UNIVERSIDAD NACIONAL AUTÓNOMA DE NICARAGUA

UNAN-LEÓN

FACULTAD DE CIENCIAS Y TECNOLOGÍAS

TESIS PARA OPTAR AL TÍTULO DE INGENIERO (A) EN SISTEMAS DE INFORMACIÓN

TEMA: PROPUESTA DE PLAN DOCENTE PARA LA ASIGNATURA ELECTIVA VIII DE LA CARRERA DE INGENIERÌA EN SISTEMAS DE INFORMACIÓN:

INTRODUCCIÓN A LA MINERÍA DE DATOS

AUTORES:

- Br. Jahaslan Ramiro Alvarado Cárcamo
- Br. Katheline Yanixa Reyes Ortiz
- Br. Brenda Carolina Ríos Díaz

TUTORES:

- Msc. Aldo René Martínez Delgadillo
- Msc. W. Milton Carvajal Herradora

León-Nicaragua, abril 2016

Dedicatoria:

*Dedicamos este trabajo monográfico a **Dios,** nuestro padre del cielo.*

*A mi madre, **Paula Cárcamo Loàisiga**, que sigue siendo mi símbolo de paciencia y humildad y a una amiga especial por su apoyo moral **Cedilia Obando**. (Jahaslan Ramiro Alvarado Cárcamo)*

*A mis **padres** que han sido mi apoyo y fortaleza para concluir mis estudios, a mi **esposo** por el apoyo brindado en esta etapa de mi carrera. (Katheline Yanixa Reyes Ortiz).*

*A mi madre, **Thelma Díaz Guevara**, quien ha sido el principal apoyo de mi vida y quien ha sido mi ejemplo de fuerza y superación. (Brenda Carolina Ríos Díaz).*

Agradecimiento:

Agradecemos a Dios por habernos regalado la vida para prepararnos y poder culminar nuestros sueños profesionales.

A nuestras familias que han sido nuestro apoyo económico, moral y de superación para ver cumplida nuestra meta.

A nuestros tutores Msc. Aldo René Martínez y Msc. W. Milton Carvajal Herradora, por su tiempo y dedicación, guiándonos hasta el final de nuestra tesis.

ÍNDICE DEL DOCUMENTO

Contenido **Página**

ÍNDICE DE ILUSTRACIÓNES

Contenido **Página**

ÍNDICE TABLAS

Contenido **Página**

1) INTRODUCCIÓN

Este documento proyecta ser el plan docente para la asignatura electiva VIII: Introducción a la Minería de Datos, asignatura que se impartiría dentro del marco de la carrera de Ingeniería en Sistemas de Información, el objetivo principal de esta asignatura es proporcionar a los alumnos los fundamentos para que sean capaces de preparar datos, aplicar técnicas de Minería de Datos y elementos disponibles cuya meta es asistir en la toma de decisiones. Centrándose en los siguientes aspectos:

- Conocimiento de los procesos de Minería de Datos y herramientas aplicadas.
- Conocimiento de metodologías utilizadas para Minería de Datos.
- Utilización y análisis de dichas herramientas para la toma de decisiones.

Este documento presenta una división de los temas teóricos que deben impartirse en dicha asignatura así como propuestas a las prácticas de laboratorios que deben de realizarse para cumplir con los objetivos. También se plantea la situación actual de la asignatura, la relación con otras asignaturas, la metodología a utilizar y la planificación temporal.

2) ANTECEDENTES

En general, la Estadística es la primera ciencia que históricamente extrae información de los datos básicamente mediante metodologías procedentes de las matemáticas.

Cuando se empezó a usar los ordenadores como apoyo para esta tarea surgió el concepto de Machine Learning traducido como aprendizaje automático. Posteriormente con el incremento de tamaño y la estructuración de los datos es cuando se empieza a hablar de Minería de Datos.

La idea de Data Mining no es nueva. Ya desde los años sesenta los estadísticos manejaban términos como Data Fishing, Data Mining o Data Archaeology con la idea de encontrar correlaciones.

A principios de los años ochenta, Rakesh Agrawal, Gio Wiederhold, Robert Blum y Gregory Piatetsky-Shapiro, entre otros, empezaron a consolidar los términos de Data Mining que a lo largo de la historia ha sido llamado de distintas maneras.

Antes de esto existió otro término que era Database Mining TM, pero esta frase fue registrada por la empresa HNC, y por ese motivo los investigadores decidieron cambiarlo por Data Mining que es el término que más se usa actualmente.

Cabe mencionar que en el Dpto. de computación de la UNAN-León no existen trabajos monográficos relacionados con Minería de Datos, únicamente se cuenta con documentación diseñada por el Dpto. de Matemática y Estadística de la Facultad de Ciencias.

3) JUSTIFICACIÓN

De acuerdo con las tendencias actuales en formación universitaria los objetivos de esta propuesta de plan docente se dirige a la adquisición de competencias profesionales de carácter teórico y práctico, principalmente mediante el uso de métodos y herramientas de trabajo sobre Minería de Datos.

Esta propuesta ha sido diseñada para estudiantes que requieren una inmersión rápida en los principales conceptos, usos y herramientas de Minería de Datos.

4) OBJETIVOS

Los objetivos del presente documento son:

General:

- Dotar al Departamento de Computación de un documento que contenga el contenido programático (teórico y práctico) del componente Curricular: Electiva VIII Introducción a la Minería de Datos.

Específicos:

- Diseñar el plan docente del componente curricular de la electiva Minería de Datos de tal forma que sirva de base para la preparación de cada una de las conferencias tanto teóricas como las de los laboratorios.

- Proporcionar al estudiante un documento de apoyo, el cual será el soporte fundamental de cada una de las conferencias teóricas y prácticas de laboratorio.

- Proponer los enunciados de las prácticas de los laboratorios que se deben desarrollar, en las que el estudiante ponga en práctica los conocimientos adquiridos en la teoría.

PARTE I: Información de la Asignatura

5) SITUACIÓN ACTUAL DE LA ASIGNATURA.

La electiva VIII Introducción a Data Mining (Minería de Datos) es una asignatura que actualmente no se imparte en la carrera de: INGENIERÍA EN SISTEMA DE INFORMACIÓN, pero sí en las carreras de Licenciatura en Matemática e Ingeniería en Estadística; todas en la Facultad de Ciencias y Tecnología de la UNAN-León

La carrera en si se desarrollara dentro de un periodo de 10 semestre que corresponden a 5 años académico/ lectivos. La electiva se impartiría durante el IX semestre (que corresponde al primer semestre de 5to año).

La asignatura consta con un total de 4 créditos académicos y un total de 60 horas las cuales incluyen tanto tiempo de teoría como de laboratorio distribuidas a los largo de 16 semanas lectivas pero considerando los días festivos, asumimos 15 semanas netas. A continuación se presenta la división de las horas entre la teoría y la práctica:

	Horas	Semanas	Horas Semestrales
Teoría	2	15	30
Práctica	2	15	30
Total de horas al Semestre			**60**

Tabla 1: División de las horas teóricas y prácticas

Esta electiva no requiere de una clase requisito pero, si de conocimientos básicos de algunas asignaturas que se imparte en las carreras en el siguiente apartado se mostrara la relación de esta electiva con otras que se imparte a lo largo de la carrera.

Además el estudiante debe de tener conocimientos previos para lograr un total aprendizaje de esta electiva. A continuación se muestra algunos conocimientos que el estudiante debe de tener bastante claros:

Teoría:

- Base de datos general

Práctica:
Manejo de tablas relacionales.

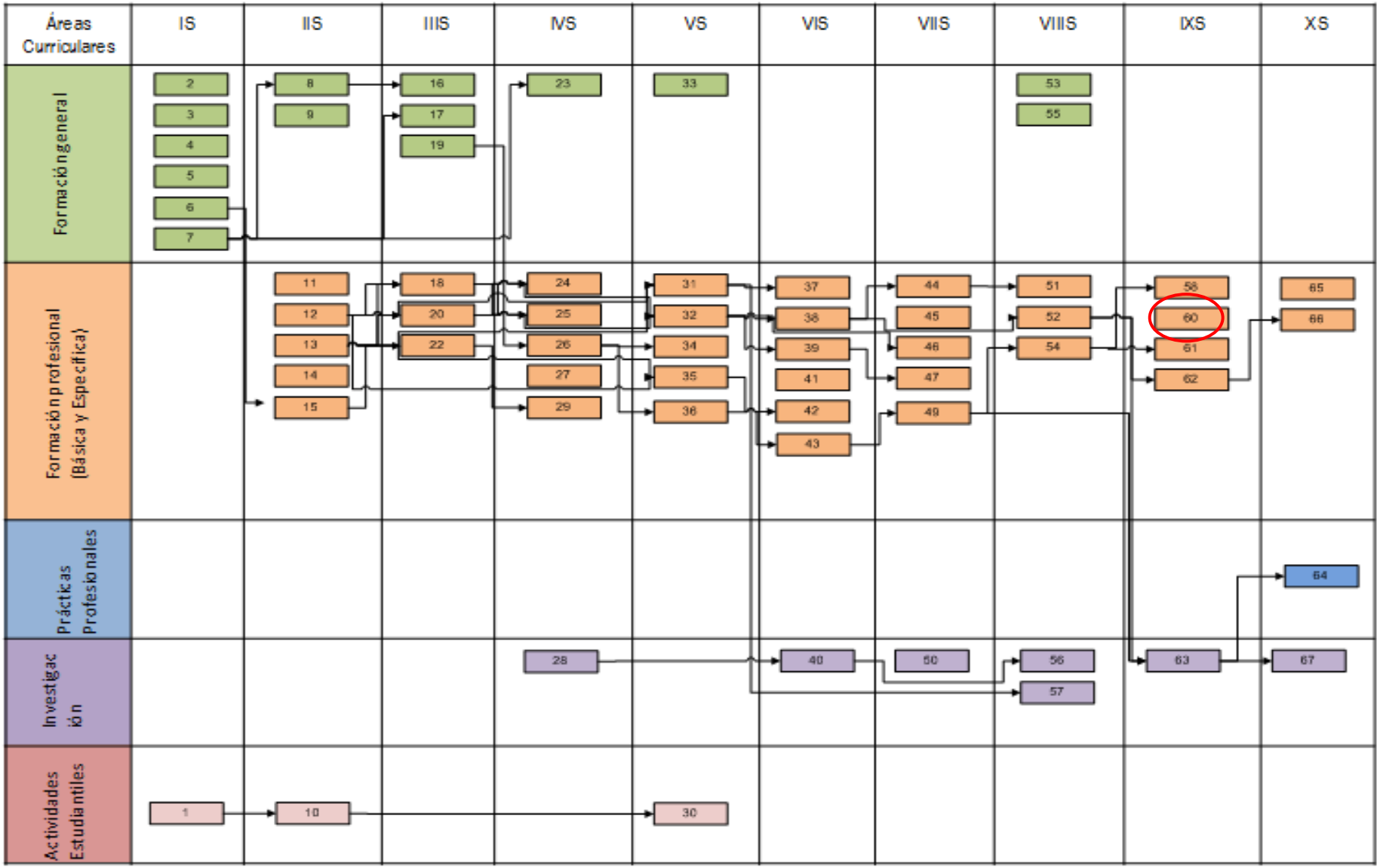


Ilustración 1: Malla curricular ISI

6) RELACIÓN CON OTRAS ASIGNATURAS

El proceso de aprendizaje de la electiva Data Mining (Minería de Datos), requiere de otros conocimientos que al llegar a esta electiva deben estar claros y afianzados para una mejor comprensión de la misma. Estos conocimientos son adquiridos por los estudiantes a lo largo de su formación académica en otras asignaturas.

Por eso se hace importante plasmar la relación que tiene la asignatura de Data Mining (Minería de Datos) con otras asignaturas, relación que se muestra primero de manera gráfica en la ilustración 2 y que a continuación se describe cada una de ellas.

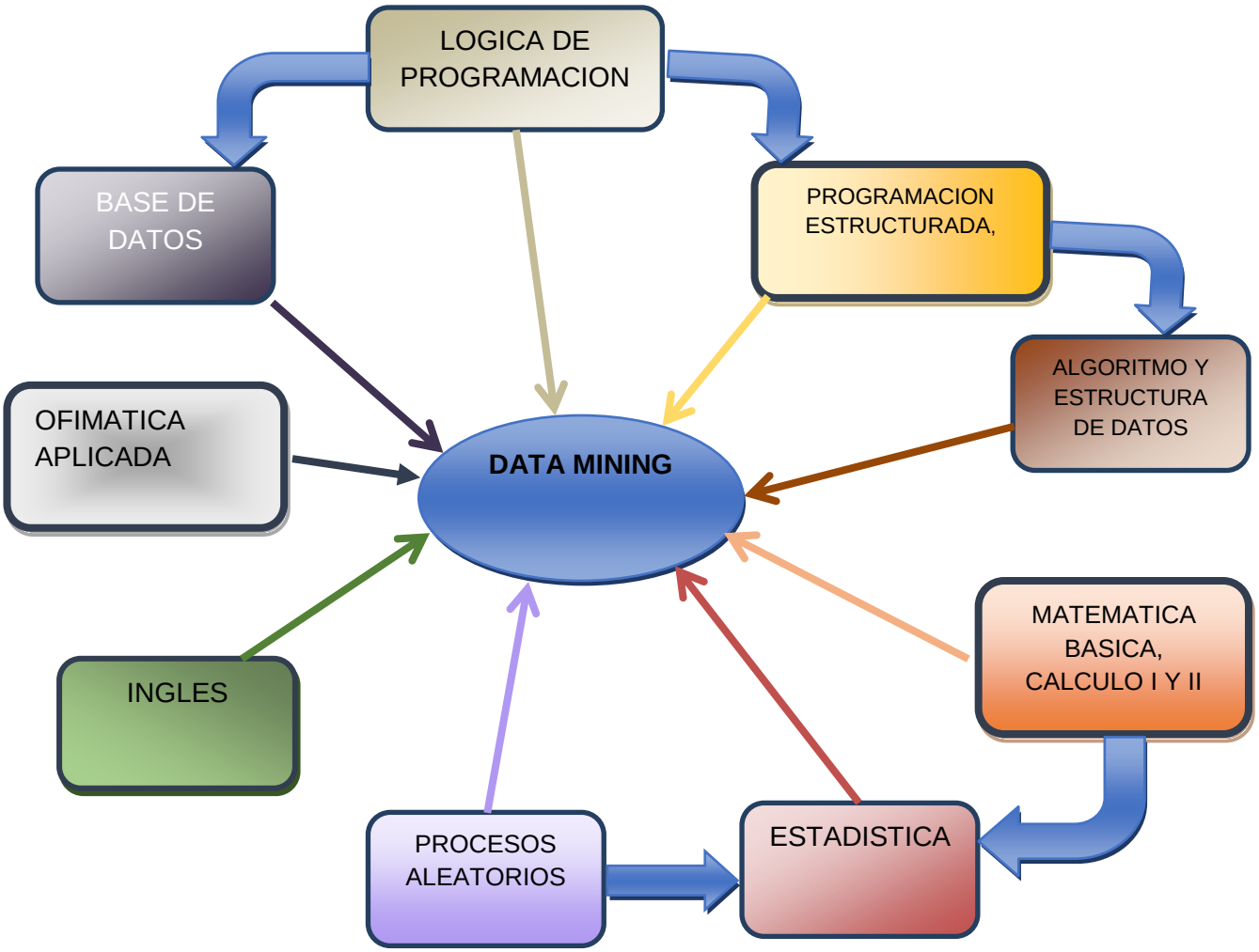


Ilustración 2: Relación con otras asignaturas

LÓGICA DE LA PROGRAMACIÓN

Este componente se ubica en el II semestre del I año. Presentará los conocimientos fundamentales de la simbología relacionada con los algoritmos con el objetivo de facilitar al estudiante el desarrollo de su capacidad analítica y creadora, proporcionándole destreza en el diseño de algoritmos que sirvan como base para la comprensión de los diferentes lenguajes de programación que aprenderá a lo largo de la carrera.

Los tópicos generales de este componente son: Pseudocódigo, sentencias de control, arreglos unidimensionales y bidimensionales, estructuras y funciones.

Relación con otros componentes: Laboratorio de lógica de programación y Programación Estructurada.

PROGRAMACIÓN ESTRUCTURADA

REQUISITOS: LÓGICA DE PROGRAMACIÓN.
LABORATORIO DE LÓGICA DE PROGRAMACIÓN

Este componente se ubica en el I semestre del II año. Presenta las bases teóricas para la solución de problemas de la vida real, utilizando un lenguaje de programación estructurado, como el Lenguaje C.

Proporciona al estudiante las habilidades básicas necesarias para realizar aplicaciones en C. El estudiante traducirá algoritmos y pseudocódigos a lenguajes de alto nivel, para completar el proceso lógico de análisis de un problema, empleando los elementos del lenguaje C.

Los tópicos generales de éste componente son: Introducción a lenguaje C, Estructuras de control, Descomposición funcional, Tipos de datos compuestos estáticos, Declaración de tipos propios, Punteros y gestión de memoria, Ficheros.

Se relaciona con las materias: Programación Orientada a Objetos, Algoritmo y Estructura de datos, para los cuales Programación Estructurada es un requisito.

DISEÑO DE BASES DE DATOS

REQUISITOS: LÓGICA DE PROGRAMACIÓN.
LABORATORIO DE LÓGICA DE PROGRAMACIÓN

Este componente se ubica en el I semestre del III año. Los estudiantes aplicarán los Conceptos básicos del diseño de base de datos utilizando el modelo relacional, enfatizando en conocimientos vinculados al diseño de un sistema en función de la realidad y el conocimiento del mismo.

Los tópicos generales de este componente son: El modelo Conceptual de un sistema de bases de datos (Diagramas ER), Transformación del modelo Conceptual al esquema relacional, El álgebra relacional, El Lenguaje SQL.

Relación con otros componentes: Sistemas gestores de bases de datos I. Sistemas gestores de bases de datos II.

ALGORITMOS Y ESTRUCTURAS DE DATOS

REQUISITOS: PROGRAMACIÓN ESTRUCTURADA
LABORATORIO DE PROGRAMACIÓN ESTRUCTURADA

Este componente se ubica en el II semestre del II año. Estudia la forma de representar un algoritmo para resolver un determinado problema de programación, se estudian las estructuras de datos más comunes así como también los algoritmos clásicos de ordenamiento ya existentes.

Los tópicos generales de este componente son: Asignación dinámica de memoria. Listas, pilas y colas. Algoritmo de ordenación y búsqueda.

Relación con otros componentes: Programación estructurada. Aplicaciones de estructuras de datos. Programación orientada a objetos.

OFIMÁTICA APLICADA

Este componente se ubica en el II semestre del I año. Enseña al estudiante sobre la manera de utilizar los programas empleados con frecuencia en la elaboración de documentos de texto, hojas de cálculo, preparación de presentaciones con diapositivas y el uso de internet en su trabajo diario.

Los tópicos generales de este componente son: Microsoft Word, Excel, Power Point, El correo electrónico, la navegación web.

Relación con otros componentes: Fundamentos de informática.

ESTADISTICA

Este componente se ubica en el II semestre del III año. Tiene el propósito de brindar herramientas básicas de estadística descriptiva e inferencial para ser aplicadas en la investigación científica.

Los tópicos generales de este componente son: Presentación y organización de datos. Tablas de frecuencia. Representaciones gráficas. Medidas de tendencia central. Medida de dispersión. Medidas de posición.

Relación con otros componentes: Matemática básica.

MATEMÁTICA BÁSICA

Tiene el propósito de brindar herramientas básicas que permitan al estudiante resolver problemas matemáticos fundamentales aplicando la lógica; teoría de conjunto; álgebra; y funciones polinomiales, racionales, exponenciales y logarítmicas. Se ubica en el I semestre del segundo año de la carrera, tiene 4 horas presenciales y un valor de 4 créditos académicos, es un curso teórico-práctico y se relaciona con Estadística. Es obligatorio.

CALCULO I

REQUISITOS: MATEMÁTICA BÁSICA.

Este componente se ubica en el II semestre del I año. Tiene el propósito de brindar herramientas básicas de cálculo diferencial e integral que permitan al estudiante resolver problemas matemáticos fundamentales aplicando conocimientos de límite, continuidad, derivada e integral.

Los tópicos generales de este componente son: Límite y continuidad, Derivadas, Aplicaciones de las derivadas, Integrales definidas.

Relación con otros componentes: Matemática básica, Matemática discreta y Cálculo II.

CÁLCULO II

REQUISITOS: CÁLCULO I.

Este componente se ubica en el I semestre del II año. Está integrado en el bloque de los componentes sobre elementos básicos que se imparten en el segundo año de la carrera, en este componente se supone que el estudiante ya tiene una base matemática para el desarrollo del cálculo integral los cuales le serán de mucha utilidad en el desarrollo de las competencias de los componentes de programación.

Los tópicos generales de este componente son: Técnicas de integración, Aplicaciones de las integrales.

Relación con otros componentes: Cálculo I, Física.

PROCESOS ALEATORIOS

El componente curricular de procesos estocásticos es un componente cognoscitivo impartido a estudiantes de las carreras de Ingeniería en Sistemas de Información ubicado en el VIII semestre. .

Con este componente el estudiante adquiere conocimientos de cadenas de Markov, procesos de Poisson y Teoría de colas entre otros temas, se le suministra las herramientas necesarias para la modelación de fenómenos del ámbito social, económico y ambiental.

Los tópicos generales de este componente: Modelos de series de tiempo y procesos de Poisson.
Relación con otros componentes: Series de tiempo

INTRODUCCIÓN A LA MINERÍA DE DATOS

Este componente curricular está ubicado en el tercer semestre de la carrera Ingeniería en Estadística y en el octavo semestre de la carrera de Matemática, es un componente de Formación General para la carrera de estadística y de formación para la carrera de Matemática.
Los tópicos generales de este componente: extraer la información trascendente de las estadísticas, técnicas multidisciplinares, indagar, preparar, procesar, interpretar, analizar y divulgar la información obtenida a partir de bases de datos

Relación con otros componentes: Análisis de Datos

MODELOS LINEALES I

Modelos Lineales I se ubica en el sexto semestre, teniendo como requisito la componente, Inferencia Estadística, en la cual el alumno ha desarrollado habilidades y destrezas en el manejo de distribuciones de probabilidad y estimación de parámetros.

Los tópicos generales de este componente La modelación de la diversidad de datos relaciones simples y con varías variables, modelo de regresión lineal múltiple.

Relación con otros componentes: Procesos aleatorios, Modelos Lineales II

ANALISIS MULTIVARIADO I

El componente curricular Análisis Multivariado I se ubica en el octavo semestre de la carrera de Ingeniería en estadística, presenta una introducción a la inferencia en Modelos Multivariante, La aplicación de estas técnicas debe de estar fundamentada en estudios de casos que le permita al estudiante poder discernir en cuanto a la técnica estadística a utilizar, así como, la interpretación adecuada de los resultados.

Los tópicos generales de este componente: Técnicas univariantes y multivariantes

Relación con otros componentes: Modelos Lineales II, Análisis Multivariado II.

SERIES DE TIEMPO

Series Temporales es una componente teórico-práctica, diseñada y vinculada al perfil del currículo de la carrera de Ingeniería Estadística, del quehacer profesional en cualquier área de las ciencias que involucre datos longitudinales.

El estudiante generará destreza y habilidades, que sirvan para que él pueda describir un conjunto de observaciones tomadas de forma secuencial, y ajustar modelos matemáticos, de entre los cuales elija aquel con la menor pérdida de información contenida en los datos.

Los tópicos generales de este componente: describir la información y elaborar pronósticos que ayuden en la toma de decisiones, aplicando modelos de Series temporales.

Relación con otros componentes: Modelos Lineales I, Probabilidad, Inferencia Estadística, Procesos Aleatorios, Minería de Datos, Modelos Lineales y Econometría.

7) CONTENIDO DEL TEMARIO

Tema 0: Presentación de la Asignatura

- Objetivos de la asignatura
- Temario
- Material de estudio
- Metodología de evaluación
- Horario de tutorías

Tema 1: Introducción a la Minería de Datos

- Surgimiento de la Minería de Datos
- Definiciones de Minería de Datos
- Ventajas y desventajas de la Minería de Datos
- Ejemplos de Minería de Datos
- Tipos de datos y tipos de modelos
    - Tipos de datos
    - Tipos de modelos
- Minería de Datos y descubrimiento del Conocimiento (KDD)
- Relación con otras disciplinas
- Aplicaciones
- Sistema y Herramientas de Minería de Datos
- ¿Qué no es Minería de Datos?

Tema 2: El Proceso de la extracción del Conocimiento

- Las Fases del Proceso de extracción del Conocimiento
- Fase de Integración y recopilación
- Fase de selección, limpieza y transformación
- Fase de Minería de Datos
- Fase de evaluación e interpretación
- Fase de difusión, uso y monitorización

Tema 3: Preparación de datos

- Introducción
- Recopilación. Almacenes de datos
- Arquitectura de los Almacenes de datos
- Carga y Mantenimiento del Almacén de Datos
- Almacén de Datos y Minería de Datos

Tema 4: Limpieza y transformación

- Introducción
- Integración y limpieza de datos
- Transformación de atributos, Creación de características
- Discretización y numerización
- Normalización de rango: escalado y centrado

Tema 5: Exploración y Selección

- Introducción
- Contexto de la vista minable
- Exploración mediante visualización
- Sumarización , descripción, generalización y pivotamiento
- Selección de datos
- Lenguajes, primitivas e interfaces de MD

Tema 6: Técnicas y Minería de Datos

- Introducción
- Tareas y métodos
- MD y aprendizaje inductivo
- Comparación de métodos
- Modelización estadística
- Modelo de regresión
- Análisis de los residuos
- Modelos Lineales
- Análisis discriminante
- Aplicabilidad

8) METODOLOGÍA DIDÁCTICA Y MATERIAL DIDÁCTICO A UTILIZAR

8.1 METODOLOGÍA DIDÁCTICA

En la teoría:

Para impartir la electiva la metodología a usar serán las lecciones presenciales con una duración de dos horas cada sección, estas serán planificadas con el contenido teórico de este plan. Estas secciones se desarrollaran en dos bloques de dos horas cada uno.

Estas lecciones se complementaran con explicaciones a través de ejemplos para ayudar a una mejor comprensión de esta electiva. Además se realizaran trabajos investigativos para profundizar en los temas y alentar el deseo de conocimiento de parte de los alumnos.

En la práctica:

El profesor deberá de dar una breve explicación de la práctica a realizar a inicio de cada sección de laboratorio, realizando ejemplos con relación a la práctica a desarrollar por los estudiantes. Prácticas que el estudiante deberá de entregar en una fecha establecida por el profesor de laboratorio cumpliendo con la puntualidad y finalización de dicho trabajo.

8.2 MATERIAL DIDÁCTICO

En ambas partes tanto en teórica como en la práctica utilizaran un material de apoyo los cuales ayudaran al profesor a exponer y ejemplificar cada tema o práctica a desarrollar:

- Pizarra para mayor captación para los alumnos.
- Diapositivas para el estudio de la electiva.
- Marcadores, borrador.
- Bibliografía básica para el estudio de esta electiva.
- Un sitio web donde se podrá colgar tanto las lecciones impartidas como las prácticas a realizar.

9) PLANIFICACIÓN TEMPORAL

La planificación de la electiva VIII, para que sea de una manera objetiva, debe de realizarse año con año en función del calendario académico del que se disponga, para poder partir de este y calcular el número de sesiones lectivas de las cuales se dispone.

La electiva VIII: Introducción a la Minería de Datos, se impartirá en el segundo semestre de IV año. Normalmente el segundo semestre consta con un total de 16 semanas hábiles, considerando los días festivos y dando margen para cualquier otra incidencia como realización de exámenes, podemos deducir que se tienen 15 semanas netas para cumplir con el contenido de la electiva.

La electiva VIII constará tanto de teoría como práctica, las cuales constan para la teoría de una sesión de 2 horas a la semana, por 15 semanas que hemos asumido como netas, suman un total de 30 horas y 30 horas para las prácticas. Debido a que para poder realizar la primera práctica de laboratorio el alumno debe conocer algunos conceptos que se imparten en el Tema 1, entonces la primera semana de clase sólo se impartirá clases teóricas aprovechando la primera sesión de laboratorio para impartir los conceptos teóricos necesarios para la realización de la primera práctica, por lo tanto, en total tendremos 30 horas netas para teoría y 30 para laboratorio.

El número de horas asignadas a cada tema y a cada práctica, se ha calculado en base a la profundidad con que se quiere abordar cada tema. De tal manera que la planificación temporal tanto para la teoría como para las prácticas puede ser la siguiente:

Planificación temporal de la parte teórica.

N° de Tema	Tema	Horas
0	Presentación de la asignatura	2
1	Introducción a la Minería de Datos	4
2	El proceso de extracción del conocimiento	4
3	Preparación de los datos	6
4	Limpieza y transformación de los datos	4
5	Exploración y selección de datos	4
6	Técnicas de Minería de Datos	6
Horas Teóricas Totales		30

Tabla 2: Tabla de planificación temporal teórica

Planificación temporal de la parte práctica

N° de Tema	Tema	Horas
0	Instalación, Ambiente y entorno R, : Exportar e Importar y Procesar Cuadros de datos	2
1	Agrupación de Funciones y operadores Aritméticos, Lógicos, Especiales y Símbolos.	2
2	Funciones de control de flujo (Estudio de caso: Máximo común divisor de dos números)	2
3	Regresión Lineal simple, múltiple y No lineal (Estudio de caso: Peso y talla de Mujeres)	6
4	Series de tiempo para Fases Exploratorias (Estudio de caso: Pasajeros procedentes de Vuelos Internacionales (AirPassengers)	6
5	Series de Tiempo con modelización (Estudio de Caso: Productos de la Canasta Básica Nicaragüense del año 2012 al año 2014)	6
6	Análisis de Conglomerados, Cluster analysis o Clustering (Estudio de Caso: Causas de Mortalidad en el Dpto. de León, Nicaragua del año 2010 al I semestre del año 2015)	6
Horas Prácticas Totales		**30**

Tabla 3: Tabla de planificación temporal práctica.

10) METODOLOGÍA DE EVALUACIÓN

La electiva VIII está compuesta por clases teóricas, y prácticas que se realizan en el laboratorio de la misma. La forma de evaluación de esta electiva será la misma que establece los estatutos de nuestra Universidad que consiste en dos evaluaciones parciales que corresponden al 60% de la nota final y el restante 40% se obtiene mediante una evaluación al final del semestre. Cada una de las evaluaciones parciales se desglosa de la manera siguiente:

- **Examen Teórico**……..……....70%
- **Evaluación Práctica**.……30%
- **Total**……100%

Las notas del primer y segundo parcial se promedian y se multiplican por 0.6 (ya que equivalen al 60% de la nota final) para obtener la nota de entrada al examen final. El examen final sólo se tendrá derecho de realizarlo si el promedio de las dos evaluaciones parciales promedian un mínimo de 50, y tiene un valor de 100 puntos que luego se multiplica por 0.4 (para obtener el 40% al que equivale) y el resultado se suma con la nota de entrada. Para aprobar la asignatura el alumno debe de obtener una nota mayor o igual que 60. Lo anterior se refleja mediante la siguiente fórmula:
(1er_Parcial + 2do_Parcial)*0.6 + Examen_Final*0.4 >= 60

Evaluación de la Teoría:

El examen teórico que corresponde al 70% de cada parcial constará de preguntas que permitan evaluar el grado de compresión de los conceptos básicos que tiene el estudiante. También contendrá preguntas de carácter analítico y ejercicios en los cuales el alumno deberá de demostrar la capacidad de resolver problemas, aplicando los conocimientos adquiridos.

Evaluación del Laboratorio:

El 30% correspondiente a las evaluaciones del laboratorio se obtendrá a través de la entrega en tiempo y forma de las prácticas de laboratorio y de la realización de un examen en el mismo periodo fijado para el examen teórico. El valor de cada práctica se distribuirá en función del número de ejercicios y la complejidad de los mismos.

PARTE II: Desarrollo Del Temario Teórico

11) TEMA 0: PRESENTACIÓN DE LA ASIGNATURA ELECTIVA VIII: INTRODUCCIÓN A LA MINERÌA DE DATOS

OBJETIVOS:

- Presentar al estudiante los principales aspectos relacionados con el desarrollo de la asignatura.
- Dar a conocer los objetivos que se desean alcanzar con esta asignatura.
- Dar a conocer la metodología para impartir y evaluar la asignatura.
- Concertar horas de tutorías, en las que el estudiante pueda acudir al profesor para plantear sus dudas e intentar resolverlas.

CONTENIDO:

En este tema se debe presentar al alumno los siguientes puntos:

- Objetivos de la asignatura
- Temario
- Material de estudio
- Metodología de evaluación
- Horario de tutorías

DESCRIPCIÓN:

Esta asignatura proporciona a los estudiantes una visión teórica y práctica acerca de los fundamentos, técnicas y problemas actuales en la Minería de Datos y el Descubrimiento de Conocimiento desde grandes bases de datos. Los estudiantes conocerán algunas herramientas básicas, recursos computacionales y algunas aplicaciones simples en Minería de Datos.

Conocimientos básicos de técnicas de aprendizaje, Bases de datos y Estadísticas son necesarios para tomar este curso.

RECOMENDACIONES

Se exponen, a continuación, algunas de las competencias que deberían poseer los alumnos antes de comenzar la asignatura:

Esenciales:

- No hay prerrequisitos esenciales.

Recomendables:

- Tener conocimientos en Sistemas de Gestión de Base de Datos

- ❖ Tener destreza en lenguajes de programación orientados a objetos de propósito general (Java, C++,…)
- ❖ Poseer conocimientos básicos de Inglés
- ❖ Poseer conocimientos básicos de estadísticas
- ❖ Poseer destreza para buscar información en la Red
- ❖ Saber manejar fuentes bibliográficas
- ❖ Tener capacidad de lectura comprensiva

RESULTADOS DE APRENDIZAJE ESPERADO

Al finalizar esta asignatura, los estudiantes deberán ser capaces de:

1. Describir los conceptos y problemas fundamentales en el proceso de minado de datos y descubrimiento de conocimiento.
2. Explicar y Distinguir los métodos y técnicas actuales en Minería de Datos.
3. Explicar y entender técnicas básicas de Minería de Datos descriptiva y predictiva.
4. Identificar y comparar modelos de evaluación en Minería de Datos.
5. Construir y Aplicar un prototipo simple de Minería de Datos utilizando herramientas computacionales

Duración: 2 horas.

12) TEMA 1: INTRODUCCIÓN A LA MINERÍA DE DATOS.

Objetivos:

- Conocer las definiciones de Minería de Datos.
- Conocer cuáles son sus ventajas.
- Conocer los tipos de Datos y Modelos.

Contenidos:

- Surgimiento de la Minería de Datos.
- Definiciones de Minería de Datos.
- Ventajas y desventajas.
- Ejemplos.
- Tipos de datos y Tipos de modelos.
- Minería de Datos y KDD.
- Relación con otras disciplinas.
- Aplicaciones.
- Sistema y Herramientas de Minería de Datos.
- ¿Qué no es Minería de Datos?

Duración: 4hrs.

Bibliografía:

- Introducción a Minería de Datos. José Hernández Orallo, Mª José Ramírez Quintana, Cesar Ferri Ramírez.
- Wikipedia

12.1 Surgimiento de la Minería de Datos.

La Minería de Datos surge debido al volumen y variedad de información que se encuentra albergada en una base de datos digital, la Minería de Datos ayudará a la mayoría de las decisiones de empresas, organizaciones e instituciones que se basan en información de experiencias pasadas extraídas de fuentes muy diversas.

También surge por la necesidad de agilizar las decisiones y abandonar la manera antigua la cual era lenta, cara y altamente subjetiva, consecuentemente muchas decisiones importantes se realizan, no sobre una base de gran cantidad de datos disponibles, sino por la intuición del usuario al no disponer de las herramientas necesarias.

El principal objetivo de la Minería de Datos es resolver problemas analizando datos presentes de las bases de datos.

12.2 Definición de Minería de Datos.

Minería de Datos es el proceso de la extracción de conocimientos útiles y comprensibles, de grandes cantidades de datos almacenados en diferentes formatos, en la Minería de Datos podemos encontrar modelos inteligibles a partir de los datos.

Los retos de la Minería de Datos son:

- Trabajar con grandes cantidades de datos.
- Utilizar técnicas para analizar y extraer conocimientos útiles.

12.3 Ventajas y Desventajas de la Minería de Datos

Las Ventajas y desventajas de Minería de Datos se nos muestran en la siguiente Ilustración.

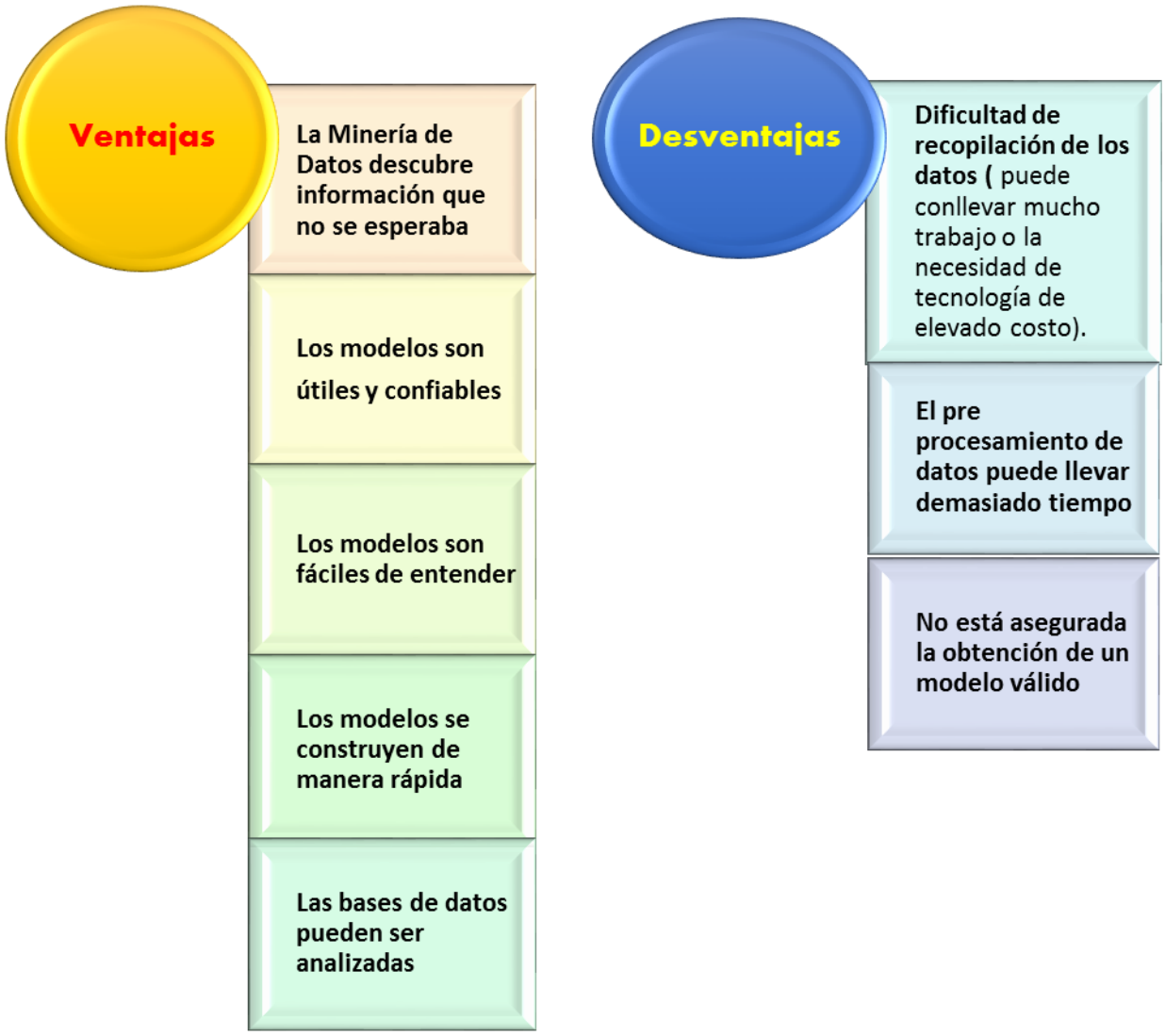


Ilustración 3: Ventajas y Desventajas de la Minería de Datos

12.4 Ejemplos

Uno de los objetivos de la Minería de Datos es convertir los datos en conocimientos. Este objetivo no sólo es ambicioso sino que muy amplio, en todo caso les mostraremos algunos ejemplos en los que se utiliza la Minería de Datos y así ayudar al alumno a captar y entender la definición de Minería de Datos.

Ejemplos de algunos usos de la Minería de Datos:

Supermercados.

En los supermercados se utiliza la Minería de Datos para saber cuál es la detección de hábito de compras en un supermercado.

Bancos
En los bancos es utilizada para poder tomar decisiones sobre a quién se le dará crédito o a quien no se le dará créditos, esta decisión se toma haciendo un estudio del estado crediticio de la que desee el préstamo.

