
213	2
214	2
215	2
216	2
217	2
218	2
219	2
220	2
221	2
222	2
223	2
224	2
225	2
226	2
227	2
228	2
229	2
230	2
231	2
232	2
233	2
234	2
235	2
236	2
237	2
238	2
239	2
240	2
241	2
242	2
243	2
244	2
245	2
246	2
247	2
248	2
249	2
250	2
251	2
252	2

Id	Grupo
253	2
254	2
255	2
256	2
257	2
258	2
259	2
260	2
261	2
262	2
263	2
264	2
265	2
266	2
267	2
268	2
269	2
270	2
271	2
272	2
273	2
274	2
275	2
276	2
277	2
278	2
279	2
280	2
281	2
282	2
283	2
284	2
285	2
286	2
287	2
288	2
289	2
290	2
291	2
292	2
293	2
294	2
295	2

Id	Grupo
296	2
297	2
298	2
299	3
300	3
301	3
302	2
303	2
304	2
305	2
306	2
307	2
308	2
309	2
310	2
311	2
312	2
313	2
314	2
315	2
316	2
317	2
318	2
319	2
320	2
321	2
322	2
323	2
324	2
325	2
326	2
327	2
328	2
329	3
330	3
331	3
332	2
333	2
334	2
335	2
336	2
337	2
338	2

Id	Grupo
339	2
340	2
341	2
342	2
343	3
344	3
345	3
346	3
347	3
348	3
349	2
350	2
351	2
352	3
353	3
354	2
355	2
356	2
357	5
358	5
359	5
360	5
361	5
362	5
363	5
364	5
365	5
366	5
367	5
368	4
369	5
370	5
371	5
372	5
373	5
374	5
375	5
376	5
377	5
378	5
379	5
380	5
381	5

Id	Grupo
382	5
383	5
384	5
385	5
386	5
387	5
388	5
389	5
390	5
391	5
392	5
393	5
394	5
395	5
396	5
397	5
398	5
399	5
400	5
401	5
402	5
403	5
404	5
405	5
406	5
407	5
408	5
409	5
410	4
411	4
412	4
413	4
414	4
415	4
416	4
417	4
418	4
419	4
420	4
421	5
422	5
423	5
424	4

Id	Grupo
425	4
426	4
427	4
428	4
429	4
430	4
431	4
432	4
433	4
434	4
435	4
436	4
437	4
438	4
439	4
440	5
441	5
442	5
443	5
444	5
445	5
446	4

Id	Grupo
447	5
448	5
449	5
450	5
451	4
452	5
453	5
454	5
455	4
456	4
457	4
458	4
459	5
460	5
461	5
462	5
463	5
464	5
465	5
466	5
467	4
468	5

Id	Grupo
469	5
470	5
471	5
472	5
473	5
474	5
475	5
476	5
477	5
478	5
479	5
480	5
481	5
482	5
483	5
484	5
485	5
486	5
487	5
488	5
489	5
490	5

Id	Grupo
491	5
492	5
493	5
494	3
495	3
496	3
497	3
498	3
499	3
500	3
501	3
502	2
503	2
504	2
505	2
506	2

Matriz de Centroides Finales

Vble\Cent	Cent 1	Cent 2	Cent 3	Cent 4	Cent 5
crim	1.963	0.235	0.625	16.346	10.911
zn	0	17.51	12.165	0	0
indus	16.708	6.692	12.134	18.1	18.573
nox	0.708	0.484	0.564	0.667	0.671
rm	5.916	6.478	6.2	6.076	5.982
age	91.818	55.911	69.799	90.126	89.914
dis	2.324	4.843	3.6	1.988	2.077
rad	4.727	4.33	4.763	24	23.02
tax	386.909	274.969	402.866	666	668.206
ptratio	17	17.849	17.793	20.2	20.195
b	187.546	388.801	382.11	55.67	371.803
lstat	17.213	9.442	12.271	21.007	17.874
class	17.018	26.045	22.551	12.9	17.429

