2 El proceso de descubrimiento de conocimiento en bases de datos

The Process of Knowledge Discovery on Databases

Resumen

En este capítulo se describen las etapas del proceso KDD y se hace énfasis en la etapa de minería de datos y en las técnicas más comúnmente utilizadas, como son la clasificación, la asociación, el agrupamiento y los patrones secuenciales. Se detalla además una de las metodologías de referencia más utilizada en el desarrollo de proyectos de minería de datos en los ambientes académico e industrial, como CRISP-DM, que está compuesta por seis fases: análisis del problema, análisis de los datos, preparación de los datos, modelado, evaluación y explotación.

Palabras clave: CRISP-DM, bases de datos, minería de datos, proceso KDD.

Abstract

The stages of the KDD process are described herein, emphasizing the data mining stage and more commonly used techniques, such as classification, association, grouping and sequential patterning. Additionally, one of the most used reference methodologies in the implementation of data mining projects in academic and industrial fields, such as CRISP-DM, is detailed. It consists of six phases: problem analysis, data analysis, data preparation, modeling, assessment and exploitation.

Keywords: CRISP-DM, databases, data mining, KDD process.

¿Cómo citar este capítulo?/How to cite this chapter?

Timarán-Pereira, S. R., Hernández-Arteaga, I., Caicedo-Zambrano, S. J., Hidalgo-Troya, A. y Alvarado-Pérez, J. C. (2016). El proceso de descubrimiento de conocimiento en bases de datos. En *Descubrimiento de patrones de desempeño académico con árboles de decisión en las competencias genéricas de la formación profesional* (pp. 63-86). Bogotá: Ediciones Universidad Cooperativa de Colombia. DOI: http://dx.doi.org/10.16925/9789587600490

Introducción

El proceso de extraer conocimiento a partir de grandes volúmenes de datos ha sido reconocido por muchos investigadores como un tópico de investigación clave en los sistemas de bases de datos, y por muchas compañías industriales como una importante área y una oportunidad para obtener mayores ganancias (Timarán, 2009). Autores como Fayyad, Piatetsky-Shapiro y Smith (1996, p. 89) lo definen como "El proceso no trivial de identificación de patrones válidos, novedosos, potencialmente útiles y fundamentalmente entendibles al usuario a partir de los datos".

El Descubrimiento de conocimiento en bases de datos (KDD, del inglés *Knowledge Discovery in Databases*) es básicamente un proceso automático en el que se combinan descubrimiento y análisis. El proceso consiste en extraer patrones en forma de reglas o funciones, a partir de los datos, para que el usuario los analice. Esta tarea implica generalmente preprocesar los datos, hacer minería de datos (*data mining*) y presentar resultados (Agrawal y Srikant, 1994) (Chen, Han y Yu, 1996) (Piatetsky Shapiro, Brachman y Khabaza, 1996) (Han y Kamber, 2001). KDD se puede aplicar en diferentes dominios, por ejemplo, para determinar perfiles de clientes fraudulentos (evasión de impuestos), para descubrir relaciones implícitas existentes entre síntomas y enfermedades, entre características técnicas y diagnóstico del estado de equipos y máquinas, para determinar perfiles de estudiantes "académicamente exitosos" en términos de sus características socioeconómicas y para determinar patrones de compra de los clientes en sus canastas de mercado.

Etapas del proceso KDD

El proceso KDD que se muestra en la figura 1 es interactivo e iterativo, involucra numerosos pasos con la intervención del usuario en la toma de muchas decisiones. Se resume en las siguientes etapas:

- Selección.
- Preprocesamiento/limpieza.
- Transformación/reducción.
- Minería de datos (*data mining*).
- Interpretación/evaluación.

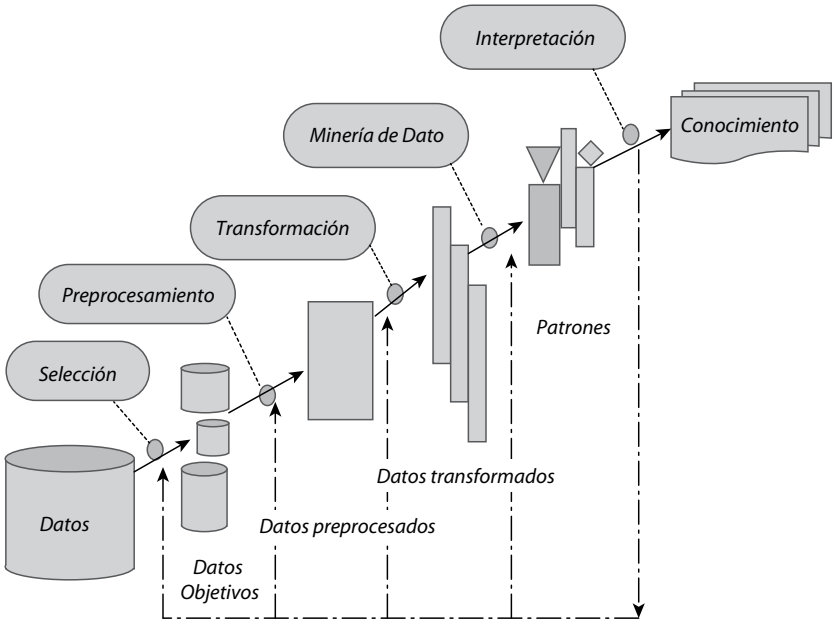

Figura 1. Etapas del proceso KDD. Elaboración propia.

Etapa de selección

En la etapa de selección, una vez identificado el conocimiento relevante y prioritario y definidas las metas del proceso KDD, desde el punto de vista del usuario final, se crea un conjunto de datos objetivo, seleccionando todo el conjunto de datos o una muestra representativa de este, sobre el cual se realiza el proceso de descubrimiento. La selección de los datos varía de acuerdo con los objetivos del negocio.

Etapa de pre-procesamiento/limpieza

En la etapa de preprocesamiento/limpieza (*data cleaning*) se analiza la calidad de los datos, se aplican operaciones básicas como la remoción de datos ruidosos, se seleccionan estrategias para el manejo de datos desconocidos (*missing* y *empty*), datos nulos, datos duplicados y técnicas estadísticas para su reemplazo. En esta etapa, es de suma importancia la interacción con el usuario o analista.

Los datos ruidosos (*noisy data*) son valores que están significativamente fuera del rango de valores esperados; se deben principalmente a errores humanos, a cambios en el sistema, a información no disponible a tiempo y a fuentes

heterogéneas de datos. Los datos desconocidos *empty* son aquellos a los cuales no les corresponde un valor en el mundo real y los *missing* son aquellos que tienen un valor que no fue capturado. Los datos nulos son datos desconocidos que son permitidos por los sistemas gestores de bases de datos relacionales (SGBDR). En el proceso de limpieza todos estos valores se ignoran, se reemplazan por un valor por omisión, o por el valor más cercano, es decir, se usan métricas de tipo estadístico como media, moda, mínimo y máximo para reemplazarlos.