Fraudes.
En casos análogos en búsqueda de detección de transacciones ilícitas como: lavado de dinero, fraude al uso de tarjetas de créditos, o de los servicios de telefonía móviles, estas operaciones suelen seguir patrones que permiten con cierto grado de probabilidad distinguirlas de las legítimas y desarrollar mecanismo que permitan tomar medidas rápidas frente a la situación.

Recursos humanos
La Minería de Datos también puede ser útil para los departamentos de recursos humanos en la identificación de las características de sus empleados de mayor éxito. La información obtenida puede ayudar a la contratación de personal, centrándose en los esfuerzos de sus empleados y los resultados obtenidos por éstos. Además, la ayuda ofrecida por las aplicaciones para Dirección estratégica en una empresa se traducen en la obtención de ventajas a nivel corporativo, tales como mejorar el margen de beneficios o compartir objetivos; y en la mejora de las decisiones operativas, tales como desarrollo de planes de producción o gestión de mano de obra.

Comportamiento en Internet

También es un área en boga el del análisis del comportamiento de los visitantes —sobre todo, cuando son clientes potenciales— en una página de Internet. O la utilización de la información —obtenida por medios más o menos legítimos— sobre ellos para ofrecerles propaganda adaptada específicamente a su perfil. O para, una vez que adquieren un determinado producto, saber inmediatamente qué otro ofrecerle teniendo en cuenta la información histórica disponible acerca de los clientes que han comprado el primero.

Análisis de la cesta de la compra

El ejemplo clásico de aplicación de la Minería de Datos tiene que ver con la detección de hábitos de compra en supermercados. Un estudio muy citado detectó que los viernes había una cantidad inusualmente elevada de clientes que adquirían a la vez pañales y cerveza. Se detectó que se debía a que dicho día solían acudir al supermercado padres jóvenes cuya perspectiva para el fin de semana consistía en quedarse en casa cuidando de su hijo y viendo la televisión con una cerveza en la mano. El supermercado pudo incrementar sus ventas de cerveza colocándolas próximas a los pañales para fomentar las *ventas compulsivas.*

12.5 Tipos de datos y Tipos de modelos.

Tipos de datos.

Debemos saber cuáles son los tipos de datos que nos permite la Minería de Datos, en principio puede aplicarse a cualquier tipo de información, siendo las técnicas de minería diferentes para cada una de ellas. Diferenciaremos entre datos estructurados que previenen de las bases de datos relacionales y datos no estructurados que provienen de la web o de otros tipos de repositorio de documentos.

a) Base de datos relacionales.

Sabemos que una base de datos relacionales es una colección de relaciones es decir de tablas, donde cada tabla consta de un conjunto de atributos lo que son las columnas o campos que esta contiene.

Una de las mayores características que la base de datos relacionales que contiene es la existencia de un esquema asociado.

Las bases de datos relacionales son las fuentes de datos para la mayoría de aplicaciones de Minería de Datos, hay técnicas de Minería de Datos que no son capaces de trabajar con todas las bases de datos, sino que solo son capaces de tratar con una sola tabla a la vez.

Es importante conocer los tipos de atributos ya que en base de datos existen varios tipos de datos:

- Los atributos Numéricos.
- Los atributos categóricos o nominales.

Existen otros tipos de base de datos:

- Las bases de datos temporales.
- Las bases de datos esenciales.
- Las bases de datos documentales.
- Las bases de datos multimedia.

b) La World Wide Web.

La World Wide Web es el repositorio de información más grande y diversa que existe en la actualidad hay una gran cantidad de datos que se pueden extraer conocimiento relevante y útiles a esto se le llama Minería Web.

Muchas páginas web contienes datos multimedia estos datos pueden residir en diversos servidores o en archivos, otro aspecto que dificulta la minería web son como determinar a qué páginas podemos acceder y como seleccionar la información, toda esta diversidad hace que la minería web se organice en tres categorías:

- Minería del contenido.
- Minería de la estructura.
- Minería de uso.

c) La minería de Texto

Se refiere al proceso de derivar información nueva de textos.

A comienzos de los años ochenta surgieron los primeros esfuerzos de minería de textos que necesitaban una gran cantidad de esfuerzo humano, pero los avances tecnológicos han permitido que esta área progrese de manera rápida en la última década.

La minería de textos es un área multidisciplinaria basada en la recuperación de información, Minería de Datos, aprendizaje automático, estadísticas y la lingüística computacional. Como la mayor parte de la información (más de un 80%) se encuentra actualmente almacenado como texto, se cree que la minería de textos tiene un gran valor comercial.

Se le presta cada vez un mayor interés a la minería de textos multilingual: la habilidad de ganar información en otros idiomas.

Aplicaciones académicas de la minería de textos

El tema de la minería de textos es de importancia para publicadores que tengan grandes bancos de data que requieran de indexación. Esto es el caso en particular para disciplinas científicas en las que hay una gran cantidad de información muy específica en forma de texto escrito. Es por ello que se han presentado iniciativas como el Open Text Mining Interface (OTMI) y el common Journal Publishing Document Type Definition (DTD) de la NIH, que ofrecerían datos semánticos para responder a preguntas muy específicas sin quitar las barreras del publicador al acceso público.

12.5.1 Tipos de modelos.

Conocemos que la Minería de Datos tiene como objetivo analizar los datos para extraer conocimientos esto puede ser de forma de:

- Relaciones.
- Patrones.
- Reglas inferidos a los datos.

Las relaciones constituyen el modelo de los datos analizados, los modelos pueden ser de dos tipos:

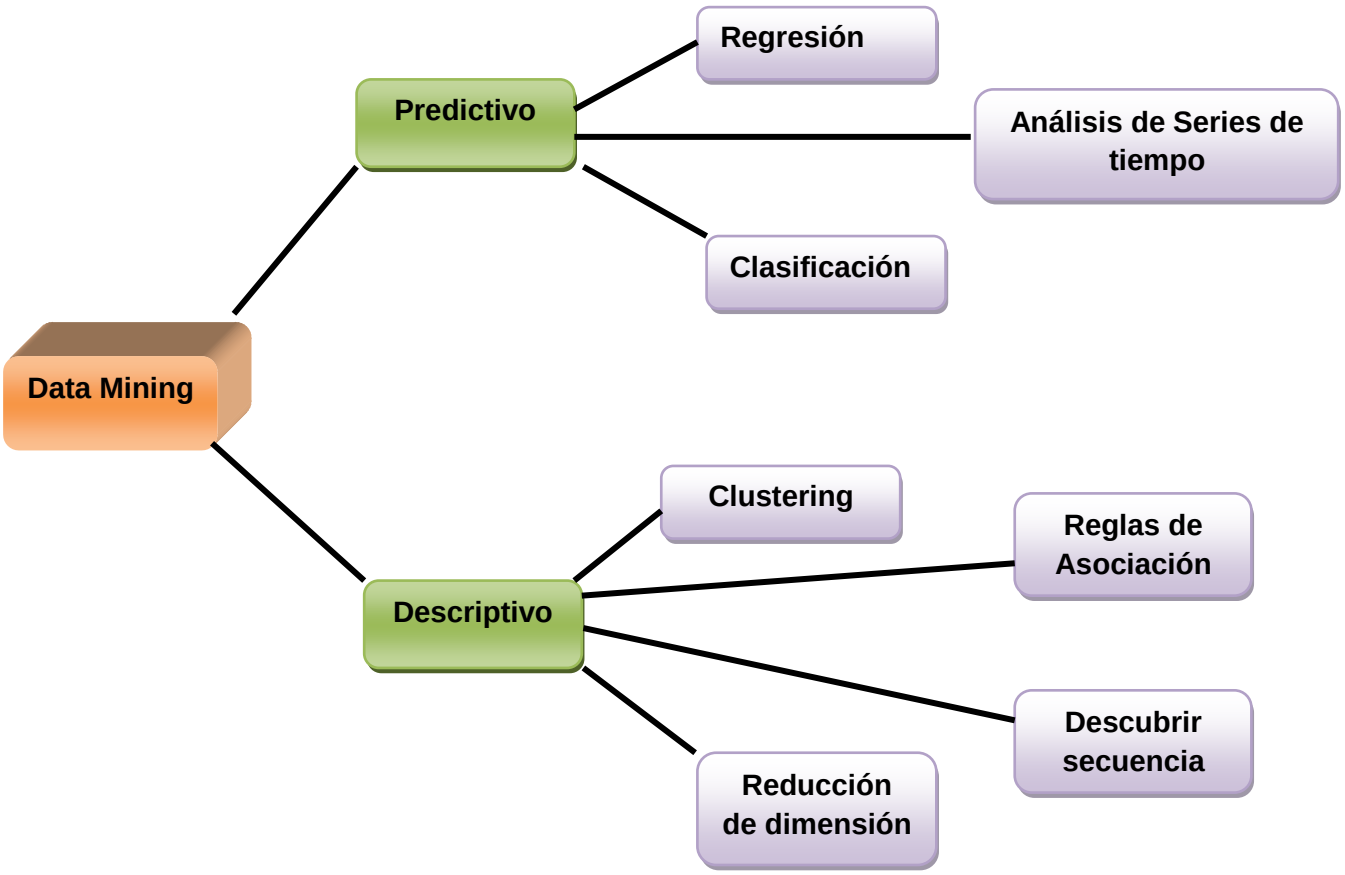


Ilustración 4: Tipos de modelos de la Minería de Datos

Los modelos predictivos: sirven para estimar valores futuros o desconocidos de variables de interés.

Los modelos descriptivos: son los que identifican patrones que explican o resumen los datos, significa que sirve para explorar las propiedades de los datos examinados.

12.6 Minería de Datos y KDD.

Existen términos frecuentemente utilizados para la Minería de Datos, se conocen como:

- Análisis de Datos.
- Descubrimiento de conocimiento de la base de datos en inglés **K**nowledge **D**iscovery in **D**ata bases o sus siglas **KDD**.

El **KDD** se define como el proceso no trivial de identificar los:

- **Patrones validos:** los patrones deben seguir siendo precisos para datos nuevos.
- **Novedosos:** aporte algo desconocido para el sistema y para el usuario.
- **Potencialmente útil:** la información debe conducir a acciones que reporten algún tipo de beneficio para el usuario.
- **Comprensibles:** información no comprensible no permiten la comprensión, interpretación, revisión, validación y uso de toma de decisiones la cual no proporcionara ningún conocimiento.

KDD es proceso complejo que incluye no solo la obtención de los modelos o patrones, la evaluación y posible interpretación de los mismos se refleja en la Ilustración siguiente:

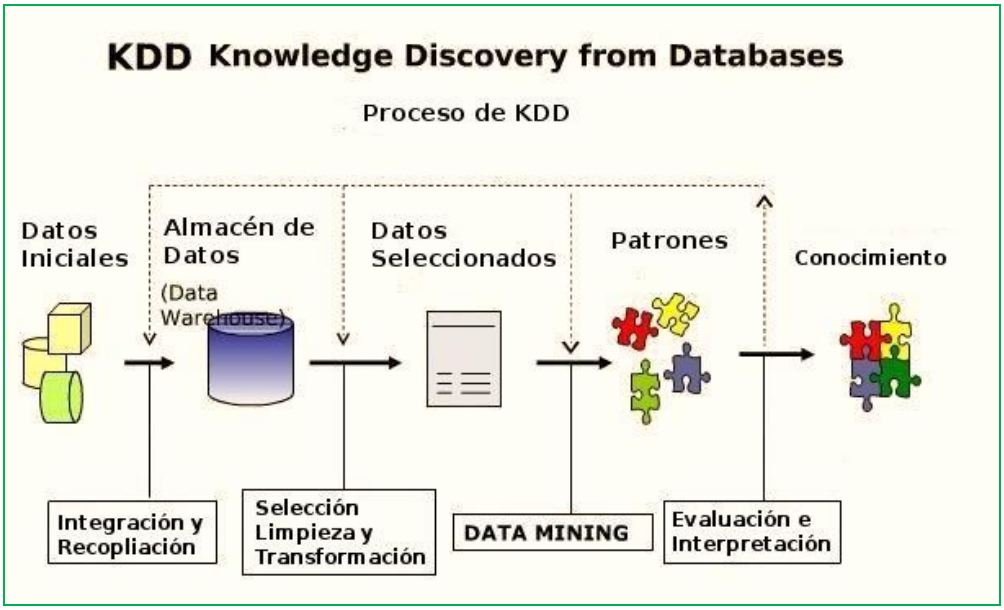


Ilustración 5: Proceso del KDD

El sistema de **KDD** considera las siguientes etapas:

- **Selección de datos:** Consiste en buscar el objetivo y las herramientas del proceso de minería, identificando los datos que han ser extraídos.
- **Limpieza de datos:** En este paso se limpian los datos sucios, incluyendo los datos incompletos
- **Integración de datos:** Combina datos de múltiples procedencias incluyendo múltiples bases de datos, que podrían tener diferentes contenidos y formatos**.**
- **Transformación de datos:** consisten principalmente en modificaciones sintácticas llevadas a cabo sobre datos sin que supongan un cambio para la técnica de minería aplicada.

- **Reducción de datos:** Reducir el tamaño de los datos, encontrando las características más significativas dependiendo del objetivo del proceso.
- **Minería de Datos:** Consiste en la búsqueda de los patrones de interés que pueden Expresarse como un modelo o simplemente que expresen dependencia de los datos.
- **Evaluación de los patrones:** Se identifican verdaderamente patrones interesantes que representan conocimiento usando diferentes técnicas incluyendo análisis estadísticos y lenguajes de consultas.
- **Interpretación de resultados:** Consiste en entender los resultados del análisis y sus implicaciones y puede llevar a regresar a algunos de los pasos anteriores.

La definición clasifica la relación entre **KDD** y la Minería de Datos, que el **KDD** es el proceso global para descubrir conocimiento útil desde la base de datos mientras que la Minería de Datos se refiere a la aplicación de los métodos de aprendizajes y estadísticos para obtención de patrones y modelos.

12.7 Relación con otras disciplinas.

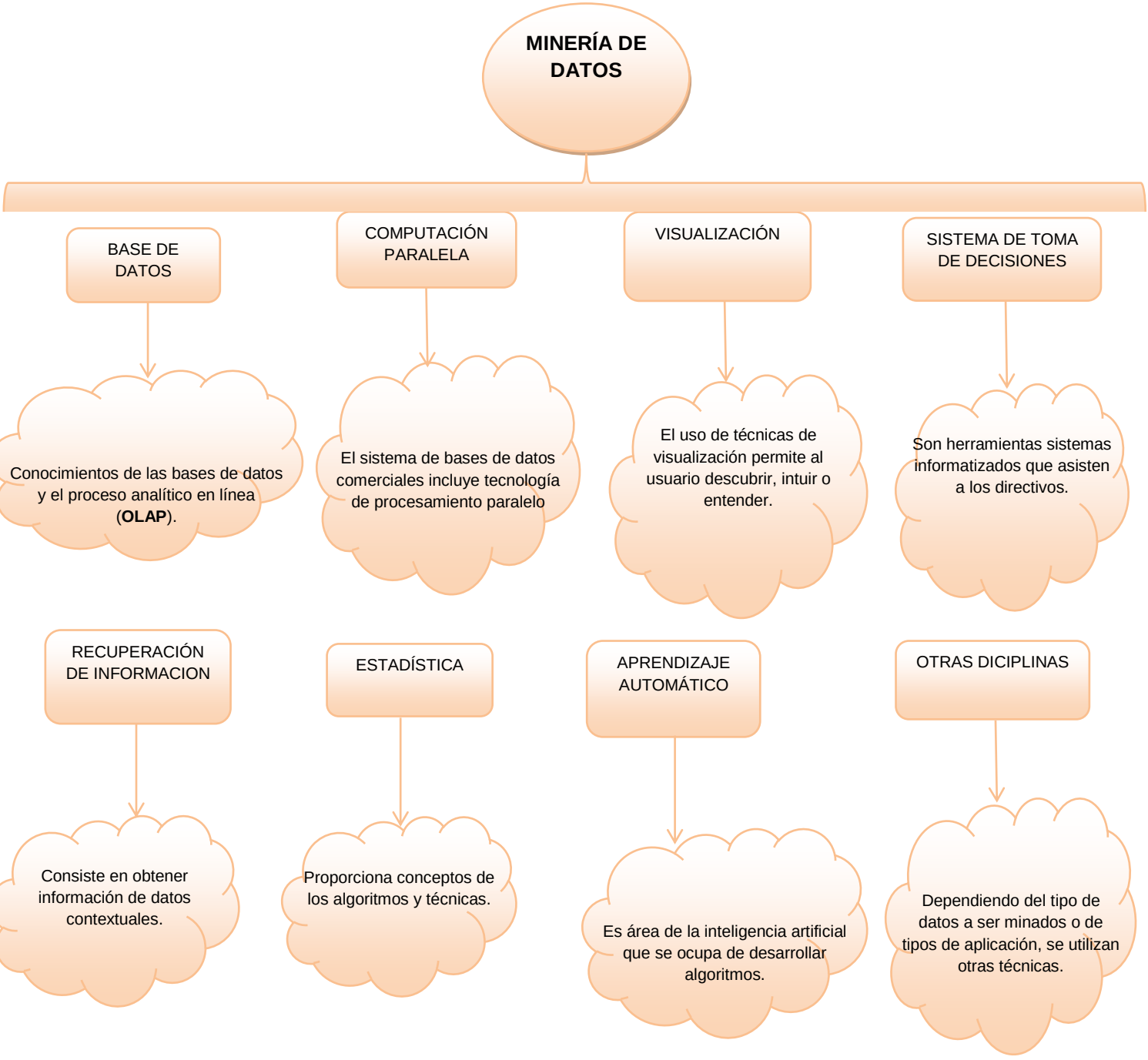


Ilustración 6: Relación con otras disciplinas.

12.8 Aplicaciones

La integración de técnicas de minera de datos se ha convertido en algo habitual en las actividades del día a día. Las áreas tradicionales en las que más se han empleado los métodos de minería son los negocios de distribución y publicidad, ya que han permitido reducir el costo o aumentar la productividad de ofertas.

He aquí una lista de algunas áreas donde se aplican:

- Aplicaciones financieras y bancas:
  - ✓ Obtención de patrones de uso fraudulentos de tarjetas de créditos.
  - ✓ Determinación de gastos de tarjetas de créditos.
  - ✓ Análisis de riesgos en créditos.
- Seguros y salud privada:
  - ✓ Determinación de los clientes que podrán ser potencialmente caros.
  - ✓ Análisis de procedimientos médicos solicitados conjuntamente.
  - ✓ Predicción de que clientes contratan nuevas pólizas.
  - ✓ Identificación de comportamiento fraudulento.
  - ✓ Predicción de los clientes que podrían ampliar su póliza.
- Educación:
  - ✓ Selección o captación de los estudiantes.
  - ✓ Detección de abandonos y de fracasos.
  - ✓ Estimación de tiempo en la estancia en la institución.
- Medicina:
  - ✓ Identificación de patologías. Diagnósticos de enfermedades.
  - ✓ Detección de paciente con riesgos de sufrir una patología concreta.
  - ✓ Gestión hospitalaria y asistencial.
  - ✓ Recomendación priorizada de fármacos para una misma patología.
- Telecomunicaciones:
  - ✓ Establecimiento de patrones de llamadas.
  - ✓ Modelos de cargas en redes.
  - ✓ Detección de fraude.
- Otras áreas:
  - ✓ Correo electrónicos y agendas personales.
  - ✓ Recursos humanos: selección de empleados.
  - ✓ Web: análisis del comportamientos de los usuarios.
  - ✓ Turismos: determinar las características socioeconómicas de los turistas.
  - ✓ Trafico: modelo de tráfico a partir de fuentes diversas.
  - ✓ Policiales: identificación de posibles terroristas en un aeropuerto.
  - ✓ Deportes: estudio de la influencia de jugadores y de cambios.

12.9 Herramientas de software

- dVelox de APARA
- KXEN
- KNIME
- Neural Designer
- OpenNN
- Orange
- Powerhouse
- Quiterian
- RapidMiner
- R
- SPSS Clementine
- SAS Enterprise Miner
- STATISTICA Data Miner
- Weka
- KEEL

12.10 Sistema y Herramientas de Minería de Datos.

Hay diversas disciplinas que contribuyen a la Minería de Datos la cual está dando lugar a una gran variedad de sistema de Minería de Datos.

En la siguiente ilustración se encuentra las clasificaciones de los sistemas y herramientas de Minería de Datos atendiendo varios criterios:

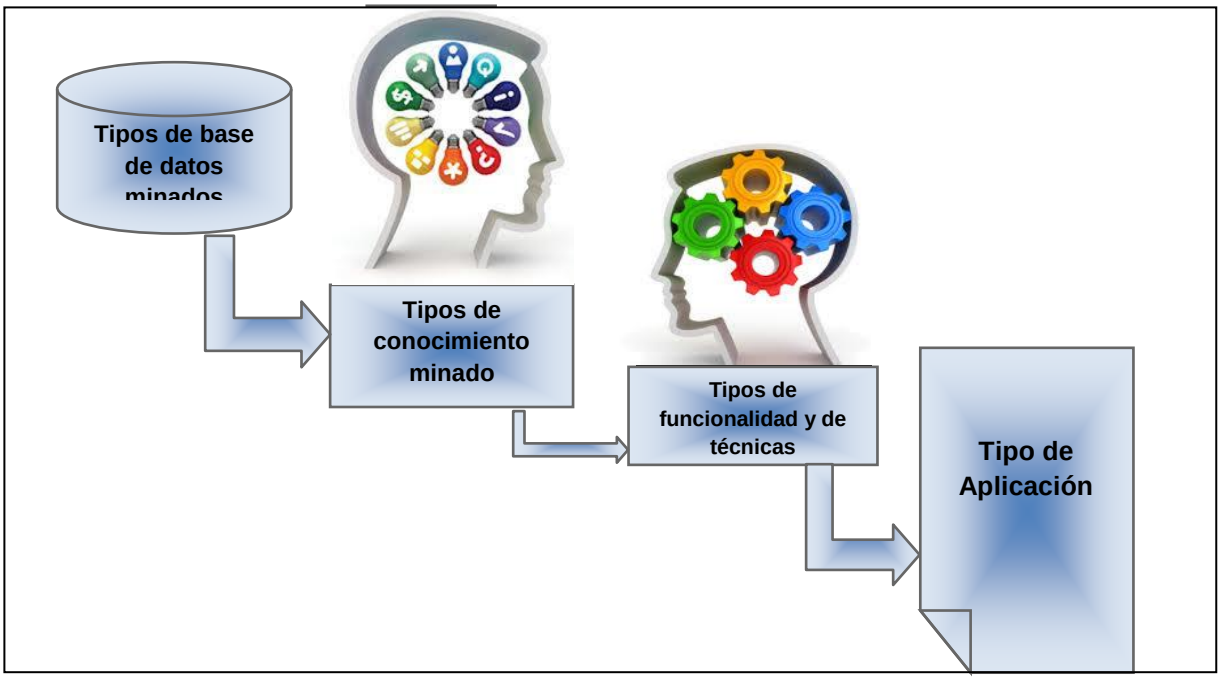


Ilustración 7: Clasificaciones de los sistemas y herramientas de Minería de Datos atendiendo varios criterios

12.11 ¿Qué no es Minería de Datos?

Ilustración 8: Qué no es Minería de Datos.

Un modelo **determinístico** es un modelo matemático donde las mismas entradas producirán invariablemente las mismas salidas, no contemplándose la existencia del azar ni el principio de incertidumbre.

Un modelo **probabilísticos** es un modelo matemático que nos ayuda a predecir la conducta de futuras repeticiones de un experimento aleatorio mediante la estimación de una probabilidad de ocurrencia de dicho evento concreto.

GUÍA DE APOYO PARA EL ESTUDIANTE

Tema 1: Introducción a la Minería de Datos

I. Complete adecuadamente.

a) La Minería de Datos surge debido al _____ y _____ de información que se encuentra albergada en una _____ de _____ digital.
b) _____ es el proceso de la extracción de conocimientos útiles y comprensibles.
c) El **KDD** es _____ complejo que incluye no solo la obtención de los _____ o _____, la _____ y posible _____ de los mismos.
d) Un modelo _____es un modelo _____ donde las mismas entradas producirán invariablemente las mismas salidas.

II. Enumere.

a) Los retos de la Minería de Datos son:
b) El **KDD** se define como el proceso no trivial de identificar los:
c) El sistema de **KDD** considera las etapas:
d) Donde se aplican la técnica de Minería de Datos:

III. Escriba **V** si es Verdadero y **F** si es Falso según convenga.

a) Una de las ventajas de Minería de Datos es que las Bases de Datos pueden ser analizados.___.
b) La Minería de Datos se usa en casos como fraudes, bancos, supermercados, etc. ___.
c) Una Base de Datos relacional es una colección de relaciones es decir de tupla.___.
d) Un modelo probabilístico en un modelo estadístico que nos ayuda a predecir la conducta de futuras repeticiones de un estudio aleatorio mediante la estimación de una probabilidad de ocurrencia de dicho evento concreto.___.

IV. Selección múltiple.

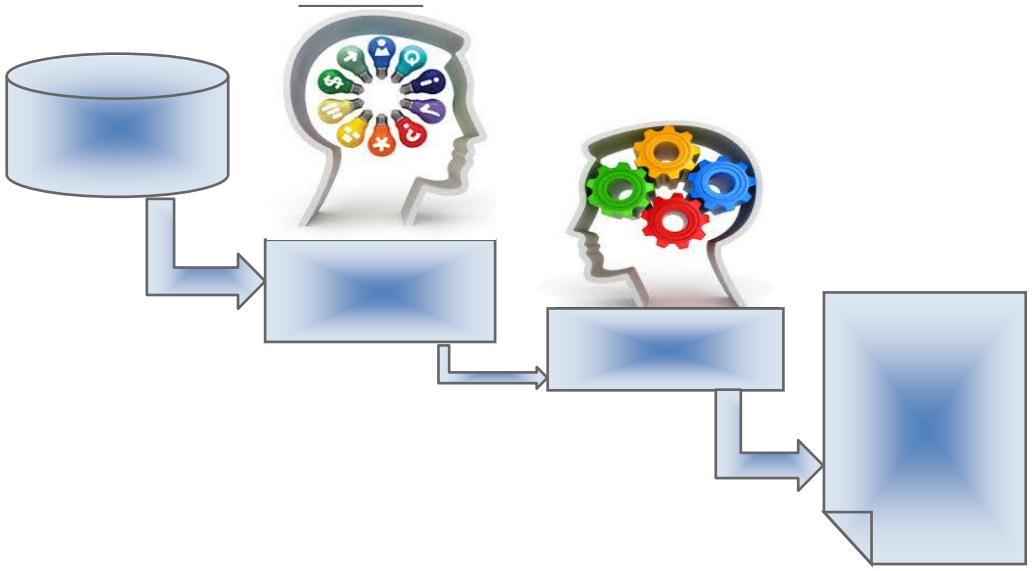
1) Los tipos de datos lo conforman:
   a) Base de Datos relacionales, la World Wide Web.
   b) Predictivo y Descriptivo.
   c) Tablas y Datos.
2) El proceso del **KDD** se refleja:
   a) Inicio, Comprensión y Asociación.
   b) Selección, Transformación, Interpretación.

c) Selección, Procesamiento, transformación, Minería, Interpretación/ Evaluación, conocimiento.

3) La relación con otras disciplinas:
   a) Aplicaciones financieras y bancas, seguro y salud privada.
   b) Base de datos, computación paralela, visualización, sistema de toma de decisiones, recuperación de información, estadística, aprendizaje automático, etc.
   c) Educación, medicina, telecomunicaciones.

V. Coloque los términos donde correspondan según clasificaciones de los sistemas y herramientas de Minería de Datos

Tipos de base de datos minados	Tipo de Aplicación	Información
Tipos de conocimiento minado	Tipos de funcionalidad y de técnicas	



SOLUCIÓN DE GUÍA DE APOYO PARA EL ESTUDIANTE

Tema 1: Introducción a la Minería de Datos

I. Complete adecuadamente.

a) La Minería de Datos surge debido al **volumen** y **variedad** de información que se encuentra albergada en una **base** de **datos** digital.

b) **La Minería de Datos** es el proceso de la extracción de conocimientos útiles y comprensibles, de grandes cantidades de datos almacenados en diferentes formatos.

c) El **KDD** es **un proceso** complejo que incluye no solo la obtención de los **modelos** o **patrones**, la **evaluación** y posible **interpretación** de los mismos.

d) Un modelo **determinístico** es un modelo **matemático** donde las mismas entradas producirán invariablemente las mismas salidas.

II. Enumere.

a) Los retos de la Minería de Datos son:

1. Trabajar con grandes cantidades de datos.
2. Utilizar técnicas para analizar y extraer conocimientos útiles.

b) El **KDD** se define como el proceso no trivial de identificar los:

1. Patrones válidos.
2. Novedosos.
3. Potencialmente útil.
4. Comprensibles.

c) El sistema de **KDD** considera las etapas:

1. Selección de datos.
2. Limpieza de datos.
3. Integración de datos.
4. Transformación de datos.
5. Reducción de datos.
6. Minería de Datos.
7. Evaluación de los patrones.
8. Interpretación de resultados.

d) Donde se aplican la técnica de Minería de Datos:

1. Aplicaciones financieras.
2. Seguro y salud privada.
3. Educación.
4. Medicina.
5. Telecomunicaciones.
6. Otras áreas.

III. Escriba **V** si es Verdadero y **F** si es Falso según convenga.

a) Una de las ventajas de Minería de Datos es que las Bases de Datos pueden ser analizados **V**.

b) La Minería de Datos se usa en casos como fraudes, bancos, supermercados, etc. **V**.

c) Una Base de Datos relacional es una colección de relaciones es decir de tupla **F**.

d) Un modelo probabilístico en un modelo estadístico que nos ayuda a predecir la conducta de futuras repeticiones de un estudio aleatorio mediante la estimación de una probabilidad de ocurrencia de dicho evento concreto **F**.

IV. Selección múltiple.

1) Los tipos de datos lo conforman:
   - **a) Base de Datos relacionales, la World Wide Web.**
   - b) Predictivo y Descriptivo.
   - c) Tablas y Datos.
2) El proceso del **KDD** se refleja:
   - a) Inicio, Comprensión y Asociación.
   - b) Selección, Transformación, Interpretación.
   - **c) Selección, Procesamiento, transformación, Minería, Interpretación/ Evaluación, conocimiento.**
3) La relación con otras disciplinas:
   - a) Aplicaciones financieras y bancas, seguro y salud privada.
   - **b) Base de datos, computación paralela, visualización, sistema de toma de decisiones, recuperación de información, estadística, aprendizaje automático, etc.**
   - c) Educación, medicina, telecomunicaciones.

V. Coloque los términos donde correspondan según clasificaciones de los sistemas y herramientas de Minería de Datos

Tipos de base de datos minados	Tipo de Aplicación	Información
Tipos de conocimiento minado	Tipos de funcionalidad y de técnicas	

Véase ilustración 7 de este documento

13) TEMA 2: EL PROCESO DE EXTRACCIÓN DE CONOCIMIENTO.

Objetivos:

- Obtener conocimientos de las fases de extracción de conocimiento.

Contenidos:

- Las fases del proceso de extracción de conocimiento.
- Fases de integración y recopilación.
- Fases de selección, limpieza y transformación.
- Fases de Minería de Datos.
- Fases de evaluación e interpretación.
- Fases de difusión, uso y monitorización.

Duración: 4 hrs.

Bibliografía:

- Introducción a Minería de Datos. José Hernández Orallo, Mª José Ramírez Quintana, Cesar Ferri Ramírez.

13.1 Las fases del proceso de extracción de conocimiento.

El proceso del **KDD** se organiza en torno a cinco fases:

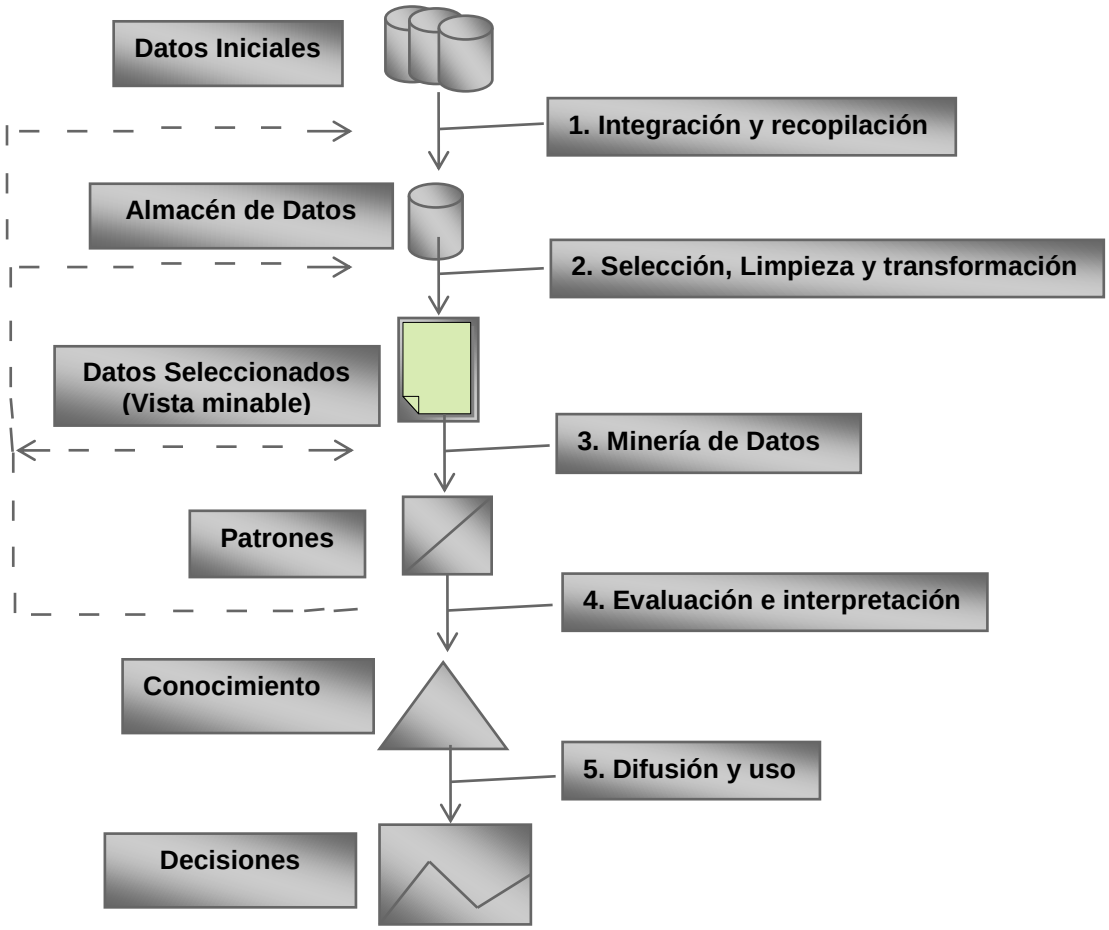


Ilustración 9: Fases del proceso de descubrimiento de conocimiento en bases de datos, KDD.

- **Integración y recopilación:** Determina las fuentes de información que pueden ser útiles y donde conseguirla.
- **Selección, limpieza y transformación:** Se eliminan o corrigen los datos incorrectos y se decide la estrategia a seguir con los datos incompletos.
- **Minería de Datos:** Se decide cual es la tarea a realizar y el método a utilizar.
- **Evaluación e interpretación:** Se evalúan los patrones y se analizan por los expertos y si es necesario vuelve a la fase anterior para una nueva iteración.
- **Difusión y uso:** Se hace uso de un nuevo conocimiento y se hace partícipe de el a todos los posibles usuarios

13.2 Fases de integración y recopilación.

Las bases de datos y las aplicaciones basadas en el procesamiento tradicional de datos, que se conoce como **procesamiento tradicional en línea (OLTP, On-Line Transaction Processing)** son suficientes para cubrir las necesidades diarias de una organización sin embargo, resultan insuficiente para otras funciones más complejas como: **análisis, la planificación y la predicción.**

La idea de integración de múltiples base de datos ha dado lugar a la tecnología de **almacenen de datos (data warehousing)**. Este término hace referencia a la tendencia actual en las empresas e instituciones de coleccionar datos de las bases de datos transaccionales y otras fuentes diversas para hacerlos accesibles para el análisis y toma de decisiones.

Ilustración 10: Integración en un almacén de datos.

Un almacén de datos es un repositorio de información coleccionada desde varias fuentes, almacenadas bajo un esquema unificado que normalmente reside en un único emplazamiento.

Los almacenes de datos son utilizados para poder agregar y cruzar eficientemente la información de maneras sofisticadas. Por ello los datos se modelan en una estructura de datos multidimensional, donde cada dimensión corresponde a un atributo o conjunto de atributos.

Esta visión multidimensional hace a los almacenes de datos adecuados para el **procesamiento analítico en línea** (**On-line analytical Processing, OLAP**).

Las operaciones **OLAP** permiten el análisis multidimensional de los datos que es superior al **SQL** para computar resúmenes y desgloses de muchas dimensiones.

13.3 ¿Cuál es la diferencia entre minería y OLAP?

La diferencia está en que el usuario de **OLAP** utiliza la herramienta para obtener información agregada a partir de información detallada, combinando la información de manera flexible. Lo que permite obtener informes y vistas sofisticadas en tiempo real.

Las herramientas **OLAP** también son usadas para comprobar rápidamente patrones y pautas hipotéticas por el usuario con el objetivo de verificarlas o rechazarla.

La **Minería de Datos** más que verificar patrones hipotéticos usa los datos para encontrar estos patrones lo cual hace al proceso que sea inductivo, ambos tipos de herramientas se complementan, podemos usar **OLAP** al principio del proceso de **KDD** para explorar los datos entre más se comprueban los datos más efectivos será el proceso de descubrir conocimientos.

13.4 Fases de selección, limpieza y transformación.

La calidad de conocimiento descubierto no sólo depende del algoritmo de minería utilizado, sino que también de la calidad de los datos, esta fase del **KDD** es seleccionar y reparar el subconjunto de datos que se va a minar, lo cual constituyen a lo que se conoce como vista minable.

Es necesario realizar este paso ya que algunos datos seleccionados en la etapa anterior son irrelevantes o innecesarios, para la tarea de minería que se desea realizar.

Además que los datos sean irrelevantes existen otros problemas que afectan la calidad de los datos:

- La presencia de valores que no se ajustan al comportamiento general de los datos (**outliers**).
- La presencia de datos faltantes o perdidos (**missing values**) pueden ser un problema pernicioso que puede conducir a resultados pocos precisos.

Estos dos problemas son únicamente dos ejemplos que muestran la necesidad de la limpieza de datos, es decir, de mejorar su calidad. Como hemos dicho no es sólo suficiente con tener una buena calidad de datos.

Ejemplo:

Supongamos que los jueces del torneo de Wimbledon desean determinar a partir de las condiciones climatológicas (nubosidad, humedad, temperatura, etc…) si se puede jugar o no al tenis, para ello cuentan con los datos recogidos de experiencias anteriores. Probablemente la base de datos contenga un atributo que identifica cada uno de los días considerados (por ejemplo, la fecha). Si consideramos este atributo en el proceso de minería, un algoritmo de generación de regla podría obtener reglas como:

SI (fecha=10/06/2003) ENTONCES (jugar_tenis=sí)

Esta fórmula aunque es correcta es inútil para realizar predicciones futuras.

Otra tarea de preparación de los datos es la construcción de atributos, la cual consiste en construir automáticamente nuevos atributos aplicando alguna operación o función a los atributos originales con el objeto de que estos nuevos atributos hagan más fácil el proceso de minería.

La motivación principal para esta tarea es fuerte cuando los atributos originales no tienen mucho poder predictivo por si solo o los patrones dependen de variaciones lineales de las variables originales.

El tipo de los datos pueden también modificarse para facilitar el uso de técnicas que requieren tipos de datos específicos, se puede numerizar, lo que reduce el espacio y permite usar técnicas numéricas.

El proceso inverso consiste en discretizar los atributos continuos, es decir transformar valores numéricos en atributos discretos o nominales. Los atributos discretizados pueden tratarse como atributos categóricos con un número más pequeño de valores. La idea básica es partir los valores de un atributo continuo en una pequeña lista de intervalos, tal que cada intervalo es visto como un valor discreto del atributo.

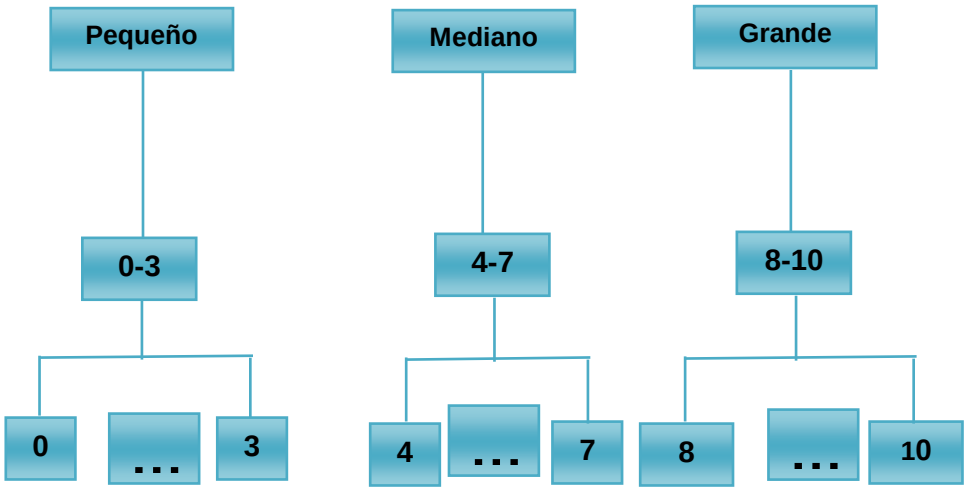


Ilustración 11: Ejemplo de discretización del atributo tamaño.