Número de Casos en cada Grupo

Grupo	# Casos
-------	---------

1	11
2	261
3	97
4	35
5	102

Anexo H. Resultados Exactos Algoritmo de K-medias en PHP

Matriz de Relación Caso-Grupo

Id	Grupo
1	5
2	5
3	5
4	5
5	5
6	5
7	5
8	5
9	5
10	5
11	5
12	5
13	5
14	5
15	5
16	5
17	5
18	5
19	5
20	5
21	5
22	5
23	5
24	5
25	5
26	5
27	5
28	5
29	5
30	5
31	5
32	5
33	5
34	5
35	5
36	5
37	5

Id	Grupo
38	5
39	5
40	5
41	5
42	5
43	5
44	5
45	5
46	5
47	5
48	5
49	5
50	5
51	5
52	5
53	5
54	5
55	1
56	5
57	5
58	5
59	5
60	5
61	5
62	5
63	5
64	5
65	5
66	5
67	5
68	5
69	5
70	5
71	5
72	5
73	5
74	5

Id	Grupo
75	1
76	1
77	1
78	1
79	1
80	1
81	5
82	5
83	5
84	5
85	5
86	5
87	5
88	5
89	5
90	5
91	5
92	5
93	5
94	5
95	5
96	5
97	5
98	5
99	5
100	5
101	1
102	1
103	4
104	1
105	1
106	1
107	1
108	1
109	1
110	1
111	1

Id	Grupo
112	1
113	1
114	1
115	1
116	1
117	1
118	1
119	1
120	1
121	5
122	5
123	5
124	5
125	5
126	5
127	5
128	1
129	1
130	1
131	1
132	1
133	1
134	1
135	1
136	1
137	1
138	1
139	1
140	1
141	1
142	1
143	1
144	1
145	1
146	1
147	1
148	1

Id	Grupo
149	1
150	1
151	1
152	1
153	1
154	1
155	1
156	4
157	4
158	1
159	1
160	1
161	1
162	1
163	1
164	1
165	1
166	1
167	1
168	1
169	1
170	1
171	1
172	1
173	5
174	5
175	5
176	5
177	5
178	5
179	5
180	5
181	5
182	5
183	5
184	5
185	5
186	5
187	5
188	1
189	1
190	1
191	1

Id	Grupo
192	1
193	1
194	5
195	5
196	5
197	5
198	5
199	5
200	1
201	1
202	5
203	5
204	5
205	5
206	5
207	5
208	5
209	5
210	5
211	5
212	5
213	5
214	5
215	5
216	5
217	5
218	5
219	5
220	5
221	5
222	5
223	5
224	5
225	5
226	5
227	5
228	5
229	5
230	5
231	5
232	5
233	5
234	5

Id	Grupo
235	5
236	5
237	5
238	5
239	5
240	5
241	5
242	5
243	5
244	5
245	5
246	5
247	5
248	5
249	5
250	5
251	5
252	5
253	5
254	5
255	5
256	5
257	5
258	5
259	5
260	5
261	5
262	5
263	5
264	5
265	5
266	5
267	5
268	5
269	5
270	5
271	5
272	5
273	5
274	5
275	5
276	5
277	5

Id	Grupo
278	5
279	5
280	5
281	5
282	5
283	5
284	5
285	5
286	5
287	5
288	5
289	5
290	5
291	5
292	5
293	5
294	5
295	5
296	5
297	5
298	5
299	1
300	1
301	1
302	5
303	5
304	5
305	5
306	5
307	5
308	5
309	5
310	5
311	5
312	5
313	5
314	5
315	5
316	5
317	5
318	5
319	5
320	5

Id	Grupo
321	5
322	5
323	5
324	5
325	5
326	5
327	5
328	5
329	1
330	1
331	1
332	5
333	5
334	5
335	5
336	5
337	5
338	5
339	5
340	5
341	5
342	5
343	1
344	1
345	1
346	1
347	1
348	5
349	5
350	5
351	5
352	1
353	1
354	5
355	5
356	5
357	3
358	3
359	3
360	3
361	3
362	3
363	3

Id	Grupo
364	3
365	3
366	3
367	2
368	4
369	3
370	3
371	3
372	3
373	3
374	3
375	3
376	3
377	3
378	3
379	3
380	3
381	3
382	3
383	3
384	3
385	2
386	3
387	3
388	3
389	3
390	3
391	3
392	3
393	3
394	3
395	3
396	3
397	3
398	3
399	3
400	2
401	3
402	3
403	3
404	3
405	2
406	3