Etapa de transformación/reducción

En la etapa de transformación/reducción de datos, se buscan características útiles para representar los datos dependiendo de la meta del proceso. Se utilizan métodos de reducción de dimensiones o de transformación para disminuir el número efectivo de variables bajo consideración o para encontrar representaciones invariantes de los datos (Fayyad et al., 1996).

Los métodos de reducción de dimensiones pueden simplificar una tabla de una base de datos horizontal o verticalmente. La reducción horizontal implica la eliminación de tuplas idénticas como producto de la sustitución del valor de un atributo por otro de alto nivel, en una jerarquía definida de valores categóricos o por la discretización de valores continuos (por ejemplo, edad por un rango de edades). La reducción vertical implica la eliminación de atributos que son insignificantes o redundantes con respecto al problema, como la eliminación de llaves, la eliminación de columnas que dependen funcionalmente (por ejemplo, edad y fecha de nacimiento). Se utilizan técnicas de reducción como agregaciones, compresión de datos, histogramas, segmentación, discretización basada en entropía, muestreo, entre otras (Han y Kamber, 2001).

Etapa de minería de datos

El objetivo de la etapa minería de datos es la búsqueda y descubrimiento de patrones insospechados y de interés, aplicando tareas de descubrimiento como clasificación (Quinlan, 1986) (Wang, Iyer y Scott, 1998), clustering (Ng y Han, 1994), (Zhang, Ramakrishnan, Livny, 1996), patrones secuenciales (Agrawal y Srikant, 1995) y asociaciones (Agrawal y Srikant, 1994), (Srikant y Agrawal, 1996), entre otras.

Las técnicas de minería de datos crean modelos que son predictivos o descriptivos. Los modelos predictivos pretenden estimar valores futuros o desconocidos

de variables de interés, que se denominan variables objetivo, dependientes o clases, usando otras variables denominadas independientes o predictivas, como por ejemplo predecir para nuevos clientes si son buenos o malos basados en su estado civil, edad, género y profesión, o determinar para nuevos estudiantes si desertan o no en función de su zona de procedencia, facultad, estrato, género, edad y promedio de notas. Entre las tareas predictivas están la clasificación y la regresión. Los modelos descriptivos identifican patrones que explican o resumen los datos; sirven para explorar las propiedades de los datos examinados, no para predecir nuevos datos, como identificar grupos de personas con gustos similares o identificar patrones de compra de clientes en una determinada zona de la ciudad. Entre las tareas descriptivas se cuentan las reglas de asociación, los patrones secuenciales, los *clustering* y las correlaciones.

Por lo tanto, la escogencia de un algoritmo de minería de datos incluye la selección de los métodos por aplicar en la búsqueda de patrones en los datos, así como la decisión sobre los modelos y los parámetros más apropiados, dependiendo del tipo de datos (categóricos, numéricos) por utilizar.

Etapa de interpretación/evaluación de datos

En la etapa de interpretación/evaluación, se interpretan los patrones descubiertos y posiblemente se retorna a las anteriores etapas para posteriores iteraciones. Esta etapa puede incluir la visualización de los patrones extraídos, la remoción de los patrones redundantes o irrelevantes y la traducción de los patrones útiles en términos que sean entendibles para el usuario. Por otra parte, se consolida el conocimiento descubierto para incorporarlo en otro sistema para posteriores acciones o, simplemente, para documentarlo y reportarlo a las partes interesadas; también para verificar y resolver conflictos potenciales con el conocimiento previamente descubierto.

Tareas de minería de datos

Dentro de la minería de datos se encuentran diferentes tipos de tareas, las cuales pueden considerarse como un tipo de problema para ser resuelto por un algoritmo de minería de datos (Hernández, Ramírez y Ferri, 2005). Entre las tareas de minería de datos más importantes están la clasificación, segmentación o *clustering*, asociación y patrones secuenciales.

Clasificación

La clasificación de datos permite obtener resultados a partir de un proceso de aprendizaje supervisado. Es, además, el proceso por medio del cual se encuentran propiedades comunes entre un conjunto de objetos de una base de datos y se los cataloga en diferentes clases, de acuerdo con el modelo de clasificación (Agrawal, Ghosh, Imielinsky, Iyer y Swami, 1992).

Este proceso se realiza en dos pasos: en el primero se construye un modelo, en el cual cada tupla de un conjunto de tuplas de la base de datos tiene una clase conocida (etiqueta), determinada por uno de los atributos de la base de datos llamado *atributo clase*. El conjunto de tuplas que sirve para construir el modelo se denomina *conjunto de entrenamiento* y se escoge randómicamente del total de tuplas de la base de datos. A cada tupla de este conjunto se denomina *ejemplo de entrenamiento* (Han y Kamber, 2001). En el segundo paso se usa el modelo para clasificar. Inicialmente, se estima la exactitud del modelo utilizando otro conjunto de tuplas de la base de datos, cuya clase es conocida, denominado *conjunto de prueba*. Este conjunto es escogido randómicamente y es independiente del conjunto de entrenamiento. A cada tupla de este conjunto se denomina *ejemplo de prueba* (Han y Kamber, 2001).

La exactitud del modelo, sobre el conjunto de prueba, es el porcentaje de ejemplos de prueba que son correctamente clasificadas por el modelo. Si la exactitud del modelo se considera aceptable, se puede usar para clasificar futuros datos o tuplas para los cuales no se conoce la clase a la que pertenecen. Se han propuesto varios métodos de clasificación: *rough sets*, árboles de decisión, redes neuronales, Bayes, algoritmos genéticos entre otros.

El modelo de clasificación basado en árboles de decisión es probablemente el más utilizado y popular por su simplicidad y facilidad para entender (Han y Kamber, 2001), (Sattler y Dunemann, 2001). Este modelo tiene su origen en los estudios de aprendizaje de máquina. Este es un método de aprendizaje supervisado que construye árboles de decisión a partir de un conjunto de casos o ejemplos denominados *conjunto de entrenamiento* (*training set*) extraídos de la base de datos. También se escoge un conjunto de prueba, cuyas características son conocidas, con el fin de evaluar el árbol.

La calidad del árbol depende de la precisión de la clasificación y del tamaño del árbol (Chen, Han y Yu, 1996). El método primero escoge un subconjunto del conjunto de entrenamiento y forma un árbol de decisión. Si el árbol no da la respuesta correcta para todos los objetos del conjunto prueba, una selección

de excepciones se adiciona al conjunto de entrenamiento y el proceso continúa hasta que se encuentra el conjunto de decisiones correctas. El eventual resultado es un árbol en el cual cada hoja lleva un nombre de la clase y cada nodo interior especifica un atributo con una rama correspondiente a cada posible valor del atributo.

Entre los algoritmos de clasificación para árboles de decisión se cuentan ID-3 (Quinlan, 1986), C4.5 (Quinlan, 1993), Sprint (Shafer, Agrawal y Metha, 1996), SLIQ (Metha, Agrawal y Rissanen, 1996) y J48 (Hall, Frank y Witten, 2011). La idea básica de estos algoritmos es construir los árboles de decisión en los que:

- Cada nodo no terminal está etiquetado con un atributo.
- Cada rama que sale de un nodo está etiquetada con un valor de ese atributo.
- Cada nodo terminal está etiquetado con un conjunto de casos, los cuales satisfacen todos los valores de atributos que etiquetan el camino desde ese nodo al nodo inicial.