En la Ilustración se ilustra una posible discretización para el atributo tamaño, con valores de 0 a 10. En la parte inferior de la Ilustración muestra la lista ordenada de los valores continuos, los cuales se han discretizado en tres intervalos a los que se les ha asignado los valores discretos pequeños, medianos y grandes, como se puede observar en la parte superior de la ilustración.

13.5 Fase de Minería de Datos.

La fase de Minería de Datos es la más característica del **KDD** y, por esta razón muchas veces se utiliza esta fase para nombrar todos los procesos.

El objetivo de esta fase es producir nuevos conocimientos que puede utilizar el usuario esto se realiza construyendo un modelo basado en los datos recopilados para este efecto el modelo es una descripción de los patrones y relaciones entre los datos que pueden usarse para hacer predicciones, para entender mejor los datos o para explicar situaciones pasadas.

Para ello es necesario tomar una serie de decisiones antes de empezar el proceso:

- Determinar qué tipo de tarea de minería es el más apropiado.
- Elegir el tipo de modelo.
- Elegir el algoritmo de minería que resuelva la tarea y obtenga el tipo de modelo que estamos buscando.

Tarea de la Minería de Datos.

Dentro de la Minería de Datos hemos de distinguir tipos de tareas las cuales puede considerarse como un tipo de problema a ser resuelto por un algoritmo de Minería de Datos. Esto significa que cada tarea tiene sus propios requisitos, y que el tipo de información obtenida con una tarea puede diferir mucho de la obtenida con otra.

Las distintas tareas pueden ser **predictivas** o **descriptivas**.

Entre las tareas predictivas encontramos:

- **La clasificación:** Es la tarea más utilizada, en ella cada instancia pertenece a una clase, la cual se indica mediante el valor de un atributo que llamamos **la clase de la instancia**. Este atributo puede tomar diferentes valores discretos, cada uno de los cuales corresponden a una clase, el resto de los atributos de la instancia se utilizan para predecir la clase. El objetivo es predecir la clase de nuevas instancias de las que se desconoce la clase.

Existen variantes de la tarea de la clasificación, como son el aprendizaje de “rankings”, el aprendizaje de preferencias, el aprendizaje de estimadores de probabilidad, etc.

- **La regresión:** Consiste en aprender una función real que asigna a cada instancia un valor real, la principal diferencia respecto a **la clasificación** es que el valor a predecir es numérico. El objetivo en este caso es minimizar el error entre el valor predicho y el valor real.

Entre las tareas descriptivas encontramos:

- **El agrupamiento (clustering):** Es la tarea descriptiva por excelencia y consiste en obtener grupos "naturales" a partir de los datos. Se habla de grupo y no de clase porque a diferencia de la clasificación, en lugar de analizar datos etiquetados con una clase, lo utiliza para generar esta etiqueta. Los datos son agrupados basándose en el principio de maximizar la similitud entre los elementos de un grupo minimizando la similitud entre los distintos grupos. Al agrupamiento se le llama también segmentación, ya que parte o segmenta los datos en grupos que pueden ser o no disjuntos.
- **Las correlaciones:** Se usa para examinar el grado de similitud de los valores de dos variables numéricas. Una formula estándar para medir la correlación es el coeficiente de correlación **r**, el cual es un valor real comprendido entre **-1** y **1**, si **r** es **1** las variables están perfectamente correlacionadas, mientras que si es **0** no hay correlación. Esto quiere decir que **r** es positivo las variables tienen un comportamiento similar y cuando **r** es negativo si una variable crece la otra decrece. El análisis de correlaciones sobre todo las negativas, pueden ser útil para establecer reglas de ítems correlacionados.
- **Las reglas de asociación:** Son similares a **las correlaciones**, tiene como objetivo identificar relaciones no explicitas entre atributos **categóricos**. Las reglas de asociación no implican una relación **causa-efecto**, es decir, no puede existir una causa para que los datos estén asociados. Esta tarea es frecuentemente utilizada para el análisis de la cesta de la compra.

En un caso especial de **reglas de asociación**, esta recibe el nombre de:

- **Reglas de asociación secuencial:** Se usa para determinar patrones secuenciales en los datos. Estos patrones se basan en secuencias temporales de acciones y difieren de las reglas de asociación en que las relaciones entre los datos se basan en el tiempo.

Técnica de minería.

La Minería de Datos es un campo muy interdisciplinar, existen diferentes paradigmas detrás de las técnicas utilizadas para esta fase, entre los cuales se destacan:

- **Técnicas de inferencia estadística.**
- **Arboles de decisión.**
- **Redes neuronales.**
- **Inducción de reglas.**
- **Aprendizaje basado en instancias.**
- **Algoritmos genéticos.**
- **Algoritmo bayesiano.**
- **Programación lógica inductiva.**
- **Y varios métodos basados en núcleos.**

Cada uno de estos 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 a lo que podríamos llamar el método universal aplicable a todo tipo de aplicación.

Existen muchos conceptos **estadísticos** que son base de la Minería de Datos. Ya que hemos mencionado la regresión lineal, como un método simple pero frecuentemente utilizado para la tarea de regresión.

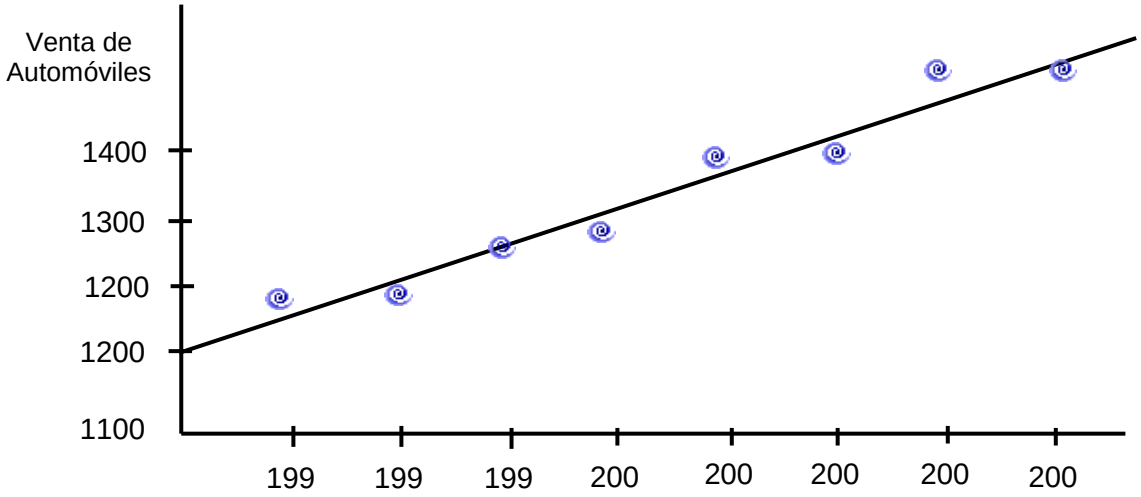


Ilustración 12: Ejemplo de regresión lineal.

Algunas técnicas de discriminantes no paramétricos tiene una relación muy estrecha con los **métodos basados en núcleo**, de los cuales las máquinas de vectores soporte son un ejemplo más representativo, en el que se busca un discriminante lineal que maximice la distancia a los ejemplos fronterizos de los distintos grupos o clases.

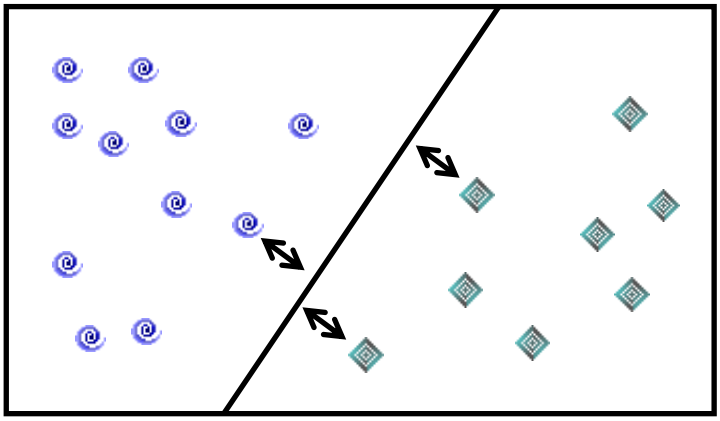


Ilustración 13: Ejemplo de un discriminante (clasificador) basado en vectores soporte.

En la Ilustración se muestra el uso de un clasificador para distinguir en una cooperativa agrícola de naranjas que son aptas para consumo directo o para zumo, cuales se preparan para conserva y procesos industriales, según su diámetro y peso.

En este caso el discriminante líneas es muy fácil de encontrar, lo interesante de esta técnica es que, en problemas no lineales, este discriminante lineal también se puede encontrar porque se utilizan núcleos para convertir el problema de un problema de mayor dimensionalidad y porque se relajan ciertas condiciones.

La idea que subyace en los **métodos bayesianos** es calcular, para una instancia dada sin clasificar, cual es la probabilidad de que se le asigne cada una de las clases, y seleccionar la de mayor probabilidad.

Uno de los métodos más utilizados es el **naive bayes**, que se basa en la regla de bayes y que ingenuamente asume la independencia de los atributos dada la clase. Este método funciona bien con bases de datos reales, sobre todo cuando se combina con otros procedimientos de selección de atributos que sirven para eliminar redundancia.

La regla de bayes establece que, si tenemos una hipótesis **H** sustentada para una evidencia **E**, entonces:

$$p(H\backslash E) = \frac{P(E\backslash H) * p(H)}{p(E)}$$

Veamos la función con un ejemplo:

Una compañía de seguros consta con los siguientes datos de sus clientes, clasificados en buenos y malos clientes.

#instancia	Edad	Hijos	Practica_deporte	Salario	Buen_cliente
1	Joven	Si	No	Alto	Si
2	Joven	No	No	Medio	No
3	Joven	Si	Si	Medio	No
4	Joven	Si	No	Bajo	Si
5	Mayor	Si	No	Bajo	Si
6	Mayor	No	Si	Medio	Si
7	Joven	No	Si	Medio	Si
8	Joven	Si	Si	Alto	Si
9	Mayor	Si	No	Medio	Si
10	Mayor	No	No	Bajo	No

Tabla 4: Datos de una compañía de seguros

Supongamos que tenemos un nuevo ejemplo con los siguientes valores:

Edad	Hijos	Practica_deporte	salario	Buen_cliente
Mayor	No	No	Medio	?

Tabla 5: Nuevo ejemplo de prácticas de deporte.

La hipótesis **H** es que buen_cliente sea **si** (o, alternativamente, **no**). La evidencia **E** es una combinación de los valores de los atributos edad, hijos, practica_deporte, y salario del dato nuevo, por lo que su probabilidad se obtiene multiplicando las probabilidades de estos valores. Es decir,

$$p(si\backslash \mathbf{E}) = \frac{[P(edad\ |sí) * p(practica\_deporte\ |sí) * p(salario\ |sí) * p(si)]}{p(E)}$$

Los arboles de decisión son una serie de decisiones o condiciones organizadas en forma jerárquica, a modo de árbol. Son muy útiles para encontrar estructuras en espacio de alta dimensionalidad y en problemas que mezclen datos categóricos y numéricos. Esta técnica se usa en tareas de **clasificación**, **agrupamiento** y **regresión**

Cuando los árboles de decisión usados para predecir variables categóricas reciben el nombre de árboles de clasificación, ya distribuyen las instancias en clase.

Cuando los árboles de decisión se usan para predecir variables continuas se llaman árboles de regresión.

Ejemplo de árbol de clasificación:

Usaremos como ejemplo un típico problema muy utilizado en el aprendizaje automático. Se trata de un conjunto de datos ficticios que muestra las condiciones climatológicas (pronóstico, humedad y viento) adecuadas para jugar un cierto deporte (por ejemplo: tenis en Wimbledon). Los datos de los que disponemos son los siguientes:

#instancia	Pronostico	Humedad	Viento	Jugar
1	Soleado	Alta	Débil	No
2	Cubierto	Alta	Débil	Si
3	Lluvioso	Alta	Débil	Si
4	Lluvioso	Normal	Fuerte	No
5	Soleado	Normal	Débil	Si
….	…	…	…	…

Tabla 6: Tabla de pronóstico.

Usando un algoritmo de aprendizaje de árboles de decisión obtendremos el siguiente árbol:

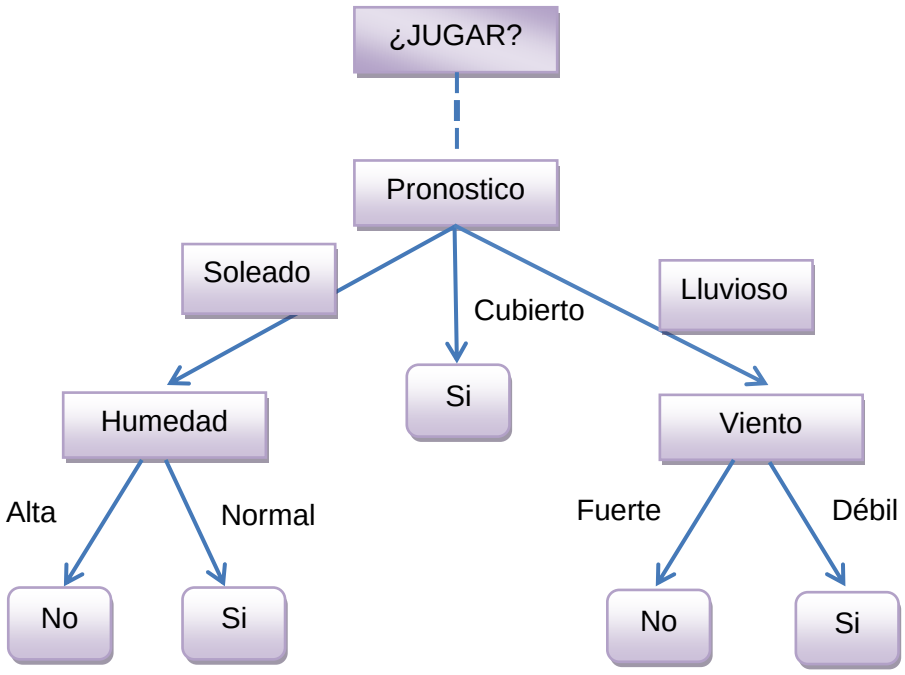


Ilustración 14: Árbol de decisión para determinar si se juega o no a un cierto deporte.

Los arboles de decisión siguen una aproximación “divide y vencerás” para partir el espacio del problema en subconjuntos. Encima del nodo raíz del árbol tenemos el problema a resolver. En nuestro ejemplo, se trata de decidir si jugar o no. Los nodos (nodos de decisión) corresponden a particiones sobre atributos particulares, como por ejemplo pronóstico, y los arcos que emanan de un nodo corresponden a los posibles valores del atributo considerado en ese nodo. Cada arco conduce a otro nodo de decisión o a un nodo hoja. Los nodos de hoja representan la precisión (o clase) del problema para todas aquellas instancias que alcanzan esa hoja.

Los arboles de decisión pueden considerase una forma de aprendizaje de reglas, ya que cada rama del árbol puede interpretarse como una regla, donde los nodos internos en el camino desde la raíz a las hojas definen el término de la conjunción que constituye el antecedente de la regla, y la clase asignada en la hoja es el consecuente.

Aunque los arboles de decisión pueden también producir un conjunto de reglas, los métodos de inducción de reglas son diferentes ya que:

- Las reglas son independientes y no tienen por qué formar un árbol.
- Las reglas generadas pueden no cubrir todas las situaciones posibles.

- Las reglas pueden entrar en conflicto en sus predicciones, en este caso, se debe de elegir que reglas se debe de seguir. Un método para resolver los conflictos consiste en asignar un valor de confianza a las reglas y usar la que tenga mayor confianza.

Algunos métodos de obtención de reglas, en especial para la tarea de reglas de asociación, se basan en el concepto de conjunto de ítems frecuentes (frequent ítem sets) y utilizan técnicas de conteo y soporte mínimo para obtener las reglas.

Las condiciones en el antecedente anterior de las reglas pueden ser comparaciones entre un atributo y uno de los valores de su dominio, o bien un par de atributos (siempre que sean el mismo dominio o de dominio compatible). Usando la terminología de la lógica, las expresiones del primer tipo forman condiciones proporcionales, mientras que las del segundo tipo forman condiciones de primer orden.

El propósito de la **programación lógica inductiva** (del inglés, Inductive logic programming, **ILP**) es permitir una representación más rica que los métodos proporcionales.

Los métodos **ILP** incorporan de forma natural conocimiento de base y pueden usarse para destruir patrones que afecten a varias relaciones.

Las **redes neuronales artificiales** son todo un paradigma de computación muy potente que permite modelizar problemas complejos en los que puede haber interacciones no lineales entre variables, como los arboles de decisión, las redes neuronales pueden usar en problemas de clasificación, de regresión y de agrupamiento. Las redes neuronales trabajan directamente con datos numéricos. Para usarlas con datos nominales estos deben numerizarse primero.

Una red neuronal puede verse como un gráfico dirigido con muchos nodos (elemento del proceso) y arcos entre ellos (sus interconexiones). Cada uno de los elementos funciona independientemente de los demás, usando datos locales (entrada y salida del nodo) para dirigir su procesamiento.

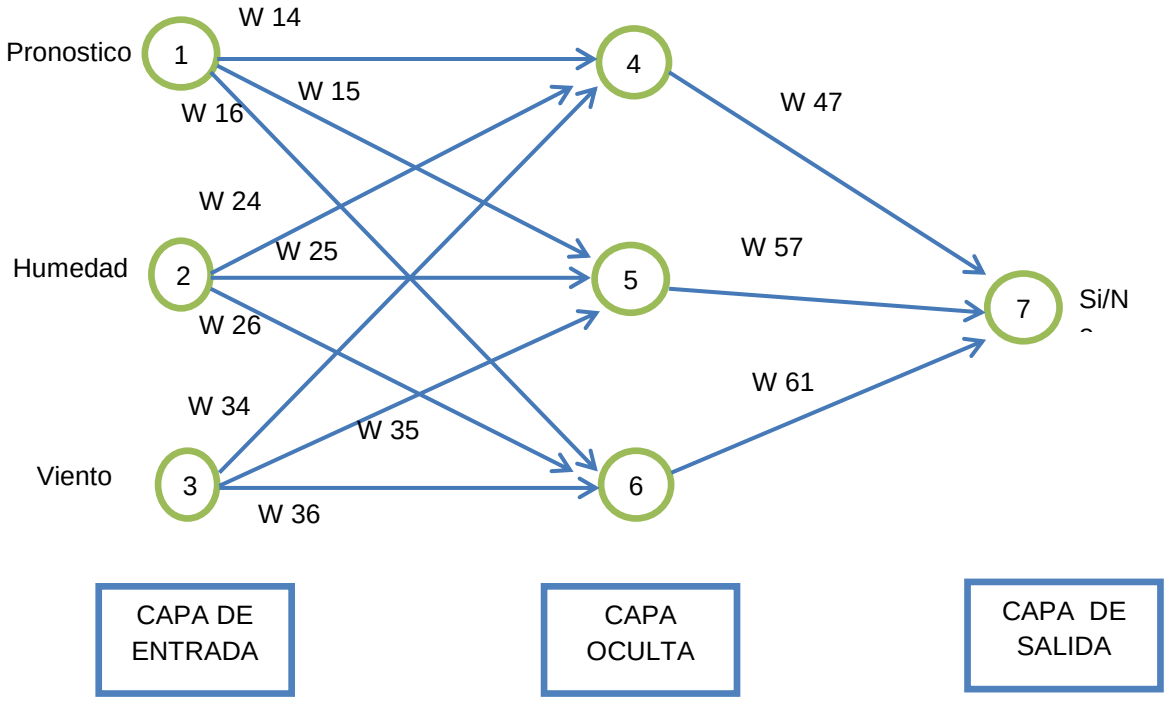


Ilustración 15: Red neuronal para el problema de jugar un cierto deporte

La organización más popular de una red neuronal consta de una capa de entrada, en la que cada nodo corresponde a una variable independiente a examinar, unos nodos internos organizados en una o varias capa oculta y una capa de salida con los nodos de salida(los posibles valores de las variables objeto).

Los pesos de conexión (W) son parámetros desconocidos que deben estimarse por un método de entrenamiento. El método comúnmente utilizado es el de propagación hacia atrás (back propagation). La idea básica es reducir el valor de error de la salida de la red.

Las redes neuronales tienen una gran capacidad de generalización para problemas no lineales, aunque requieren bastantes datos para su entrenamiento.

Su mayor desventaja no obstante que el modelo aprendido es difícilmente comprensible.

En el **aprendizaje basado en instancias o casos**, las instancias se almacenan en memoria, de tal forma que cuando llega una nueva instancia cuyo valor es desconocido se intenta relacionar esta con las instancias almacenadas.

Todo el trabajo en el aprendizaje basado en instancias o casos se hace cuando llega una instancia a clasificar y no cuando se procesa el conjunto de entrenamiento. En este sentido se trata de un método retardado o perezoso, ya que retrasa el trabajo real tanto como sea posible, a diferencia de los otros métodos vistos hasta el momento que tiene un

comportamiento anticipativo o voraz. Produciendo generalizaciones en cuanto reciben los datos de entrenamiento.

En el aprendizaje basado en instancias, cada nueva instancia se compara con las existentes usando una métrica de distancia, y la instancia más próxima se usa para asignar su clase a la nueva instancia.

La variante más sencilla de este método de clasificación es conocido como “el vecino más próximo” (nearest-neighbor). Otra variante, conocida como el método de los “k vecinos más próximos” (k-nearest-neighbor).

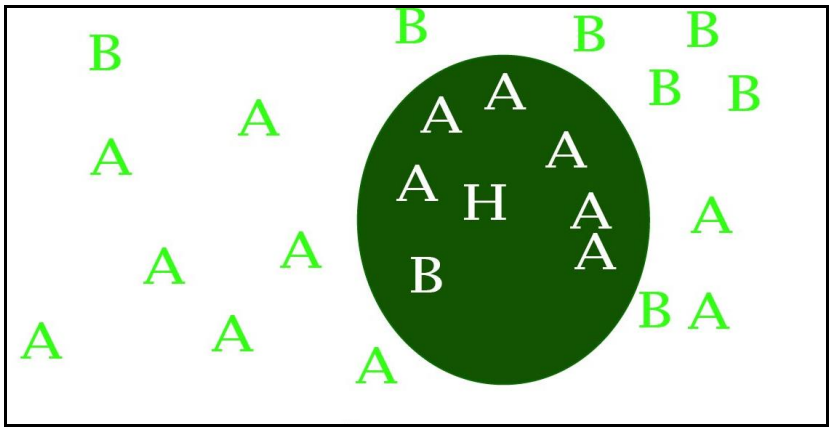


Ilustración 16: K-vecinos más próximos

El aprendizaje basado en instancias es muy útil para trabajar sobre tipos no estándar, como los textos o multimedia.

Los **algoritmos evolutivos** son métodos de búsqueda colectiva en el espacio de soluciones. Dada una población de potenciales soluciones a un problema, la computación evolutiva expande esta población con nuevas y mejores soluciones.

El nombre se debe a que siguen los patrones de la evolución biológicas. En la Minería de Datos, los algoritmos genéticos se pueden usar para el agrupamiento, la clasificación y las reglas de asociación, así como para la selección de atributos.

Los algoritmos genéticos también pueden usarse para guiar a otros algoritmos de Minería de Datos en el proceso de aprendizaje.

13.6 Fases de evaluación e interpretación.

Medir la calidad de los patrones descubiertos por un algoritmo de Minería de Datos no es un problema trivial, ya que esta medida puede atañer a varios criterios, algunos de ellos muy subjetivos.

Los patrones descubiertos deben de tener tres cualidades:

- **Ser precisos.**
- **Comprensibles.**
- **Interesantes.**

Técnicas de evaluación
Para entrenar y probar un modelo se parten los datos en dos conjuntos:

- **El conjunto de entrenamiento (training set)**
- **El conjunto de prueba o de test (test set)**

Esta separación es necesaria para garantizar que la validación de la precisión del modelo es una medida independiente. Si no se usan conjuntos diferentes de entrenamiento y prueba, la precisión del modelo será sobre estimada, ósea, tendremos estimaciones muy óptimas.

En los modelos predictivos, el uso de esta separación entre entrenamiento y prueba es fácil de interpretar.

Por ejemplo:

Para la tarea de clasificación, después de generar el modelo con el conjunto de entrenamiento, este se puede usar para predecir la clase de los datos de prueba (test) entonces, la razón de precisión, se obtiene dividiendo el número de clasificaciones correctas por el número total de instancias.

La precisión es una buena estimación de cómo se comportara el modelo para datos futuros similares a los de test. Esta forma de proceder no garantiza que el modelo sea correcto, sino que simplemente indica que si usamos la misma técnica con una base de datos con datos similares a los de prueba, la precisión media será bastante parecida a la obtenida con estos.

El método de evaluación más básico, **la validación simple**, reserva un porcentaje de la base de datos como conjunto de prueba, y no lo usa para construir el modelo. Este porcentaje suele variar entre cinco y el 50 por ciento. La división de los datos en estos dos grupos debe ser aleatoria para que la estimación sea correcta.

Si tenemos una cantidad no muy elevada de datos para construir el modelo, puede que no podamos permitirnos el lujo de reservar parte de los mismos para la etapa de evaluación. En estos casos se usa un método conocido como **validación cruzada (Cross validation)**.

Los datos se dividen aleatoriamente en dos conjuntos equitativos con los que se estima la precisión predictiva del modelo. Para ello primero se construye un modelo con el primer conjunto y se usa para predecir los resultados del segundo conjunto y calcular así un ratio de error (o de precisión). A continuación se construye un modelo con el segundo conjunto y se usa para predecir los resultados del primer conjunto, obteniéndose un segundo ratio de error. Finalmente se construye un modelo con todos los datos, se calcula un promedio de los ratios de error y se usa para estimar mejor su precisión.

El método que se usa normalmente es la **validación cruzada con *n* pliegues (n-flod Cross validation)**. En este método se divide aleatoriamente en *n* grupos. Un grupo se reserva para el conjunto de prueba y con los otros *k-1* restantes (juntando todos sus datos) se construye un modelo y se usa para predecir el resultado de los datos del grupo reservado, este proceso se repite *n* veces, dejando cada vez un grupo diferente para la prueba. Esto significa que se calculan *n* ratios de error independientes. Finalmente se construye un modelo con todos los datos y se obtienen sus ratios de error y precisión promediando los *n* ratios de errores disponibles.

Otra técnica para estimar error de un modelo cuando se disponen de pocos datos es la conocida como **boot strapping**. Esta consiste en construir primero un modelo con todos los datos iniciales. Entonces, se crean numerosos conjuntos de datos, llamados **boot strap simples** haciendo un muestreo con los datos originales con reemplazo, es decir se van seleccionando instancias del conjunto inicial, pudiendo seleccionar la misma instancia varias veces.

Medidas de evaluación de modelos

La medida de evaluación de modelos dependerá de la tarea de Minería de Datos ya que existen diferentes medidas de evaluación de los modelos.

Por ejemplo:

La clasificación lo normal es evaluar la calidad de los patrones encontrados con respecto a su **precisión predictiva**. La cual se calcula como el número de instancias de conjunto de prueba clasificada correctamente dividiendo por el número de instancia totales en el conjunto de prueba. El objetivo es obtener la mayor precisión posible sobre el conjunto de test.

Ya que obtener el 100 por ciento de precisión sobre el conjunto de entrenamiento es trivial, bastaría con generar una regla para cada instancia usando una conjunción como sus variables valores, como antecedente de la regla (parte “SI”) y el valor a predecir como consecuente (parte “ENTONCES”)

Por ejemplo: Para la cuarta instancia del ejemplo de la tabla podríamos generar una regla como:

SI #instancia =4 **Y** edad=joven **Y** hijos=si **Y** practica_deporte=no **Y** salario=bajo **ENTONCES** buen_cliente=si

O incluso, usando algún atributo que pueda servir como clave primaria, por ejemplo:

SI #instancia =4 **ENTONCES** buen_cliente=si

En el caso que la tarea sea de **reglas de asociación**, se suele evaluar de forma separada cada una de las reglas con objeto de restringirnos a aquellas que pueden aplicarse a un mayor número de instancias y que tienen una precisión relativamente alta sobre estas instancias. Esto se hace en base a dos conceptos:

- Cobertura: número de instancia a las que la regla se aplica y predice correctamente.

- Confianza: proporción de instancias que la regla predice correctamente, es decir la cobertura dividida por el número de instancias que se puede aplicar la regla.

Siguiendo con el ejemplo de determinar si se juega a un deporte, consideremos:

#instancia	Pronostico	Humedad	Viento	Jugar
1	Soleado	Alta	Débil	No
2	Cubierto	Alta	Débil	Si
3	Lluvioso	Alta	Débil	Si
4	Lluvioso	Normal	Fuerte	No
5	Soleado	Normal	Débil	Si

Tabla 7: Tabla de pronóstico utilizando reglas de asociación.

Que son los mismos valores anteriormente vistos, entonces la regla

SI pronostico=soleado **Y** viento=débil **ENTONCES** jugar=si

Tendrá una cobertura de 1, es decir, el número de días soleados y con viento débil en los que se recomienda jugar (instancia 5) y una confianza de ½ (ya que la regla también se puede aplicar a la instancia 1).

Si la tarea es **regresión**, la salida del modelo es un valor numérico, la manera más habitual de valorar el modelo es mediante el error cuadrático medio del valor predicho respecto al valor que se utiliza como validación. Esto promedia los errores y tiene más en cuenta aquellos errores que se desvían más del valor predicho (ponderación cuadrática). Aunque se puede utilizar otras medidas del error en regresión, esta es quizá la más utilizada.

Para la tarea de **agrupamiento**, las medidas de evaluación suelen depender del método utilizado, aunque suelen ser función de la cohesión de cada grupo y de la separación entre grupos. La cohesión y separación entre grupos se puede formalizar. **Por ejemplo**: utilizando la distancia media al centro del grupo de los miembros de un grupo y la distancia media entre grupos, respectivamente. El concepto de distancia y de densidad son dos aspectos cruciales tanto en la construcción de modelos de agrupamiento como en su evaluación.

El concepto de distancia y de densidad son dos aspectos cruciales tanto en la construcción de modelos de agrupamiento como en su evaluación.

Además de las medidas comentadas, existen otras medidas más subjetivas, como:

- El interés.
- La novedad.
- La simplicidad.

- La comprensibilidad.

13.7 Fases de difusión, uso y monitorización.

Una vez construido y validado el modelo puede usarse principalmente con dos finalidades:

- Para que un analista recomiende acciones basándose en el modelo y en sus resultados.
- Para aplicar el modelo a diferentes conjuntos de datos.

También se pueden incorporar a otras aplicaciones como por ejemplo:
A un sistema de análisis de crédito bancario, que asista al empleado bancario a la hora de evaluar a los solicitantes de los créditos, o incluso automáticamente, como los filtros de **spam** o la detección de compras con tarjetas de créditos fraudulentas.

Tanto en el caso de una aplicación manual o automática del modelo, es necesario su difusión, es decir que se distribuyan y se comunique a los posibles usuarios, ya sea por cauces habituales dentro de la organización, reuniones, intranet, etc.

El nuevo conocimiento extraído debe integrar el **know-how** de la organización. Es importante medir lo bien que el modelo evoluciona. Aun cuando el modelo funcione bien debemos continuamente comprobar las prestaciones del mismo. Esto se debe principalmente que los patrones pueden cambiar.

Ejemplo:
En el caso de las ventas todos sabemos que se ven afectada por factores externos como la tasa de inflación, la cual altera el comportamiento de compra de la gente. Por lo tanto el modelo deberá ser monitorizado, lo que significa que de tiempo en tiempo el modelo tendrá que ser re-evaluado, re-entrenado y posiblemente reconstruido completamente.

GUÍA DE APOYO PARA EL ESTUDIANTE

Tema 2: El Proceso de extracción del conocimiento

I. Enumere

a) El entorno del **KDD** se organiza en torno a cinco fases:
b) Entre las tareas descriptivas encontramos :
c) Enumere las técnicas de Minería :

II. Complete.

a) La _____ es la tarea más utilizada, en ella cada instancia pertenece a una clase, la cual se indica mediante el valor de un atractivo que llamamos _____.
b) La fase de Minería de Datos es la más característica del _____ y, por esta razón muchas veces se utiliza esta fase para nombrar todos los procesos.
c) Las herramientas _____ también son usadas para comprobar rápidamente patrones y pautas hipotéticas por el usuario con el objetivo de verificarlas o rechazarla.

III. Conteste.

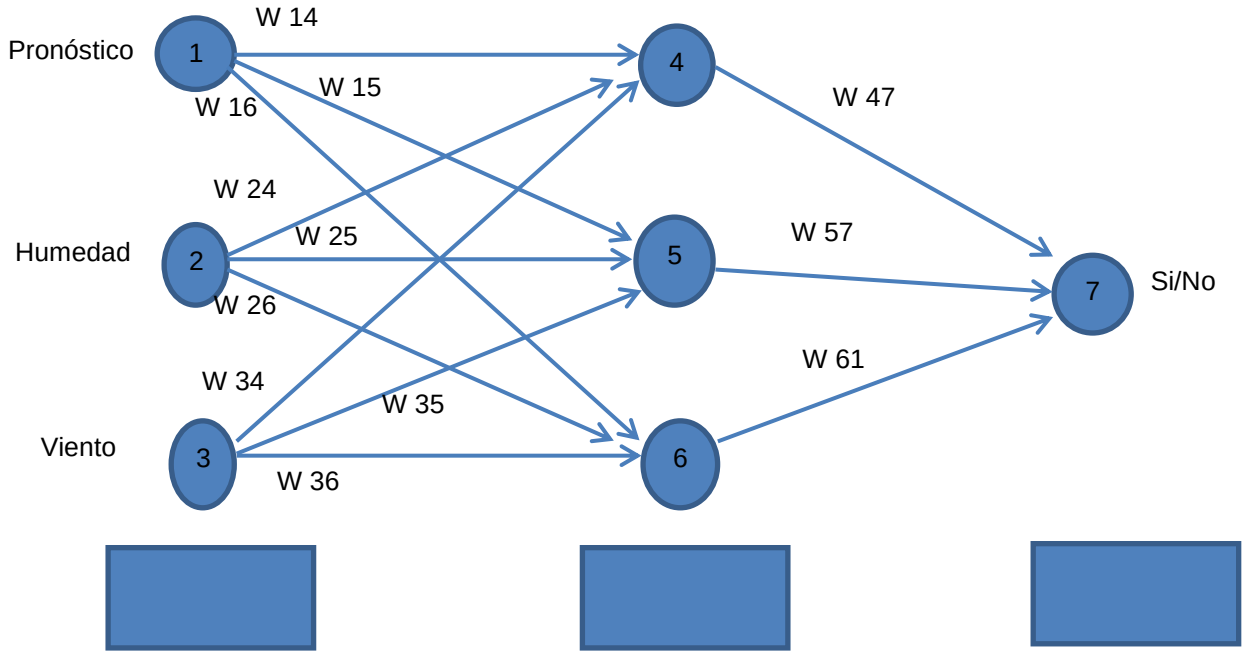
1) ¿Qué son los arboles de decisión?
2) ¿Qué permiten las redes neuronales artificiales?
3) ¿Cuál es el propósito de la lógica de programación inductiva?

IV. Escriba **V** si es Verdadero y **F** si es Falso según convenga.

a) Los métodos **ILP** incorporan de forma natural conocimiento de base y pueden usarse para destruir patrones que afecten a varias relaciones___.
b) Los arboles de decisión siguen una aproximación “resta y vencerás” para partir el espacio del problema en subconjuntos____.
c) La medida de evaluación de modelos dependerá de la tarea de Minería de Datos ya que existen diferentes medidas de evaluación de los modelos___.
d) La clasificación lo normal es evaluar la variabilidad de los patrones encontrados con respecto a su precisión predictiva ___

V. Coloque los términos donde correspondan.

Capa oculta	Capa salida	Capa de transición	Capa entrada



SOLUCIÓN DE GUÍA DE APOYO PARA EL ESTUDIANTE

Tema 2. El proceso de la extracción del conocimiento

I. Enumere

a) El entorno del **KDD** se organiza en torno a cinco fases:
   1. Integración y recopilación.
   2. Selección, limpieza y transformación.
   3. Minería de Datos.
   4. Evaluación e interpretación
   5. Difusión y Uso.

b) Entre las tareas descriptivas encontramos :
   1. El agrupamiento (clustering).
   2. Las correlaciones.
   3. Las reglas de asociación.

c) Enumere las técnicas de Minería
   1. Técnicas de inferencia estadística.
   2. Arboles de decisión.
   3. Redes neuronales.
   4. Inducción de reglas.
   5. Aprendizaje basado en instancias.
   6. Algoritmos genéticos.
   7. Algoritmo bayesiano.
   8. Programación lógica inductiva.
   9. Y varios métodos basados en núcleos.

II. Complete.

a) La **clasificación** es la tarea más utilizada, en ella cada instancia pertenece a una clase, la cual se indica mediante el valor de un atractivo que llamamos **la clase de la instancia**.

b) La fase de Minería de Datos es la más característica del **KDD** y, por esta razón muchas veces se utiliza esta fase para nombrar todos los procesos.

c) Las herramientas **OLAP** también son usadas para comprobar rápidamente patrones y pautas hipotéticas por el usuario con el objetivo de verificarlas o rechazarla.

III. Conteste.

1) ¿Qué son los arboles de decisión?

Rta: son una serie de decisiones o condiciones organizadas en forma jerárquica, a modo de árbol. Son muy útiles para encontrar estructuras en espacio de alta dimensionalidad y en problemas que mezclen datos categóricos y numéricos. Esta técnica se usa en tareas de clasificación, agrupamiento y regresión.

2) ¿Qué permiten las redes neuronales artificiales?
Rta: permite modelizar problemas complejos en los que puede haber interacciones no lineales entre variables, como los arboles de decisión, las redes neuronales pueden usar en problemas de clasificación, de regresión y de agrupamiento. Las redes neuronales trabajan directamente con datos numéricos. Para usarlas con datos nominales estos deben numerizarse primero.

3) ¿Cuál es el propósito de la lógica de programación inductiva?
Rta: es permitir una representación más rica que los métodos proporcionales

IV. Escriba **V** si es Verdadero y **F** si es Falso según convenga.

a) Los métodos **ILP** incorporan de forma natural conocimiento de base y pueden usarse para destruir patrones que afecten a varias relaciones **V**.
b) Los arboles de decisión siguen una aproximación "resta y vencerás" para partir el espacio del problema en subconjuntos **F**.
c) La medida de evaluación de modelos dependerá de la tarea de Minería de Datos ya que existen diferentes medidas de evaluación de los modelos **V**.
d) La medida de evaluación de modelos dependerá de la tarea de Minería de Datos ya que existen diferentes medidas de evaluación de los modelos **V**.
e) **La clasificación** lo normal es evaluar la variabilidad de los patrones encontrados con respecto a su **precisión predictiva F**.

VI. Coloque los términos donde correspondan.

Capa oculta	Capa salida	Capa de transición	Capa entrada

Véase ilustración 15 de este documento.

14) TEMA 3: PREPARACIÓN DE LOS DATOS

Objetivos:

- Comprender la importancia de la preparar datos en Data Mining.
- Conocer el proceso de preparación de los datos
- Distinguir dos usos diferentes del sistema de información: el procesamiento transaccional y el procesamiento analítico.

Contenido:

- Introducción
- Necesidad de los Almacenes de Datos
- OLTP y OLAP
- Datamarts
- Explotación de un almacén de datos. Operadores
- Implementación del almacén de datos. Diseño
- Carga y mantenimiento del almacén de datos
- Almacenes de datos y Minería de Datos

Duración: 6 hrs.

Bibliografía:

- Introducción a Minería de Datos. José Hernández Orallo, Mª José Ramírez Quintana, Cesar Ferri Ramírez.
- www.sinnexus.com/business_intelligence/datamining.aspx

- **Preparación de los Datos**

En esta parte se detallan las primeras fases: recopilación, limpieza y transformación. Para ello se introduce unas series de tecnologías: almacenes de datos, OLAP, técnicas simples del análisis multivariante, tratamiento de la dimensionalidad, de datos faltantes y anómalos, visualización básica de consulta, etc.