Id	Grupo
407	3
408	2
409	2
410	2
411	4
412	4
413	4
414	2
415	4
416	4
417	4
418	4
419	4
420	4
421	2
422	2
423	2
424	4
425	4
426	4
427	4
428	4
429	4
430	4
431	4
432	4
433	4
434	4
435	4
436	4
437	4
438	4
439	4
440	3
441	3
442	3
443	3
444	3
445	2
446	4
447	2
448	3
449	3

Id	Grupo
450	2
451	4
452	3
453	3
454	3
455	4
456	4
457	4
458	4
459	2
460	3
461	2
462	3
463	3
464	3
465	3
466	2
467	4
468	2
469	3
470	3
471	3
472	3
473	3
474	3
475	3
476	2
477	3
478	3
479	3
480	3
481	3
482	3
483	3
484	3
485	3
486	3
487	3
488	3
489	3
490	3
491	2
492	3

Id	Grupo
493	3
494	1
495	1
496	1

Id	Grupo
497	1
498	1
499	1
500	1

Id	Grupo
501	1
502	5
503	5
504	5

Id	Grupo
505	5
506	5

Matriz de Centroides Finales

Vble\Cent	Cent 1	Cent 2	Cent 3	Cent 4	Cent 5
crim	0.741	11.06	11.128	14.867	0.241
zn	9.949	0	0	0	17.817
indus	12.984	18.582	18.559	17.917	6.669
nox	0.582	0.668	0.67	0.678	0.483
rm	6.19	5.832	6.019	6.069	6.465
age	73.289	91.485	89.78	89.344	55.705
dis	3.332	1.947	2.095	2.02	4.874
rad	4.827	23	23.048	22.417	4.313
tax	406.082	668.25	668.143	643.556	276.549
ptratio	17.666	20.195	20.195	19.914	17.873
b	371.664	295.694	385.719	50.154	387.814
lstat	12.715	20.398	17.322	20.478	9.538
class	22.379	15.825	17.918	12.639	25.865

Número de Casos en cada Grupo

Grupo	# Casos
1	98
2	20
3	84
4	36
5	268

Anexo I. Resultados en SPSS para Componentes Principales

Total Variance Explained							
	Component	Initial Eigenvalues(a)			Rotation Sums of Squared Loadings		
		Total	% of Variance	Cumulative %	Total	% of Variance	Cumulative %
Raw	1	30910.013	80.457	80.457	27792.426	72.342	72.342
	2	6250.814	16.271	96.728	9368.401	24.386	96.728
	3	822.463	2.141	98.869			
	4	267.296	.696	99.565			
	5	77.041	.201	99.765			
	6	46.683	.122	99.887			
	7	17.062	.044	99.931			
	8	13.558	.035	99.966			
	9	8.909	.023	99.989			
	10	2.721	.007	99.997			
	11	1.102	.003	99.999			
	12	.218	.001	100.000			
	13	.003	.000	100.000			
Rescaled	1	30910.013	80.457	80.457	4.294	33.027	33.027
	2	6250.814	16.271	96.728	1.447	11.133	44.160
	3	822.463	2.141	98.869			
	4	267.296	.696	99.565			
	5	77.041	.201	99.765			
	6	46.683	.122	99.887			
	7	17.062	.044	99.931			
	8	13.558	.035	99.966			
	9	8.909	.023	99.989			
	10	2.721	.007	99.997			
	11	1.102	.003	99.999			
	12	.218	.001	100.000			
	13	.003	.000	100.000			