La aplicación de un atributo *A* como criterio de selección clasifica los casos en distintos conjuntos (tantos como valores discretos del atributo). Se trata de construir el árbol de decisión más simple que sea consistente con el conjunto de entrenamiento T. Para ello hay que ordenar los atributos relevantes, desde la raíz a los nodos terminales, de mayor a menor poder de clasificación. El poder de clasificación de un atributo *A* es su capacidad para generar particiones del conjunto de entrenamiento que se ajuste en un grado dado a las distintas clases posibles; de esta forma se introduce un orden en dicho conjunto. El orden o el desorden (ruido) de un conjunto de datos son medibles. El poder de clasificación de un atributo se mide de acuerdo con su capacidad para reducir la incertidumbre o entropía (grado de desorden de un sistema). Esta métrica se denomina *ganancia de información*. El atributo con la más alta ganancia de información se escoge como el atributo que forme un nodo en el árbol (Quinlan, 1993) (Agrawal et al., 1992).

El árbol de decisión se construye de la siguiente forma:

- Calcular la entropía que puede reducir cada atributo.
- Ordenar los atributos de mayor a menor capacidad de reducción de entropía.
- Construir el árbol de decisión siguiendo la lista ordenada de atributos.

La ganancia de información obtenida por el particionamiento del conjunto T, de acuerdo con el atributo *A* se define como:

$$Gain(T, A) = I(T) - E(A)$$

Donde, *I(T)* es la entropía del conjunto *T*, compuesto de *s* ejemplos y *m* distintas clases *Ci (i=1,m)* y se calcula:

$$I(T) = - \Sigma\, p_i \log_2 (p_i)$$

Donde, $p_i = s_i/s$ es la probabilidad que un ejemplo cualquiera pertenezca a una clase C_i y si es el número de ejemplos de *T* de la clase C_i.

E(A) es la entropía del conjunto *T* si es particionado por los *n* diferentes valores del atributo *A* en *n* subconjuntos, $\{S_1, S_2, \ldots S_n\}$, donde S_j contiene esos ejemplos de *T* que tienen el valor a_j en *A* y s_{ij} el número de ejemplos de la clase C_i en el subconjunto S_j.

E(A) se calcula:

$$E(A) = \Sigma\, s_{ij} / s * I(S_{ij})$$

Donde, s_{ij} el número de ejemplos de la clase C_i en el subconjunto S_j

$$I(S_{ij}) = -\Sigma\, p_{ij} \log_2 (p_{ij})$$

Donde $p_{ij} = s_{ij} / |s_j|$ es la probabilidad de que un ejemplo de S_j pertenezca a la clase C_i.

En otras palabras, *Gain (T, A)* es la reducción esperada de la entropía causada por el particionamiento de *T* de acuerdo con el atributo *A*.

Finalmente, las reglas de clasificación se obtienen recorriendo cada rama del árbol desde la raíz hasta el nodo terminal. El antecedente de la regla es la conjunción de los pares recogidos en cada nodo y el consecuente es el nodo terminal.

Segmentación o clustering

El proceso de agrupar objetos físicos o abstractos en clases de objetos similares se llama segmentación o *clustering* o clasificación no supervisada (Chen, Han y Yu,

1996). Básicamente, el *clustering* agrupa un conjunto de datos (sin un atributo de clase predefinido) basado en el principio de maximizar la similitud intraclase y minimizar la similitud interclase. El análisis de *clustering* ayuda a construir particiones significativas de un gran conjunto de objetos basado en la metodología divide y conquista, la cual descompone un sistema de gran escala en pequeños componentes para simplificar el diseño y la implementación.

La meta del *clustering* en una base de datos es la partición de esta en segmentos o *clusters* de registros similares que comparten un número de propiedades y son considerados homogéneos. Los registros en diversos *clusters* son diferentes y estos últimos tienen una alta homogeneidad interna (dentro del *cluster*) y una alta heterogeneidad externa (entre *clusters*). Por homogeneidad se entiende que los registros en un *cluster* están próximos unos a otros; allí la proximidad se expresa por medio de una medida, dependiendo de la distancia de los registros al centro del segmento. Por heterogeneidad se entiende que los registros en diferentes segmentos no son similares de acuerdo con una medida de similaridad (Cabena, Hadjinian, Stadler, Verhees y Zanasi, 1998).

La segmentación, típicamente, permite descubrir subpoblaciones homogéneas: por ejemplo, se aplica a una base de datos de clientes, para mejorar la exactitud de los perfiles, identificando subgrupos de clientes que tienen un comportamiento similar al comprar.

El algoritmo de *clustering* segmenta una base de datos sin ninguna indicación por parte del usuario sobre el tipo de *clusters* que va a encontrar en la base de datos, y desecha cualquier sesgo o intuición por parte del usuario; así potencia el verdadero descubrimiento de conocimiento. Por esta razón, al método de segmentación o *clustering* se lo denomina *aprendizaje no supervisado*. Algunos de los algoritmos utilizados para *clustering* son: K-Means (Han y Kamber, 2001), Clarans (*Clustering Large Applications based upon Randomized Search*) (Ng y Han, 1994), y Birch (*Balanced Iterative Reducing and Clustering using Hierarchies*) (Zhang, Ramakrishnan y Livny, 1996).

El *clustering* se utiliza por ejemplo en el análisis de flujo de efectivo para un grupo de clientes que paga en un período del mes en particular, para hacer segmentación de mercado y para descubrir grupos de afinidades. También se utiliza para descubrir subpoblaciones homogéneas de consumidores en bases de datos de *marketing*.

Asociación

La tarea de asociación descubre patrones en forma de reglas, que muestran los hechos que ocurren frecuentemente juntos en un conjunto de datos determinado. El problema fue formulado por Agrawal et al. (1992), y a menudo se referencia como el problema de canasta de mercado (*market-basket*). En este problema se da un conjunto de ítems y una colección de transacciones que son subconjuntos (canastas) de estos ítems. La tarea es encontrar relaciones entre los ítems de esas canastas para descubrir reglas de asociación que cumplan unas especificaciones mínimas dadas por el usuario, expresadas en forma de soporte y confianza. Las cantidades de ítems comprados en una transacción no se toman en cuenta, lo que significa que cada ítem es una variable binaria que representa si un ítem está presente o no en una transacción.

Formalmente, sea I={i1, i2, ...,im} un conjunto de literales, llamados ítems; sea D un conjunto de transacciones, donde cada transacción T es un conjunto de ítems tal que $T \subseteq I$. Cada transacción se asocia con un identificador llamado TID. Sea X un conjunto de ítems. Se dice que una transacción T contiene a X si y solo si $X \subseteq T$.

Una regla de asociación es una implicación de la forma $X \Rightarrow Y$, donde X y Y son conjuntos de ítems que $X \subset I$, $Y \subset I$ y $X \cap Y = \phi$.