- **Recopilación. Almacenes de Datos**

Para poder comenzar a analizar y extraer algo útil de los datos es preciso, en primer lugar, disponer de ellos. Esto en algunos casos puede parecer trivial; se parte de un simple archivo de datos a analizar. En otros, la diversidad y tamaño de las fuentes hace que el proceso de recopilación de datos sea una tarea compleja, que requiere una metodología y una tecnología propia. En general, el problema de reunir un conjunto de datos que posibilite la extracción de conocimiento requiere decidir, entre otros aspectos, de qué fuentes, internas y externas, se van a obtener los datos, cómo se van a organizar, cómo se van a mantener con el tiempo y, finalmente, de qué forma se van a poder extraer parcial o totalmente, en detalle o agregados, con distintas "vistas minables" a las que podamos aplicar las herramientas concretas de Minería de Datos.

En este tema nos centramos en las tecnologías para realizar esta recopilación e integración. En particular introducimos la tecnología de los almacenes de datos y algunos conceptos relacionados, como las herramientas OLAP (On-Line Analytical Processing). Los almacenes de datos no son estrictamente necesarios para realizar Minería de Datos, aunque si extremadamente útiles si se va a trabajar con grandes volúmenes de datos, que varían con el tiempo y donde se desea realizar tareas de Minerías de Datos variadas, abiertas y cambiantes. Es importante destacar las diferencias entre análisis que se puede realizar con técnicas OLAP y con Minería de Datos (aunque exista un cierto solapamiento entre ambas), así comprender que ambas tecnologías son complementarias. Finalmente, la selección, la transformación y limpieza de datos serán tratadas más adelantes, aunque algunas de estas operaciones pueden hacerse antes o durante el proceso de recopilación e integración.

14.1 Introducción

El primer paso en el proceso de extracción de conocimiento a partir de datos es precisamente reconocer y reunir los datos con los que se va a trabajar. Si esta recopilación se va a realizar para una tarea puntual y no involucra muchas cantidades y variedades de datos simples, es posible que el sentido común sea suficiente para obtener un conjunto de datos con la calidad suficiente para poder empezar a trabajar. En cambio, si requerimos datos de distintas fuentes, tanto externas como internas a la organización, con datos complejos y variados, es posiblemente en grandes cantidades y además cambiantes, con los que se desee realizar a medio o largo plazo diversas tareas de Minería de Datos, es posible que nuestro sentido común no sea suficiente para hacer una recopilación e integración en condiciones.

Al igual que la tecnología de bases de datos ha desarrollado una serie de modelos de datos (como el relacional), de lenguajes de consulta y actualización, de reglas de actividad, etc.

Para trabajar con la información transaccional de una organización, veremos que existe una tecnología relativamente reciente, denominada “almacenes de datos” (data warehouses) que pretende proporcionar metodologías y tecnología para recopilar e integrar los datos históricos de una organización, cuyo fin es el análisis, la obtención de resúmenes e informes complejos y la extracción de conocimientos. Esta tecnología está diseñada especialmente para organizar grandes volúmenes de datos de procedencia generalmente estructurada (base de datos relaciones, por ejemplo), aunque el concepto general es útil para la organización de pequeños conjuntos de datos en aplicaciones de Minería de Datos más modestas.

Supóngase que en una compañía bien implantada en el ámbito europeo queremos analizar aquellos países y gamas de productos en los que las ventas vayan excepcionalmente bien (con el objetivo , por ejemplo, de premiar a las oficinas comerciales de cada gama y producto) o, dicho de una manera más técnica , averiguar si la penetración relativa (teniendo en cuenta la permeabilidad del país en cuestión) de una gama de productos es significativamente mayor que la media de penetración en el conjunto del continente. La compañía dispone, por supuesto, de una base de datos transaccional sobre la que operan todas las aplicaciones de la empresa: producción, ventas, facturación, proveedores, nominas, etc. Lógicamente, de cada venta se registra la fecha, la cantidad y el comprador y, de este, el país. Con toda esta información histórica nos podemos preguntar: ¿es esta información suficiente para realizar el análisis anterior? La respuesta, a primera vista, quizá de manera sorprendente, es negativa. Pero, aparentemente, si tenemos detalladas las ventas de tal manera que una consulta SQL puede calcular las ventas por países de todos los productos y gamas, ¿Qué más puede hacer falta?

Sencillamente, la respuesta hay que buscarla fuera de la base de datos, en el contexto donde se motiva el análisis. La penetración de un producto depende de las ventas por habitante. Si no tenemos en cuenta la población de cada país la respuesta del análisis estará sesgada; será muy probable que entre los países con mayor penetración siempre esté Alemania y entre los países con menor penetración se encuentra San Marino. Pero no solo eso, es posible que, si deseamos hacer un análisis más perspicaz, nos interese saber la renta per cápita de cada país incluso la distribución por edad de cada país. Dependiendo de la gama, nos puede interesar información externa verdaderamente específica. Por ejemplo, las horas de sol anuales de cada país pueden ser una información valiosísima para la compañía de cosméticos. Lógicamente es más difícil vender bronceadores en Lituania que en Grecia o, dicho más técnicamente, Lituania tiene menos permeabilidad a la gama de bronceadores que Grecia. Pero este hecho, que nos parece tan lógico, sólo podrá ser tan descubierto por nuestras herramientas de Minería de Datos si somos capaces de incorporar información relativa a las horas de sol o, al menos, cierta información climática de cada país.

Evidentemente, cada organización deberá recoger diferente información que le pueda ser útil para la tarea de análisis, extracción de conocimiento y, en definitiva, de toma de decisiones.

En la ilustración 17, se muestran las fuentes de datos que pueden ser requeridas en el caso anterior, un proceso de extracción de conocimiento satisfactorio. Sólo conociendo el contexto de cada organización o de cada problema en particular se puede determinar que fuentes

externas van a ser necesarias. Además, este proceso es generalmente iterativo. A medida que se va profundizando en un estudio, se pueden ir determinando datos externos que podrían ayudar y se pueden ir añadiendo a nuestro "repositorio de datos". Por tanto, la tarea de mantener un "repositorio" o un "almacén" con toda la información necesaria cobra mayor relevancia y complejidad.

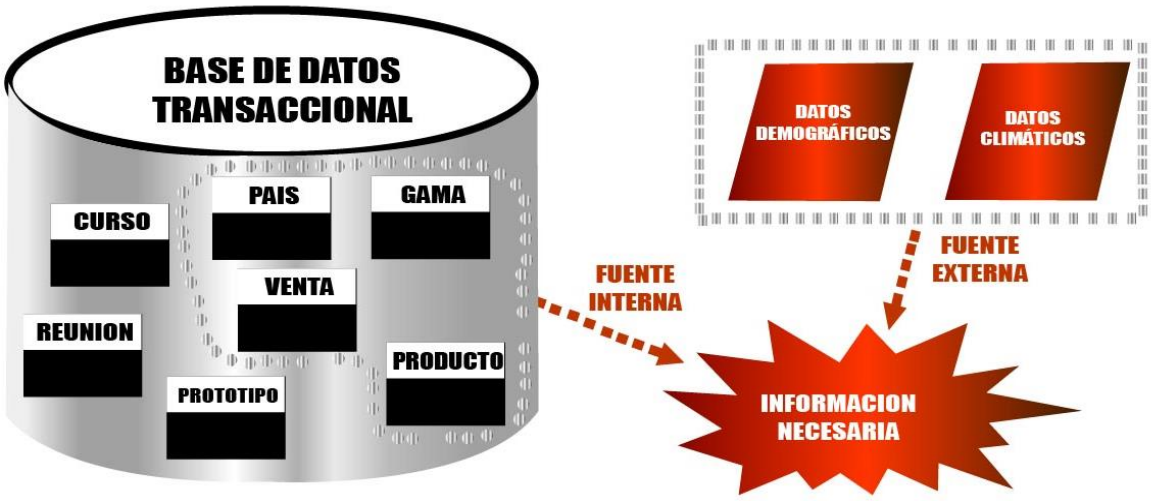


Ilustración 17: Fuentes de datos requeridas para responder "países con mayor penetración de bronceadores".

El mantenimiento de esta información planteas cuestiones técnicas. En Primer lugar, se requerirá añadir, puede que frecuentemente, nueva información a nuestro repositorio, tanto proveniente de actualizaciones de la propia organización como fuentes externas, ya sean actualizaciones como nuevas incorporaciones. En segundo lugar, y la que resulta la cuestión principal, ¿hay que almacenar toda esta información en la base de datos transaccional? Puestos en el ejemplo anterior, ¿requieren las aplicaciones diarias de la organización almacenar en una tabla de la base de datos la temperatura media de Lituania?

Estas y otras cuestiones, como veremos a continuación, han motivado el desarrollo de una tecnología nueva y específica, denominada "almacenes de datos" (Data Warehouses).

14.2 Necesidad de los Almacenes de Datos

La proliferación de sistemas de información sostenidos en bases de datos ha generalizado el uso de herramientas que permiten obtener informes complejos, resúmenes e incluso estadísticas globales sobre la información almacenada con el objetivo de asistir en la toma de decisiones. La mayoría de sistemas comerciales de gestión de bases de datos incluyen herramientas de "informes avanzados", "inteligencia de negocio" (business intelligence), sistemas de información ejecutivos (EIS, Executive Information Systems) y otras, que pese a sus nombre variados intentan realizar un procesamiento analítico de la información, más que

el procesamiento transaccional habitual realizado por las aplicación del día a día de la organización.

Por tanto, cada día es más necesario distinguir dos usos diferentes del sistema de información: el procesamiento transaccional y el procesamiento analítico.

14.2.1 OLTP y OLAP

Con las siglas OLTP y OLAP se denominan dos tipos de procesamiento bien diferentes:

- OLTP (On Line Transactional Processing). El procesamiento transaccional en tiempo real constituye el trabajo primario en un sistema de información. Este trabajo consiste en realizar transacciones, es decir, actualizaciones y consultas a la base de datos con un objetivo operacional: hacer funcionar las aplicaciones de la organización, proporcionar información sobre el estado del sistema de información y permitir actualizarlo conforme va variando la realidad del contexto de la organización. Muestras de este tipo de trabajo transaccional son, por ejemplo, en el caso de una empresa , la inserción de un nuevo cliente , el cambio de sueldo de un empleado , la tramitación de un pedido , el almacenamiento de un venta, la impresión de una factura , la baja un producto, etc. Es el trabajo diario y para el que inicialmente se ha diseñado la base de datos.

- OLAP (On Line Analytical Processing). El procesamiento analítico en tiempo real engloba un conjunto de operaciones, exclusivamente de consulta, en las que se requiere agregar y cruzar gran cantidad de información. El objetivo de estas consultas es realizar informes y resúmenes, generalmente para el apoyo en la toma de decisiones. Ejemplos de este tipo de trabajo analítico pueden ser resúmenes de venta mensuales, los consumos eléctricos por días, la espera media de los pacientes en cirugía digestiva de un hospital, el producto cuyas ventas han crecido más en el último trimestre, las llamadas por horas, etc. Este tipo de consultas suelen emanarse de los departamentos de dirección, logística o prospectiva y requieren de muchos recursos.

  Una característica de ambos procesamientos es que se pretende que sean “on line”, es decir, que sean relativamente “instantáneos” y se puedan realizar en cualquier momento (en tiempo real). Esto parece evidente e impredecible para la OLTP, pero no esta tan claro que esto sea posible para algunas consultas muy complejas realizadas por el OLAP.

  La práctica general, hasta hace pocos años, y todavía existente en muchas organizaciones y empresas, es que ambos tipos de procesamientos (OLTP y OLAP) se realizaran sobre la misma base de datos transaccional. De hecho una de las máximas de la tecnología de bases de datos era la eliminación de redundancia, con lo que parecía lo más lógico que ambos procesamientos trabajaran sobre una única base de datos general (aunque pudiera tener diferentes vistas para diferentes aplicaciones, procesamientos o servicios).

Esta práctica plantea dos problemas fundamentales:

- Las consultas OLAP perturban el trabajo transaccional diario de los sistemas de información originales. Al ser consultas complejas y que involucran muchas tablas y agrupaciones, suelen consumir gran parte de los recursos del sistema de gestión de base de datos. El resultado es que durante la ejecución de estas consultas, las operaciones transaccionales normales (OLTP), se resienten: las aplicaciones van más lentas, las actualizaciones se demoran muchísimo y el sistema puede incluso llegar a colapsarse. De este hecho viene el nombre familiar que se les da a las consultas OLAP: "Killer queries" (consultas asesinas). Como consecuencia, muchas de estas consultas se deben realizar por la noche o en fines de semana, con lo que en realidad dejan de ser "on line".

- La base de datos está diseñada para el trabajo transaccional, no para el análisis de los datos. esto significa que, aunque tuviéramos el sistema dedicado exclusivamente para realizar una consulta OLAP, dicha consulta puede requerir mucho tiempo, pero no solo por ser compleja intrínsecamente, sino porque el esquema de las base de datos no es el más adecuado para este tipo de consultas.

Ambos problemas implican que van a ser prácticamente imposible (a un costo de hardware razonable, lógicamente) realizar un análisis complejo de la información en tiempo real si ambos procesamientos se realizan sobre la misma base de datos.

Afortunadamente, debido a que los costos de almacenamientos masivo y conectividad se han reducido drásticamente en los últimos años, parece razonable recoger (copiar) los datos en un sistema unificado y diferenciado del sistema tradicional transaccional u operacional. Aunque esto vaya contra la filosofía general de base de datos, son muchas más las ventajas que los inconvenientes, como veremos a continuación. Desde esta perspectiva, se separa definitivamente la base de datos con fines transaccionales de la base de datos con fines analíticos. Nacen los almacenes de datos.

14.2.2 Almacenes de datos y bases de datos transaccionales

Un almacén de datos es un conjunto de datos históricos , internos y externos, 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.

La ventaja fundamental de un almacén de datos es su diseño específico y su separación de la base de datos transaccional. Un almacén de datos:

- Facilita el análisis de los datos en tiempo real (OLAP).
- No disturba el OLTP de las base de datos originales.

A partir de ahora, por tanto, diferenciamos claramente entre bases de datos transaccionales (u operacionales) y almacenes de datos. Dicha diferencia, además, se ha ido marcando más

profundamente a medida que las tecnologías propias de ambas bases de datos (y en especial la de almacenes de datos) se han ido especializando. De hecho, hoy en día, las diferencias son claras, como se muestran en la tabla.

Las diferencias mostradas en la tabla, como veremos, distinguen claramente la manera de estructurar y diseñar almacenes de datos respecto a la forma tradicional de hacerlos con bases de datos transaccionales.

	BASE DE DATOS TRANSACCIONAL	ALMACÉN DE DATOS
Propósito	Operaciones diarias. Soporte a las aplicaciones.	Recuperación de información, informes, análisis y Minería de Datos.
Tipos de datos	Datos de funcionamiento de la organización.	Datos útiles para el análisis , la sumarización, etc.
Características de los datos	Datos de funcionamiento, cambiantes, internos, incompletos…	Datos históricos, datos internos y externos, datos descriptivos…
Modelos de datos	Datos normalizados.	Datos en estrella, en copo de nieve, parcialmente desnormalizados, multidimensionales…
Número y tipo de usuarios	Cientos/miles: aplicaciones, operarios, administrador de la base de datos.	Decenas: directores, ejecutivos, analistas (granjeros, mineros).
Acceso	SQL. Lectura y escritura.	SQL y herramientas propias (slice&dice, drill, roll, pivot...). Lectura.

Tabla 8: Tabla Diferencias entre la base de datos transaccional y el almacén de datos.

Aunque ambas fuentes de datos (transaccional y almacén de datos) están separadas, es importante destacar que gran parte de los datos que se incorporan en un almacén de datos provienen de la base de datos transaccional. Esto supone desarrollar una tecnología de volcado y mantenimiento de datos desde la base de datos transaccional al almacén de datos. Además, el almacén de datos debe integrar datos externos, con lo que en realidad debe estar actualizándose frecuentemente de diferentes fuentes. El almacén de datos pasa a ser un integrado o recopilador de información de diferentes fuentes, como se observa en la *Ilustración 18.*

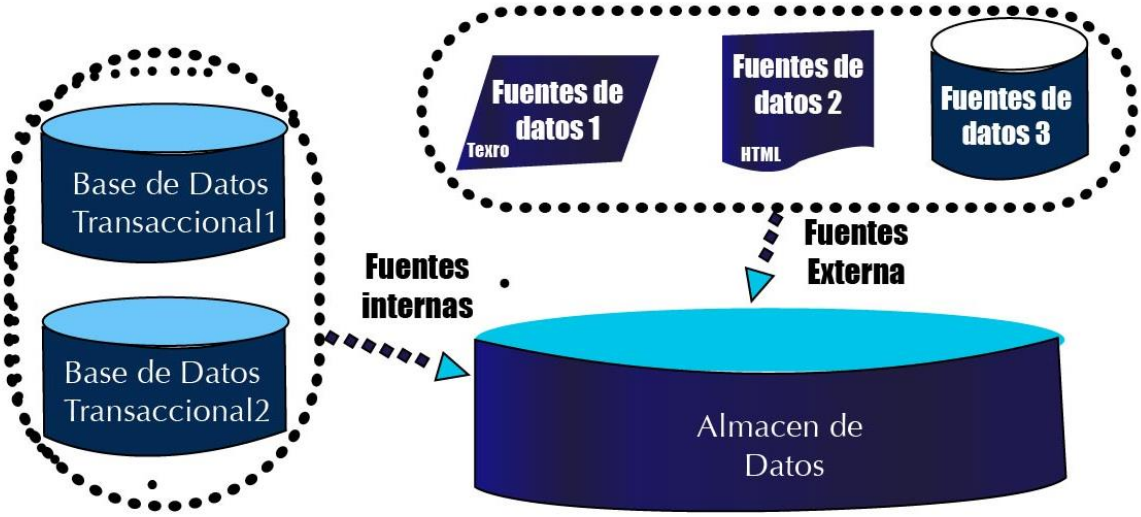


Ilustración 18: El almacén de datos como integrador de diferentes fuentes de datos.

La organización y el mantenimiento de esta información plantean cuestiones técnicas, fundamentalmente sobre como diseñar el almacén de datos, como cargarlo inicialmente, como mantenerlo y preservar su consistencia. No obstante, son muchas más las ventajas de esta separación que sus inconvenientes. Además, esta separación facilita la incorporación de fuentes externas, que, en otro caso, sería muy difícil de encajar en la base de datos transaccional.

14.3 Arquitectura de los almacenes de datos

Un almacén de datos recoge, fundamentalmente, datos históricos, es decir, hechos, sobre el contexto en el que se desenvuelve la organización. Los hechos son, por tanto, el aspecto central de los almacenes de datos.

14.3.1 Modelo multidimensional

El modelo conceptual de datos más extendido para los almacenes de datos es el modelo multidimensional. Los datos se organizan en torno a los hechos, que tienen unos atributos o medidas que pueden verse en mayor o menor detalle según ciertas dimensiones. Por ejemplo, una gran cadena de supermercados puede tener como hechos básicos las ventas. Cada venta tiene unas medidas: importe, cantidad, número de clientes, etc., y se puede detallar o agregar en varias dimensiones: tiempo de la venta, producto de la venta, lugar de la venta, etc. Es esclarecedor comprobar que las medida responden generalmente a la preguntas “cuánto”, mientras que las dimensiones responderán al “cuándo”, “qué”, “dónde”, etc.

Lo realmente interesante del modelo es que ha de permitir, de una manera sencilla, obtener información sobre hechos a diferentes niveles de agregación. Por ejemplo, el hecho “El día 20 de Mayo de 2003 la empresa vendió en España 12.327 unidades de productos de la categoría insecticidas” representa una medida (cantidad ,12.327 unidades) de una venta con granularidad día para la dimensión tiempo (20 de mayo 2003), con granularidad país para la

dimensión lugar (España) y con granularidad categoría (insecticidas) para la dimensión de productos. Del mismo modo, el hecho “El primer trimestre de 2004 la empresa vendió en Valencia por un importe de 22.000 euros del producto Androbrio 33cl.” representa una medida (importe, 22.000 euros) de una venta con granularidad trimestre para la dimensión tiempo (primer trimestre de 2004), con granularidad ciudad para la dimensión lugar (Valencia) y con granularidad articulo (Androbrio 33 cl.) para la dimensión de productos.

En la siguiente ilustración se representa parte de un almacén de datos con estructura multidimensional de donde se pueden extraer estos dos hechos.

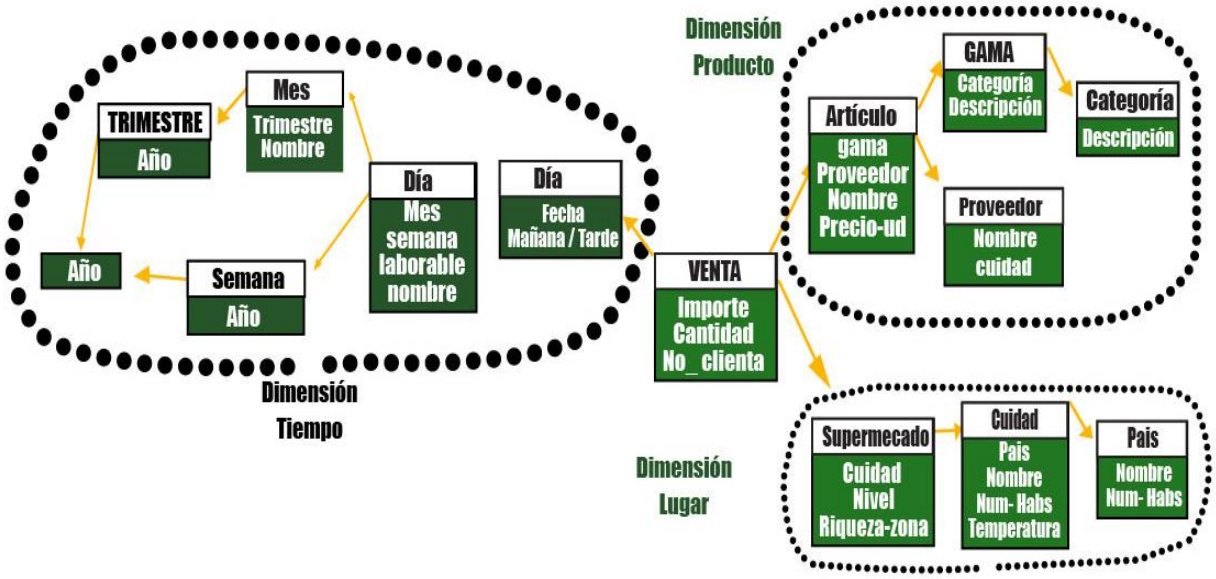


Ilustración 19: Información sobre ventas en un almacén de datos representado bajo un modelo multidimensional.

La ilustración 19 no se basa en ningún modelo de datos en particular (por ejemplo el relacional). Nótese que no estamos hablando de que cada rectángulo de la dicha ilustración sea una tabla o que las flechas sean claves ajenas. Al contrario, simplemente estamos representando datos de una manera conceptual. Mostramos los hechos “venta” y tres dimensiones con varios niveles de agregación. Las flechas se pueden leer como “se agrega en”. Como se observa en la ilustración 19, cada dimensión tiene una estructura jerárquica pero no necesariamente lineal. Por ejemplo, en las dimensiones tiempo y producto hay más de un camino posible de agregación (ruta de agregación). Incluso, en el caso de los productos, el nivel de agregación mayor puede ser diferente (hacia categoría o hacia proveedor). Esto permite diferentes niveles y caminos de agregación para las diferentes dimensiones, posibilitando la definición de hechos agregados con mucha facilidad. La forma que tienen estos conjuntos de hechos y sus dimensiones hace que se llamen popularmente almacenes de datos en “estrella simple” (cuando no hay caminos alternativos en las dimensiones) o de “estrella jerárquica” o “copo de nieve” (cuando si hay caminos alternativos en las dimensiones, como el ejemplo anterior).

Cuando el número de dimensiones no excede de tres (o se agregan completamente el resto) podemos representar cada combinación de niveles de agregación como un cubo. El cubo está formado por casillas, con una casilla para cada valor entre los posibles para cada dimensión a su correspondiente nivel de agregación. Sobre esta "vista", cada casilla representa un hecho. Por ejemplo, en la ilustración 20 se representa un cubo tridimensional donde las dimensiones producto, lugar y tiempo se han agregado por artículo, ciudad y trimestre. La representación de un hecho como el visto anteriormente corresponde, por tanto, a una casilla en dicho cubo. El valor de la casilla es la medida observada (en este caso, tanto importe de las ventas).

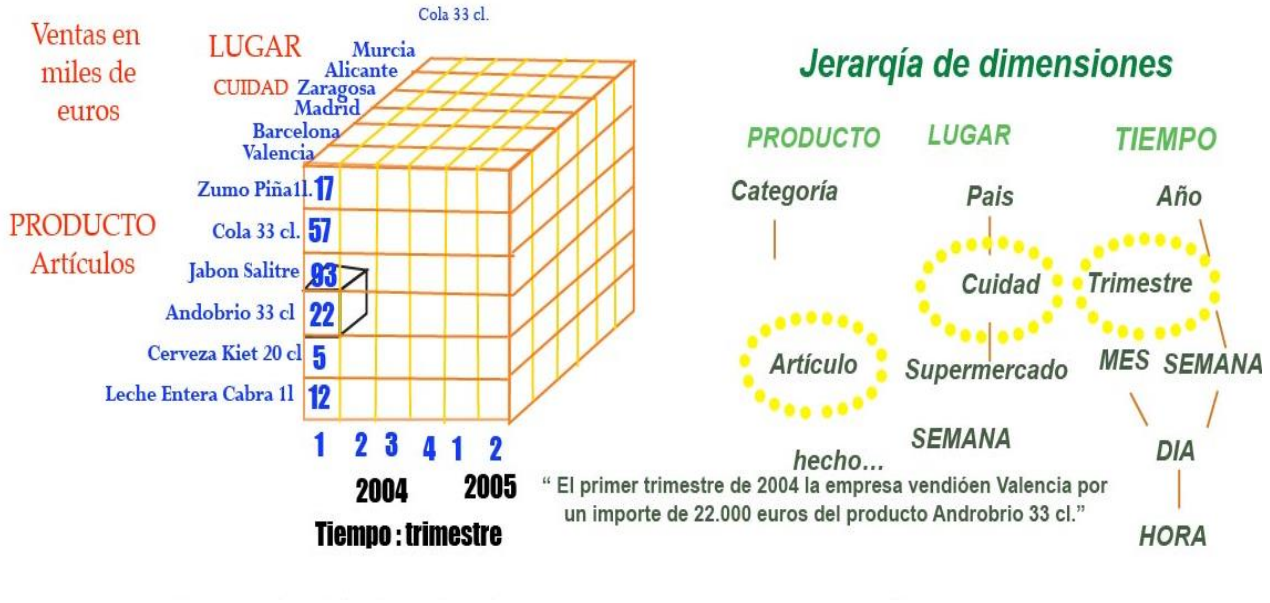


Ilustración 20: Visualización de un hecho en un modelo multidimensional.

Esta visualización hace que, incluso cuando tengamos más de tres dimensiones, se hable de un "cubo"(o más propiamente de "hipercubo") como un conjunto de niveles de agregación para todas las dimensiones.

Esta estructura permite ver de una manera intuitiva la sumarización/agregación (varias casillas se fusionan en casillas más grandes), la disgregación (las casillas se separan en casillas con mayor detalle) y la navegación según las dimensiones de la estrella.

14.3.2 Datamarts

En algunos casos puede parecer intuitivo organizar la información en dimensiones. El Caso de las ventas es el ejemplo más ilustrativo. En general, cierta información es más fácilmente representable de esta forma, pero siempre se puede llegar a una estructura de este tipo. Lo que no es posible, en general, es la representación de todo el almacén de datos como una sola estrella, ni jerárquica. Por ejemplo, la información de personal de una empresa (empleados, departamentos, proyectos, etc.) es difícilmente integrable en la misma estrella

que las ventas. Incluso, en ámbitos más relacionados de una organización (por ejemplo ventas y producción) esto tampoco es posible. La idea general es que para cada sub-ámbito de la organización se va a construir una estructura de estrella. Por tanto, el almacén de datos estará formado por muchas estrellas (jerárquica o no), formando una "constelación". Por ejemplo, aparte de la estrella jerárquica para las ventas, podríamos tener otra estrella para personal. En este caso, los hechos podrían ser que un empleado ha dedicado ciertos recursos en un proyecto durante un periodo en un departamento.

Los hechos podrían llamarse "participaciones". Las medidas o atributos podrían ser "horas de participación", "número de participantes", "presupuesto", "nivel de éxito del proyecto", etc. y las dimensiones podrían ser "tiempo" (para representar el periodo en que ha estado involucrado), "departamento" (para representar un empleado, equipo, departamento o división a la que se ha desarrollado) y el "proyecto" (sub-proyecto, proyecto o programa).

Cada una de estas estrellas que representan un ámbito específico de la organización se denomina popularmente "datamarts" (mercado de datos). Lógicamente, cada datamarts tendrá unas medidas y unas dimensiones propias y diferentes de los demás. La única dimensión que suele aparecer en todos los datamarts es la dimensión tiempo, ya que el almacén de datos representa información histórica y, por tanto, siempre es de interés ser capaz de agregarlo por intervalos de diferente detalle.

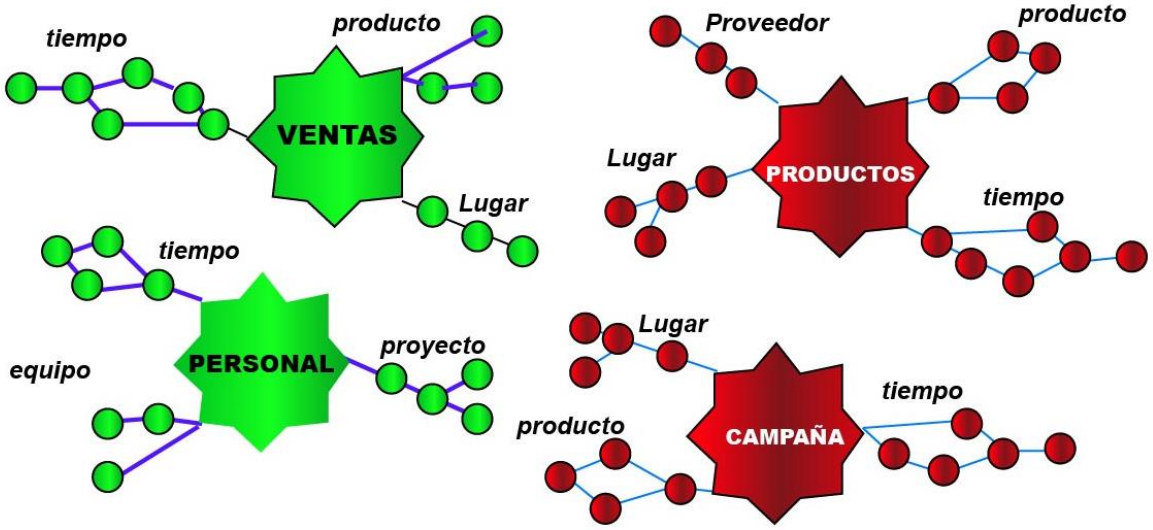


Ilustración 21: Representación icónica de un almacén de datos compuesto por varios datamarts.

- Aparentemente, da la impresión de que el almacén de datos puede contener mucha información redundante, especialmente sobre la dimensión. Aunque, en general, los almacenes de datos contienen información redundante, la estructura anterior es la estructura externa, visible o conceptual. Esta estructura no determina la manera de implementarlo ni lógica ni físicamente, como veremos.

14.3.3 Explotación de un almacén de datos. Operadores

En realidad, un modelo de datos se compone de unas estructuras y unos operadores sobre dicha estructuras. Acabamos de ver que el modelo multidimensional se basa en un conjunto de datamarts, que, generalmente, son estructuras de datos en estrella jerárquica.

Para completar el modelo multidimensional debemos definir una serie de operadores sobre la estructura. Los operadores más importantes asociados a este modelo son:

- **Drill:** se trata de disgregar los datos (mayor nivel de detalle o desglose, menos sumarización) siguiendo los caminos de una o más dimensiones.
- **Roll:** se trata de agregar los datos (menor nivel de detalle o desglose, más sumarización o consolidación) siguiendo los caminos de una o más dimensiones.
- **Slice & Dice**: se seleccionan y se proyectan datos. Este operador permite escoger parte de la información mostrada, no por agregación sino por selección.
- **Pivot:** reorienta las dimensiones. Permite cambiar algunas filas en columnas.

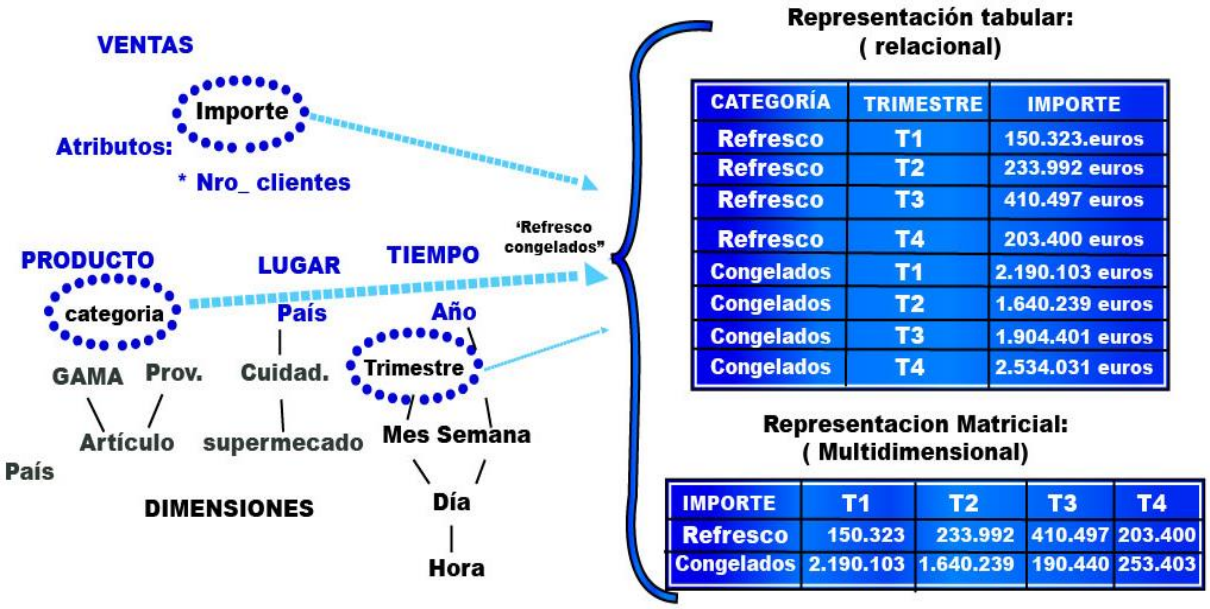
Normalmente, estos operadores se llaman operadores OLAP, operadores de análisis de datos u operadores de almacenes de datos. Para explicar estos operadores hemos de pensar que partimos, además, de unos operadores genéricos básicos, que permiten realizar consultas, vistas o informes sobre la estructura estrella, generalmente de una forma gráfica.
Estos operadores básicos permiten realizar las mismas consultas de proyección, selección y agrupamiento que se pueden hacer en SQL. En muchos casos, de hecho, se puede editar la consulta SQL correspondiente, aunque esta se haya hecho gráficamente.

Por tanto, el primer paso para poder utilizar los operadores propios del modelo multidimensional es definir una consulta. En realidad, como veremos a continuación, los operadores drill, roll, slice & dice y pivot, son modificadores o refinadores de consulta y solo pueden aplicarse sobre una consulta realizada previamente.

Consideremos por ejemplo la consulta “obtener para cada categoría y trimestre el total de ventas” para el datamart de la ilustración 22. . En un entorno gráfico, dicha consulta se podría realizar eligiendo el nivel “categoría” para la dimensión “producto” (obteniendo además sólo dos categorías: “refrescos” y “congelados”), el nivel “trimestre” para la dimensión “tiempo” y no escogiendo la dimensión “lugar” (o considerando que se considera el nivel más agregado, es decir, todo el datamart). Además, se elegiría la propiedad que se desea (“importe”).
Dependiendo del sistema o de la manera que hayamos elegido, el resultado se nos puede mostrar de manera tabular o de manera matricial, como se observa en el siguiente gráfico (Construcción de una consulta seleccionando niveles de dimensión), aunque la información mostrada es la misma. En este caso, como hay pocas dimensiones, la representación matricial parece más adecuada.

La existencia de dimensiones y atributos facilita, en gran medida, la realización de consultas y estas se suelen hacer arrastrando con el ratón las medidas y dimensiones deseada. No es necesario mucho más para realizar informes sencillos sobre ese datamart.

No obstante, lo interesante empieza justamente cuando intentamos modificar el informe (una consulta, al fin y al cabo). A veces, queremos mayor nivel de detalle, otras veces menos, o bien desearemos añadir o quitar alguna dimensión, o modificar el informe en cualquier otro sentido.



CATEGORÍA	TRIMESTRE	IMPORTE
Refresco	T1	150.323.euros
Refresco	T2	233.992 euros
Refresco	T3	410.497 euros
Refresco	T4	203.400 euros
Congelados	T1	2.190.103 euros
Congelados	T2	1.640.239 euros
Congelados	T3	1.904.401 euros
Congelados	T4	2.534.031 euros

IMPORTE	T1	T2	T3	T4
Refresco	150.323	233.992	410.497	203.400
Congelados	2.190.103	1.640.239	190.440	253.403

Ilustración 22: Construcción de una consulta creando niveles de dimensión.

Por ejemplo, supongamos ahora que queremos ver las ventas de refrescos y desglosado por ciudades (en particular por dos: Valencia y León) con el objetivo de ver si los hábitos estacionales (hay más consumo de refrescos en estaciones calurosas) son generalmente en todas las áreas geográficas. Una forma obvia de hacer esto sería realizar un nuevo informe. Lo interesante de los nuevos operadores drill, roll, slice&dice y pivot, es que permiten modificar la consulta realizada, sin necesidad de realizar otra. En realidad son “navegadores” de informes, más que operadores por sí mismos. Por ejemplo, en el caso anterior, podemos utilizar el operador drill. Este operador permite entrar más al detalle en el informe. En particular, solo es necesario que desglose la información por ciudades (en concreto, restringiéndose a sólo Valencia y León) y además seleccionando sólo la categoría “refrescos”. La transformación producida tras esta operación se refleja en la ilustración 23, en la que se muestra cómo cambia la vista tanto en la representación relacional como en la multidimensional (la información mostrada en ambas representaciones siempre es la misma).

En el resultado se puede observar que la distribución de ventas en Valencia (claramente estacional) difiere claramente de la de León (prácticamente no estacional).

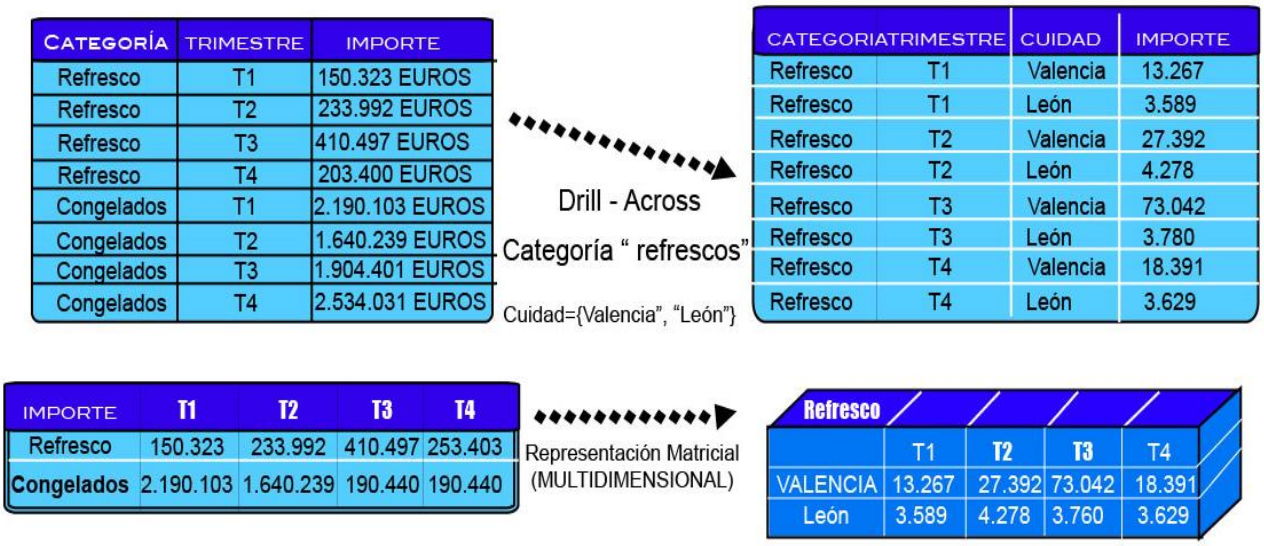


Ilustración 23: Ejemplo del operador "drill".