Reproduced Covariances

	crim	zn	indus	nox	rm	age	dis	rad	tax	ptratio	b	lstat	class
Reproduced Covariance	26.804 ^b	-39.443	25.445	.407	-1.045	75.491	-5.925	40.374	847.473	4.968	-303.519	21.033	-23.484
zn	-39.443	58.799 ^b	-38.223	-1.006 ^b	1.581	-112.781	8.845	-60.704	-1281.896	-7.571	385.832	-30.912	34.351
indus	25.445	-38.223	24.958 ^b	.394	-1.036	73.408	-5.755	39.659	840.384	4.984	-225.416	19.926	-22.080
nox	.407	-.606	.394	.006 ^b	-0.16	-3.039	-.091	.625	13.192	.078	-4.058	.319	-.355
rm	-1.045	1.581	-1.036	-0.16	.043 ^b	-3.039	238	-1.647	-2.08	8.380	-818	.904	
age	75.491	-112.781	73.408	1.162	-1.647	216.404 ^b	-16.969	116.603	14.574	-193.141	-353.274	663.207	-4.282
dis	-5.925	8.845	-5.755	-.091	1.331 ^b	-16.969	1.331 ^b	-9.140	56.911	-4.642	31.615	-35.021	
rad	40.374	-60.704	39.659	.625	-1.647	116.603	-9.140	63.022 ^b	1335.991	7.927	663.207	16.507 ^b	20.633 ^b
tax	847.473	-1281.896	840.384	13.192	-2.08	2464.743	-193.141	1335.991	28397.104 ^b	7.927	169.041	663.207	-732.982
ptratio	4.968	-7.571	4.984	.078	8.380	14.574	-1.142	7.927	169.041	-1.010 ^b	-35.356	3.885	
b	-303.519	385.832	-225.416	-4.058	-818	-718.707	56.911	-353.274	-6795.900	3.885	8334.207 ^b	-241.344	282.613
lstat	21.033	-30.912	19.926	.319	-818	59.151	-4.642	31.615	663.207	3.885	-241.344	16.507 ^b	-18.439
class	-23.484	34.351	-22.080	-.355	.904	-65.679	5.156	-35.021	-732.982	-4.282	282.613	-18.439	20.633 ^b
Residuals	crim	zn	indus	nox	rm	age	dis	rad	tax	ptratio	b	lstat	class
	-7.73	-7.73	-1.453	.012	-2.80	9.914	-.952	6.473	-2.651	.431	1.137	6.953	-7.234
	-1.453	-47.190	-47.190	-.790	3.532	-261.120	23.785	-2.644	45.442	-12.206	-12.111	-37.871	42.964
	.012	-.790	.213	.213	-.852	51.106	-4.474	-4.109	-7.023	.708	1.836	9.654	-8.441
	-.280	3.532	-.852	-.008	51.106	-1.713	-.097	-.008	-1.46	-.030	.038	.170	-.100
	9.914	-261.120	51.106	1.224	-1.713	-27.360	.065	3.64	4.33	-3.32	-1.65	-2.262	3.580
	-.952	23.785	-4.474	-.097	51.106	-27.360	-27.360	-4.832	-62.052	1.363	15.766	61.927	-31.910
	6.473	-2.644	-4.109	-.008	3.64	-.27.360	.072	.072	3.476	.082	-.871	-2.831	-.316
	-2.651	45.442	-7.023	-.008	-1.46	-62.052	3.476	-.234	-888	-.888	-2.011	-8.493	4.459
	.431	-12.206	.708	-.030	8.33	-62.052	.082	.833	2.97	-.888	2.97	1.898	6.726
	1.137	-12.111	1.836	.038	1.65	15.766	-8.71	-.003	2.677	1.898	2.677	-2.623	-5.829
	6.953	-37.871	9.654	.170	-2.262	61.927	-2.831	-1.230	-30.008	1.898	2.677	-30.008	-30.008
	-7.234	42.964	-8.441	-.100	3.580	-31.910	-.316	4.459	-30.008	-5.829	-2.623	-30.008	-30.008