El significado intuitivo de tal regla es que las transacciones de la base de datos que contienen X tienden a contener Y. La regla $X \Rightarrow Y$ se cumple en el conjunto de transacciones D con una confianza c si el c% de las transacciones en D que contienen X también contienen Y. La regla $X \Rightarrow Y$ tiene un soporte *s* en el conjunto de transacciones D si el *s*% de las transacciones en D contienen $X \cup Y$.

La confianza denota la fuerza de la implicación y el soporte indica la frecuencia de ocurrencia de los patrones en la regla. Las reglas con una confianza alta y soporte fuerte son referidas como reglas fuertes (*strong rules*) (Agrawal et al., 1992). El problema de encontrar reglas de asociación se descompone en los siguientes pasos:

- Descubrir los *itemsets* frecuentes, i.e., el conjunto de ítems que tienen el soporte de transacciones por encima de un predeterminado soporte *s* mínimo.
- Usar los *itemsets* frecuentes para generar las reglas de asociación para la base de datos.

Después de que los *itemsets* frecuentes son identificados, las correspondientes reglas de asociación se pueden derivar de una manera directa. Un ejemplo de una regla de asociación es "el 30% de las transacciones que contienen cerveza también contienen pañales; el 2% de todas las transacciones contienen a ambos ítems" (Agrawal et al., 1996, p. 244). Aquí el 30% es la confianza de la regla y el 2%, el soporte de la regla.

Según Han y Kamber (2001), existen varios criterios para clasificar las reglas de asociación, uno de estos es el de las dimensiones que estas abarcan. De acuerdo con este criterio, las reglas de asociación pueden ser unidimensionales y multidimensionales. Una regla de asociación es unidimensional, si los ítems o atributos de la regla hacen referencia a un solo predicado o dimensión. Por ejemplo, se tiene la siguiente regla de asociación:

Cerveza ^ papas fritas => pañales, que se puede reescribir como:

Compra (cerveza) ^ compra (papas fritas) => compra (pañales), hace referencia a una sola dimensión: compra.

Una regla de asociación es multidimensional, si los ítems o atributos de la regla hacen referencia a dos o más criterios o dimensiones. Por ejemplo, está la siguiente regla de asociación:

Edad (30...39) ^ ocupación (ingeniero) =>compra (laptop), contiene tres predicados*: edad, ocupación y compra.*

Un uso clásico de asociaciones es el análisis de la canasta de mercado, en la cual la asociación es una lista de afinidades de productos. Por ejemplo, observar los pedidos individuales de clientes para suministros de oficina puede generar una regla: el 70% de los clientes que ordenan plumas y lápices también ordenan libretas.

Otras aplicaciones de reglas de asociación son los análisis de demandas médicas para determinar procedimientos médicos que se realizan al mismo tiempo o a lo largo de un período, para un diagnóstico en particular. También se aplican para el análisis de textos, diseño de catálogos, segmentación de clientes basado en patrones de compra, en mercadeo, entre otros.

Patrones secuenciales

Los patrones secuenciales buscan ocurrencias cronológicas. El problema de descubrimiento de patrones secuenciales se trata en Agrawal y Srikant (1995). Se aplica principalmente en el análisis de la canasta de mercado y su objetivo es descubrir en los clientes ciertos comportamientos de compra en el tiempo. El dato de entrada es un conjunto de secuencias llamado *data-secuencia*. Cada una de estas últimas es una lista de transacciones, en las que cada transacción es un conjunto de ítems (literales). Típicamente, hay un tiempo asociado con cada transacción.

Un patrón secuencial también se compone de una lista de conjuntos de ítems. El problema es encontrar todos los patrones secuenciales que cumplan con un soporte mínimo especificado por el usuario, en el cual el soporte es el porcentaje de *data-secuencias* que contiene el patrón. Por ejemplo, en una base de datos de una librería, cada *data-secuencia* puede corresponder a todas las selecciones de libros de un cliente y cada transacción, a los libros seleccionados por el cliente en una orden.

Un patrón secuencial puede ser "El 5% de los clientes que compran '*Foundation*', después '*Foundation y Empire*' y luego '*Second Foundation*' (Agrawal y Srikant, 1995, p. 3). La *data-secuencia* correspondiente al cliente, quien compró otros libros conjuntamente o después de estos libros, contiene este patrón secuencial. La *data-secuencia* puede también tener otros libros en la misma transacción, así como uno de los libros del patrón. Elementos de un patrón secuencial pueden ser conjuntos de ítems; por ejemplo, "'*Foundation*' y '*Ringworld*', seguido por '*Foundation* y *Empire*' y '*Ringworld Engineers*'". Sin embargo, todos los ítems en un elemento de un patrón secuencial deben estar presentes en una transacción simple para que la *data-secuencia* soporte al patrón (Agrawal y Srikant, 1995).

Los patrones secuenciales, en el dominio de la medicina, se pueden utilizar por ejemplo para ayudar a identificar síntomas y enfermedades que preceden a otras enfermedades.

Áreas relacionadas con el proceso KDD

KDD se ha desarrollado y continúa desarrollándose con base en las investigaciones realizadas en los campos del aprendizaje de máquina, reconocimiento de patrones, bases de datos, estadística, inteligencia artificial, sistemas expertos, visualización

de datos y computación de alto rendimiento. La meta común es la extracción de conocimiento de los datos en el contexto de grandes bases de datos.

KDD se relaciona con aprendizaje de máquina y reconocimiento de patrones en el estudio de teorías y algoritmos de minería de datos para modelamiento de datos y extracción de patrones. Así mismo, se enfoca en la extensión de estas teorías y algoritmos al problema de encontrar patrones entendibles que puedan ser interpretados como conocimiento útil o interesante, y hace un fuerte énfasis sobre el trabajo con grandes conjuntos de datos del mundo real.

KDD tiene que ver con la estadística, particularmente con el análisis exploratorio de datos. La inferencia de conocimiento a partir de los datos tiene un componente estadístico fundamental (Elder y Pregibon, 1996). La estadística provee un lenguaje y una estructura para cuantificar el grado de certeza de los resultados cuando se trata de inferir patrones generales sobre una muestra particular de toda una población.

Data warehousing es otra área con la que se relaciona KDD y se refiere a la actual tendencia de los negocios de coleccionar y limpiar datos transaccionales con el fin de que se encuentren disponibles para el análisis en línea y el soporte de decisiones. Un popular método para el análisis de bodegas de datos (*data warehouse*) es OLAP (*On-line Analytical Processing*) (Gill y Rao, 1996). Las herramientas de OLAP se enfocan en proveer análisis de datos multidimensional y están destinadas hacia la simplificación y soporte interactivo de análisis de datos, mientras que el objetivo de las herramientas de DCBD es automatizar el proceso, tanto como sea posible.

Metodología CRISP-DM

Generalidades

En 1993, líderes de la industria como Daimler Benz, SPSS de Inglaterra, OHRA de Holanda, NCR de Dinamarca, consorcio de empresas Europeas, y AG de Alemania construyeron el acrónimo CRISP-DM (*Cross- Industry Standard Process for Data Mining*), el cual tiene como finalidad proporcionar nuevas ideas a los que decidan trabajar con minería de datos. Esta metodología tiene la ventaja de que no ha sido construida de manera teórica y académica, sino que se basa en experiencias reales de cómo la gente hace proyectos (Moro, Laureano y Cortez, 2011) (Martínez y Podestá, 2014) (Raus, Vegega, Pytel y Pollo-Cattaneo, 2014).