Lo importante de estos operadores es que modifican el informe en tiempo real y no generan uno nuevo. Lógicamente, para que esto sea eficiente el almacén de datos ha de estar diseñado e implementado para este tipo de operaciones utilicen ciertas estructuras intermedias que permitan agregar y disgregar con facilidad.

Veamos ahora un ejemplo de la operación roll. Simplemente la operación roll es la inversa del drill y el objetivo es obtener información más agregada.

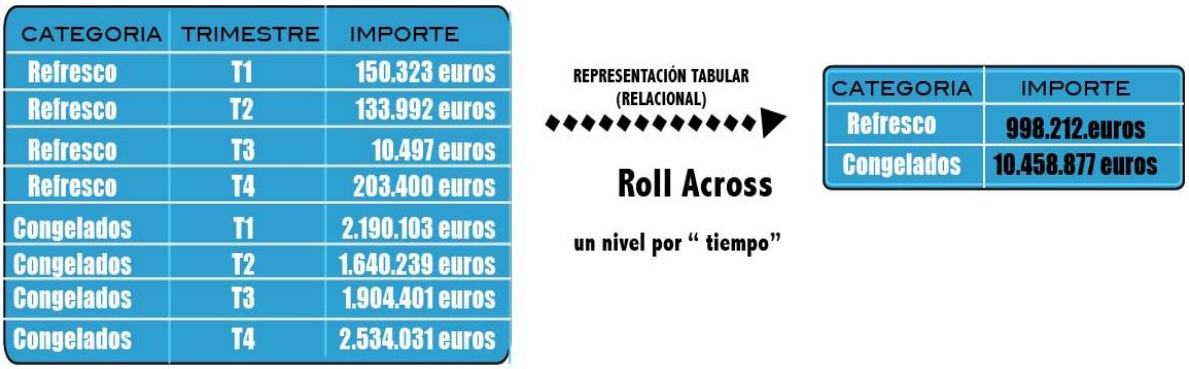


Ilustración 24: Ejemplo del operador roll.

Por ejemplo, si quisiéramos obtener los totales de las categorías "refrescos" y "congelados", simplemente sería necesario aplicar el operador roll-across a la consulta original, sin necesidad de crear una nueva, como se observa en la ilustración 24.

Vistos los operadores drill y roll, cabe preguntarse por qué a veces se utiliza la notación "-across" (como hemos hecho nosotros) y a veces la notación "-up" (que incluso es más frecuente). Aunque en realidad es una cuestión meramente terminológica y no universalmente respetada, las correspondencias son las siguientes:

- Drill- down y roll-up: representan agregaciones o disgregaciones en otras dimensiones ya definida inicialmente en la consulta.
- Drill-across y roll-across: representan agregaciones o disgregaciones en otras dimensiones de las definidas inicialmente en la consulta o hacen desaparecer alguna de las dimensiones.

Finalmente, veamos los otros dos operadores: pivot y slice& dice. Estos dos operadores se utilizan exclusivamente cuando se hace una representación matricial o al menos una representación mixta.

Veamos en primer lugar el operador pivot. Supongamos que tenemos la consulta en la situación en la que estamos mostrando el importe para las categorías "refrescos" y "congelados", las ciudades "Valencia" y "León", y todos los trimestres. La posible representación (mixta, entre tabular y multidimensional) es la que se muestra en la parte izquierda de la Ilustración

CATEGORIA	TRIMESTRE	VALENCIA	LEÓN
REFRESCOS	T1	13.267	3.589
REFRESCOS	T2	27.392	4.278
REFRESCOS	T3	73.042	3.780
REFRESCOS	T4	18.391	3.629
CONGELADOS	T1	150.242	4.793
CONGELADOS	T2	173.105	3.564
CONGELADOS	T3	163.240	4.309
CONGELADOS	T4	190.5739	4.812

REPRESENTACION MIXTA:

PIVOT

Categorías x Cuidad

CATEGORIA	TRIMESTRE	REFRESCO	CONGELADOS
VALENCIA	T1	13.267	150.242
VALENCIA	T2	27.392	173.105
VALENCIA	T3	73.042	163.240
VALENCIA	T4	18.391	190.573
LEÓN	T1	3.589	4.789
LEÓN	T2	4.278	3.564
LEÓN	T3	3.780	4.309
LEÓN	T4	3.629	4.812

Ilustración 25: Ejemplo del operador pivot

El operador pivot permite cambiar algunas filas por columnas. Esta operación, aparentemente sencilla, no está generalizada en muchos sistemas de bases de datos (en SQL- 92 no existía, por ejemplo). No obstante, su inclusión es prácticamente imprescindible para poder realizar análisis de datos y muy en particular, Minería de Datos. Como veremos, este cambio permite que valores de columnas pasen a ser nombre de nuevas columnas y viceversa.

Como se ve en parte derecha de la ilustración 26. Ejemplo del operador slice& dice, esto supone que ciertos métodos de aprendizaje proposicionales sean capaces de extraer patrones sobre la consulta de la izquierda y no sean capaces de hacerlo sobre la segunda y viceversa.

Ilustración 26: Ejemplo de slice& dice

Veamos finalmente el operador slice& dice. En realidad este operador permite escoger parte de la información mostrada, no por agregación sino por selección. En la ilustración 26 se muestra un ejemplo de este operador.

Los operadores vistos son los básicos para refinar una consulta o informe, aunque distintos sistemas propietarios pueden añadir más operadores, maneras diferentes de representar los datos, de interpretar la petición de aplicación de operadores (mediante arrastre de dimensiones utilizando el ratón), etc. En los dos temas siguientes veremos como estos operadores pueden facilitar en gran medida la transformación y adecuación de datos de cara a obtener una "vista minable" que sea idónea para aplicar técnicas de Minería de Datos.

14.3.4 Implementación del almacén de datos. Diseño

Recordemos que una de las razones para crear un almacén de datos separado de la base de datos operacional era conseguir que el análisis se pudiera realizar de una manera eficiente.

El hecho de que la estructura anterior y los operadores vistos permitan trabajar sencillamente y combinar dimensiones, en detallar o agregar informes, etc., y todo ello de manera gráfica, no asegura que esto sea eficiente.

Con el objetivo de obtener la eficiencia deseada, los sistemas de almacenes de bases de datos pueden implementarse utilizando dos tipos de esquemas físicos:

- ROLAP (Relational OLAP): físicamente, el almacén de datos se construye sobre una base de datos relacional.
- MOLAP (Multidimensional OLAP): físicamente, el almacén de datos se construye sobre estructuras basadas en matrices multidimensionales.

Las ventajas del ROLAP son, en primer lugar, que se pueden utilizar directamente sistemas de gestión de bases de datos genéricos y herramientas asociadas: SQL, restricciones, disparadores, etc. En segundo lugar, la información y el costo necesario para su implementación es generalmente menor. Las ventajas del MOLAP son su especialización, la correspondencia entre el nivel lógico y el nivel físico. Esto hace que el MOLAP sea generalmente más eficiente, incluso aunque en el caso de ROLAP se utilicen ciertas técnicas de optimización, como comentaremos más abajo.

No todos los sistemas, libros y manuales son consistentes respecto a si la diferencia ROLAP/MOLAP se produce a nivel físico o a nivel lógico. En algunos textos se habla de que si el sistema representa los resultados de los informes/consultas como tablas, el sistema es ROLAP y si los representa como matrices el sistema es MOLAP. Según nuestra definición (y la de muchos otros autores) tanto ROLAP como MOLAP se refieren a la implementación y son independientes de la manera en la que, externamente, se vean las herramientas del sistema de almacenes de datos o el sistema OLAP. Por tanto un sistema puede tener una representación de las consultas relacional y estar basado en un MOLAP o puede tener una representación completamente multidimensional y estar basado en un ROLAP. Algunos ejemplos de sistemas ROLAP son Microstrategy, Informix Metacube u Oracle Discoverer. El primero, por ejemplo, tiene una interfaz completamente multidimensional mientras que por debajo existe un sistema relacional. Ejemplos de sistemas MOLAP son Oracle Express o el Hyperion Enterprise.

Como hemos dicho, la ventaja de los ROLAP es que pueden utilizar tecnología y nomenclatura de los sistemas de bases de datos relacionales. Esto tiene el riesgo de que en algunos casos se pueda decidir mantener parte de la base de datos transaccional o inspirarse en su organización (manteniendo claves ajenas, claves primarias, conservando parte de la normalización, etc.). En general, aunque esto pueda ser cómodo inicialmente, no es conveniente a largo plazo. De hecho, una de las maneras más eficientes de implementar un datamart multidimensional mediante bases de datos relacionales se basa en ignorar casi completamente la estructura de los datos en las fuentes de origen y utiliza una estructura nueva denominada starflake [Kimball1996]. Esta estructura combina los esquemas en estrella, start y en estrella jerárquica o copo de nieva, snowflake.

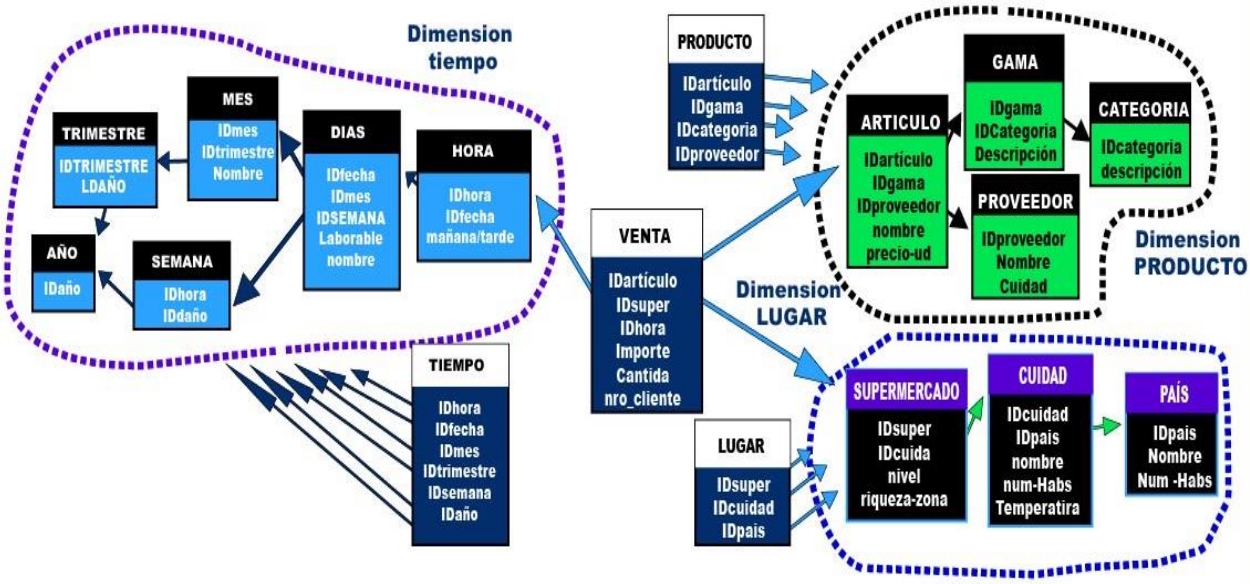


Ilustración 27: Implementación de un datamart utilizando la tecnología relacional ROLAP.

Para construir esta estructura se construyen en tres tipos de tablas:

- tablas de copo de nieve (snowflake tables): para cada nivel de agregación de una dimensión se crea una tabla. Cada una de estas tablas tiene una clave primaria (señalada en la ilustración 27 en negrita) y tantas claves ajenas como sean necesarias para conectar con los niveles de agregación superiores. En la ilustración 27. , las tablas "mes", "día", "articulo", "ciudad" y "país" (entre otras) son tablas copo de nieve.

- Tablas de hechos (fact tables): se crea una única tabla de hechos por datamart. En esta tabla se incluye un atributo para cada dimensión, que será clave ajena (foreign key) a cada una de las tablas copo de nieve de mayor detalle de cada dimensión. Además, todos estos atributos forman la clave primaria. Adicionalmente, pueden existir atributos que representen información de cada hecho, denominados generalmente medidas. En la ilustración 27, la tabla "VENTA" es la tabla de hechos.

- Tabla estrella (star tables): para cada dimensión se crea una tabla que tiene un atributo para cada nivel de agregación diferente en la dimensión. Cada uno de estos atributos es una clave ajena que hace referencia a tablas copo de nieve. Todos los atributos de la tabla forman la clave primaria (señalados en negrita).

En la Ilustración 27, las tablas "TIEMPO", "PRODUCTO" y "LUGAR" son tablas estrella. Las tablas estrella son, en realidad, tablas de apoyo, ya que no representan ninguna información que no esté en las demás.

Este diseño proporciona la realización de consultas OLAP de una manera eficiente, así como la aplicación de los operadores específicos:

- Las tablas copo de nieve permiten realizar vistas o informes utilizando diferentes grados de detalle sobre varias dimensiones. Al estar normalizadas permiten seleccionar datos dimensionales de manera no redundante. Esto es especialmente útil para los operadores drill, slice& dice y pivot.
- Las tablas estrella son, como hemos dicho, tablas de apoyo, que representan "pre concatenaciones" o "pre-junciones" (pre-joins) entre las tablas copo de nieve. El propósito de las tablas estrella es evitar concatenaciones costosas cuando se realizan operaciones de roll- up.

Además de la estructura anterior, los sistemas ROLAP se pueden acompañar de estructuras especiales: índices de mapa de bits, índices de JOIN, optimizadores de consultas, extensiones de SQL (POR EJEMPLO "CUBE"), etc., así como técnicas tan variadas como el pre calculo y almacenamiento de valores agregados que vayan a utilizarse frecuentemente (totales por año, por producto, etc.). Además, se pueden desactivar los locks de lectura/escritura concurrente (ya que sólo hay lecturas), muchos índices dinámicos se pueden sustituir por estáticos o por hashing (ya que las tablas no van a crecer frecuentemente), etc.

Todas estas extensiones y ajustes hacen que el sistema de gestión de bases de datos subyacente se adapte mejor a su nuevo cometido que ya no es una base de datos operacional sino un almacén de datos y proporcione la eficiencia necesaria.
Por el contrario los sistemas MOLAP almacenan físicamente los datos en estructuras multidimensionales de forma que la representación externa e interna coincidan. Las estructuras de datos utilizadas para ello son bastante específicas, lo que permite rendimientos mayores que los ROLAP. En cambio, los sistemas MOLAP tienen algunos inconvenientes:

- Se necesitan sistemas específicos. Esto supone un costo de software mayor y generalmente compromete la portabilidad, al no existir estándares sobre MOLAP tan extendidos como los estándares del modelo relacional.
- Al existir un gran acoplamiento entre la visión externa y la implementación, los cambios en el diseño del almacén de datos obligan a una restructuración profunda del esquema físico y viceversa.
- Existe más desnormalización que en las ROLAP. En muchos casos un almacén de datos MOLAP ocupa más espacio que su correspondiente ROLAP.

Tanto para los sistemas ROLAP y MOLAP existen numerosos aspectos que influyen en el diseño físico. Además, existen metodologías y modelos conceptuales para asistir en el diseño conceptual y lógico y, de ahí, al diseño físico. Existen extensiones del modelo entidad-relación (ER) [Sapia et al.199; Tryfona et al. 1999] o de modelos orientados a objetos [Trujillo et al. 2001], así como modelos específicos [Golfarelli et al. 1998]. Respecto a la representación, en especial si se utiliza una metodología orientada a objetos o se es familiar

con el UML (Unified Modeling Language), existe un estándar Common Warehouse Metadara (CWM) del OMG (Object Management Group, http://www.omg.org). Se trata de una extensión del UML para modelar almacenes de datos.

Quizá la parte de diseño de almacenes de datos es una de las áreas más abiertas y donde existe menos convergencia. Las razones son múltiples pero, fundamentalmente, se resumen en que los almacenes de datos se han originado principalmente desde el ámbito industrial y no académico, que al fin inicial del almacén de datos era realizar OLAP eficiente, con lo que el énfasis recaía fundamentalmente en los niveles lógico y físico. A pesar de todo esto, podemos identificar cuatro pasos principales a la hora de diseñar un almacén de datos (en realidad estos pasos se han de seguir para cada datamart):

1. Elegir para modelar un "proceso" o "dominio" de la organización sobre el que se deseen realizar informes complejos frecuentemente, análisis o Minería de Datos. Por ejemplo, se puede hacer un datamart sobre pedidos, ventas, facturación, etc.
2. Decidir el hecho central y el "granulo" (nivel de detalle) máximo que se va a necesitar sobre él. Por ejemplo, ¿se necesita información horaria para el tiempo?, ¿se necesita saber las cajas del supermercado o es suficiente el supermercado como unidad mínima?, etc. En general, siempre hay que considerar gránulos finos por si más adelante se fueran a necesitar, a no ser que haya restricciones de tamaño importantes. Precisamente, el almacén de datos se realiza, entre otras cosas, para poder agregar eficientemente, por lo que un almacén de datos demasiado detallado no compromete, en principio, la eficacia.
3. Identificar las dimensiones que caracterizan el "dominio" y su grafo o jerarquía de agregación, así como los atributos básicos de cada nivel. No se deben incluir atributos descriptivos más que lo imprescindibles para ayudar en la visualización. En cambio, atributos informativos del estilo "es festivo", "es fin de semana", "es estival", etc., son especialmente interesantes de cara a agregaciones y selecciones que detecten patrones. Las dimensiones varían mucho de un dominio a otro, aunque respondan a preguntas como "qué", "quién", "dónde", "de dónde", "cuándo", "cómo", etc. El tiempo siempre es una (o más de una) de las dimensiones presentes.
4. Determinar y refinar las medidas y atributos necesarios para los hechos y las dimensiones. Generalmente las medidas de los hechos son valores numéricos agregables (totales, cuentas, medias…) y suelen responder a la pregunta "cuánto". Revisar si toda la información que se requiere sobre los hechos está representada en el almacén de datos.

Existen muchas otras dimensiones que hay que tener en cuenta durante el diseño. Por ejemplo, no hay que obsesionarse por el espacio (algunas normalizaciones no van a mejorar la eficiencia y el ahorro en espacio no es considerable). Tampoco hay que orientarse demasiado en la estructura de la base de datos transaccional. Por ejemplo, no se debe utilizar la misma codificación de claves primarias que en la base de datos transaccional.

14.4 Carga y mantenimiento del almacén de datos

Finalmente, si se ha decidido diseñar un almacén de datos, y ya este implementado mediante tecnología ROLAP o MOLAP, el siguiente pase es cargar los datos. El proceso tradicional de base de datos más parecido a la carga de un almacén de datos es el proceso de “migración”, aunque a diferencia de él, existe un “mantenimiento” posterior.

En realidad, la carga y mantenimiento de un almacén de datos es uno de los aspectos más delicados y que más esfuerzo requiere(alrededor de la mitad del esfuerzo necesario para implantar un almacén de datos), y, de hecho, suele existir un sistema especializado para realizar estas tareas , denominado sistema ETL (Extraction, Transformation, Load).

Dicho sistema no se compra en el supermercado ni se descarga de Internet, sino que:

- La construcción del ETL es responsabilidad del equipo de desarrollo del almacén de datos y se realiza específicamente para cada almacén de datos.

Afortunadamente, aunque un ETL se puede construir realizando programas específicos, también se puede realizar adaptando herramientas genéricas (por ejemplo triggers), herramientas de migración o utilizando herramientas más específicas que van apareciendo cada vez más frecuente.

El sistema ETL se encarga de realizar muchas tareas:

- Lectura de datos transaccionales: se trata generalmente de obtener los datos mediantes consultas SQL sobre la base de datos transaccional. Generalmente se intenta que esta lectura sea en horarios de poca carga transaccional (fines de semana o noches). Para la primera carga los datos pueden encontrarse en históricos y es posible que en distintos formatos. Este hecho condiciona muchas veces el número de años que se puede incluir en el almacén de datos.
- Incorporación de datos externos: generalmente aquí se deben incorporar otro tipo de herramientas, como wrappers, para convertir texto, hojas de cálculo o HTML en XML o en tablas de base de datos que se puedan integrar en el almacén de datos.
- Creación de claves: en general se recomienda crear claves primarias nuevas para todas las tablas que se vayan creando en el almacenamiento intermedio o en el almacén de datos.
- Integración de datos: consiste en muchos casos en la fusión de datos de distintas fuentes, detectar cuando representan los mismos objetos y generar las referencias y restricciones adecuadas para conectar la información y proporcionar integridad referencial.
- Obtención de agregaciones: si se sabe que cierto nivel de detalle no es necesario en ningún caso, una primera fase de agregación se puede realizar aquí.
- Limpieza y transformación de datos: aunque de estas dos tareas nos dedicaremos en el tema siguiente, parte de la limpieza y la transformación necesaria para organizar el

almacén se realiza por el ETL. Se trata, como veremos, de evitar datos redundantes, inconsistentes, estandarizar medidas, formatos, fechas, tratar valores nulos, etc.

- Integración de datos: consiste en muchos casos en la fusión de datos de distintas fuentes, detectar cuando representan los mismos objetos y generar las referencias y restricciones adecuadas para conectar la información y proporcionar integridad referencial.
- Obtención de agregaciones: si se sabe que cierto nivel de detalle no es necesario en ningún caso, una primera fase de agregación se puede realizar aquí.
- Planificación de la carga y mantenimiento: consiste en definir las fases de carga, el orden, para evitar violar restricciones de integridad, del mismo modo que se realizan las migraciones, y las ventajas de carga, con el objetivo de poder hacer la carga sin saturar ni la base de datos transaccional, así como el mantenimiento sin paralizar el almacén de datos.
- Indización: finalmente se han de crear índices sobre las claves y atributos del almacén de datos que se consideren relevantes (niveles de dimensiones, tablas de hechos, etc.).
- Pruebas de calidad: en realidad se trata de definir métricas de calidad de datos, así como implantar un programa de calidad de datos, con un responsable de calidad que realice un seguimiento, especialmente si el almacén de datos se desea utilizar para el apoyo en decisiones estratégicas o especialmente sensibles.

Generalmente, para realizar todas estas tareas, los sistemas ETL se basan en un repositorio de datos intermedio, como se muestra en la ilustración 28 esto puede parecer que ya es abusar de recursos, al tener además de la base de datos transaccional y el almacén de datos un tercer repositorio de datos de similar magnitud. Sin embargo, este almacenamiento intermedio es extremadamente útil, ya que hay tareas que no se pueden realizar en el sistema transaccional ni en el almacén de datos. Por ejemplo, la limpieza y transformación de datos se pueden realizar tranquilamente en este repositorio intermedio, ciertos metadatos pueden almacenarse ahí y valores agregados intermedios también pueden residir ahí, así como los valores integrados de fuentes externas. Con ello, muchos procesos del ETL, incluidos el mantenimiento, se pueden realizar en gran medida sin paralizar ni la base de datos transaccional ni el almacén de datos.

Esta estructura basada en un “almacenamiento intermedio” se muestra en la Ilustración 28 y sitúa más claramente las siglas del acrónimo ETL.

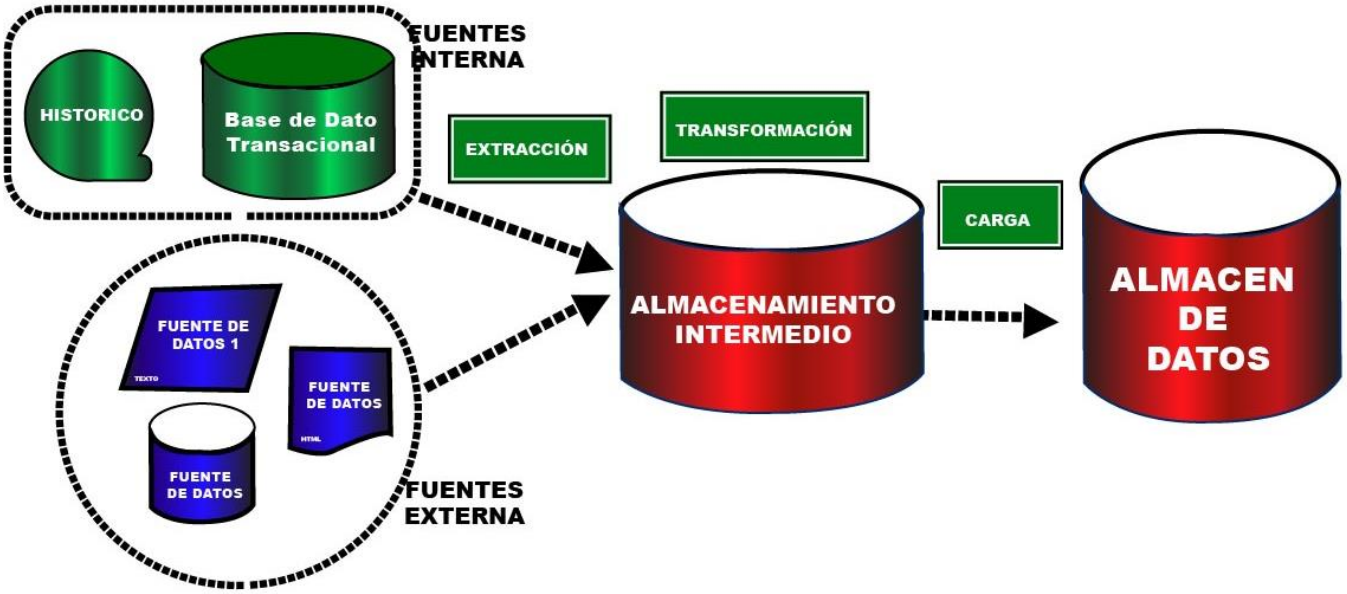


Ilustración 28: El sistema ETL basado en un repositorio intermedio.

La organización con almacenamiento intermedio es especialmente indicada para integrar la información externa. Como dijimos en la introducción, dicha información externa es especialmente importante para encontrar patrones o aspectos significativos en muchos casos, por lo que no nos podemos limitar a la información de la base de datos transaccional. Por ejemplo, la ilustración 29 Muestra que se pueden malinterpretar los datos si no se compara (o se integra) con información externa; una tendencia que parece atisbar una recuperación puede verse como una pérdida de mercado si se compara con la competencia.

Del mismo modo, sin esta información externa, puede ser prácticamente imposible extraer patrones.

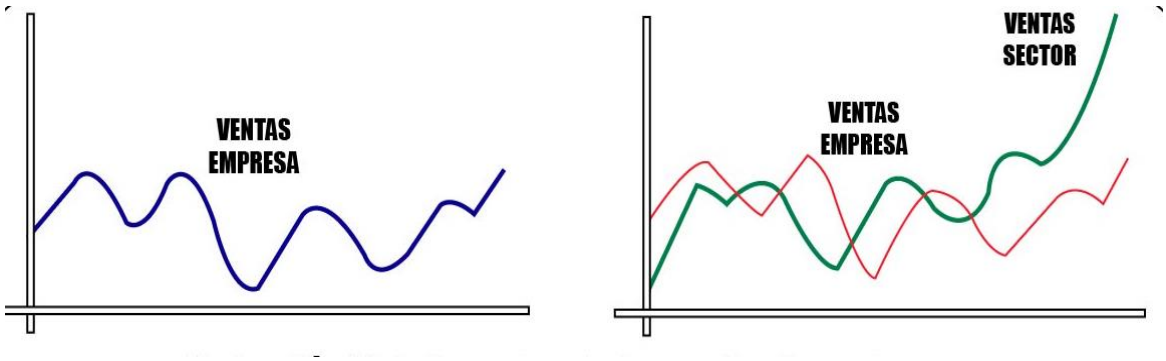


Ilustración 29: La importancia de usar fuentes externas.

En general, existe información que suele ser apropiada para muchos almacenes de datos: demografías (censo), datos resumidos de áreas geográficas, distribución de la competencia, evolución de la economía, información de calendarios y climatológicas, programaciones

televisivas-deportivas, catástrofes, páginas amarillas, psicografías sociales, información de otras organizaciones, datos compartidos en una industria o ara de negocio, organizaciones y colegios profesionales, catálogos, etc.

El valor de estas fuentes externas ha propiciado la existencia de un mercado de este tipo de base de datos. En aquellos casos en los que no se puede obtener de una manera gratuita (por ejemplo, desde algún organismo público), algunas bases de datos se pueden comprar a compañías especializadas. Gran parte de estas bases externas no tienen información personal y por tanto no infringen leyes de protección de datos. En el caso de que esto pudiera ser así, es muy importante estar al tanto de la legalidad al respecto.

14.5 Almacenes de datos y Minería de Datos

El concepto de almacenes de datos nace hace más de una década ligado al concepto de EIS (Executive Information System), el sistema de información ejecutivo de una organización. En realidad, cuando están cubiertas las necesidades operacionales de las organizaciones se plantean herramientas informáticas para asistir o cubrir en las necesidades estratégicas.

La definición original de almacén de datos es una “colección de dato, orientada a un dominio, integrada, no volátil y variante en el tiempo para ayudar en las decisiones de dirección” [Inmon 1992; Inmon 2002]. A raíz de esta definición, parecía que los almacenes de datos son sólo útiles en empresas o instituciones donde altos cargos directivos tengan que tomar decisiones. A partir de ahí, y de la difusión cada vez mayor de las herramientas de bussines intelligence y OLAP, podríamos pensar que los almacenes de datos no se aplican en otros ámbitos: científicos, médicos, ingenieriles, académicos, donde no se tratan con las variables y problemáticas típicas de las organizaciones y empresas.

Al contrario, en realidad, los almacenes de datos pueden utilizarse de muy diferentes maneras, y pueden agilizar muchos procesos diferentes de análisis. En la ilustración 30, se pueden observar las distintas aplicaciones y usos que se puede dar a un almacén de datos: herramientas de consultas e informes, herramientas EIS, herramientas OLAP y herramientas de Minería de Datos.

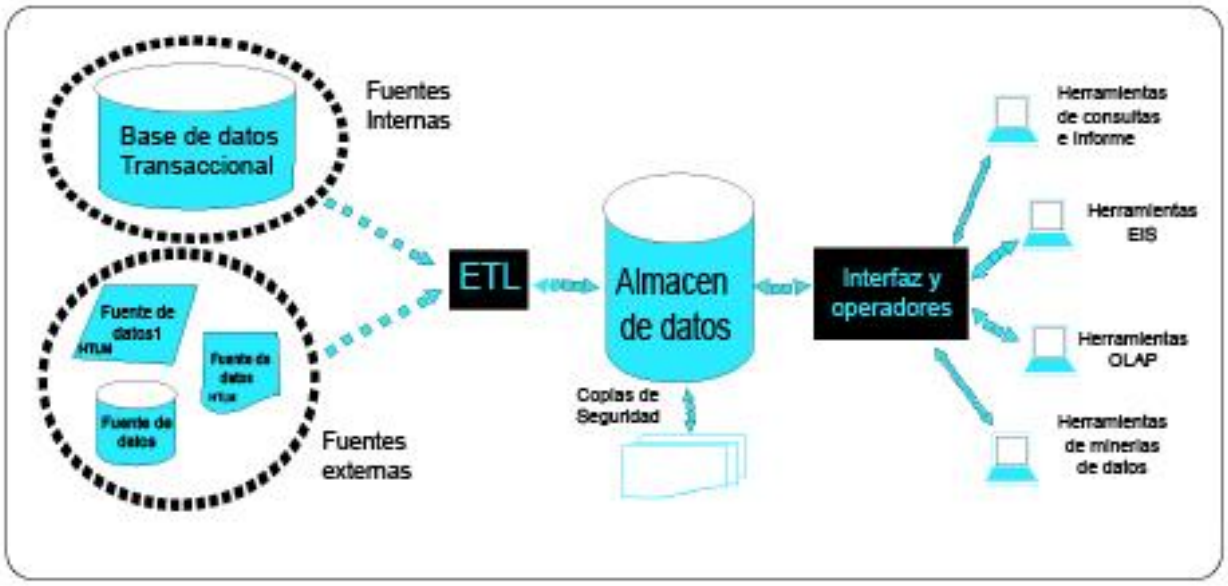


Ilustración 30: Perspectiva general y usos de un almacén de datos.

La variedad de usos que se muestran en la Ilustración anterior sugiere también la existencia de diferentes grupos de usuarios: analistas, ejecutivos, investigadores, etc. Según el carácter de estos usuarios se les puede catalogar en dos grandes grupos:

- "picapedreros" (o "granjeros"): se dedican fundamentalmente a realizar informes periódicos, ver la evolución de indicadores, controlar valores anómalos, etc.
- "exploradores": encargados de encontrar nuevos patrones significativos utilizando técnicas OLAP o de Minería de Datos. La estructura del almacén de datos y sus operadores facilita la obtención de diferentes vistas de análisis o "Vistas Minables".

Esta diferencia, y el hecho de que se catalogue como "exploradores" a aquellos que utilizan técnicas OLAP o Minería de Datos, no nos debe hacer confundir las grandes diferencias de un análisis clásico, básicamente basado en la agregación, la visualización y las técnicas descriptivas estadísticas con un uso genuino de Minería de Datos que no transforma los datos en otros datos (más o menos agregados) sino que transforma los datos en conocimiento (o más humildemente, en reglas o modelos).

Un aspecto a destacar es que el nivel de agregación para los requerimientos de análisis OLAP puede ser mucho más grueso que el necesario para la Minería de Datos. Por ejemplo, para el análisis OLAP puede ser insuficiente usar como unidad mínima de lugar el supermercado. En cambio, para la Minería de Datos puede ser interesante tener un nivel más fino (por caja o por cajera).

Los almacenes de datos no son imprescindibles para hacer extracción de conocimiento a partir de datos. En realidad, se puede hacer Minería de Datos sobre un simple archivo de datos. Sin embargo, las ventajas de organizar un almacén de datos se amortizan sobradamente a medio y largo plazo. Esto es especialmente patente cuando nos

enfrentamos a grandes volúmenes de datos, o estos aumentan con el tiempo, o provienen de fuentes heterogéneas o se van a querer combinar de maneras arbitrarias y no predefinidas. Tampoco es cierto que un almacén de datos sólo tenga sentido si tenemos una base de datos transaccional inicial. Incluso si todos los datos originalmente no provienen de bases de datos puede ser conveniente la realización de un almacén de datos.

En gran medida, un almacén de datos también facilita la limpieza y la transformación de datos (en especial para generar “Vistas Minables” en tiempo real). La limpieza y transformación de datos se tratan precisamente en el tema siguiente.

GUÍA DE APOYO PARA EL ESTUDIANTE
Tema 3: Preparación de los datos

I. CONTESTE

1) Diferencias entre las técnicas OLAP y OLT

2) ¿Cuáles son la ventaja fundamental de un almacén de datos?

3) ¿Qué recoge un almacén de datos?

4) ¿Cuáles son las funciones de los operadores drill, roll, slice& dice y pivot?

5) ¿Qué importancia tienen los operadores drill, roll, slice& dice y pivot?

II. FALSO O VERDADERO

1) ROLAP (Relational OLAP): físicamente, el almacén de datos se construye sobre una base de datos relacional._____

2) MOLAP (Multidimensional OLAP): representan agregaciones o disgregaciones en otras dimensiones ya definida inicialmente en la consulta. _____

3) No todos los sistemas, libros y manuales son consistentes respecto a si la diferencia ROLAP/MOLAP se produce a nivel físico o a nivel lógico _____

4) Un ETL se puede construir realizando programas específicos, también se puede realizar adaptando herramientas genéricas, herramientas de migración o utilizando herramientas más específicas que van apareciendo cada vez más frecuente. _____

III. COMPLETE

1) La _____con almacenamiento intermedio es especialmente indicada para integrar la información externa.

2) El concepto de _____nace hace más de una década ligado al concepto de EIS (Executive Information System), el sistema de información ejecutivo de una organización.

3) ---------- encargados de encontrar nuevos patrones significativos utilizando técnicas OLAP o de Minería de Datos.

4) Un aspecto a destacar es que el nivel de agregación para los requerimientos de análisis _____puede ser mucho más grueso que el necesario _____

SOLUCION DE GUÍA DE APOYO PARA EL ESTUDIANTE

Tema 3: Preparación de los datos

I. CONTESTE

1) Diferencias entre las técnicas OLAP y OLT

a) OLTP (On Line Transactional Processing). El procesamiento transaccional en tiempo real constituye el trabajo primario en un sistema de información. Este trabajo consiste en realizar transacciones, es decir, actualizaciones y consultas a la base de datos con un objetivo operacional:
b) OLAP (On Line Analytical Processing). El procesamiento analítico en tiempo real engloba un conjunto de operaciones, exclusivamente de consulta, en las que se requiere agregar y cruzar gran cantidad de información

2) ¿**Cuáles son la ventaja fundamental de un almacén de datos?**

a) Facilita el análisis de los datos en tiempo real (OLAP).
b) No disturba el OLTP de las base de datos originales.

3) ¿Qué recoge un almacén de datos?

Datos históricos, es decir, hechos, sobre el contexto en el que se desenvuelve la organización. Los hechos son, por tanto, el aspecto central de los almacenes de datos.

4) ¿Cuáles son las funciones de los operadores drill, roll, slice& dice y pivot?

Modificadores o refinadores de consulta y solo pueden aplicarse sobre una consulta realizada previamente.

5) ¿Qué importancia tienen los operadores drill, roll, slice& dice y pivot?

Lo importante de estos operadores es que modifican el informe en tiempo real y no generan uno nuevo

II. FALSO O VERDADERO

1) ROLAP (Relational OLAP): físicamente, el almacén de datos se construye sobre una base de datos relacional. **V**

2) MOLAP (Multidimensional OLAP): representan agregaciones o disgregaciones en otras dimensiones ya definida inicialmente en la consulta. **F**

3) No todos los sistemas, libros y manuales son consistentes respecto a si la diferencia ROLAP/MOLAP se produce a nivel físico o a nivel lógico V

4) Un ETL se puede construir realizando programas específicos, también se puede realizar adaptando herramientas genéricas, herramientas de migración o utilizando herramientas más específicas que van apareciendo cada vez más frecuente. **V**

III. COMPLETE

1) La organización con almacenamiento intermedio es especialmente indicada para integrar la información externa.

2) El concepto de almacenes de datos nace hace más de una década ligado al concepto de EIS (Executive Information System), el sistema de información ejecutivo de una organización.

3) "Exploradores": encargados de encontrar nuevos patrones significativos utilizando técnicas OLAP o de Minería de Datos.

4) Un aspecto a destacar es que el nivel de agregación para los requerimientos de análisis OLAP puede ser mucho más grueso que el necesario para la Minería de Datos.

15) TEMA 4: LIMPIEZA Y TRANSFORMACIÓN

Objetivos:

- Conocer el proceso de limpieza y transformación de datos
- Conocer técnicas de detección y tratamiento de los datos anómalos o erróneos.
- Conocer técnicas de discretización y numerización
- Conocer métodos para la normalización de rango

Contenido:

- Introducción
- Integración y limpieza de datos
- Transformación de atributos, Creación de características
- Discretización y numerización
- Normalización de rango: escalado y Centrado

Duración: 4 hrs.

Bibliografía:

- Introducción a Minería de Datos. José Hernández Orallo, Mª José Ramírez Quintana, Cesar Ferri Ramírez.

15.1 Introducción

El concepto de “calidad de datos” se asocia cada vez más frecuente a los sistemas de información. Aunque se ha avanzado mucho en el diseño y desarrollo de restricciones de integridad de los sistemas de información, éstos han crecido de tal manera en las últimas décadas, que el problema de calidad de datos, en vez de resolverse, en muchos casos empeora.

En la mayoría de Bases de Datos (BD) existe mucha información que es incorrecta respecto al dominio de la realidad que se desea cubrir, datos inconsistentes. Estos problemas se acentúan cuando realizamos integración de distintas fuentes. No obstante, mientras los datos erróneos crecen de manera lineal respecto al tamaño de los datos recopilados, los datos inconsistentes se multiplican; varias fuentes diferentes pueden afirmar distintas cosas distintas sobre el mismo objeto.

La recopilación de datos debe ir acompañada de una limpieza e integración de los mismos, para que éstos estén en condiciones para su análisis. Los beneficios del análisis y la extracción de conocimiento a partir de datos dependen, en gran medida, de la calidad de los datos recopilados. El éxito de un proceso de Minería de Datos (MD) depende, no sólo de tener todos los datos necesarios (una buena recopilación) sino de que estos estén íntegros, completos y consistentes (una buena limpieza e integración).

15.2 Integración y limpieza de datos

Existen problemas de calidad en los sistemas de información. Problemas que pueden verse agravados por el proceso de integración de distintas fuentes si no se hace con esmero. Por ejemplo: existen datos faltantes que suelen ser originados muchas veces al integrar fuentes diferentes, para los cuales no existen soluciones fáciles, pero hay otros casos, como los valores duplicados que sí pueden y deben ser detectados durante la integración.

La integración es generalmente un proceso que se realiza durante la recopilación de datos y, si se realiza un almacén de datos, durante el proceso de carga, mediante el sistema ETL. La limpieza de datos (data cleaning /cleansing) puede, en muchos casos, detectar y solucionar problemas de datos no resueltos durante la integración.

Lógicamente estos procesos, limpieza e integración pueden ser más rudimentarios cuando no se desee crear un almacén de datos. En cualquier caso es un aspecto que no se debe descuidar, particularmente si se desea realizar Minería de Datos de una manera sistemática.