Anexo J. Resultados Algoritmo de Componentes Principales en PHP

Matriz de covarianza

////	crim	zn	indus	nox	rm	age	dis	rad	tax	ptratio	b	lstat	class
crim	73.98658	-40.21596	23.99234	0.41959	-1.32504	85.40532	-6.87672	46.84776	844.82154	5.39933	-302.38182	27.98617	-30.71851
zn	-40.21596	543.93681	-85.41265	-1.39615	5.11251	-373.90155	32.6293	-63.34869	-1236.45374	-19.77657	373.7214	-68.78304	77.31518
indus	23.99234	-85.41265	47.06444	0.60707	-1.88796	124.5139	-10.2281	35.54997	833.36029	5.6921	-223.57976	29.58027	-30.52082
nox	0.41959	-1.39615	0.60707	0.01343	-0.0246	2.38593	-0.1877	0.61693	13.04629	0.0474	-4.02057	0.48895	-0.45541
rm	-1.32504	5.11251	-1.88796	-0.0246	0.49367	-4.75193	0.30366	-1.28381	-34.58345	-0.54076	8.21501	-3.07974	4.49345
age	85.40532	-373.90155	124.5139	2.38593	-4.75193	792.3584	-44.32938	111.77085	2402.69012	15.93692	-702.94033	121.07772	-97.58902
dis	-6.87672	32.6293	-10.2281	-0.1877	0.30366	-44.32938	4.43402	-9.06825	-189.66459	-1.05977	56.04036	-7.47333	4.84023
rad	46.84776	-63.34869	35.54997	0.61693	-1.28381	111.77085	-9.06825	75.81637	1335.75658	8.76072	-353.27622	30.38544	-30.56123
tax	844.82154	-1236.45374	833.36029	13.04629	-34.58345	2402.69012	-189.66459	1335.75658	28404.75949	168.15314	-6797.91122	654.71452	-726.25572
ptratio	5.39933	-19.77657	5.6921	0.0474	-0.54076	15.93692	-1.05977	8.76072	168.15314	4.68699	-35.05953	5.78273	-10.11066
b	-302.38182	373.7214	-223.57976	-4.02057	8.21501	-702.94033	56.04036	-353.27622	-6797.91122	-35.05953	8334.75226	-238.66752	279.98983
lstat	27.98617	-68.78304	29.58027	0.48895	-3.07974	121.07772	-7.47333	30.38544	654.71452	5.78273	-238.66752	50.99476	-48.44754
class	-30.71851	77.31518	-30.52082	-0.45541	4.49345	-97.58902	4.84023	-30.56123	-726.25572	-10.11066	279.98983	-48.44754	84.58672

Diagonal Principal de valores propios

Diagonal Principal de valores propios	
crim	30910.0128
zn	6250.8142
indus	822.4632
nox	0.0045
rm	77.0409
age	267.2956
dis	17.0548
rad	46.6819
tax	0.222
ptratio	8.9085
b	13.5569
lstat	1.1307
class	2.698

Matriz de Vectores Propios

Vble\Vble	crim	zn	indus	nox	rm	age	dis	rad	tax	ptratio	b	lstat	class
crim	0.0275	0.0002	-0.0001	0	0	0	-0.0001	0	0	0	0.0001	0.0002	0
zn	-0.0372	-0.0002	0.0147	0	0.026	-0.0266	-0.0004	0.0269	0.0296	-0.0015	0.0008	0.0003	0.0018
indus	-0.0283	0.0001	0.0715	0.0001	0.0003	0.0005	0.0092	-0.0005	0.0005	-0.0047	0.0021	0.0029	0.0057
nox	-0.0004	0	-0.0018	1	0	0	0	0	0	0	0	0	0
rm	0.0012	0	0.0049	0	0.0435	-0.0003	-0.0007	0.0004	0.0524	-0.0007	-0.0006	-0.0006	0.0011
age	0.0586	0.0003	-0.0184	0	-0.03	0.0307	0.0002	-0.0311	-0.0342	0.0017	-0.0008	-0.0003	-0.0021
dis	0.0066	-0.0002	0.0446	0.0001	0.036	0.0001	0.0741	-0.0003	0.0001	-0.0041	0.0098	0.0108	-0.0039
rad	-0.0387	-0.0086	0.0147	0	0.0255	-0.0263	-0.0033	0.0266	0.0292	-0.0024	-0.0002	-0.0006	0.0035
tax	-0.6809	0.2809	0.2155	0.0001	0.365	-0.3695	0.0939	0.3719	0.413	0.011	0.0428	0.0383	-0.0324
ptratio	-0.0056	0.0025	-0.0117	0	0.0289	0.03	-0.0082	-0.0331	0.0333	0.103	0.0086	-0.0032	-0.1835
b	0.2088	0.9278	-0.0749	0	-0.085	0.1026	0.3164	-0.1081	-0.1083	0.1092	0.1057	0.1117	-0.1964
lstat	-0.023	-0.0063	-0.0774	-0.0001	0.0434	-0.0419	0.2347	0.0448	0.0458	0.1525	-0.3858	-0.4851	-0.2713
class	0.0256	0.0084	0.0705	0.0001	0.0597	-0.0562	0.3361	0.0599	0.063	0.2023	-0.5163	-0.6494	-0.3636