Este modelo es uno de los más utilizados como guía de referencia en el desarrollo de proyectos de minería de datos. La metodología CRISP-DM consiste en un conjunto de tareas que están organizadas en cuatro niveles de abstracción: fases, tareas generales, tareas especializadas e instancias de proceso (véase figura 2). Dichos niveles están establecidos respetando jerarquías en tareas; inician en el nivel más general hasta llegar, finalmente, a los casos más específicos (Chapman et al., 2000).

Ciclo de vida de la metodología CRISP-DM

El modelo provee una representación completa del ciclo de vida de un proyecto de minería de datos. El proceso es dinámico e iterativo, por lo que la ejecución de los procesos no es estricta y con frecuencia se puede pasar de uno a otro proceso, de atrás hacia delante y viceversa. Estos dependen del resultado de cada fase o la planeación de la siguiente tarea por ejecutar.

Estas fases ayudan a las organizaciones a entender el proceso y proveen de un "mapa del camino" que se debe seguir, así: conocimiento del negocio, conocimiento de los datos, preparación de los datos, modelado, evaluación, despliegue (véase figura 3).

Cada fase se estructura en varias tareas generales, las tareas generales se proyectan en tareas específicas, en las cuales finalmente se describen las acciones que deben ser desarrolladas para situaciones definidas (Larose y Larose, 2014).

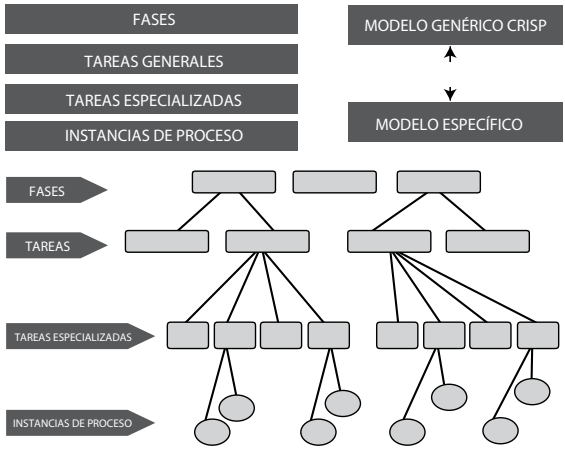

Figura 2. Esquema de los cuatro niveles de CRISP-DM. Tomado de *CRISP-DM 1.0 Step-by-Step Data Mining Guide* (P. Chapman et al.), 2000.

Fases de la metodología

Fase 1. Comprensión del negocio o problema. Comprende los requisitos y objetivos del proyecto desde una perspectiva empresarial o institucional para convertirlos en objetivos técnicos y en un plan de proyecto, para lo cual es necesario comprender de manera completa el problema por resolver (ver figura 4).

- **Determinar los objetivos.** Se determina cuál es el problema que se quiere resolver y por qué se usa minería de datos para dicho propósito; también se deben fijar los criterios de éxito. En cuanto a estos últimos, pueden ser de tipo cualitativo o de tipo cuantitativo; por ejemplo, si el problema es detectar fraude en el uso de tarjetas de crédito, el criterio de éxito cuantitativo sería el número de detecciones de fraude.
- **Evaluar la situación actual.** En esta tarea se evalúan antecedentes y requisitos del problema, tanto en términos del negocio como en términos de la minería de datos. Algunos de los aspectos por tener en cuenta pueden ser el conocimiento previo acerca del tema, la cantidad de datos requeridos para resolver el problema, ventajas de aplicar minería de datos al problema, entre otros.

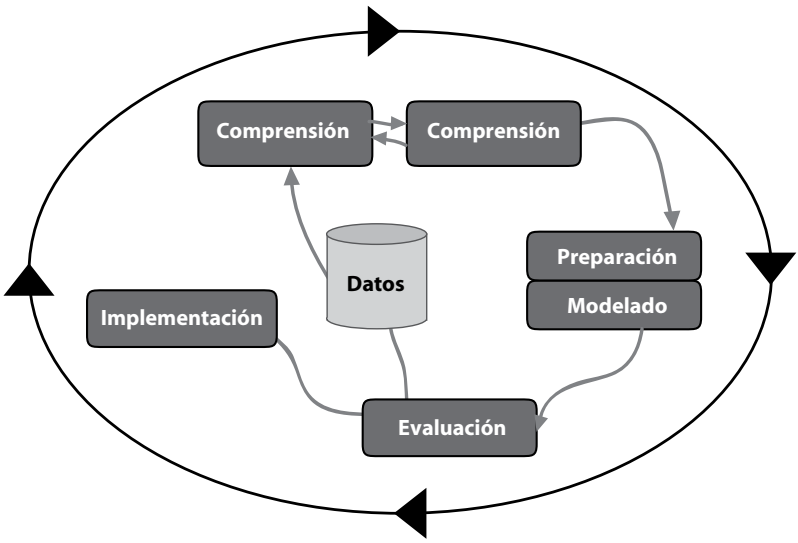

Figura 3. Ciclo de vida de CRISP-DM. Tomado de *CRISP-DM 1.0 Step-by-Step Data Mining Guide* (P. Chapman et al.), 2000.

• **Determinar los objetivos de la minería de datos.** El objetivo de esta tarea es representar los objetivos del negocio en términos de las metas del proyecto de minería de datos. Por ejemplo, si el objetivo del negocio es el desarrollo de una campaña publicitaria para incrementar asignación de créditos hipotecarios, la meta de la minería de datos sería determinar el perfil de los clientes respecto a su capacidad de endeudamiento.

• **Producir un plan de proyecto**. La última tarea de esta fase tiene como objetivo desarrollar el plan de proyecto considerando los pasos que se deben seguir y los métodos por emplear en cada paso.

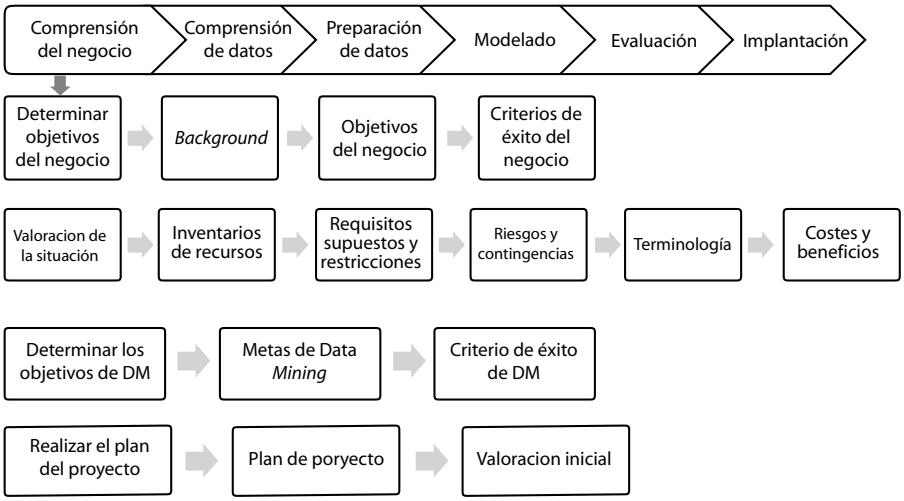

Figura 4. Fase de comprensión del negocio. Tomado de *CRISP-DM 1.0 Step-by-Step Data Mining Guide* (P. Chapman et al.), 2000.