15.2.1. Integración:

El primer paso a la hora de realizar una integración de distintas fuentes de datos es identificar los objetos, es decir, conseguir que datos sobre el mismo objeto se unifiquen y datos de diferentes objetos permanezcan separados. Este problema se conoce como el problema de esclarecimiento de identidad. Existen 2 tipos de errores que pueden ocurrir en esta integración:

- Dos o más objetos diferentes se unifica: los datos resultantes mezclarán patrones de diferentes individuos y serán un problema para extraer conocimiento. esto será más grave cuanto más diferentes sean los dos objetos unificados.
- Dos o más fuentes de objetos iguales se dejan separadas: los patrones del mismo individuo aparecerán repartidos entre varios individuos parciales. Este problema genera menos "ruido" que el anterior, aunque es especialmente problemático cuando se usan valores agregados (el total de compras será mucho menor si consideramos que un individuo real como dos individuos en la base de datos, por ejemplo)

En general el primer problema es más frecuente que el segundo, ya que la unificación se realiza generalmente por identificadores ***externos*** a la base de datos: número de identidad, número de pólizas, matrículas, tarjetas de crédito, etc.

Esta tarea es más difícil de lo que pueda parecer, ya que se utilizan claves internas para identificar objetos, hay que mirar los identificadores externos y éstos, muchas veces, varían de formato (por ejemplo, un ciudadano español puede identificarse por su DNI, el NIF y el pasaporte, que coinciden con 8 dígitos, pero el NIF añade una letra y el pasaporte añade alguna letra y digito adicional).

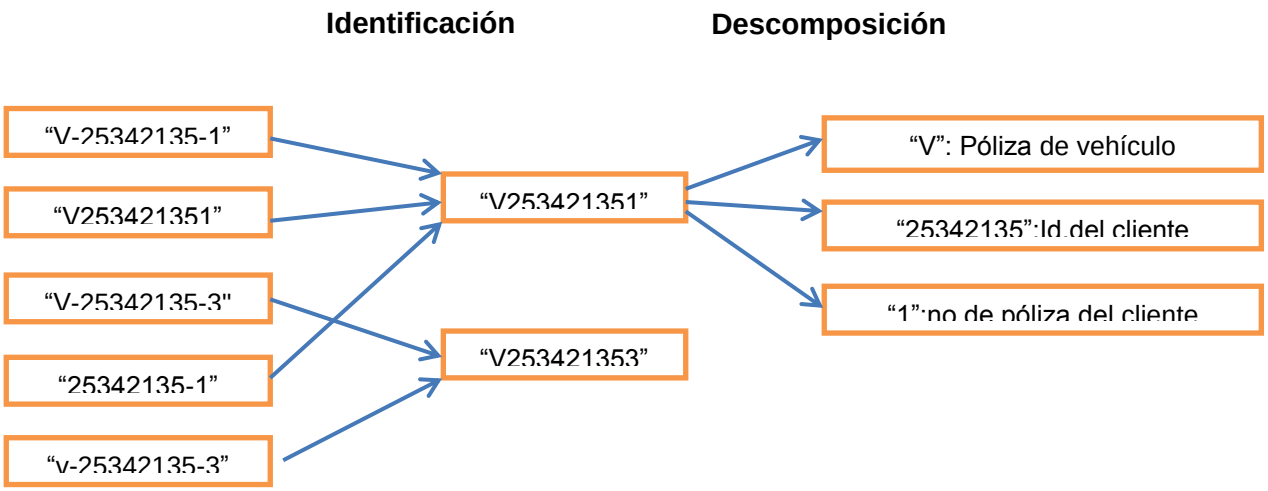


Ilustración 31: Ejemplo de integración: identificación y descomposición

Las claves internas de sistemas diseñados pueden entrañar información no normalizada, que es preciso detectar en el proceso de integración. Este proceso se denomina ***descomposición de claves.*** La Parte derecha de la ilustración 31 muestra un ejemplo.

Cuando se integran (correctamente) dos fuentes de datos de distintos objetos suele suceder que puedan aparecer datos faltantes (el dato se registra de una fuente pero no en la otra) o datos inconsistentes (el dato es diferente en una fuente y otra). Esto se observa en la siguiente Ilustración.

DNI	EDAD	COD.POSTAL	ESTADO	AÑOS_CARNÉ
...	...	...	...	...
25342135	35	45019	Casado	13
98525925	23	25004	Soltero	1
...	...	...	...	...

DNI	FECHA_NAC	CIUDAD	ESTADO	CARNÉ
...	...	...	...	...
77775252	1/1/1950	Barcelona	SI	A2
25342135	18/11/1971	Valencia	NO	B1
...	...	...	...	...

Fuente 1

Fuente 2

DNI	EDAD	FECHA_NAC	CIUDAD	COD.POSTAL	ESTADO	CASADO	AÑOS_CARNÉ	CARNÉ
...	...	...	...	...	...	...	...	...
25342135	35	18/11/1971	Valencia	45019	Casado	NO	13	B1
98525925	23	-	-	25004	Soltero	-	1	-
77775252	-	1/1/1950	Barcelona	-	-	SI	-	A2
...	...	...	...	...	...	...	...	...

Ilustración 32: Ejemplos de integración de atributos de distintas fuentes

Lógicamente aparecen campos redundantes total o parcialmente: “edad” y “fecha_nac”, “ciudad” y “cod_postal”, “estado” y “casado”. Cuando sea posible se intentarán fusionar. En muchos casos los datos inconsistentes se convierten en faltantes; por ejemplo si el mismo clientes tiene estados civiles diferentes en cada fuente, es preferible dejar el valor faltante que elegir al azar uno de los dos (o hacer la media).

Otro caso muy frecuente es la integración de formatos diferentes, que se produce si tenemos codificadores diferentes (casado / matrimonio), idiomas diferentes, medidas diferentes, etc.

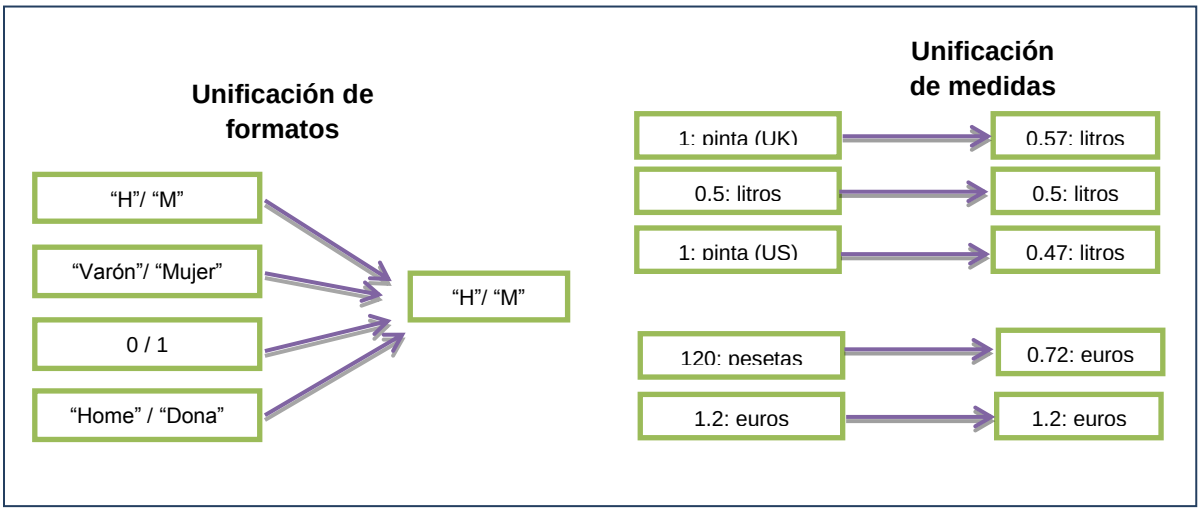


Ilustración 33: Ejemplos de integración: unificación de formatos y medidas

15.2.2 Reconocimiento:

Cuando tenemos integrados los datos lo primero que podemos realizar es un resumen de las características (o informe de estado) de atributos (ya sea tabla a tabla o para toda la base o almacén de datos). En este tipo de tablas se muestran las características generales de los atributos (medias, mínimos, máximos posibles valores). Se puede distinguir entre valores nominales y numéricos (y hacer dos tablas) o integrarlo todo en la misma tabla.

Atributos Nominales: Podemos detectar, Valores redundantes (Hombre, Varón), valores despreciables (agrupar valores como otros).

Atributos Numéricos: Podemos detectar, Valores anómalos (Distribuciones en los datos).

Por ejemplo, para una compañía de seguros, tenemos los datos referidos a las pólizas de vehículos. La siguiente tabla muestra (parcialmente) un resumen de los atributos de la base de datos:

ATRIBUTO	TIPO	# TOTAL	# NULLS	# DIST	MEDIA	DESV.	MODA	MIN	MAX
Código postal	**Nominal**	**10320**	**150**	**1672**	**-**	**-**	**“46003”**	**“01001”**	**“50312”**
Sexo	**Nominal**	**10320**	**23**	**6**	**-**	**-**	**“V”**	**“E”**	**“M”**
Estado Civil	**Nominal**	**10320**	**317**	**8**	**-**	**-**	**Casado**	**“Casado”**	**“Viudo”**
Edad	**Numérico**	**10320**	**4**	**66**	**42,3**	**12,5**	**37**	**18**	**87**
Total póliza p/a	**Numérico**	**17523**	**1325**	**142**	**737,24€**	**327€**	**680€**	**375€**	**6200€**
Asegurados	**Numérico**	**17523**	**0**	**7**	**1,31**	**0,25**	**1**	**0**	**10**
Matricula	**Nominal**	**16324**	**0**	**16324**	**-**	**-**	**-**	**“A-0003-BF”**	**“Z-9835-AF”**
Modelo	**Nominal**	**16324**	**1321**	**2429**	**-**	**-**	**“O.Astra”**	**“Audi A3”**	**“VW Polo”**
…	**…**	**…**	**…**	**…**	**…**	**…**	**…**	**…**	**…**

Tabla 9: Tabla resumen de atributos

La tabla anterior es sencilla de construir (se puede hacer incluso a partir de un conjunto de consultas en SQL) y da mucha información de un simple vistazo. Además de ver cuántos clientes, pólizas y vehículos tenemos, podemos observar el total de nulos de cada atributo, después el número de valores distintos (incluyendo los nulos), el mínimo y el máximo (para los valores nominales, el mínimo y el máximo se interpretan como en SQL, alfabéticamente).

La tabla de resumen de atributos proporciona bastante información sobre los atributos numéricos. Un segundo paso sobre estos valores es también un ***histograma.*** Por ejemplo, observemos la *Ilustración 34* un histograma para los valores del atributo “total póliza por año” (los nulos no se muestran en el histograma):

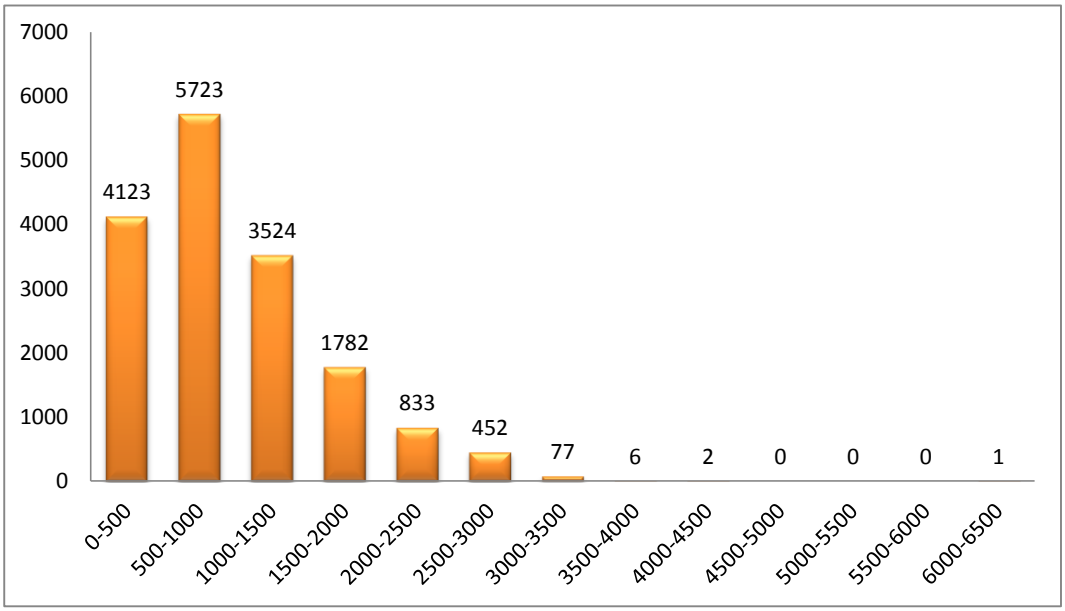


Ilustración 34: Histograma representando la frecuencia de un atributo.

Una alternativa a los histogramas, a la hora de estudiar la frecuencia de una variable son los diagramas de caja (boxplot) o de bigotes (whiskerplots).

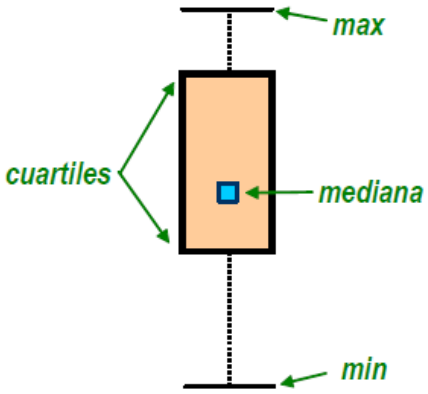


Ilustración 35: Diagramas de caja o de bigotes

La caja va del primer al tercer cuartil (de 25 por ciento de los datos al 75 por ciento de los datos). Es decir, la caja muestra el rango intercuartil, que contiene el 50 por ciento de los datos. La mediana se representa con un cuadrito (también se puede hacer otro tipo de marca). Los Bigotes (o líneas acabadas en un segmento) muestran el resto de los datos hasta los valores más extremos. En cierto modo, los diagramas de caja son un resumen de los histogramas y permiten, en un mismo gráfico, representar varias variables.

Otra alternativa especialmente útil para los atributos numéricos son las gráficas de dispersión.

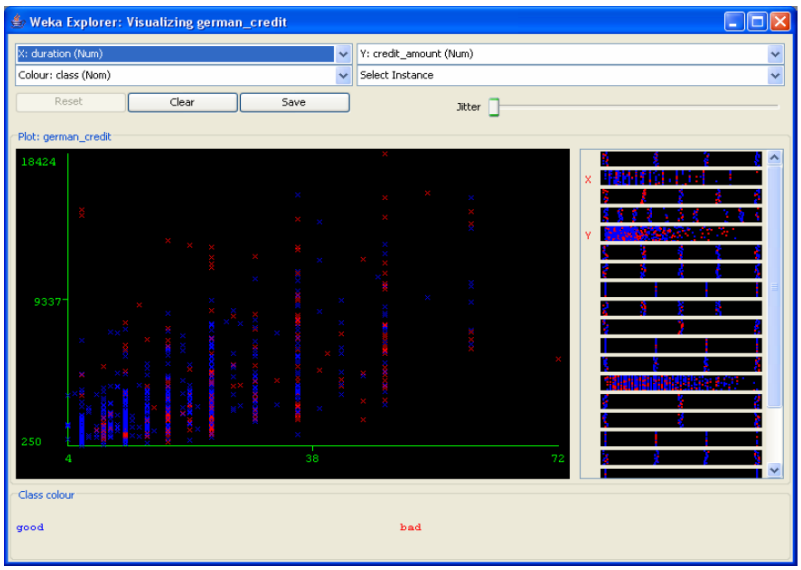


Ilustración 36: Ejemplo de gráficas de dispersión

En las gráficas de dispersión se puede mostrar una tercera dimensión. Cuando tenemos más de dos variables el gráfico anterior se puede repetir para todas las combinaciones posibles

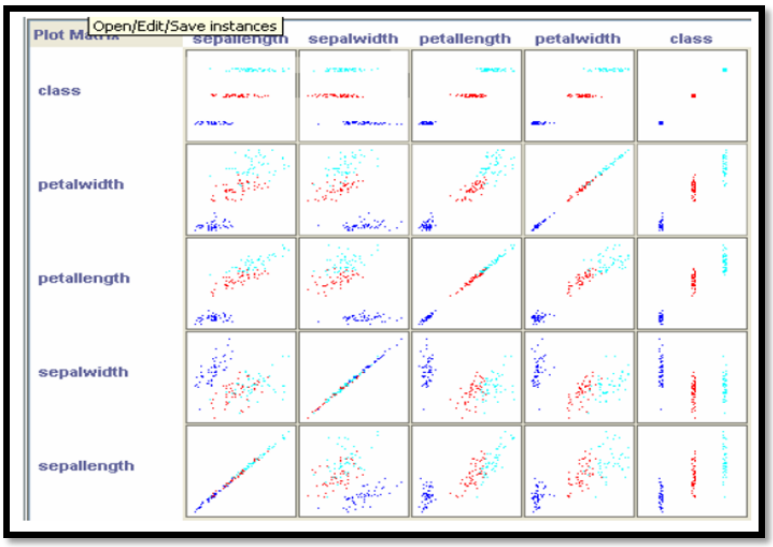


Ilustración 37: Matriz de graficas de dispersión etiquetadas (plotmatrix).

15.2.3 Valores Faltantes

Los valores faltantes, perdidos o ausentes (missing values) pueden ser reemplazados por varias razones. En primer lugar, el método de Minería de Datos que utilicemos puede no tratar bien los campos faltantes. En segundo lugar, podemos querer agregar los datos (especialmente los numéricos) para realizar otras Vistas Minables y que los valores faltantes no nos permitan agregar correctamente (totales, medias, etc.). En tercer lugar, si el método es capaz de tratar campos faltantes es posible que ignore todo el ejemplo (produciendo un sesgo) o es posible que tenga métodos de sustitución de campos faltantes que no sea adecuado debido a que no se conoce el contexto asociado al atributo faltante.

A la hora de hablar de campos faltantes, debemos tratar de su detección y de su tratamiento. La detección de campos faltantes puede parecer sencilla. Si los datos proceden de una base de datos, basta mirar la tabla de resumen de atributos/características y ver la cantidad de nulos que tiene cada atributo. El problema es que a veces los campos faltantes no están representados como nulos. Por ejemplo, aunque hay campos en los que las restricciones de integridad del sistema evitan introducir códigos fuera del formato para representar los valores faltantes, esto al final ocurre en muchos otros, especialmente en campos sin formato: direcciones o teléfono como “no tiene”, códigos postales, esto al final o números de tarjeta de crédito con valor -1 , etc. A veces son las propias restricciones de integridad las que causan el problema. Por ejemplo, ¿Cuántos sistemas obligan a introducir necesariamente los dos apellidos y fuerzan a que los extranjeros, por tanto, les pongamos un (“-”) en el segundo apellido. Este tipo de situaciones complica de sobremanera la detección de los valores faltantes. No obstante, son precisamente aquellos casos más difíciles en los que merece la pena realizar más esfuerzo, ya que muchas veces son este tipo de “nulos camuflados” los que pueden introducir sesgo en el conocimiento extraído.

Tanto para la detección, como para su tratamiento posterior, es importante saber el porqué de los valores faltantes.

- **Algunos valores faltantes:** expresan características relevantes: p.ej. la falta de teléfono puede representar en muchos casos un deseo de que no se moleste a la persona en cuestión, o un cambio de domicilio reciente.

- **Valores no existentes:** muchos valores faltantes existen en la realidad, pero otros no. P.ej. el cliente que se acaba de dar de alta no tiene consumo medio de los últimos 12 meses.

- **Datos incompletos:** si los datos vienen de fuentes diferentes, al combinarlos se suele hacer la unión y no la intersección de campos, con lo que muchos datos faltantes representan que esas tuplas vienen de una/s fuente/s diferente/s al resto.

Finalmente si se han conseguido establecer los datos faltantes e, idealmente, sus causas, procederemos a su tratamiento. Las posibles acciones sobre datos faltantes (missing values) son:

- **Ignorar** (dejar pasar): algunos algoritmos son robustos a datos faltantes (p.ej. árboles de decisión).

- **Filtrar** (eliminar o reemplazar) toda la columna (es decir quitar el atributo apara todos los ejemplos): solución extrema, pero a veces la cantidad de nulos es tan alta que la columna no tiene arreglo. Otras veces, existe otra columna dependiente con datos de mayor calidad.

- **Filtrar la fila:** claramente sesga los datos, porque muchas veces las causas de un dato faltante están relacionadas con casos o tipos especiales.

- **Reemplazar el valor**: por medias. A veces se puede predecir a partir de otros datos, utilizando cualquier técnica de Machine Learning.

- **Segmentar:** se segmentan las tuplas por los valores que tienen disponibles. Se obtienen modelos diferentes para cada segmento y luego se combinan.

- **Modificar la política de calidad de datos** y esperar hasta que los datos faltantes estén disponibles.

Quizás una de las soluciones anteriores más frecuentes cuando el algoritmo a utilizar no maneja bien los nulos sea reemplazar el valor. En este caso debemos de tener en cuenta que, primero, perdemos información y segundo inventamos información, con el riesgo de que pueda ser errónea. El primer problema también ocurre en el caso que eliminemos toda la columna.

La solución a ambos problemas pasa por crear un nuevo atributo lógico (booleano) indicando si el atributo original era nulo o no. Esto permite al proceso y el método de Minería de Datos, si son bastantes perspicaces, saber que el dato era faltante y, por tanto, el valor hay que tomarlo con cautela. En el caso en que el atributo original sea nominal no es necesario crear un nuevo atributo, basta con añadir un valor adicional, denominado “faltante”. Esta es una solución (sin ninguna acción adicional) bastante frecuente con atributos nominales.

Existen casos excepcionales, por ejemplo, cuando tratamos con datos un poco densos en el sentido de que casi todos los atributos tienen unos niveles muy altos de valores faltantes (por encima del 50 por ciento, por ejemplo). Reemplazar o filtrar no parece ser la solución. En estos casos existen soluciones particulares, por ejemplo intentar aglutinar varios ejemplos similares en uno solo y describir porcentaje de valores faltantes, valores medios, desviaciones, etc.

15.2.4 Valores erróneos. Detección de valores anómalos.

Del mismo modo que para los campos faltantes, para los campos erróneos o inválidos, hemos de distinguir entre la detección y el tratamiento de los mismos. La detección de los campos erróneos se puede realizar de maneras muy diversas, dependiendo del formato y origen del campo. En el caso de los datos nominales, la detección dependerá fundamentalmente de conocer el formato o de los posibles valores del campo. Por ejemplo si estamos tratando con un atributo con un formato fijo, como la matrícula de un vehículo, podemos establecer si una determinada matricula es errónea si no se ajunta al formato o los formatos permitidos en un cierto país. Parece evidente que aquellos datos erróneos que si se ajunten al formato serán más difíciles de (o imposibles) de detectar.

Resulta especialmente el hecho de que los sistemas de información originales fueron diseñados con buenas restricciones de integridad (no permitiendo, por ejemplo matrículas con un formato no correcto), todos estos campos erróneos no habrán podido introducirse. Este caso, obviamente es mucho más aconsejable.

En algunos casos, afortunadamente, es posible detectar campos erróneos por su contenido. Siguiendo con el ejemplo de las matrículas, podemos comparar (de una manera automática o semiautomática) los modelos de vehículos con las matrículas. Si las matrículas se numeran incrementalmente (“BFG 8940” después de “BFG 8939”), podemos observar si alguna matrícula supuestamente antigua está asociada a un modelo de vehículo muy moderno. Esta situación puede denotar un posible error.

En el caso de la detección de valores erróneos en atributos numéricos, ésta suele empezar por buscar valores anómalos, atípicos o extremos (outliers), también llamados datos aislados, exteriores o periféricos. Es importante destacar que un valor erróneo y un valor anómalo no son lo mismo. Como hemos dicho, existen casos en los que los valores extremos se categorizan anómalos estadísticamente pero son correctos, es decir, representan un dato fidedigno de la realidad. No obstante, así todo, puede ser un inconveniente para algunos métodos que se basan en el ajuste de pesos por ejemplo, como las redes neuronales. De modo similar, y mucho más frecuentemente, puede haber datos erróneos que caen en la “normalidad” y, por tanto, no pueden ser detectados.

Es importante destacar que no detectar un valor anómalo puede ser un problema importante si el atributo se normaliza posteriormente (entre cero y uno por ejemplo), ya que la mayoría de los datos estarán en un rango muy pequeño (entre 0 y 0,1 por ejemplo) y puede haber poca precisión o sensibilidad para algunos métodos de Minería de Datos.

Finalmente, llegamos al tratamiento de valores anómalos o erróneos (los que decidamos tratar de ambos casos, visto que anómalo no siempre indica un error). Los tratamientos son muy similares a los de los datos faltantes (outliers):

- **Ignorar (dejar pasar)**: algunos algoritmos son robustos a datos anómalos (p.ej. árboles).

- **Filtrar** (eliminar o reemplazar) la columna: solución extrema, pero a veces existe otra columna dependiente con datos de mayor calidad. Preferible a eliminar la columna es reemplazarla por una columna discreta diciendo si el valor era normal outlier (por encima o por debajo). En el caso de los anómalos se puede sustituir por no anómalo, anómalo superior anómalo inferior.

- **Filtrar la fila**: claramente se pueden sesgar los datos, porque muchas veces las causas de un dato erróneo están relacionadas con casos o tipos especiales.

- **Reemplazar el valor**: por el valor 'nulo' si el algoritmo lo trata bien o por máximos o mínimos, dependiendo por donde es el outlier, o por medias. A veces se puede *predecir* a partir de otros datos, utilizando cualquier técnica de ML.

- **Discretizar**: transformar un valor continuo en uno discreto (p.ej. muy alto, alto, medio, bajo, muy bajo) hace que los outliers/anómalos caigan en 'muy alto' o 'muy bajo' sin mayores problemas.

Los atributos erróneos son mucho más graves cuando afectan un atributo que va a ser utilizado con clase o como campo de salida de una tarea predictiva de Minería de Datos. De estos casos existe un tipo peculiar todavía más grave: todos los atributos de dos o más registros son idénticos excepto en la clase (esto es frecuente si existen atributos numéricos). Este tipo de errores se consideran frecuentemente como "inconsistencias" e incluso algunos métodos de Minería de Datos no pueden "digerirlos" (llegando a dar errores de ejecución), Por lo que se deben eliminar unificando (siempre que se pueda) los registros en una única clase.

15.3 Transformación de atributos. Creación de características.

La transformación de datos engloba, en realidad, cualquier proceso que modifique la forma de los datos. Prácticamente todos los procesos de preparación de datos entrañan algún tipo de transformación.

Por transformación entendemos aquellas técnicas que transforman un conjunto de atributos en otros, o bien derivan nuevos atributos, o bien cambian el tipo (mediante numerización y discretización) o el rango (mediante escalado).La selección de atributos (eliminar los menos relevantes) en realidad no transforma atributos y, en consecuencia, no entra en este grupo de técnicas.

15.3.1 Reducción de dimensionalidad por transformación

La alta dimensionalidad es, muchas veces, un gran problema a la hora de aprender de los datos. Si tenemos muchas dimensiones (atributos) respecto a la cantidad de instancias, pueden existir demasiados grados de libertad, por lo que los patrones extraídos pueden ser poco robustos.

Este problema se conoce popularmente como "la maldición de la dimensionalidad" *("the curse of dimensionality")*. Una manera de intentar resolver este problema es mediante la reducción de dimensiones.

La reducción se puede realizar por selección de un subconjunto de atributos, o bien la sustitución del conjunto de atributos iniciales por otros diferentes, que, geométricamente, se denomina proyección.

Análisis de componentes principales

La técnica más conocida para reducir la dimensionalidad por transformación se denomina "análisis de componentes principales" *("principal component analysis"),* PCA.

PCA transforma los *m* atributos originales en otro conjunto de atributos *p* donde $p \leq m$. Este proceso se puede ver geométricamente como un cambio de ejes en la representación (proyección).

Los nuevos atributos se generan de tal manera que son independientes entre sí y, además, los primeros tienen más relevancia (más contenido informacional) que los últimos. La transformación nos asegura que si ignoramos los últimos *p* atributos, estaremos descartando la información menos relevante.

Otros métodos de reducción de dimensionalidad por transformación

Existen otros métodos para reducir la dimensionalidad que no se basan en una selección de los atributos sino en una transformación de los mismos y pueden considerarse similares al análisis de componentes. Muchos de estos métodos se encuentran dentro de un área de técnicas más heterogéneas que se conoce como análisis factorial (*factorial analysis*). Por ejemplo existen métodos de **análisis factorial** para reducir el número de atributos basados en mínimos cuadrados (ponderados o no), basados en máxima verisimilitud, basados en factorización alfa y de imagen.

Sin embargo, las técnicas anteriores, sólo son capaces de realizar una reducción atendiendo a las relaciones lineales entre las variables originales. Para capturar relaciones no lineales hay que optar por modelos no lineales.

Una de las grandes limitaciones de la reducción de dimensionalidad por transformación es que los nuevos atributos no son representativos, en el sentido de que no son los atributos originales del problema. Por tanto, un método comprensible obtenido a partir de atributos transformados de esta manera (por ejemplo, un árbol de decisión o un modelo de regresión) no puede ser entendido por un experto, ya que las reglas están definidas en función de los atributos transformados y no de los originales.

Otra limitación es que el análisis de componentes principales clásico solo es aplicable a atributos numéricos.

Ejemplo de reducción de dimensionalidad por transformación

Ejemplo: Data set Segment, 19 Atributos

El método Principal Components genera 10 atributos cubriendo el 97% de la varianza

eigenvalue	proportion	cumulative	
7.6214	0.42341	0.42341	-0.357rawblue-mean-0.355value-mean-0.351intensity-mean-0.348rawred-mean-0.343rawgreen-mean...
2.91666	0.16204	0.58545	0.495hedge-sd+0.481vegde-sd+0.472hedge-mean+0.466vedge-mean+0.257short-line-density-2...
1.7927	0.09959	0.68504	0.596hue-mean+0.428exgreen-mean+0.373region-centroid-row-0.363saturation-mean-0.192exblue-mean...
1.05431	0.05857	0.74362	0.714short-line-density-5-0.677region-centroid-col+0.127short-line-density-2-0.07vegde-sd-0.057exgre...
0.93564	0.05198	0.7956	-0.63region-centroid-col-0.462short-line-density-5-0.453short-line-density-2+0.213exgreen-mean-0...
0.90907	0.0505	0.8461	-0.694short-line-density-2+0.483short-line-density-5+0.323region-centroid-col+0.282vegde-sd...
0.72745	0.04041	0.88651	0.456region-centroid-row-0.434saturation-mean-0.428short-line-density-2-0.387exgreen-mean...
0.56163	0.0312	0.91771	-0.514vedge-mean-0.438saturation-mean+0.406exred-mean+0.357hedge-sd-0.331region-centroid-row...
0.53996	0.03	0.94771	0.491saturation-mean-0.438vedge-mean+0.418hedge-mean+0.317hedge-sd-0.297exred-mean...
0.39511	0.02195	0.96966	0.498hedge-mean-0.473vegde-sd-0.424region-centroid-row+0.392vedge-mean-0.29hedge-sd...

Ejemplo de reducción de dimensionalidad por transformación

15.3.2 Aumento de dimensionalidad por transformación o construcción

Aunque hay muchos casos donde, razonablemente reducir la dimensionalidad es positivo, podemos ver otros casos donde nos interesa aumentar el número de atributos. Incluso, hablaremos en muchos casos de generar nuevos atributos que son combinación de otros y, en cierto modo, son atributos redundantes.

Aumento de dimensionalidad mediante núcleos

En el apartado anterior se mencionaron muchos métodos que aprovechan los atributos originales en otro conjunto de atributos nuevos y diferentes, con la propiedad de que el número de atributos después de la proyección sea menor que el inicial, también se puede hacer una transformación sea mayor que el original.

El objetivo, en estos casos, es aprovecharse de la “maldición de la dimensionalidad”, como no hay mal que por bien no venga, si aumentamos la dimensionalidad conseguimos, como

hemos dicho anteriormente, que los datos se separen en el espacio, facilitando, por aumento de dimensionalidad adecuado, podemos convertir algunos problemas no lineales e incluso irresolubles n problemas lineales, al “aclararse” el espacio.

Esto no es sencillo, pero se ha avanzado mucho en los últimos años. El concepto fundamental en este tipo de aumento de dimensionalidad es la utilización de funciones de núcleo (kernels), que permiten realizar estas proyecciones de manera adecuada.

Como hemos visto ya anteriormente, existe la posibilidad de utilizar un método de aprendizaje automático, por ejemplo, un agrupamiento, para generar un nuevo atributo, en este caso, el grupo al que pertenece el atributo. Esta es la base del **análisis discriminante lineal** y de los núcleos utilizados para añadir o transformar atributos.

Creación de características.

La creación o agregación de características consiste en crear nuevos atributos para mejorar la calidad, visualización o comprensibilidad del conocimiento extraído. La mayoría o todos los atributos originales se preservan, con lo que hablamos de *añadir* nuevos atributos y no de sustituirlos, como en el caso anterior. Este proceso recibe muchos nombres: construcción, creación o descubrimiento de características, inducción constructiva, invención de atributos, escoge y mezcla (*pick and mix*).

La importancia de añadir atributos se muestra patente cuando existen patrones complejos en los datos que no pueden ser adquiridos por el método de Minería de Datos utilizado. Por ejemplo, supongamos que tenemos los datos de la evolución de apertura de pólizas en los últimos 21 meses. Los datos se representan en la siguiente Ilustración como pequeños círculos:

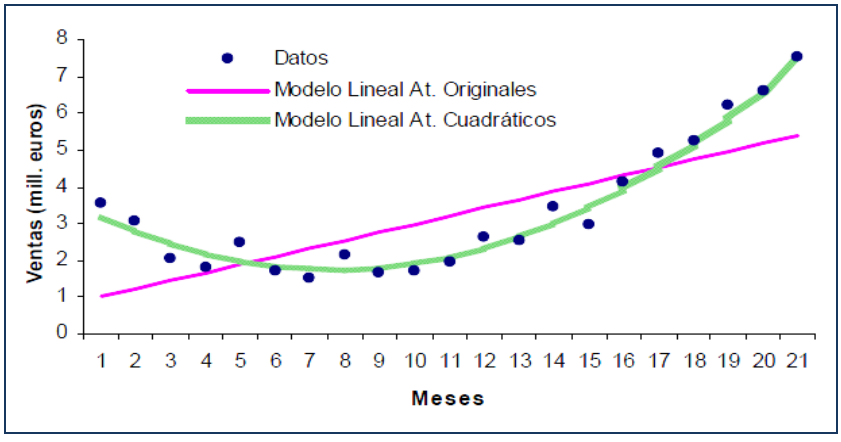


Ilustración 38: La importancia de crear características

- La regresión lineal no se aproxima a la solución
- El modelo de componentes cuadráticos se ajusta mucho mejor a los datos
- Añadiendo un nuevo atributo z=meses^2 se obtiene un buen modelo

En el ejemplo anterior, un simple atributo adicional, como el cuadrado de un atributo original, ha sido suficiente para ajustar un buen modelo con una técnica relativamente sencilla. Sin embargo disponemos de muchos atributos (tanto nominales como numéricos) y de métodos de aprendizaje más sofisticados. La "idea feliz" para crear nuevos atributos no resulta fácil cuando existen muchas posibles combinaciones.

En muchos casos la creación de buenas características o atributos depende no sólo de los propios datos, sino también del conocimiento que se tenga de los datos y del método de Minería de Datos que vaya a utilizar. De hecho, a veces se caracterizan las técnicas según están guiadas por los (*data -driven*), guiadas por la hipótesis o el modelo (*hypothesis-driven),* o guidas por el conocimiento (*knowledge-driven*).

Lógicamente, se pueden utilizar combinaciones muy complejas. Afortunadamente, es más útil (y más comprensible) el uso de relaciones simples entre los atributos. Dependiendo de si son numéricos o nominales podemos distinguir las técnicas:

- Atributos numéricos: se utilizan generalmente, operaciones matemáticas básicas de uno o más argumentos: suma, resta, producto, división, máximo, mínimo, media, cuadrado, raíz cuadrada, senos, cosenos, etc. Todas ellas retornan un valor numérico. También se pueden generar valores nominales a partir de valores numéricos, por ejemplo, crear una variable lógica que indique si un determinado valor numérico es mayor o menor que un determinado valor o que otro atributo, la igualdad estricta o aproximada entre atributos o valores, la desigualdad, etc.
- Atributos nominales: se utilizan, generalmente, operaciones lógicas: conjunción, disyunción, negación, implicación, condiciones M-de- M (M de N es cierto si y sólo si al menos M de las N condiciones son ciertas), igualdad o desigualdad, etc. Todas ellas retornan un valor nominal. También se pueden generar valores numéricos a partir de los valores nominales, por ejemplo, las variables X-de-N (se retorna el numero X de las N condiciones que son ciertas).

Un tipo especial de atributos "numéricos" son las fechas. Hay que ser cautelosos con ellas, ya que generalmente la fecha (incluso la hora) da muy poca información. En general, las fechas hay que transformarlas en nuevos atributos más significativos. La primera idea es generar tres atributos: día, mes y año, aunque se pueden generar otros relacionados.

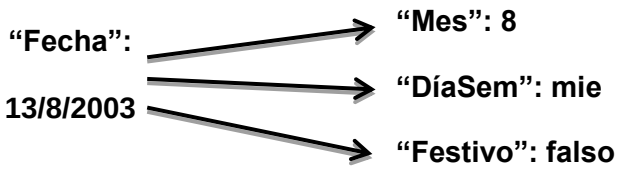


Ilustración 39: Convirtiendo fechas en atributos más significativos

Finalmente, el conocimiento del dominio es el factor que más determina la creación de buenos atributos derivados. La siguiente tabla muestra algunos ejemplos de atributos derivados en distintos dominios:

Atributo Derivado	Fórmula
Índice de obesidad	Altura ^{2}/ peso
Hombre familiar	Casado, varón e "hijo>0"
Síntomas SARS	3-de-5 (fiebre alta, vómitos, tos, diarrea, dolor de cabeza)
Riesgo póliza	X-de-N (edad < 25, varón, años de carné < 2, vehículo deportivo)
Beneficios brutos	Ingresos – Gastos
Beneficios netos	Ingresos – Gastos - Impuestos
Desplazamiento	Pasajeros * kilómetros
Duración media	Segundos de llamada / número de llamadas
Densidad	Población / Área
Retardo compra	Fecha compra – Fecha campaña

Tabla 10: Atributos derivados

15.4 Discretización y numerización

Uno de los aspectos más importantes, sino el que más, de un atributo es el tipo. El hecho de que un atributo sea nominal o numérico determina en gran cantidad como va a ser tratado por las herramientas de Minería de Datos. En algunas ocasiones puede ser conveniente convertir un numérico a nominal (discretización) o viceversa (numerización).

15.4.1 Discretización

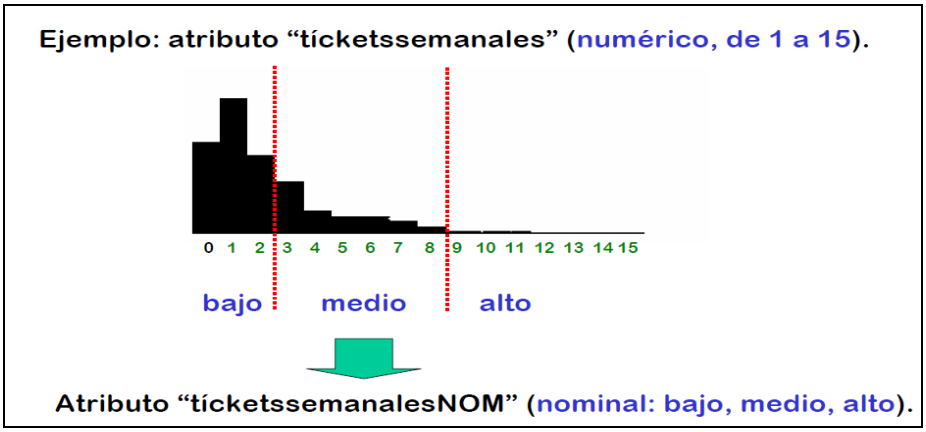


Ilustración 40: ejemplo de discretización y numerización

La discretización, o cubanización (también llamada "*binning*") es la conversión de un valor numérico en un valor nominal ordenado (que representa un intervalo o "*bin*").

Ejemplo:

f **= {3, 2, 1, 5, 4, 3, 1, 7, 5, 3}**

Ordenado:

F= {1, 1, 2, 3, 3, 3, 4, 5, 5, 7}

Particionado en **3 BINs**:

{1, 1, 2, 3, 3, 3, 4, 5, 5, 7}

Representación usando la moda:

{1, 1, 1, 3, 3, 3, 5, 5, 5, 5}

Usando media:

{1.33, 1.33, 1.33, 3, 3, 3, 5.25, 5.25, 5.25, 5.25}

Remplazando por el límite más cercano:

{1, 1, 2, 3, 3, 3, 4, 4, 4, 7}

- Valores numéricos que pueden ser ordenados de menor a mayor.
- Particionar en grupos con valores cercanos
- Cada grupo es representado por un simple valor (media, la mediana o la moda).
- Cuando el número de bins es pequeño, el límite más cercano puede ser usado para representar el bin.