Porcentaje de varianza total asociada con cada componente

- Comp 1 = 80.46% seleccionado
- Comp 2 = 16.27%
- Comp 3 = 2.14%
- Comp 4 = 0%
- Comp 5 = 0.2%
- Comp 6 = 0.7%
- Comp 7 = 0.04%
- Comp 8 = 0.12%
- Comp 9 = 0%
- Comp 10 = 0.02%
- Comp 11 = 0.04%
- Comp 12 = 0%
- Comp 13 = 0.01%

Correlación entre variables y componentes

Vble\Cor	Comp 1	Comp 2	Comp 3	Comp 4	Comp 5	Comp 6	Comp 7	Comp 8	Comp 9	Comp 10	Comp 11	Comp 12	Comp 13
crim	0.5625	0.0018	-0.0002	0	0	0	0	0	0	0	0.0001	0	0
zn	-0.2806	-0.0006	0.0181	0	0.0098	-0.0187	-0.0001	0.0079	0.0006	-0.0002	0.0001	0	0.0001
indus	-0.7247	0.0011	0.2988	0	0.0004	0.0011	0.0055	-0.0005	0	-0.0021	0.0011	0.0004	0.0014
nox	-0.6822	0	-0.4458	0.5795	-0.0002	0	0	0.0001	0	0	0	0	0
rm	0.2932	-0.0021	0.2016	0	0.5431	-0.0079	-0.0039	0.0035	0.0351	-0.0029	-0.003	-0.0009	0.0025
age	0.3661	0.0008	-0.0188	0	-0.0093	0.0179	0	-0.0075	-0.0006	0.0002	-0.0001	0	-0.0001
dis	0.5496	-0.0079	0.6074	0	0.1501	0.0005	0.1453	-0.001	0	-0.0058	0.0171	0.0055	-0.003
rad	-0.7807	-0.0777	0.0483	0	0.0257	-0.0495	-0.0016	0.0209	0.0016	-0.0008	-0.0001	-0.0001	0.0007
tax	-0.7103	0.1318	0.0367	0	0.019	-0.0358	0.0023	0.0151	0.0012	0.0002	0.0009	0.0002	-0.0003
ptratio	-0.4543	0.0917	-0.1545	0	0.1173	0.2268	-0.0156	-0.1046	0.0073	0.1421	0.0146	-0.0015	-0.1392
b	0.4021	0.8035	-0.0235	0	-0.0082	0.0184	0.0143	-0.0081	-0.0006	0.0036	0.0043	0.0013	-0.0035
lstat	-0.5658	-0.0695	-0.3106	0	0.0534	-0.0958	0.1357	0.0428	0.003	0.0637	-0.1989	-0.0722	-0.0624
class	0.4885	0.0723	0.2199	0	0.057	-0.0998	0.1509	0.0445	0.0032	0.0656	-0.2067	-0.0751	-0.0649

Anexo K. Mapa del Sitio Completo.

Introducción

El “Mapa del Sitio” se divide en 4 sub-mapas que corresponden a una sesión de usuario diferente cada uno incluyendo la página normal, es decir, la sección en la que el que el usuario del sitio no se autentifica con usuario y contraseña para acceder a los contenidos.