Fase 2. Comprensión de los datos. Corresponde a la recolección inicial de los datos para establecer un primer contacto con el problema; esta fase, junto con la fase 3 y la fase 4, demanda mayor esfuerzo y tiempo (véase figura 5).

Las principales tareas que se deben desarrollar en la fase de comprensión de los datos son: recolectar datos iniciales, describir los datos, explorar los datos y verificar la calidad de los datos.

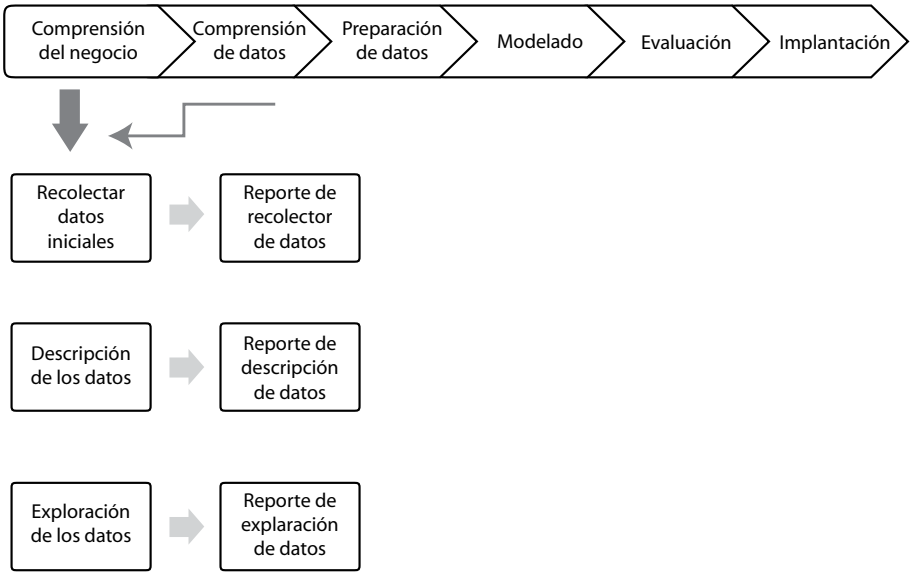

Figura 5. Fase de comprensión de los datos. Tomado de *CRISP-DM 1.0 Step-by-Step Data Mining Guide* (P. Chapman *et al.*), 2000.

• **Recolectar datos iniciales.** Tiene como objetivo principal la recolección de datos iniciales y su adecuación para su posterior procesamiento. Se deben elaborar informes con una lista de los datos adquiridos, su localización, las técnicas utilizadas en su recolección y los problemas y soluciones inherentes a este proceso.

• **Describir los datos.** Se deben describir los datos iniciales obtenidos, tales como número de registros y campos por registro, su identificación, el significado de cada campo y la descripción del formato inicial.

• **Explorar los datos.** Su finalidad es descubrir una estructura general para los datos. Involucra la aplicación de pruebas estadísticas básicas, que revelen propiedades en los datos, se crean tablas de frecuencia y se construyen gráficos de distribución. Se crea un informe de exploración de datos.

• **Verificar la calidad de los datos.** Se realiza la verificación de los datos para determinar la consistencia de los valores de los campos, la cantidad y distribución de los valores nulos, encontrar valores fuera de rango que pueden

ser ruido para el proceso. Se tiene como objetivo asegurar la completitud y corrección de los datos.

Fase 3. Preparación de los datos. Se usa para adaptarlos a la técnica de minería de datos, mediante la visualización de los datos y la búsqueda de relaciones entre las variables. Esta fase es la de modelado, ya que los datos requieren ser procesados de diferentes formas; por ende, las fases de preparación y modelado interactúan permanentemente (véase figura 6).

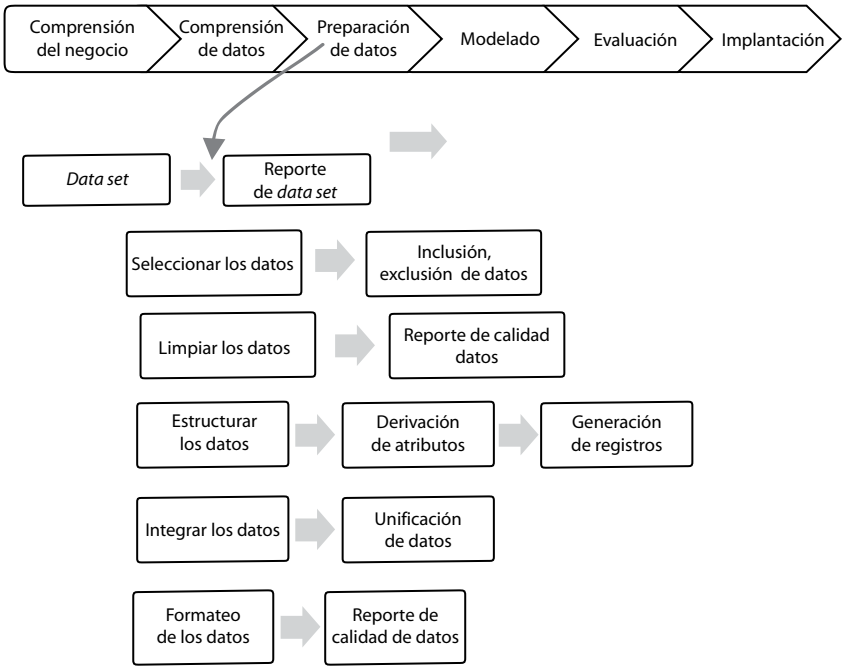

Figura 6. Fase de preparación de los datos. Tomado de *CRISP-DM 1.0 Step-by-Step Data Mining Guide* (P. Chapman et al.), 2000.

Los pasos que se consideran para la preparación de los datos son: seleccionar, limpiar, estructurar, integrar y formatear los datos.

- **Seleccionar los datos.** Se selecciona un subconjunto de datos considerando la calidad de los datos, la limitación en el volumen o en los tipos de datos que están relacionados con las técnicas de minería de datos seleccionados.

- **Limpiar los datos.** Existe una diversidad de técnicas aplicables a esta tarea con el fin de optimizar la calidad de los datos en la perspectiva de prepararlos para la fase de modelación. Algunas de las técnicas son: normalización de los datos, discretización de campos numéricos, tratamiento con valores vacíos, reducción del volumen de datos.
- **Estructurar los datos.** Algunas de las operaciones por realizar en esta tarea son la generación de nuevos atributos a partir de atributos ya existentes, integración de nuevos registros o transformación de valores para atributos existentes.
- **Integrar los datos.** Involucra la creación de nuevas estructuras; por ejemplo, crear nuevos campos, nuevos registros, fusión de tablas o nuevas tablas.
- **Formatear los datos.** Consiste principalmente en transformar sintácticamente los datos sin modificar su significado con el fin de permitir o facilitar, en particular, el empleo de alguna técnica de minería de datos; por ejemplo, eliminar comas, tabuladores, caracteres especiales, espacios, máximos y mínimos para las cadenas de caracteres, etc.