La discretización se debe realizar cuando:

- El error en la medida puede ser grande o existen umbrales significativos (por ejemplo, notas de calificación, hay países como El Salvador en donde la calificación de aprobado empieza en 6 y no en 5 como en España)
- En ciertas zonas el rango de valores es más importante que en otras (interpretación no lineal)
- Aplicar ciertas tareas de MD que sólo soportan atributos nominales (por ejemplo, reglas de asociación).

En general, si no conocemos el atributo que queremos discretizar o queremos realizar la discretización de muchos atributos nos podemos ayudar de técnicas que nos indican, principalmente, donde hay que separar los intervalos y cuántos hay que obtener , ya que existen infinitas posibilidades. Cuando no existe una tarea asignada para los datos el proceso es más complejo, porque no se sabe muy bien con respecto a qué evaluar (qué discretizaciones son mejores o peores). Se pueden utilizar técnicas similares al análisis de correspondencias para ver si ciertas discretizaciones tienen un efecto mayor o menor sobre el resto de variables.

Cuando la tarea final que se pretende es clasificación, los métodos de discretización son más sencillos y se basan en medidas de separabilidad o entropía. Entropía es una medida global que es menor para con Ilustraciones ordenadas, y grande para con Ilustraciones desordenadas. Desventaja: complejidad, Implementación paralela.

15.4.2 Numerización

Es el proceso inverso a la discretización. Aunque es menos común que la discretización, también existen casos donde puede ser extremadamente útil. Un primer caso obvio donde la numerización puede ser útil es cuando el método de MD que vamos a utilizar no admite datos nominal. Un claro ejemplo es si pretendemos utilizar regresión lineal para obtener la influencia de cada "factor". En general, es preciso para muchos de los métodos de modelización estadística, como la regresión logística (o multinominal), análisis ANOVA, los modelos log-linear, los discriminantes paramétricos o no paramétricos, etc.

En todos estos casos lo que se suele hacer es lo que se denomina **numerización "1 a n"**: Si una variable nominal *x* tiene posibles valores creamos *n* variables numéricas, con valores 0 ó 1 dependiendo de si la variable nominal toma ese valor o no. Podemos también prescindir del último atributo pues es dependiente del resto (numerización "1 a n-1"). Por ejemplo; si en una inmobiliaria tenemos un atributo para el tipo de inmueble, que puede tomar el valor "adosado/chalet/piso/negocio/solar", si queremos estudiar qué tipo de inmuebles son más interesantes para una constructora, la partición de dos reglas formadas por las condiciones "inmueble_solar= V" e "inmueble_solar=F" es mucho más breve y general que la partición en cinco reglas dada por las condiciones "inmueble = solar", "inmueble = adosado", "inmueble = chalet", "inmueble = piso" e "inmueble = negocio" . Por ejemplo, un árbol de decisión sin particiones nominales binarias repetiría cuatro modelos idénticos para los casos en que el inmueble no es solar, cuando todo esto se fusionaría para el caso de las etiquetas numerizadas.

En algunos casos es posible identificar un cierto orden o magnitud en los valores de un atributo nominal y entonces podemos realizar una numerización denominada **numerización "1 a 1":** Se aplica si existe un cierto orden o magnitud en los valores del atributo nominal. Por ejemplo, si tenemos categorías del estilo {niño, joven, adulto, anciano} podemos numerar los valores de 1 a 4.

15.5 Normalización de rango: escalado y Centrado

En muchos métodos de MD no es necesario centrar los datos ni escalar. Es decir normalizar el rango de un valor numérico. De hecho se usan métodos como árboles de decisión o los sistemas de reglas, que son comprensibles, escalar previamente hace que los modelos resultantes sean más difíciles de interpretar (hay que invertir el escalado final).

Sin embargo, en muchas técnicas es necesario normalizar todos los atributos al mismo rango. Por ejemplo, es preciso normalizar a la misma medida cuando el atributo proviene de distintas fuentes (euros, pesetas). En el análisis de componentes principales (PCA), se

requiere normalizar el rango entre cero y uno. También algunos métodos de aprendizaje funcionan mejor con los atributos normalizados. Por ejemplo, los métodos basados en distancias. Ya que las distancias debidas a diferencias de un atributo que van entre 0 y 100 serán mucho mayores que las distancias debidas a diferencias que va entre 0 y 10.

La normalización más común es la **normalización lineal uniforme** y se normaliza a una escala genérica entre 0 y 1 utilizando la siguiente fórmula:

$$V' = \frac{v - min}{\max - min}$$

El resultado de esta normalización es que la relación (el coeficiente) entre los valores se mantiene, para realizar esta normalización sólo es necesario conocer el máximo y mínimo de los valores dados para ese atributo.

Sin embargo, esta normalización es muy sensible a la presencia de valores anómalos (outliers).

Una solución a este problema es el escalado *softmax* o **escalado *sigmoidal*,** en el que no se usa una transformación lineal, sino una transformación que es más pronunciada en el centro y más aplanada en los bordes. Para ello se utiliza una función sigmoidal, como por ejemplo: *1/exp (-20(x-0,5)).*

La siguiente Ilustración muestra una posible función para realizar un escalado Softmax (suponiendo que se le ha hecho un escalado lineal previamente o la variable original va entre 0 y 1.)

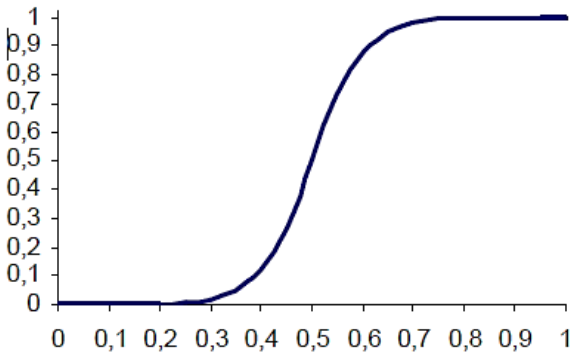


Ilustración 41: Funcion sigmoidal (o logística para realizar un escalado softmax.

Normalización de un atributo, por ejemplo, podemos tener tres atributos: atrubuto1 en euros, atributo2 en euros y atributo3 en metros. Se puede optar por normalizar independientemente los 3 entre cero y uno, o se puede optar por normalizar el atributo1 y el atributo2 usando el máximo y el mínimo encontrados en ambos atributos, para mantener la relación.

GUÍA DE APOYO PARA EL ESTUDIANTE

Tema 4: Limpieza y transformación de los datos

I. COMPLETA EL ESPACIO EN BLANCO DE LAS SIGUIENTES AFIRMACIONES CON LA RESPUESTA CORRECTA:

a) El éxito de un proceso de Minería de Datos (MD) depende no sólo de una buena recopilación sino de que estos estén_____, _____y_____ _____

b) Las claves internas de sistemas diseñados pueden entrañar información no normalizada, que es preciso detectar en el proceso de integración. Este proceso se denomina _____

c) La _____ es la conversión de un valor numérico en un valor nominal ordenado

d) La _____ es la conversión de un valor nominal ordenado en un valor numérico

II. CONTESTE VERDADERO O FALSO A LAS SIGUIENTES AFIRMACIONES:

a) El primer paso a la hora de realizar una integración de distintas fuentes de datos es: conseguir que datos sobre el mismo objeto se unifiquen y datos de diferentes objetos permanezcan separados. _____

b) Existen 2 tipos de errores que pueden ocurrir en esta integración: que dos o más objetos diferentes se unifiquen y dos o más fuentes de objetos iguales se dejan separadas.

c) Se puede decir que un valor erróneo y un valor anómalo son lo mismo._____

d) El proceso de transformación se encarga de cambiar los formatos de datos del sistema fuente al sistema destino, así como de realizar la integración de las fuentes y la estandarización de los datos_____

III. MENCIONA:

a) Las posibles acciones sobre datos faltantes (missing values):

_____

_____

b) Los tratamientos de valores anómalos o erróneos:

_____

_____

SOLUCIÓN DE GUÍA DE APOYO PARA EL ESTUDIANTE

Tema 4: Limpieza y transformación de los datos

I. COMPLETA EL ESPACIO EN BLANCO DE LAS SIGUIENTES AFIRMACIONES CON LA RESPUESTA CORRECTA:

a) El éxito de un proceso de Minería de Datos (MD) depende no sólo de una buena recopilación sino de que estos estén **íntegros**, **completos** y **consistentes**
b) Las claves internas de sistemas diseñados pueden entrañar información no normalizada, que es preciso detectar en el proceso de integración. Este proceso se denomina ***descomposición de claves.***
c) La **discretización** es la conversión de un valor numérico en un valor nominal ordenado
d) La **numerización** es la conversión de un valor nominal ordenado en un valor numérico

II. CONTESTE VERDADERO O FALSO A LAS SIGUIENTES AFIRMACIONES:

a) El primer paso a la hora de realizar una integración de distintas fuentes de datos es: conseguir que datos sobre el mismo objeto se unifiquen y datos de diferentes objetos permanezcan separados. **V**

b) Existen 2 tipos de errores que pueden ocurrir en esta integración: que dos o más objetos diferentes se unifiquen y dos o más fuentes de objetos iguales se dejan separadas **V**
c) Se puede decir que un valor erróneo y un valor anómalo son lo mismo. **F**

d) El proceso de transformación se encarga de cambiar los formatos de datos del sistema fuente al sistema destino, así como de realizar la integración de las fuentes y la estandarización de los datos**V**

III. MENCIONE

a) Las posibles acciones sobre datos faltantes (missing values): Ignorar (dejar pasar), filtrar (eliminar o reemplazar), filtrar la fila, reemplazar el valor, segmentar, modificar la política de calidad de datos y esperar hasta que los datos faltantes estén disponibles.
b) Los tratamientos de valores anómalos o erróneos: Ignorar (dejar pasar), filtrar, filtrar la fila, reemplazar el valor y discretizar.

16) TEMA 5: EXPLORACIÓN Y SELECCIÓN DE DATOS

Objetivos:

- Conocer algunas técnicas simples del análisis exploratorio de datos
- Conocer técnicas de visualización previa, de agrupamiento exploratorio y técnicas de selección.

Contenido:

- Introducción
- Contexto de la vista minable
- Exploración mediante visualización
- Sumarización , descripción, generalización y pivotamiento
- Selección de datos
- Lenguajes, primitivas e interfaces de MD

Duración: 4hrs.

Bibliografía:

- Introducción a Minería de Datos. José Hernández Orallo, Mª. José Ramírez Quintana, Cesar Ferri Ramírez.

16.1 Introducción. Contexto de la vista minable

Una vez los datos están recopilados, integrados y limpios, todavía no estamos listos (en muchos casos) para realizar una tarea de MD. Es necesario, además, realizar un reconocimiento o análisis exploratorio de los datos con el objetivo de conocerlos mejor de cara a la tarea de MD. Incluso esta fase es imprescindible cuando se realiza Minería de Datos "abierta", ya que tenemos todo el volumen de datos pero hemos de determinar los datos a seleccionar y las tareas a realizar sobre esos datos.

Usted se preguntará, entre otras cosas, lo siguiente:

- ¿Qué parte de los datos es pertinente analizar?
- ¿Qué tipo de conocimiento se desea extraer y cómo debe presentárselo?
- ¿Qué conocimiento puede ser válido, novedoso e interesante?
- ¿Qué tipo de conocimiento previo me hace falta para realizar esta tarea?

Lógicamente usted no será capaz de extraer conocimiento si no se le responde a dichas preguntas. Del mismo modo una herramienta de Minería de Datos, no puede digerir un conjunto de datos y producir algo razonable, si no se le orienta. La razón fundamental del porqué esto es así radica no sólo en la incapacidad actual de las herramientas a realizar algunas de estas tareas de una manera completamente automática, sino, fundamentalmente, en que la extracción de conocimiento viene a cubrir unas necesidades y expectativas, que deben indicarse, en cierto modo, de forma interactiva. Usted puede realizar la compra en un supermercado, por Internet, o se la puede hacer su mayordomo, pero, en ningún caso, podrá realizar una compra si no indica lo que quiere.

Por tanto, es necesario expresar y proporcionar las respuestas a las 4 preguntas anteriores, ya sea mediante lenguajes de MD, interactivamente en herramientas especializadas o seleccionando aquellas herramientas necesarias. Incluso conociendo los datos y el dominio del que provienen, responder a algunas de ellas no es sencillo. Es necesario, en muchos casos, *explorar los datos*, el contexto y los usuarios de la información.

Las 4 preguntas anteriores son, en realidad, una manera de clasificar el conjunto de preguntas que se podrían realizar, ya que, son preguntas interrelacionadas. Por ejemplo, si no se sabe el conocimiento que puede ser útil no se puede decidir que parte de los datos lo pueden proporcionar. Por el contrario, si no se selecciona y se estudia un conjunto de los datos no se puede saber qué validez pueden tener los modelos extraídos y si finalmente van a ser útiles. Otro ejemplo similar es determinar el modelo de MD; observando los datos puedo ver qué método puede ser más aconsejable. Sólo determinar el método se puede saber si hay ciertos atributos que hará falta cambiar o eliminar. De modos diversos se interrelacionan estas preguntas acerca del qué, del dónde y del cómo.

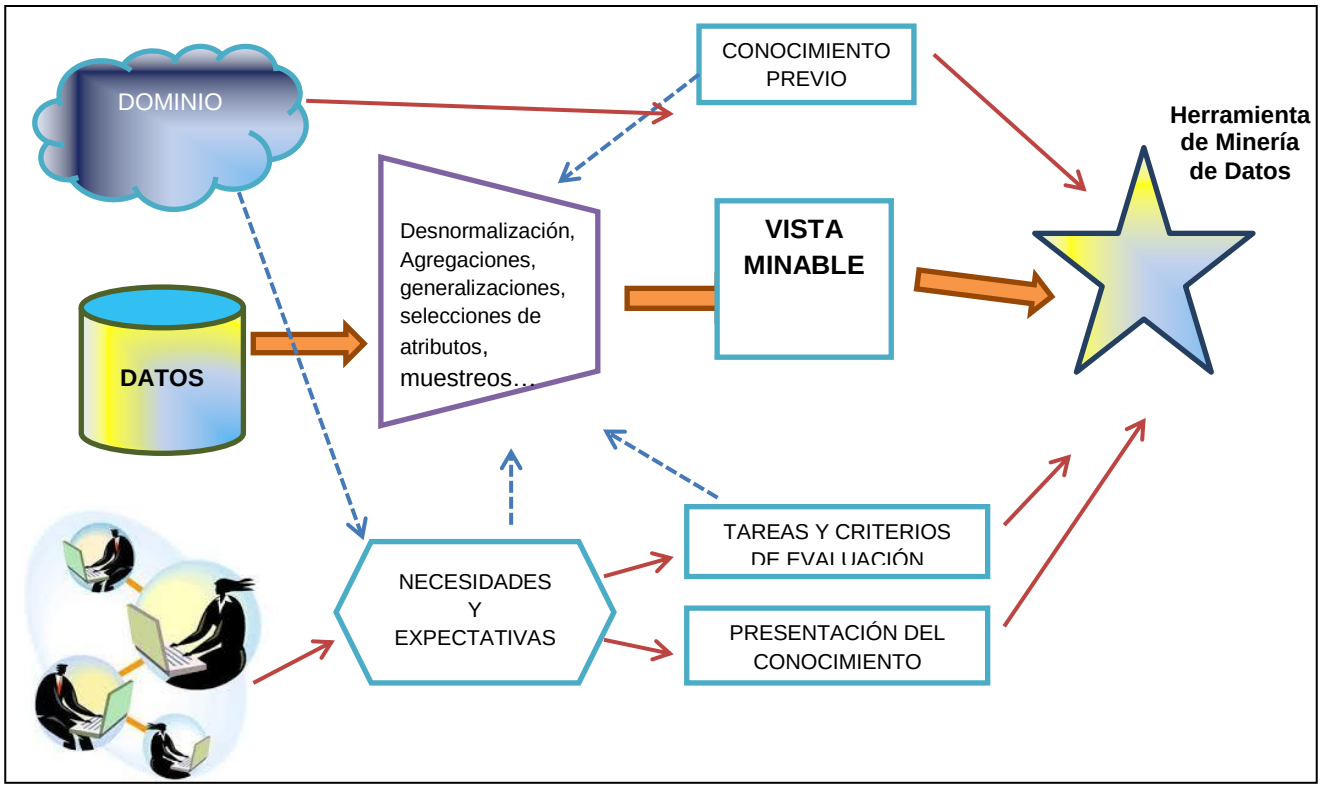


Ilustración 42: De los datos, dominio y usuarios a la vista minable y elementos asociados.

La ilustración 42 intenta esquematizar el proceso que lleva de los datos, del conocimiento del dominio y de los usuarios a los 4 aspectos anteriores que son necesarios para llevar a cabo la fase propia de MD.

Como vemos, no es sólo necesario obtener la vista minable (una tabla con los atributos relevantes) sino que debe ir acompañada de la tarea a realizar sobre ella y cómo evaluarla, así como la forma de presentar el resultado final y, en su uso, el conocimiento previo necesario. Demos nombres y entendamos las 4 preguntas anteriores

- **Vista minable**: ¿qué parte de los datos es pertinente analizar? Una vista minable consiste en una vista en el sentido más clásico de BD: una tabla. La mayoría de métodos de Minería de Datos, como veremos, son sólo capaces de tratar una tabla en cada tarea. Por tanto, la vista minable ha de recoger toda (y sólo) la información necesaria para realizar la tarea de MD.

- **Tarea, método y presentación**: ¿qué tipo de conocimiento se desea extraer y cómo debe presentárselo? Se trata de decidir qué tarea (clasificación, regresión,

agrupamiento, reglas de asociación, etc.) cuáles son las entradas y salidas (en las tareas predictivas), con qué método, entre los existentes para cada tarea (árboles de decisión, redes neuronales, regresión logística, etc.) y de qué manera se van a presentar o se van a navegar los resultados (gráficamente, como un árbol, como un conjunto de reglas, etc.)

- **Criterios de calidad**: ¿qué conocimiento puede ser válido, novedoso e interesante? En muchos casos hay que establecer unos criterios de comprensibilidad de los modelos (número de reglas máximo), criterios de fiabilidad (basados en medidas como la confianza para las reglas de asociación, el error cuadrático medio para la regresión, etc.), criterios de utilidad (basados en medidas de cuándo son aplicables, como soporte, qué beneficios se obtiene, a partir de matrices de costos, etc.), y criterios de novedad o interés (basados en medidas más o menos subjetivas).

- **Conocimiento previo:** ¿qué tipo de conocimiento previo me hace falta para realizar esta tarea? Tanto a la hora de construir la vista minable final o para ayudar al propio algoritmo de Minería de Datos puede ser necesario establecer e incluso expresar de una manera formal cierto conocimiento previo. Por ejemplo, las jerarquías de conceptos o de dimensiones OLAP permiten trabajar con los datos y generar atributos, existen funciones que pueden utilizarse en medidas de similitud o a la hora de expresar los modelos, se pueden añadir otras tablas como conocimiento previo o incluso se pueden añadir otros modelos anteriores como apoyo o sobre los cuales revisar o construir uno nuevo.

Por ejemplo, supongamos que hemos recolectado la información sobre diagnósticos y recetas de atención primaria de toda una zona sanitaria. Nuestro objetivo es extraer conocimiento de estos datos. En primer lugar, antes incluso que mirar los datos, establecemos una serie de entrevistas con los jefes de servicio de atención primaria de la zona. Entre las cosas que salen a la luz en las entrevistas es su preocupación por que una multitud de nuevos medicamentos han aparecido recientemente para una serie de dolencias crónicas, y la mayoría de médicos prescriben de una manera aleatoria de entre medicamentos generalmente efectivos, o como mucho, siguiendo patrones globales de éxito de cada medicamento (prueba el “a” antes que el “b”, etc.). Esto tiene como consecuencia que, en muchos casos, a los pocos días el paciente vuelve a la consulta, y el médico le receta otro medicamento, hasta que dan con el medicamento realmente efectivo y que no muestre contraindicaciones no previstas. Entre las reuniones que aparecen se encuentra la de realizar modelos que determinen, según el paciente, qué medicamentos prescribir primero, con el objetivo de resolver cuanto antes el problema sanitario del paciente, evitar nuevas visitas de los pacientes “(reducción de visitas) y reducción de costos farmacéuticos.

A partir de este ejemplo, se pueden estudiar varias patologías, si nos centramos en una sola, tendremos que la vista minable va a formarse a partir de los diagnósticos de dichas patologías y los medicamentos prescritos. El medicamento satisfactorio es el último prescrito, ya que, se supone, que si no hay más registros del mismo paciente y patología, el último medicamento fue bien. Por tanto habrá que realizar un tipo de consulta que nos seleccione el

último medicamento prescrito a los pacientes de una patología (excluyendo los de menos de un mes, para tener más perspectiva). Los factores que vamos a incluir de los antecedentes son todos aquellos existentes del historial del paciente: parámetros generales: edad, tensión…, análisis de sangre, etc.

La tarea a realizar es una tarea de clasificación, ya sea completa o parcial (por ejemplo se podría realizar un subconjunto de reglas de asociación que ayudaran en los casos más claros). Debido a la características de los usuarios (médicos) y a la exigencia de comprensibilidad de los modelos (para su evaluación facultativa), se decide que los patrones extraídos estarán expresados en forma de árboles de decisión, ya que los médicos están acostumbrados a seguir este tipo de árboles a la hora de hacer diagnósticos o prescribir medicamentos.

Qué se puede hacer para conocer mejor los datos, el contexto y los usuarios. Englobemos el reconocimiento en dos aspectos: reconocimiento del dominio y de los usuarios y, reconocimiento y exploración de los datos

16.1.1 Reconocimiento del dominio y de los usuarios:

Para conocer qué se puede hacer con unos ciertos datos es necesario conocer el dominio y los usuarios. Si usted es el gerente o un directivo de una empresa o departamento que conoce bien, probablemente no necesite realizar este reconocimiento. Pero si usted va a dedicarse a la MD de varios clientes, usted será ajeno al dominio. Una de las tareas a realizar será, por tanto, conocer y reconocer el dominio y los usuarios.

Para ello, realizaremos preguntas del estilo: ¿Qué aspectos son cruciales en su negocio? ¿Qué reglas o modelos de decisión está utilizando? ¿Se pueden mejorar dichas reglas? ¿Existen decisiones que se toman de manera arbitraria o basándose en reflexiones personales no explicitas? ¿Existe documentación sobre decisiones anteriores? ¿Quiénes toman las decisiones? ¿Qué decisiones son críticas? ¿Los modelos deben ser comprendidos y validados por expertos? ¿Qué otros requerimientos exigiríamos a los patrones extraídos? ¿Qué conocimiento previo suele utilizar para apoyarse en sus decisiones? ¿Utiliza otras fuentes de datos externas para fundamentarse en sus decisiones? Y un largo etcéteras de preguntas de este estilo. Algunas de estas cuestiones también son útiles y se pueden realizar a la hora de construir un almacén de datos en el momento de integración. Este reconocimiento se puede establecer como una fase previa a la MD, en la que se establecen los *requerimientos y objetivos del negocio*.

El resultado de este "*reconocimiento"* puede resumirse en una documentación u organizarse de una manera esquemática, estableciendo prioridades de análisis, destacando aquellas reglas de decisión importantes, que pueden mejorarse de manera significativa y para las cuales parece que disponemos de datos.

De este modo, se van descubriendo mayores posibilidades a medida que va conociendo el dominio. Como hemos dicho, sin este reconocimiento es imposible esclarecer las tareas, los métodos, los criterios de calidad, explorar los datos y el conocimiento previo

16.1.2 Reconocimiento y exploración de los datos

Además del reconocimiento del dominio, debemos reconocer los datos. Para ello, lógicamente debemos conocer lo que significan y esto sólo es posible si quien lo realiza conoce el dominio o los datos (ya sea porque son sus propios datos y dominio o porque ha hecho el reconocimiento del dominio). El reconocimiento de los datos por tanto viene guiado por el interés y las necesidades establecidas en el reconocimiento de dominio. Sin este no se puede saber qué datos son relevantes ni que tareas pueden ser útiles.

El reconocimiento de datos se suele conocer con distintos nombres en inglés (data survey, exploratory data analysis, data fishing…). De modo similar, en castellano, también se puede utilizar términos diversos: exploración, prospección…

De los datos seguimos transformando y seleccionando con el objetivo de obtener una "vista minable", lista ya para ser tratada por las herramientas de MD. Las herramientas de exploración y selección requieren saber las expectativas y necesidades del dominio o, de una forma más concreta, la tarea y el conocimiento previo pueden influir más en estas transformaciones y selecciones.

16.2 Exploración mediante visualización

El objetivo de la exploración para la MD es obtener una vista minable, con tarea asignada. Para ello, se pueden utilizar distintas técnicas para obtener o refinar dicha vista: visualización, descripción, generalización, agregación y selección.

En los temas anteriores vimos algunos tipos de tablas, como la tabla de resumen de características, y algunas gráficas, como los histogramas y las gráficas de distribución. Estas gráficas en general se centran en uno o dos atributos, a lo sumo, y el objetivo principal era la limpieza de datos. En este apartado veremos algunas gráficas más con un objetivo diferente, intentar sugerir tareas de MD o patrones que puedan extraerse. Las gráficas que veremos en este apartado se pueden caracterizar por dos aspectos: o bien son interactivas y permiten una exploración activa, o bien son multidimensionales, con lo que permiten observar muchos atributos a la vez.

Recientemente ha aparecido el término "Minería de Datos visual" (visual Data Mining) con el significado de una MD que se realiza manejando e interactuando con gráficos.

Lo que caracteriza la Minería de Datos de técnicas anteriores o de la perspectiva clásica del análisis de datos es que los modelos son extraídos por algoritmos y, por tanto, no son vistos o descubiertos por el usuario (y posteriormente simplemente validados estadísticamente).

Las técnicas de visualización de datos se utilizan fundamentalmente con dos objetivos:

- Aprovechar la gran capacidad humana de ver patrones descubiertos y tendencias a partir de imágenes y facilitar la compresión de los datos.
- Ayudar al usuario a comprender más rápidamente patrones descubiertos automáticamente por un sistema de KDD.

Estos dos objetivos marcan dos momentos diferentes del uso de la visualización de los datos (no excluyentes):

- **Visualización previa**: (esta es la normalmente recibe el nombre de Minería de Datos visual) se utiliza para entender mejor los datos y sugerir posibles patrones o qué tipo de herramienta de KDD utilizar.

  Las herramientas gráficas requieren mayor experiencia para seleccionar qué gráfico nos interesa utilizar entre los cientos de gráficas que proporcionan los sistemas actuales.

  Ejemplo: segmentación mediante funciones de densidad, generalmente representadas tridimensionalmente.

  Los seres humanos ven claramente los segmentos (clusters) que aparecen con distintos parámetros

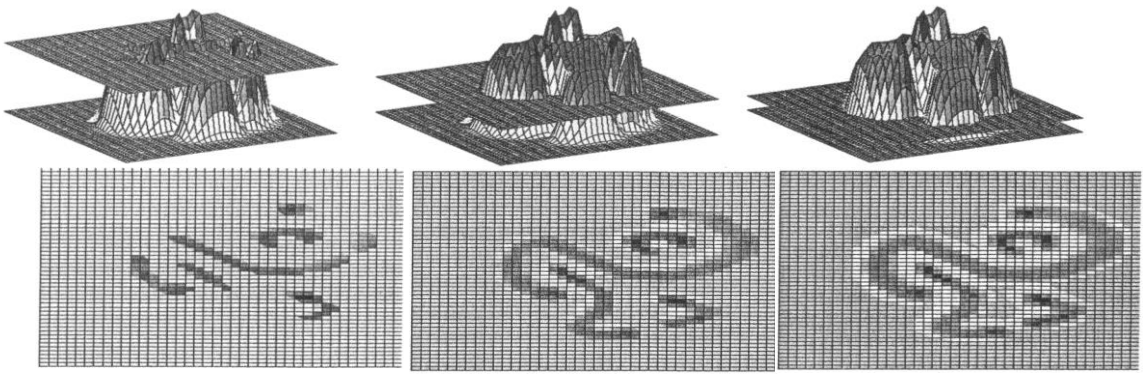


Ilustración 43: ejemplo de grafica de visualización previa

- **Visualización posterior**: al proceso de MD se utiliza para mostrar los patrones y entenderlos mejor. La visualización posterior se utiliza frecuentemente para validar a los expertos y mostrar los resultados de la extracción de conocimiento.

Ejemplo:
Se muestra el grado de asociación según la línea que conecta los valores (continúa gruesa, continua, discontinua o inexistente):

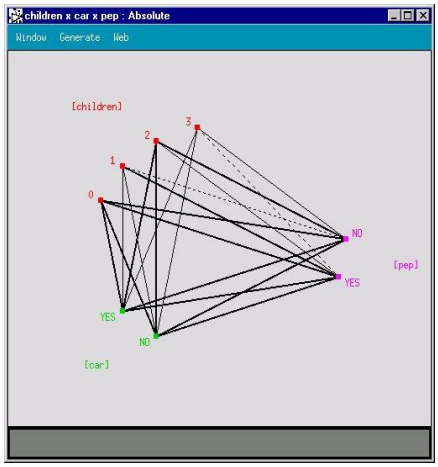


Ilustración 44: Ejemplo de grafica visualización posterior

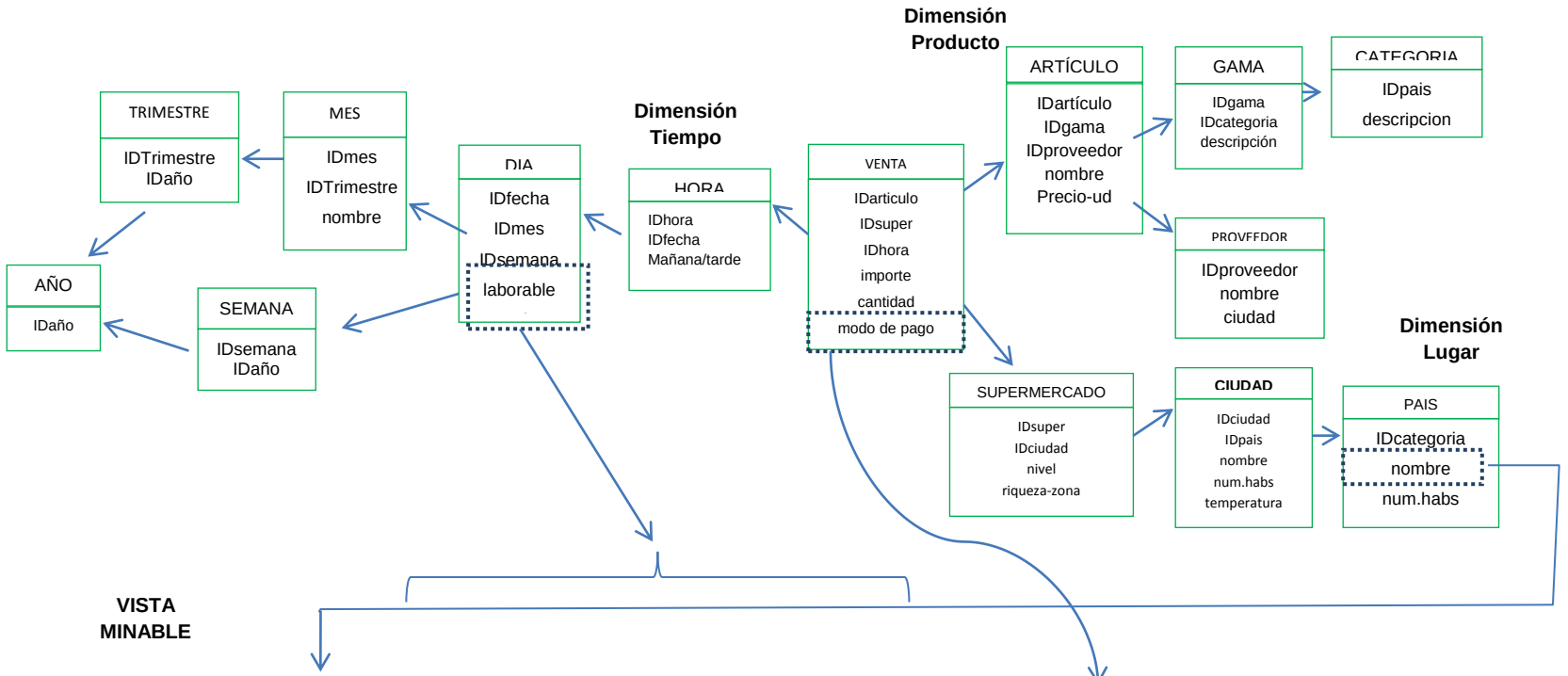
16.3 Sumarización, descripción, generalización y pivotamiento

La construcción de una vista minable es un proceso iterativo que pasa por conocer y visualizar los datos, combinados de diferentes maneras. Para esta combinación podemos utilizar operadores de consultas de bases de datos y operadores OLAP. Los datos con los que trabaja la MD son, muy frecuentemente, datos históricos que, por tanto pueden agregarse a diferentes niveles de detalle temporal. Si, además, la estructura de los datos es multidimensional (por ejemplo un *datamart) o* existen campos de agregación podemos obtener diferentes vistas concatenando diferentes tablas y agregando al nivel que deseemos.

Una pregunta que aparece generalmente en el entorno de la MD es la siguiente: “si ya he decidido qué tablas y atributos son relevantes ¿por qué debo construir una única tabla derivada, denominada vista minable? ¿No es suficiente con marcar dichos atributos y dejar a la herramienta de MD que trabaje sobre las BD?”. Existen 2 razones fundamentales para contestar esta pregunta. La primera es que dadas varias tablas, incluso aunque tenga claves ajenas definidas, existen muchas maneras de concatenarlas, es decir, de combinar información que contienen. Por tanto es más difícil definir tareas concretas si no se clarifica exactamente la información sobre la que se van a definir. La segunda razón es quizás más importante: la mayoría de métodos de MD sólo tratan con una sola tabla. La mayoría de técnicas sólo son capaces de trabajar con representaciones del estilo atributo- valor, es decir, una tabla.

Por tanto, debemos definir una consulta o vista minable. Para ello, las operaciones necesarias son aquellas de un lenguaje relacional (como por ejemplo SQL), concatenaciones (*join*sen inglés que significa unir), selecciones, proyecciones,

agrupamiento/agregaciones, etc. La I lustración muestra precisamente la construcción de una vista minable a partir de un conjunto de tablas. Aunque las tablas tienen una estructura multidimensional y podamos apoyarnos en herramientas OLAP, en realidad, las operaciones necesarias son las típicas de una consulta SQL: concatenación, selección, proyección y agrupamiento.



Pais.Nombre	Dia.Nombre	Laborable	% (modo de pago)=tarjeta
España	Lunes	Sí	34,5%
España	Martes	Sí	32,1%
España	Miercoles	Sí	33,8%
España	Jueves	Sí	31,7%
España	Viernes	Sí	36,3%
España	sábado	No	53,2%
España	Domingo	No	61,4%

Nombre = 'España'

Ilustración 45: Selección de tablas, atributos, condiciones y niveles de agregación para obtener una vista minable

Es quizás la concatenación de tablas aquella que permite juntar en una tabla la información proveniente de varias. Este proceso generalmente obtiene vistas desnormalizados, en el que por ejemplo, la tabla ciudad y país se funden en una sola, donde aparece el nombre de la ciudad y el nombre del país. Este tipo de desnormalizaciones contienen redundancia y, por tanto, patrones. Por tanto, hay que ser consciente de ellos, porque si incluyéramos todos los atributos para reglas de asociación, por ejemplo, tendremos patrones *redescubiertos* del estilo de dependencias funcionales como "ciudad →país" o, en el ejemplo anterior "dia.nombre→laborable".

16.3.1 Sumarización

La sumarización o agregación muestra los datos de una manera más resumida, permitiendo, precisamente, calcular valores agregados, que no son los datos directos registrados, sino derivados de ellos. Se puede considerar, en cierto modo, una generalización de los datos y, por tanto, suele facilitar el aprendizaje. Pero no sólo es una cuestión de eficiencia, sino, muchas veces, una necesidad. En la mayoría de los eventos físicos cuando más se detallan los datos menos patrones suelen encontrarse. Esto es patente, por ejemplo, en los fenómenos meteorológicos: no podemos establecer un patrón de si hará sol el día 31 de diciembre del 2015 en Valparaíso (chile) pero sí que podemos afirmar que en el mes de julio. Tampoco podemos predecir cuántos pacientes ingresarán en urgencias el sábado que viene por la noche de 3:00h a 4:00h, pero podemos asegurar que los sábados por la noche hay muchas más urgencias que el resto de las noches de la semana. Para establecer estos patrones, debemos agregar los datos.

Además, de la agregación aparecen nuevos atributos, que pueden ser mucho más significativos que los atributos más detallados. Por ejemplo podemos tener los atributos típicos de los clientes (edad, estado civil, dirección…). En otras tablas podemos tener información sobre los productos comprados anteriormente y cuándo. Nos puede interesar generar nuevos atributos como por ejemplo "gasto medio por mes", "número de productos comprados por año", etc. estos atributos son, en realidad, nuevos atributos por agregación y numerización (*featurine aggregation*).

La sumarización se puede utilizar no sólo para construir la vista minable directamente, sino para realizar un análisis exploratorio, similar, en cierto modo a las gráficas del punto anterior. Una aplicación muy práctica de la sumarización es la **comparación** o **discriminación de clases.** Consiste en sumarizar para dos o más clases, es decir, agrupar, y ver las características para las submuestras formadas. Las clases pueden ser las que finalmente van a servir para una tarea de clasificación o pueden ser cualquier atributo nominal (o numérico separado por intervalos) elegido para realizar el "contraste".

Por ejemplo, pacientes de enfermedades cardiovasculares, podríamos querer ver cuál es la distribución por sexos de algunos de los otros atributos:

		EDAD		Colesterol		Obesidad		Fumador	
Sexo	Card.	Media	Desv.	Media	Desv.	Media	Desv.	Sí	No
H	2341 (80,6%)	47,6	12,3	190,1	51,1	1,8	0,6	1803 (77%)	538 (23%)
M	563 (19,4%)	58,1	9,7	230	58,7	2,3	0,4	242 (43%)	321(57%)
Todos	2904	49,6	11,7	197,8	54,2	1,9	0,5	2045 (70,4%)	859(29,6%)

Tabla 11: pacientes de enfermedades cardiovasculares

Esto sirve en muchos casos para comparar cardinalidades (en este caso hay muchos más hombres que mujeres con este tipo de enfermedades), para comparar las distribuciones de otros atributos numéricos (por ejemplo, la obesidad parece ser más determinante en mujeres que en los hombres) o las frecuencias de valores en los atributos (el tabaquismo es mucho más alto en los hombres que en las mujeres), etc.

Este tipo de operaciones se realizan más eficientemente con técnicas OLAP y de consultas que permitan calcular medias, cuentas, etc., eficientemente, es de resaltar que este tipo de operaciones se pueden realizar con SQL básico.

Otra forma de realizar sumarización es determinar los “tipos” de individuos más frecuentes de cada clase.

Edad: 30-45, Colesterol > 200.Abarca el 13,9% de los hombres.
Edad: 46-60, Colesterol > 150, Fumador = no. Abarca el 6,6% de los hombres.
Resto: 7% de los hombres.
Edad: 46-60, Colesterol > 150, Fumador = Sí. Abarca el 50,2% de los hombres.
Edad: 60-inf. Abarca el 22,3% de los hombres.

Y lo mismo para la otra clase, para ver los típicos de cada clase.

16.3.2 Generalización y descripción

La generalización se puede realizar fundamentalmente mediante agregación por dimensiones (mediante herramientas OLAP u otros medios más tradicionales) o se puede realizar mediante lo que a veces se llama “inducción orientada al atributo” (*concept description*) e incluye: generalización multinivel, sumarización, caracterización y comparación. Es un tipo de técnicas de consulta explorativas que puedan considerarse como un paso previo a la MD.

Este tipo de generalización multinivel o descripción de conceptos se basa fundamentalmente en la definición de niveles conceptuales o jerarquías.

Este tipo de jerarquías de valor se pueden definir, además de las jerarquías de la dimensión. Esto permite que algunas operaciones se puedan automatizar, en especial para los atributos nominales:

- Si un atributo tiene muchos valores y se puede generalizar, considerar las posibles generalizaciones. Por ejemplo, si tenemos (muy numerosas lógicamente), pero podemos generalizar en “días festivos”, “días laborales”, o meses del año, etc., se debe considerar de estas generalizaciones cual se ajusta más a la tarea a realizar.
- Si un atributo tiene muchos valores y no se puede generalizar, borrar el atributo. Por ejemplo, los números de teléfono, si no tiene generalización, tal vez sea conveniente borrarlos.
- Si un atributo tiene poco valores, dejar el atributo intacto. Por ejemplo, el estado civil de un cliente puede no requerir generalización.
- Si después de las generalizaciones existen atributos con muchos valores que ya no se pueden generalizar, considerar borrar el atributo.
- Para los atributos numéricos se puede utilizar generalización por intervalos, en varias escalas significativas.