Página normal

Inicio (con botones que corresponden a las categorías de establecimientos)

- Lista de establecimientos pertenecientes a la categoría escogida
 - Lista de productos del establecimiento seleccionado

Registrarse

Quiénes somos

Términos de uso

FAQs

Contáctenos

Cientes Registrados

Inicio (con botones que corresponden a las categorías de establecimientos)

- Lista de establecimientos pertenecientes a la categoría escogida
 - Lista de productos del establecimiento seleccionado

Actualizar datos personales

Actualizar datos de cuenta (En caso de ser titular de la cuenta)

Quiénes somos

Términos de uso

FAQs

Contáctenos

Ver estado de pedidos

Carrito de compras

- Compras realizadas

Promociones

- Imprimir promoción

Establecimientos

Inicio

Quiénes somos

Términos de uso

FAQs

Contáctenos

Actualizar datos del establecimiento

Crear promociones

- [Cuantas características va a tener en cuenta](#)
 - [Ventana para escoger las características en orden de importancia y seleccionar el número de personas a las cuales se enviará la promoción.](#)
 - [Resultado: listado de clientes.](#)

Mis clientes (se generarán diferentes listados de clientes del establecimiento)

- [Clientes Frecuentes](#)
- [Clientes por Producto](#)

- Hábitos de Compra²²

Ver estado de cuenta

Sucursales (listado de las sucursales existentes).

- Insertar sucursal.
- Modificar sucursal o eliminar sucursal (Haciendo click sobre el icono de su respectiva operación).

Productos (listado de productos activos en el establecimiento).

- Insertar producto.
- Modificar producto o Eliminar producto (Haciendo click sobre el icono de su respectiva operación).

Administrador

Inicio

Establecimientos (listado de establecimientos existentes)

- Insertar establecimiento.
- Modificar establecimiento o eliminar establecimiento (Haciendo click sobre el icono de su respectiva operación).
- Tipos de Característica para establecimientos (lista).
 - Insertar tipo de característica para establecimientos.
 - Modificar tipo de característica para establecimientos o eliminar tipo de característica para establecimientos (Haciendo click sobre el icono de su respectiva operación).
- Tipos de establecimientos (lista).
 - Insertar un tipo de establecimiento.

²² Nota de pie: La parte azul representa las secciones pertinentes al presente trabajo de grado.

- Modificar un tipo de establecimiento o Eliminar un tipo de establecimiento (Haciendo click sobre el icono de su respectiva operación).

Operadores (listado de operadores registrados)

- Insertar operador.
- Modificar operador o eliminar operador (Haciendo click sobre el icono de su respectiva operación).
- Tipos de Característica para operadores (lista).
 - Insertar tipo de característica para operadores.
 - Modificar tipo de característica para operadores o eliminar tipo de característica para operadores (Haciendo click sobre el icono de su respectiva operación).

Tipos de características

Tipo de características para productos

- Insertar tipo de característica para productos.
- Modificar tipo de característica para productos o eliminar tipo de característica para productos (Haciendo click sobre el icono su respectiva operación).

Tipo de características para cuentas (lista)

- Insertar tipo de característica para cuentas.
- Modificar tipo de característica para cuentas o eliminar tipo de característica para cuentas (Haciendo click sobre el icono su respectiva operación).

Tipo de características para sucursales (lista)

- Insertar tipo de característica para sucursales.
- Modificar tipo de característica para sucursales o eliminar tipo de característica para sucursales (Haciendo click sobre el ícono su respectiva operación).

Tipo de características para contratos (lista)

- Insertar tipo de característica para contratos.
- Modificar tipo de característica para contratos o eliminar tipo de característica para contratos (Haciendo click sobre el ícono su respectiva operación).

Tipo de características para promociones (lista)

- Insertar tipo de característica para promociones.
- Modificar tipo de característica para promociones o eliminar tipo de característica para promociones (Haciendo click sobre el ícono su respectiva operación).

Otras funciones de administrador

Formas de pago (lista de formas de pago)

- Insertar forma de pago.
- Modificar forma de pago o eliminar forma de pago (Haciendo click sobre el ícono su respectiva operación).

Países (listado de países registrados)

- Insertar país.
- Modificar país o eliminar país (Haciendo click sobre el ícono de su respectiva operación).
- Ciudades (listado de ciudades correspondientes a ese país)
 - Insertar ciudad.
 - Modificar ciudad o eliminar ciudad (Haciendo click sobre el ícono de su respectiva operación).