Fase 4. Modelado. Corresponde a la selección de un modelo adecuado y específico; para ello se usan técnicas que cumplan los siguientes criterios (véase figura 7):

- Ser apropiada para el problema.
- Disponer de datos adecuados.
- Cumplir con los requisitos del problema.
- Técnica adecuada para obtener un modelo.
- Conocimiento pleno de la técnica.

Por ejemplo, si el problema es de clasificación, podemos elegir entre árboles de decisión, *k-nearest neighbour* o razonamiento basado en caos (CBR).

- **Generar plan de prueba.** Se debe generar un plan para probar la calidad y validez del modelo construido; por ejemplo, en una tarea como la clasificación es posible usar la razón de error como medida de la calidad. Entonces, típicamente se separan los datos en dos conjuntos, uno de entrenamiento y otro de prueba.
- **Construir el modelo.** Se ejecuta la técnica seleccionada sobre los datos preparados para generar uno o más modelos. Todas las técnicas del modelado tienen un conjunto de parámetros que determinan características del modelo

por generar. La tarea de selección de los mejores parámetros es iterativa, basada en los resultados generados. Estos deben ser interpretados y su rendimiento justificado.

• **Evaluar el modelo.** Se deben interpretar los modelos de acuerdo con el conocimiento del dominio y los criterios de éxitos preestablecidos.

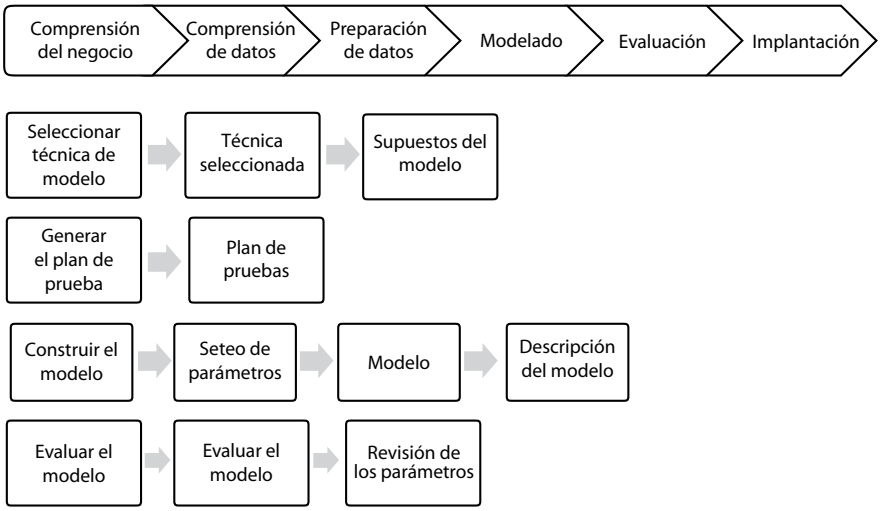

Figura 7. Fase de modelado. Tomado de *CRISP-DM 1.0 Step-by-Step Data Mining Guide* (P. Chapman et al.), 2000.

Fase 5. Evaluación. Evalúa el modelo teniendo en cuenta el cumplimiento de los criterios de éxito del problema; para ello se emplean múltiples herramientas para la interpretación de los resultados, entre ellas matrices de confusión Edelstein 1999, que es una tabla que indica cuántas clasificaciones se han hecho para cada tipo. La diagonal de la tabla representa las clasificaciones correctas (figura 8).

Si es válido lo anterior, se procede a la explotación del modelo, que es el mantenimiento de la aplicación y la posible difusión de los resultados.

Una vez que el modelo ha sido construido y validado, se transforma el conocimiento obtenido en acciones dentro del proceso de negocio; la retroalimentación generada por la monitorización y mantenimiento puede indicar si el modelo está siendo utilizado apropiadamente.

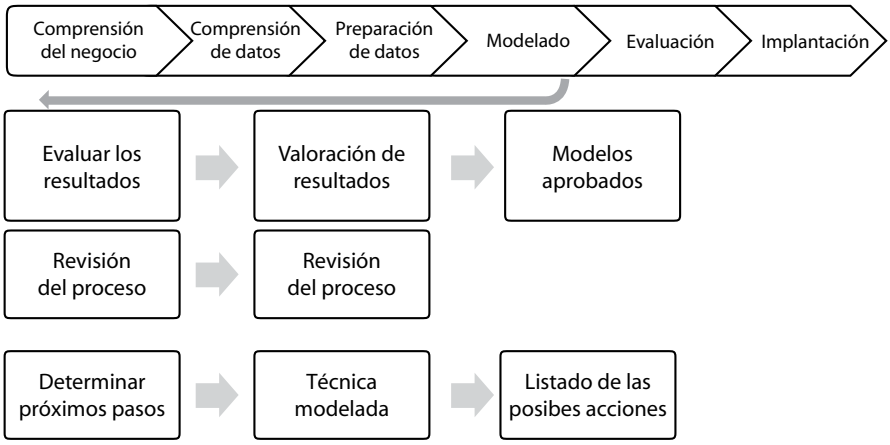

Figura 8. Fase de evaluación. Tomado de *CRISP-DM 1.0 Step-by-Step Data Mining Guide* (P. Chapman et al.), 2000.

Fase 6. Implementación. Es aquí donde el conocimiento obtenido se transforma en acciones dentro del proceso de negocio, ya sea observando el modelo y resultados, o aplicándolo a múltiples grupos de datos o como parte del proceso. Las tareas que se efectúan son: planear la implementación, monitorizar y mantener, informe final y revisar el proyecto.

- **Planear la implementación.** Esta tarea toma los resultados de la evaluación y concluye una estrategia para su implementación. Si un procedimiento general se ha identificado para crear el modelo, debe estar documentado para su posterior implementación (véase figura 9).
- **Monitorizar y mantener.** Se deben preparar estrategias de monitorización y mantenimiento para ser aplicadas sobre los modelos.
- **Informe final.** Dependiendo del plan de implementación, este puede ser un resumen de los puntos importantes del proyecto y la experiencia lograda, o puede ser una presentación final que incluya y explique los resultados logrados con el proyecto.
- **Revisar el proyecto.** Se evalúa lo correcto y lo incorrecto.

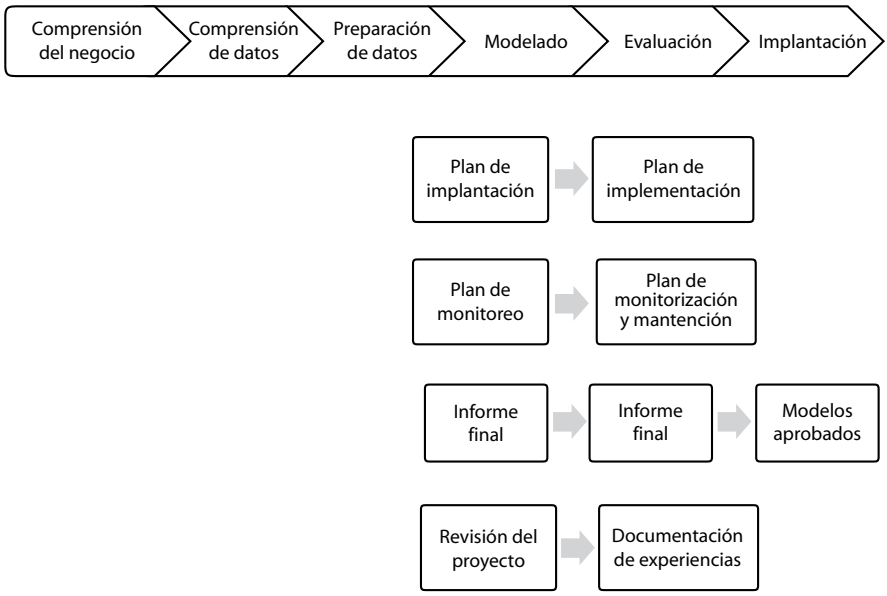

Figura 9. Fase de implementación. Tomado de *CRISP-DM 1.0 Step-by-Step Data Mining Guide* (P. Chapman et al.), 2000.

En este capítulo se describieron los conceptos fundamentales del proceso de descubrimiento de conocimiento en bases de datos, haciendo énfasis en la etapa de minería de datos donde se especificaron las tareas y técnicas de minería de datos más importantes. De igual forma se describió la metodología CRISP-DM que sirvió de base para esta investigación.

Referencias

Agrawal, R. y Srikant, R. (1994). Fast Algorithms for Mining Association Rules. *VLDB Conference*, Santiago de Chile.

Agrawal, R. y Srikant, R. (1995). Mining Sequential Patterns. *The 11th International Conference on Data Engineering ICDE*, Taipei, República de China.

Agrawal, R., Ghosh S., Imielinski, T., Iyer, B. y Swami, A. (1992). An Interval Classifier for Database Mining Applications. *Proceedings* VDLB *Conference*, Vancouver.

Cabena, P., Hadjinian, P., Stadler, R., Verhees, J. y Zanasi A. (1998). *Discovering Data Mining from Concept to Implementation,* Prentice Hall. Recuperado de http://dl.acm.org/citation.cfm?id=270298

Chapman, P., Clinton, J., Randy, K., Khabaza, T., Reinartz, T., Shearer, C. y Wirth, R. (2000). CRISP-DM 1.0 Step-by-Step Data Mining Guide. Recuperado de http://www.crisp-dm.org/CRISPWP-0800.pdf

Chen, M., Han, J. y Yu, P. (1996). Data Mining: An Overview from Database Perspective. *IEEE Transactions on Knowledge and Data Engineering*.

Elder, J. y Pregibon, D. (1996). A Statistical Perspective on Knowledge Discovery in Databases. En *Advances in Knowledge Discovery and Data Mining, AAAI Pres/ The MIT Press*.

Fayyad, U., Piatestky-Shapiro, G. y Smyth, P. (1996). The KDD Process for Extracting Useful Knowledge from Volumes of Data. *Comunications of the ACM*, *39*(11), 27-34.

Gill, H. y Rao, P. (1996). *Data warehousing: la integración de información para la mejor toma de decisiones*. Prentice-Hall.

Hall, M., Frank, E., y Witten, I. (2011). Practical Data Mining: Tutorials. University of Waikato. Disponible en: www.micai.org/2012/tutorials/Weka%20tutorials%20Spanish.pdf

Han, J. y Kamber, M. (2001). *Data Mining Concepts and Techniques*. San Francisco: Morgan Kaufmann Publishers.

Hernández, J., Ramírez, M. y Ferri, C. (2005). *Introducción a la Minería de Datos*. Madrid: Editorial Pearson Educación SA.

Larose, D. y Larose, Ch. (2014). *Discovering Knowledge in Data: An Introduction to Data Mining* (2da. ed.). New Jersey: John Wiley & Sons.

Martínez, D, y Podestá, C. (2014). Metodología de estudio del rendimiento académico mediante la minería de datos. *Campus Virtuales, 3*(1), 56-73.

Metha M., Agrawal R., Rissanen J. (1996). SLIQ: A Fast Scalable Classifier for Data Mining. *Proceedings EDBT96*. Avignon, France.

Moro, S., Laureano, R. y Cortez, P. (2011). Using Data Mining for Bank Direct Marketing: An Application of the CRISP-DM Methodology. *Proceedings of European Simulation and Modelling Conference -ESM'2011*, 117–121. Recuperado de http://sci2s.ugr.es/docencia/in/pdf/MoroCortezLaureano_DMApproach4DirectMKT.pdf

Ng, R. y Han, J. (1994). Efficient and Effective Clustering Method for Spatial Data Mining. *VLDB Conference*. Santiago de Chile, Chile.

Piatetsky-Shapiro, G., Brachman, R. y Khabaza, T. (1996). *An Overview of Issues in Developing Industrial Data Mining and Knowledge Discovery Applications*. Association for the

Advancement of Artificial Intelligence [AAAI], MIT Press. Recuperado de http://www.aaai.org/Papers/KDD/1996/KDD96-015.pdf

Quinlan, J. (1986). Induction of Decision Trees. *Machine Learning Journa*l, *1*(1), 81-106.

Raus, N., Vegega, C., Pytel, P. y Pollo-Cattaneo, M. (2014). Metodología propuesta para la predicción de deserción universitaria mediante explotación de información (pp. 1014-1158). *WICC 2014 XVI Worshop de investigadores en ciencias de la computación*. Recuperado de http://sedici.unlp.edu.ar/bitstream/handle/10915/43835/Documento_completo.pdf?sequence=1

Srikant R. y Agrawal, R. (1996). *Mining quantitative association rules in large relational tables, ACM SIGMOD, Montreal*. Recuperado de http://rakesh.agrawal-family.com/papers/sigmod-96qassoc.pdf

Sattler, K. y Dunemann, O. (2001). SQL database primitives for decision tree classifiers. En *Proceedings of the tenth international conference on Information and knowledge management* (pp. 379-386). Atlanta: CIKM. Recuperado de http://dl.acm.org/citation.cfm?id=502650

Shafer J., Agrawal R., Metha M. (1996). SPRINT: A Scalable Parallel Classifier for Data Mining. *Proceedings of the VLDB Conference*. Bombay, India.

Timarán, R. (2009). Una mirada al descubrimiento de conocimiento en bases de datos. *Revista Ventana Informática*, *20*, 39-58.

Wang, M., Iyer, B. y Scott, J. (1998). Scalable Mining for Classification Rules in Relational Databases. I*nternational Database Engineering and Application Symposium - Ideas*. Cardiff, Wales.

Zhang, T., Ramakrishnan, R. y Livny, M. (1996). BIRCH: An Efficient Data Clustering Method for Very Large Databases. *ACM SIGMOD International Conference on Management of Data*. Montreal, Canadá.