El problema de la técnica anterior es, lógicamente, definir las jerarquías, para las jerarquías de dimensiones y para los atributos de valor. La definición de estas jerarquías requiere, en muchos casos, obtener datos externos. Por ejemplo, buscar el calendario de festividades de los años de análisis (para saber si las fechas eran festivas o no), obtener datos del nivel de ventas del mismo ramo para saber si era año de recesión o año de bonanza, etc. Este proceso de creación de jerarquías ayudado por información externa se llama enriquecido (enrichment o enhancement).

16.3.3 Pivotamiento

Una operación muy usual a la hora de preparar la vista minable se conoce como pivotamiento y, forma parte de los operadores OLAP. La operación de pivotamiento cambia filas por columnas y, por tanto realiza un cambio verdaderamente radical para una representación basada en pares “atributo-valor”.

El ejemplo más clásico de pivotamiento es de la clase compra. Supongamos que unos grandes almacenes guardan una gran cantidad de cestas de la compra, donde cada atributo indica si el producto se ha comprado o no. Existen unos 10,000 productos en los grandes almacenes y millones de cestas semanales. El objetivo del análisis es ver qué productos se compran conjuntamente (regla de asociación).

Lógicamente los datos no caben en la memoria, con lo que hay que ir trabajando en disco. Para tener algo de fiabilidad en las reglas hay que mirar al menos la raíz cuadrada de todas las cestas, eso obliga a seleccionar unas 1000 filas (aleatoriamente) de la tabla para cada dos atributos que queramos evaluar.

Si este tipo de análisis se van a realizar frecuentemente, puede merecer la pena cambiar filas por columnas, como se muestra en la Ilustración 46

Ahora, para observar si dos productos están asociados es sólo necesario tomar dos filas de la tabla y realizar, por ejemplo, un “o exclusivo” (XOR) entre las dos filas, para ver si están asociadas o no.

#Cesta	Prod 1	Prod 2	Prod 3	…	Prod10,000
1	Si	No	No	…	No
2	No	No	No	…	Si
3	Si	Si	No	…	No
4	Si	No	No	…	No
5	No	Si	Si	…	Si
…	…	…	…	…	…
10,000,00	No	No	Si	…	Si

PIVOT →

#Prod	Cesta 1	Cesta2	Cesta3	…	Cesta10.000.000
1	Si	No	Si	…	No
2	No	No	Si	…	No
3	No	No	No	…	Si
4	No	Si	No	…	Si
5	Si	Si	No	…	No
…	…	…	…	…	…
10,000,00	No	Si	No	…	Si

Ilustración 46: Pivotamiento: cambio de filas por columnas

16.4 Selección de datos

La selección de datos es algo más que decidir que tablas (o archivos, en su caso) se van a necesitar para la Minería de Datos y de qué manera concatenarlas. Esto podría estar ya decidido, pero todavía no sabemos qué atributos/variables necesitamos y cuántas instancias (ejemplos) van a ser necesarias. Dicho de otra manera, puede que no todas las columnas, ni todas las filas sean necesarias. El problema existente es precisamente que si seleccionamos como “vista minable” todo aquello que pueda ser relevante podemos acabar con una vista minable de cientos de columnas/atributos y millones de filas/registros. El tamaño de una tabla como ésta desborda la capacidad de muchas de las técnicas de MD.

La selección de datos no tiene únicamente como objetivo la reducción del tamaño para obtener una MD más rápida sino que, en muchos casos, puede permitir mejorar el resultado (tanto en precisión o en costo, por ejemplo utilizando muestreo estratificado o en comprensibilidad, por ejemplo utilizando reducción de dimensionalidad).

El proceso de selección de datos muchas veces se engloba dentro de un concepto más amplio, denominado reducción de datos (*data reduction*), aunque este término también puede incluir agregación (por ejemplo si pasamos de instancias cada día a instancias agregadas mensualmente), la generalización (por ejemplo si reemplazamos el atributo ciudad por región, siguiendo por ejemplo la jerarquía de alguna dimensión), o incluso la compresión de datos (por ejemplo, eliminando datos redundantes).

En general, cuando tratamos de datos del estilo atributo-valor (es decir, una tabla), hay dos tipos de selección aplicables: selección horizontal (muestreo), donde se eliminan algunas filas (individuos) y selección vertical (reducción de dimensionalidad), donde se eliminan características de todos los individuos.
Veamos estos dos tipos de selección:

16.4.1 Técnicas de muestreo

La manera más directa de reducir el tamaño de una población o conjunto de individuos es realizar el muestreo. La gran mayoría de medidas y técnicas y de sus aplicaciones, se basan en el concepto de muestra.

En el caso de la Minería de Datos nos podemos plantear dos situaciones, dependiendo de la disponibilidad de la población:

- Se dispone de la población: en este caso se ha determinar qué cantidad de datos son necesarios y cómo hacer la muestra. En muchos casos por ejemplo, una muestra aleatoria no es lo más aconsejable por ejemplo en el caso de una encuesta, se intenta escoger al menos un mínimo de individuos de cada tipología (edades, sexos, nivel económico, rural/urbano, etc.). No obstante, esta es la situación ideal, ya que se dispone (con mayor o menor facilidad) de la población.

- Los datos son ya una muestra de realidad, por ejemplo, los datos recogidos en una base de datos y solo representan una parte de la realidad. Por ejemplo, las reclamaciones realizadas a través de la web quedan registradas, mientras el resto de reclamaciones no se registran.

Ya sea uno u otro caso existen otras razones para querer realizar un muestreo sobre los datos disponibles (ya sean, a su vez, un muestreo o el total de la población). Ente las razones están principalmente, la reducción del tamaño (con el objetivo de facilitar y agilizar los algoritmos de Minería de Datos), pero existen otras muchas (facilitar la visualización)

Dependiendo de estas razones existen varios tipos de muestreo: aleatorio, estratificado por grupos o exhaustivo.

- **Muestreo Aleatorio Simple**: Cualquier instancia tiene la misma probabilidad de ser extraída en la muestra. Dos versiones, con reemplazamiento y sin reemplazamiento.

- **Muestreo Aleatorio Estratificado**: El objetivo de este muestreo es obtener una muestra balanceada con suficientes elementos de todos los estratos, o grupos. Una versión simple es realizar un muestreo aleatorio simple sin reemplazamiento de cada estrato hasta obtener los *n* elementos de ese estrato. Si no hay suficientes elementos en un estrato podemos utilizar en estos casos muestreo aleatorio simple con reemplazamiento (sobre muestreo).

- **Muestreo de Grupos:** El muestreo de grupos consiste en elegir sólo elementos de unos grupos. El objetivo de este muestreo es generalmente descartar ciertos grupos que, por diversas razones, pueden impedir la obtención de buenos modelos.

- **Muestreo Exhaustivo:** Para los atributos numéricos (normalizados) se genera al azar un valor en el intervalo posible; para los atributos nominales se genera al azar un valor

Entre los posibles. Con esto obtenemos una instancia ficticia y buscamos la instancia real más similar a la ficticia. Se repite este proceso hasta tener n instancias. El objetivo de este método es cubrir completamente el espacio de instancias.

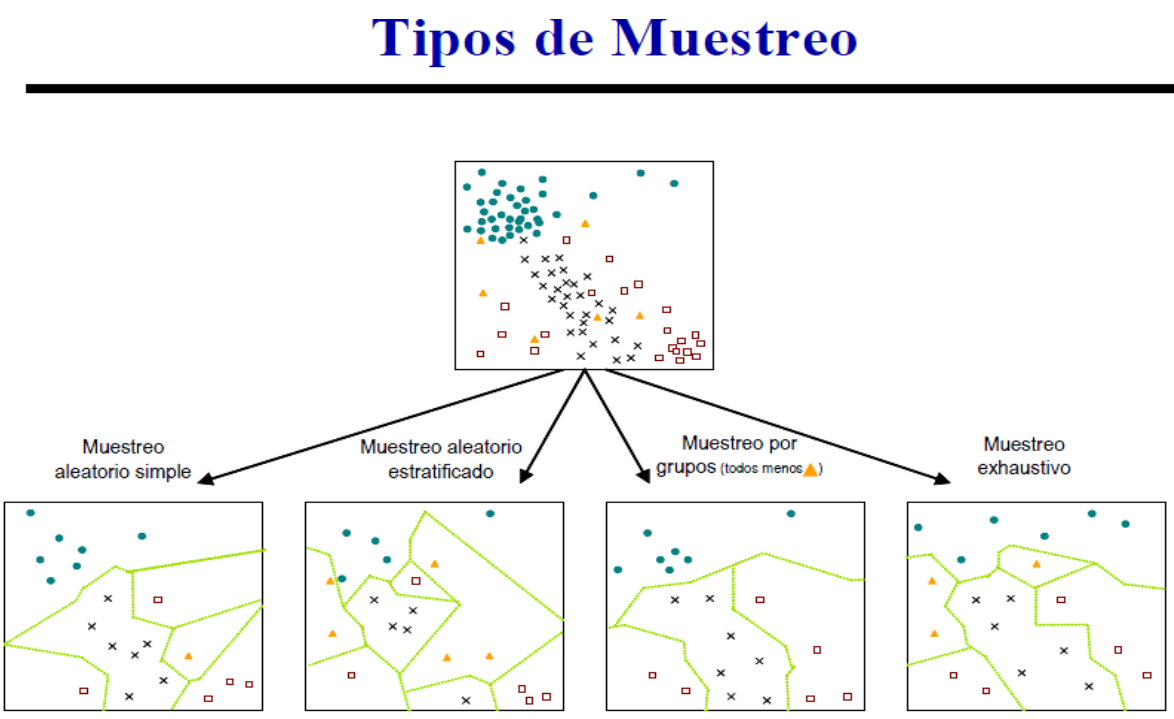


Ilustración 47: Tipos de Muestreo

A veces se pueden realizar varios muestreos en cascada, el primero, por ejemplo aleatorio simple, para analizar mejor los datos y determinar grupos y, el segundo, estratificado, dependiendo del resultado anterior.

16.4.2 Selección de características relevantes. Reducción de dimensionalidad

La selección de características, variables o atributos (*feature selection*, en inglés) tiene 4 objetivos principales:

1- Permite reducir el tamaño de los datos, al eliminar características o atributos de todos los registros que puedan ser irrelevantes o redundantes

2- Una buena selección de las características puede mejorar la calidad del modelo, al permitir al método de MD centrarse en las características relevantes.
3- Una buena selección de características permite expresar el modelo resultante en función de menos variables, esto es especialmente importante cuando se desean modelos comprensibles (árboles de decisión, regresión lineal, etc.).

4- Se puede requerir una reducción de dimensionalidad a dos o tres características exclusivamente, con el propósito de representar los datos visualmente. Existen otras razones para eliminar atributos, por ejemplo, cuando existen muchos datos erróneos o faltantes en un atributo, y es preferible deshacerse de él. También es necesario para realizar análisis de componentes principales (PCA, visto en el tema anterior), porque si no eliminamos atributos identificadores estos pesarán más que ningún otro.

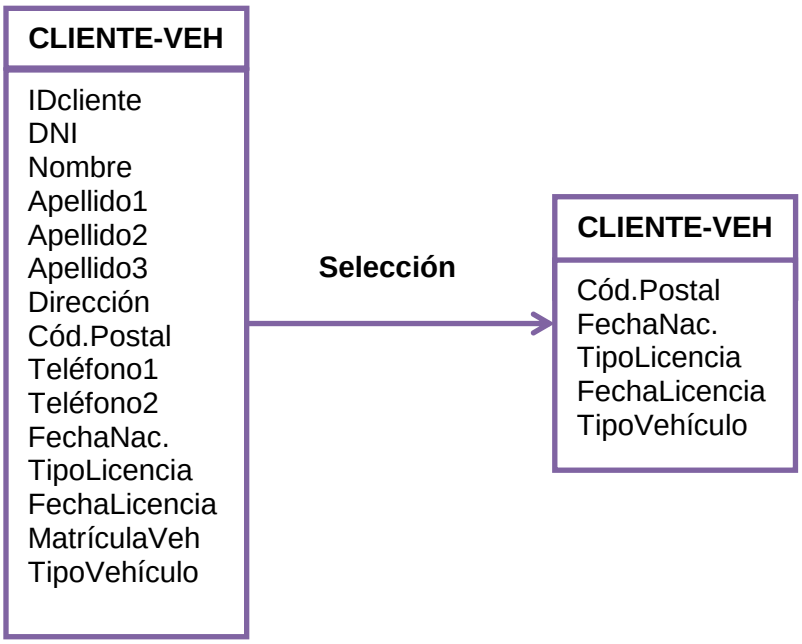


Ilustración 48: Selección de características (atributos)

En principio, si disponemos de un problema con muchos atributos siempre será mejor que cuando tenemos pocos atributos. Por ejemplo, cuanta más información tengamos de cada cliente puede parecer que podemos hacer mejores decisiones sobre las ofertas o productos que se le puedan dirigir. El problema es que la gran mayoría de métodos de MD pueden perderse entre tantas características en un espacio que al tener alta dimensionalidad resulta estar más desierto (especialmente si las hay irrelevantes, redundantes o con valores erróneos) y obtener modelos que se ajustan a particularidades de los datos de entrenamiento y no de los datos en general. Esto ocurre especialmente cuando existen muchas dimensiones pero no tenemos un número suficiente de ejemplos para *reducir* los "*grados de libertad*"

Algunos atributos son fáciles de eliminar, por ejemplo si un atributo es constante, es decir, tiene el mismo valor para todas las instancias es claramente eliminable.

Existen dos reglas generales para eliminar características, en especial los atributos nominales, que resultarán familiares a aquellos que conozcan la tecnología y terminología de BD:

- **Eliminación de claves candidatas:** La regla general es eliminar cualquier atributo que pueda ser clave primaria de la tabla (o que sea clave candidata o incluso parte de clave candidata, parcial o totalmente).Por ejemplo, hay que eliminar números de documentos de identificación, códigos internos, nombres y apellidos, direcciones.
- **Eliminación de atributos dependientes:** En la teoría de la normalización de bases de datos, cuando existen dependencias funcionales entre atributos se intenta normalizar en varias tablas...Por ejemplo, cuando se tiene el código postal, la ciudad, la región, con la región tenemos el país, con lo que podemos establecer una serie de dependencias funcionales que se deben normalizar en cuantas tablas sean necesarias.

Aunque determinar los atributos anteriores (claves candidatas y atributos dependientes) no es sencillo en muchos casos, es mucho más fácil que determinar otras características, que tampoco son relevantes o redundantes. En general una vez detectados los atributos irrelevantes y redundantes, debemos utilizar técnicas más sofisticadas si queremos seguir reduciendo la dimensionalidad, en particular para los atributos numéricos. Existen dos tipos generales de métodos de selección de características (atributos):

- **Métodos de filtro o métodos previos:** Se filtran los atributos irrelevantes antes de cualquier proceso de Minería de Datos y, en cierto modo, son fundamentalmente estadísticas (medidas de información, distancia, dependencia o inconsistencia). El criterio para establecer el subconjunto de características “Optimo” se basa en medidas de calidad previa que se calcula a partir de los mínimos datos.

- **Métodos basados en modelo o métodos envolventes (wrapper):** La bondad de la selección de atributos se evalúa respecto a la calidad de un método de Minería de Datos o estadístico extraído a partir de los datos (utilizando, lógicamente, algún método de validación). Lógicamente, este tipo de técnicas requieren mucho más tiempo que las otras, ya que para evaluar hay que haber entrenado un modelo. Nótese que el método de Minería de Datos utilizado para hacer la selección de atributos no tiene por qué ser el método de Minería de Datos que se utilizará finalmente. Por ejemplo se puede utilizar una red neuronal para determinar los atributos más significativos y, una vez determinados, utilizar un árbol de decisión para obtener un modelo comprensible utilizando sólo las características seleccionadas para generar las reglas del modelo.

Sea de filtro o basados en modelo, ambos modelos pueden ser iterativos, es decir, se van eliminado atributos y se va viendo el resultado. Se va recuperando o eliminando atributos de una manera iterativa, hasta que se obtiene una combinación que maximiza la calidad.

Existen muchas maneras de realizar este procedimiento, por ejemplo, empezando con un atributo y elegir el que de mayor calidad de selección con atributos, después añadir el atributo que de mayor calidad de selección con dos atributos, y así hasta que no se mejore la calidad o se llegue al número deseado de atributos (estrategia forward). La manera inversa (comenzar con todos e ir eliminado) o maneras mixtas se pueden utilizar. El objetivo de los dos casos es el mismo, obtener un subconjunto de atributos representativo (que contenga la

mayor parte de la información que exista en el conjunto inicial) y que no exista la menor correlación entre los atributos seleccionados.

16.5 Lenguajes, Primitivas e interfaces de Minería de Datos:

Para realizar una serie de exploraciones, transformaciones, vínculos, modificaciones selecciones, etc., con el objetivo de obtener una vista minable, acompañada de una tarea a realizar y un método para llevarla a cabo, una evaluación de calidad, un conocimiento previo y una manera de presenta los patrones obtenidos, todo ello es necesario para realizar el proceso de Minería de Datos. Pero ¿cómo especificamos este conjunto de elementos? Para ello, podemos utilizar 3 tipos de medios para especificar los componentes: lenguajes de consulta, conjunto de primitivas o interfaces integrados

- **Lenguajes de consulta:** Llamados también lenguajes de consulta inductivos o de MD permite enfocar el proceso de MD de una manera similar al proceso de consulta de BD.
- **Conjunto de primitivas o interfaces middleware:** En vez de proporcionar un lenguaje, proporciona una serie de primitivas que, junto a un lenguaje de programación (C++, Java, Python, etc.), permiten especificar los componentes o realizar todos los pasos previos a la MD.
- **Interfaces o entornos integrados visuales:** Basados en la idea de flujo de datos/información/conocimiento, presentan una serie de nodos que tienen una serie de entradas y salidas, lo que permite interconectarlos. Esto permite ver todo el proceso como un flujo que se origina en la información y que termina en los patrones o en su evaluación. Las herramientas gráficas, de selección , de transformación, de modelo, de evaluación, etc., son “nodos del sistema”

16.5.1 Lenguajes de consulta de MD

: Cuando hablamos de vista minable nos puede parecer que podríamos utilizar un lenguaje de consulta como el SQL y alguna herramienta “reports” para realizar estas Vistas Minables. Presentadas las transformaciones y exploración requerida, podríamos considerar utilizar herramientas OLAP, que permiten transformar y agregar los datos de una manera más flexible, cómoda y sobretodo, eficiente.

16.5.2 Conjunto de primitivas de MD

:

Los lenguajes de consulta de MD pueden utilizarse interactivamente o pueden utilizarse dentro de algún lenguaje de programación. Este es el caso de, por ejemplo, el “OLE DB

for Data Mining" o es el SQL/MM. Si lo que deseamos es realizar MD a través de una aplicación puede ser preferible disponer de un conjunto de primitivas que se pueden utilizar a forma de API, en nuestros programas. Esta suele ser la elección más aconsejable y eficiente para programadores. Otras veces necesitan interfaces entre aplicaciones y servidores, constituyendo estándares, en cierto modo de lo que se ha venido llamando *middle weare*

16.5.3 Interfaces visuales de MD:

Uno de los aspectos que ha hecho popularizar la MD en la última década es la aparición de unas interfaces visuales que faciliten en gran medida la realización de todo el proceso de extracción de conocimiento. Recordemos que parte de los usuarios potenciales de la MD son informáticos ni profesionales de las tecnologías, sino que pueden ser directivos o analistas. En vez de tener que aprender lenguajes al estilo del SQL que, probablemente les costó aprender, o, todavía peor, conjuntos primitivas, existen actualmente herramientas visuales que periten realizar el proceso de extracción de una manera visual, indicando con el ratón qué transformaciones, procesos y procedimientos aplicar, de una manera relativamente sencilla.

Los ejemplos más paradigmáticos de esta manera de trabajar quizá sean las interfaces de SPSS Clementine y del SAS Enterprise Miner, ambas muy similares. Por ejemplo en la Ilustración siguiente se muestra un ejemplo de un proceso de extracción de conocimiento con SPSS Clementine. Como se puede ver en la parte superior izquierda, la información parte de un nodo denominado "titanic.dat" y se le van aplicando nodos, transformándose, analizando, seleccionando, partiendo, visualizando en distintas ramas, que se pueden seguir por las flechas que conectan los nodos. Esto hace ver el proceso de MD como un flujo de trabajo ("workflow"), donde cada nodo transforma información en otra información.

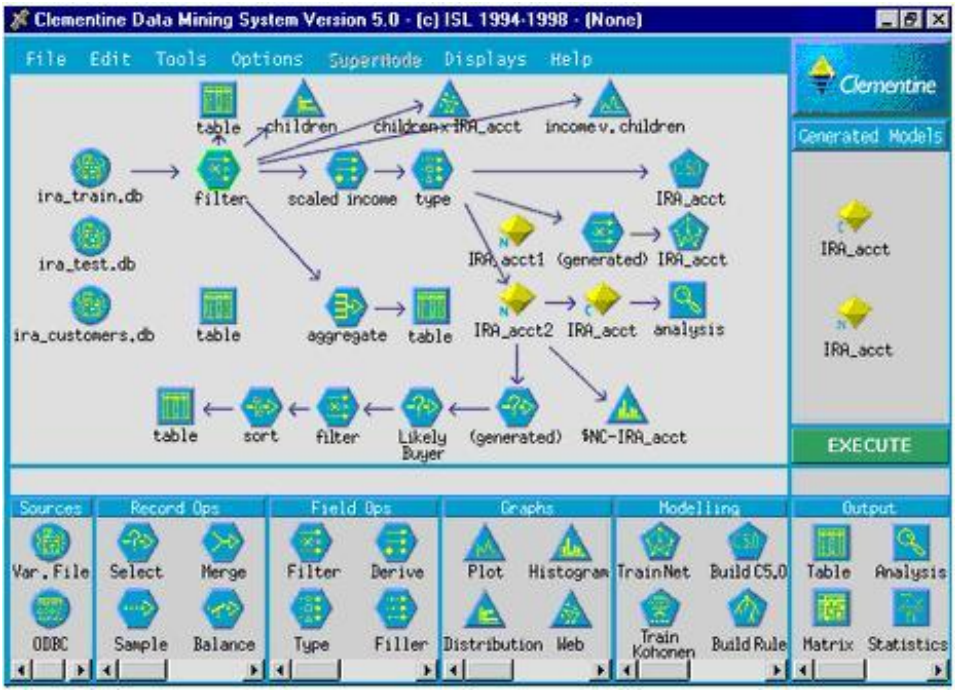


Ilustración 49: Ejemplo de interfaz visual del paquete de Minería de Datos SPSS Clementine

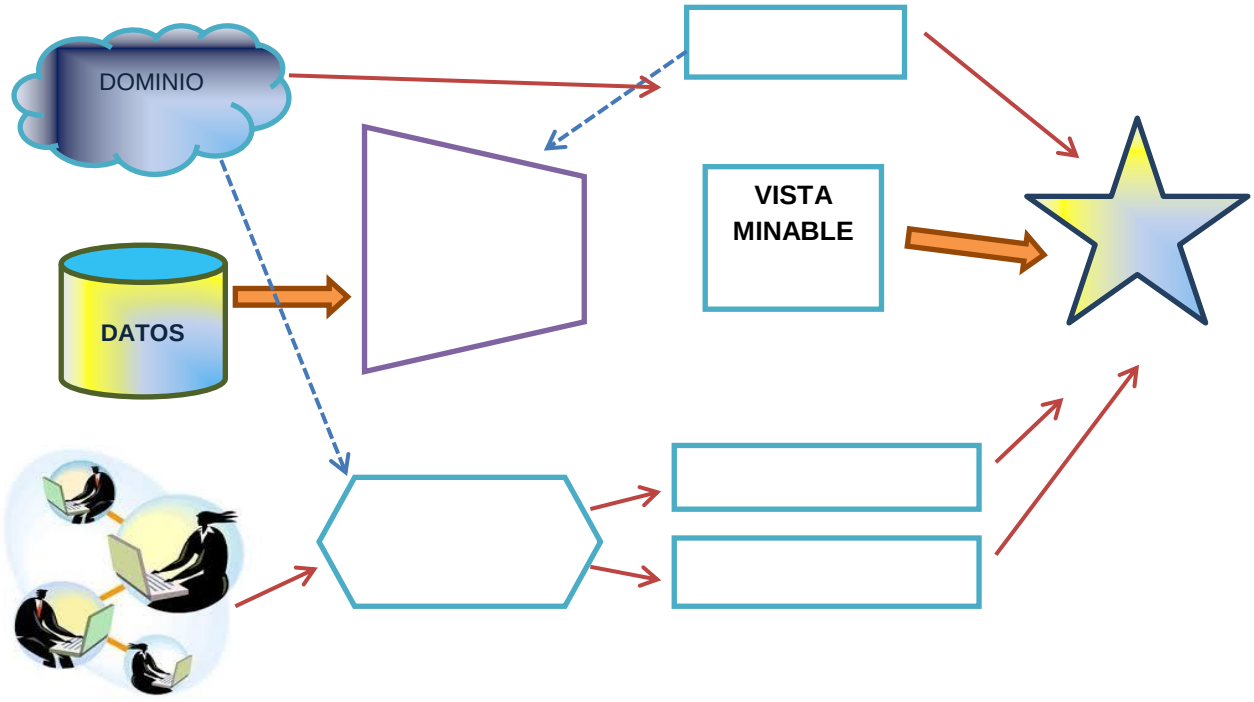
Existen otro tipo de funciones que se pueden hacer de una manera visual, incluidas en otros sistemas, como la definición de jerarquías y el establecimiento de conocimiento previo, el diseño de experimentos y de validación, operadores OLAP, etc. A medida avanzan las versiones de los sistemas, van incorporando cada vez más herramientas (traducidas en nodos u opciones de los mismos)

Las interfaces visuales tienen sus desventajas: la primera es que el usuario se acostumbra a utilizar una herramienta y acaba dependiendo de ella. Además, se dificulta en gran medida el poder portar a otras herramientas el flujo o trabajo de MD realizado. En segundo lugar, estos entornos visuales están diseñados para que las operaciones y el flujo se vayan construyendo a mano.

GUÍA DE APOYO PARA EL ESTUDIANTE

Tema 5: exploración y selección de los datos

I. Completa el esquema sobre el proceso que lleva de los datos, del conocimiento del dominio y de los usuarios que son necesarios para llevar a cabo la fase propia de MD.



II. Completa el espacio en blanco de las siguientes afirmaciones con la respuesta correcta:

a) Para conocer qué se puede hacer con unos ciertos datos es necesario conocer el y _____y_____

b) Una aplicación muy práctica de la sumarización es la _____o _____ _____

c) La manera más directa de reducir el tamaño de una población o conjunto de individuos es realizar el _____

d) _____Cualquier instancia tiene la misma probabilidad de ser extraída en la muestra.

e) _____ El objetivo de este muestreo es obtener una muestra balanceada con suficientes elementos de todos los estratos, o grupos.

f)_____consiste en elegir sólo elementos de unos grupos.

g) _____El objetivo de este método es cubrir completamente el espacio de instancias.

III. Responda correctamente

a) Las técnicas de visualización de datos se utilizan fundamentalmente con dos objetivos:

_____

_____

_____

IV. Conteste verdadero o falso a las siguientes afirmaciones:

a) La generalización se puede realizar mediante herramientas OLAP u otros medios más tradicionales _____

b) La operación de pivotamiento cambia filas por columnas y, por tanto realiza un cambio verdaderamente radical para una representación basada en impares "atributo-valor" _____

c) La selección de datos tiene únicamente como objetivo la reducción del tamaño para obtener una MD más rápida sino que, en muchos casos, puede permitir mejorar el resultado____

SOLUCIÓN DE GUÍA DE APOYO PARA EL ESTUDIANTE

Tema 5: Exploración y selección de datos

I. Completa el esquema sobre el proceso que lleva de los datos, del conocimiento del dominio y de los usuarios que son necesarios para llevar a cabo la fase propia de MD.

Véase ilustración 42 de este documento

II. Completa el espacio en blanco de las siguientes afirmaciones con la respuesta correcta:

a) Para conocer qué se puede hacer con unos ciertos datos es necesario conocer el **dominio** y **los usuarios**.
b) Una aplicación muy práctica de la sumarización es la **comparación** o **discriminación de clases**
c) La manera más directa de reducir el tamaño de una población o conjunto de individuos es realizar el **muestreo.**
d) **Muestreo Aleatorio Simple** Cualquier instancia tiene la misma probabilidad de ser extraída en la muestra.
e) **Muestreo Aleatorio Estratificado** El objetivo de este muestreo es obtener una muestra balanceada con suficientes elementos de todos los estratos, o grupos.
f) **Muestreo de Grupos** consiste en elegir sólo elementos de unos grupos.
g) **Muestreo Exhaustivo** El objetivo de este método es cubrir completamente el espacio de instancias.

III. Responda correctamente

a) Las técnicas de visualización de datos se utilizan fundamentalmente con dos objetivos:
Aprovechar la gran capacidad humana de ver patrones descubiertos y tendencias a partir de imágenes y facilitar la compresión de los datos. Y Ayudar al usuario a comprender más rápidamente patrones descubiertos automáticamente por un sistema de KDD.

IV. Conteste verdadero o falso a las siguientes afirmaciones:

a) La generalización se puede realizar mediante herramientas OLAP u otros medios más tradicionales **V**
b) La operación de pivotamiento cambia filas por columnas y, por tanto realiza un cambio verdaderamente radical para una representación basada en pares “atributo-valor” **F**
c) La selección de datos no tiene únicamente como objetivo la reducción del tamaño para obtener una MD más rápida sino que, en muchos casos, puede permitir mejorar el resultado **F**

17) TEMA 6: TÉCNICAS DE MINERA DE DATOS.

Objetivos:

- Obtener conocimientos sobre las técnicas de la Minería de Datos
- Obtener conocimiento acerca de la tarea y los métodos

Contenidos:

- Introducción.
- Tareas y métodos.
- Minería de Datos y aprendizaje inductivo.
- El lenguaje de los patrones. Expresividad
- Breve comparación de métodos.

Duración: 6 hrs

Bibliografía:

- Introducción a Minería de Datos. José Hernández Orallo, Mª José Ramírez Quintana, Cesar Ferri Ramírez.

17.1 Introducción

La extracción de conocimiento a partir de datos tiene como objetivos descubrir patrones que entre otras cosas, deben ser:

- Válidos.
- Novedosos.
- Interesantes.
- Entendibles.

Los seres humanos tenemos la capacidad innata de ver patrones a nuestro alrededor, incluso donde no lo hay (borreguitos en las nubes, cuadrigas en las estrellas, pautas en las ruletas, mariposas en manchas de tinta, etc.). A veces nos parece tan sencillo que no caemos en cuenta de la complejidad intrínseca que existe en procesos tan rutinarios como estimar a qué hora se pone el despertador para llegar temprano al trabajo.

Las técnicas de Minería de Datos (en especial las heredadas del aprendizaje automático y del reconocimiento de formas) han querido emular, en un primer momento, e ir más allá, más tarde y en cierto aspecto, a estas capacidades de aprendizaje.

Se dice que el proceso de Minería de Datos convierte datos en conocimientos, tal cual un alquimista pudiera convertir espigas de trigo en lingotes de oro. Por si esto no fuera poco, en algunos casos se llega a decir que el objetivo es extraer "verdad a partir de basura".

Veamos la siguiente ilustración

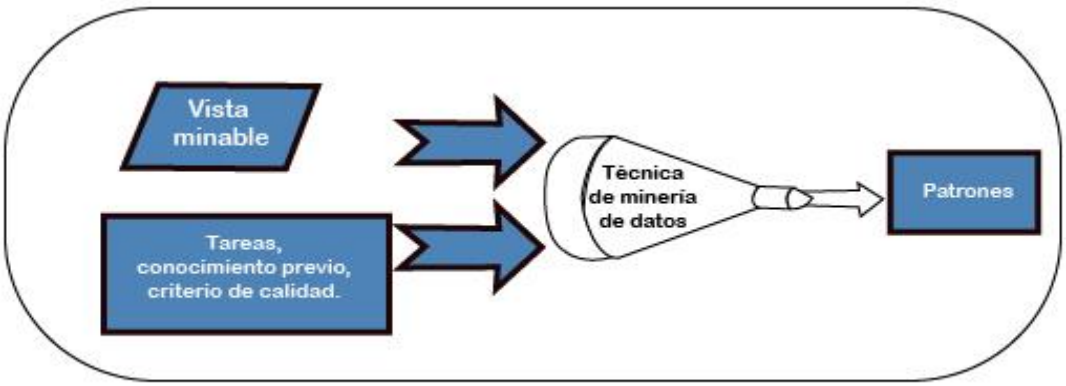


Ilustración 50: Técnicas de Minería

En la Ilustración las técnicas de Minería de Datos aparecen como un colador pasapurés que, al introducirle los datos (en forma de vista minable junto a ciertos elementos asociados) produce, sin grumos y atasco alguno, una serie de patrones lustrosos y relucientes.

Existen tareas, presentaciones de tareas, instancias de tareas, que son más sencillas que otras. Por ejemplo la extracción de reglas de asociación es un problema más sencillo, computacionalmente hablando, que la clasificación.

Además de los datos y de la tarea, existen otros aspectos que influyen en el aprendizaje, que suelen denominarse conjuntamente **bias**. Quizás las **bias** que más influye en esta complejidad sea la manera de expresar o definir los patrones (**bias del lenguaje**)

Por ejemplo:

No es lo mismo una regresión lineal que una regresión realizada por una red neuronal multicapa. Ambos métodos permiten realizar la misma tarea, pero la expresividad y la mayor capacidad de la segunda se paga, de alguna manera, con un mayor tiempo de espera para obtener el modelo.

El conocimiento previo es otro tipo de **bias**, que puede ayudar a refinar el espacio de búsqueda (**bias de búsqueda**)

17.2 Tareas y métodos

Una de las cosas que debemos clasificar antes de continuar es diferenciar una tarea de un método, así como destacar las tareas y métodos más relevantes. Una (un tipo de) tarea de Minería de Datos en un (tipo de) problema de Minería de Datos. Es muy importante distinguir el problema de los métodos para solucionarlo.

17.2.1 Tarea

Los tipos de tareas más importantes:

- Clasificación.
- Regresión.
- Agrupamiento.
- Reglas de asociación. Etc.

Para definir las tareas debemos definir el conjunto de ejemplo con los que se van a tratar. Definimos **E** como el conjunto de todos los posibles elementos de entrada. Las instancias posibles dentro de **E** generalmente se representan como un conjunto de valores para una serie de atributos (sean nominales o numéricos) es decir $\mathbf{E}=A_1 * A_2 * \ldots * A_N$ y un ejemplo **e** es una tupla$<a_1, a_2, .., a_n>$ tal que $a_i \in A_i$

Veamos a continuación las tareas más importantes en Minería de Datos:

- **Predictivas:** Se trata de problemas y tareas en los que hay que predecir uno o más valores para uno o más ejemplos. Los ejemplos en la evidencia van acompañados de una salida (clase, categoría o valor numérico) o un orden entre ellos. Dependiendo de cómo sea la correspondencia entre los ejemplos y los valores de salida y la presentación de los ejemplos podemos definir varias tareas predictivas:

- **Clasificación:** Los ejemplos se presentan como un ejemplo de pares de elementos de dos conjuntos, ∞= {<e, s>: e €E, s€S} donde S es el conjunto de valor de salida. Los ejemplos e, al ir acompañados de un valor de S, se denomina comúnmente ejemplos etiquetados <e, s> y, en consecuencia ∞ se denomina conjunto de datos etiquetados. El objetivo es aprender una función A: E -> S, denominada clasificador, que represente la correspondencia existente en los ejemplos, es decir, para cada valor de E tenemos un único valor para S.
- **Clasificación Suave:** La presentación del problema es la misma que la clasificación, pares de elementos de dos conjuntos, ∞= {<e, s>: e €E, s€S}. además de la función A: E -> S se aprenderá otra función e: E->Æ que significa el grado de certeza de la predicción hecha por la función A lógicamente, es preferible tener un clasificador suave que acompañe a las predicciones de una medida de certeza o de fiabilidad de dichas predicciones.
- **Estimación de probabilidad de clasificación:** Se trata de una generalización de la clasificación suave. La presentación del problema es la misma que la de la clasificación normal y suave, pares de elementos de dos conjuntos, ∞= {<e, s>: e €E, s€S}. la función a aprender, sin embargo es distinta de la clasificación y de la función suave. Se trata exclusivamente m funciones e_1: E->$Æ_1$, donde m es el número de clases.
- **Categorización:** No se trata de aprender una función, sino una correspondencia. Es decir, cada ejemplo de ∞= {<e, s>: e €E, s€S}; así como la correspondencia a aprender A: E -> S, se puede asignar varias categorías asociadas. La categorización también se puede presentar también en forma de categorización suave o en forma de un estimador de probabilidades.
- **Preferencia o priorización:** El aprendizaje de preferencia consiste en determinar a partir de dos o más ejemplos, un orden de preferencia. Cada ejemplo es en realidad una secuencia <e_1…e_k>, e € E, k≥2, donde el orden de la secuencia representa la predicción. Un conjunto de datos para este problema es, por tanto, un conjunto de secuencia ∞= {<e, s>: e €E, s€S}. otra manera alternativa de presentar los datos es mediante un orden parcial, que se puede considerar un caso particular de la anterior, donde la secuencia solo tiene dos elementos (k ≥2).
- **Regresión:** Es quizás la tarea más sencilla de definir. El conjunto de evidencia son correspondencias entre dos conjuntos ∞: E -> S, donde S es el conjunto de valores de salida. Al igual que en la clasificación, el objetivo es aprender una función A: E -> S que represente la correspondencia existente en los ejemplos, es decir, para cada valor de E tenemos un único valor para S.

- **Descriptivas:** se presentan como un conjunto ∞ = {e: e€E}, sin etiquetar ni ordenar de manera de ninguna manera. El objetivo, por tanto, no es predecir nuevos datos sino describir los existentes. Esto se puede hacer de muchas maneras y la variedad de tareas se dispara. Algunas técnicas ya vistas se pueden considerar tareas descriptivas. Veamos las tareas descriptivas más delimitadas:
    - **Agrupamiento (clustering):** el objetivo de esta tarea es obtener grupos o conjuntos entre los elementos de ∞, de tal manera que los elementos asignados

al mismo grupo sean similares. Lo importante del agrupamiento respecto a la clasificación es que son precisamente los grupos y la pertenencia a los grupos. En algunos casos se puede proporcionar el número de grupos que se desea obtener, otras veces este número se determina por el algoritmo de agrupamiento, según las características de los datos. La función a obtener es idéntica a la de la **clasificación** A: E -> S, con la diferencia de que los valores de S y sus miembros se **crean** o **inventan** durante el proceso de **aprendizaje**. Averiguar que las instancias se agrupan en varios segmentos es útil porque podemos determinar el comportamiento de una nueva instancia viendo a que grupo de instancia pertenece. El **agrupamiento** se conoce muy frecuentemente por su término en inglés: **clustering**, o por traducciones alternativas como: segmentación, aglomeración, racionamiento algunos autores consideran el agrupamiento con este objetivo como una tarea nueva, llamada **sumarización** de modo similar si buscamos subgrupos que se separen del resto de la población, tenemos lo que a veces se conoce como "descubrimiento del subgrupo" (subgroup discovery), que también se puede considerar una tarea en sí misma.

- **Correlaciones y factorizaciones:** Los estudios correlaciones y factoriales se centran exclusivamente en los atributos numéricos (ya sean inicialmente numéricos o después de una numerización).
- **Reglas de asociación:** esta es una de las tareas reinas de la Minería de Datos y que ha evolucionado conjuntamente, desde mediados de los 90, con la propia Minería de Datos. El objetivo es similar a los estudios correlacionales y factoriales, pero para los atributos nominales, muy frecuente en las bases de datos. Este tipo de estudio reciben, además del nombre de análisis de asociaciones, el nombre de análisis de vínculos (**link analysis**), este término también se utiliza en el agrupamiento jerárquico. Dado los ejemplos del conjunto **E**=A_1 * A_2*…*A_N una regla de asociación se define generalmente de la siguiente forma: "si A_i=a^A_j=b^…^A_k=h entonces A_t= u^A_s=v ^…^ A_z=w", donde todos los atributos son nominales y las igualdades se definen utilizando algún valor de las posibles para cada atributo. La regla anterior está orientada es decir es una regla de asociación direccional.
- **Dependencias funcionales:** el descubrimiento de dependencias funcionales generalmente se suelen incluir dentro de la variedad de tareas con el nombre de "**reglas de asociación**". Una dependencia funcional podría ser "dada la edad, el nivel de ingresos, el código postal, si está casado o no". Las dependencias funcionales en particular cuando hay un atributo a cada lado, pueden ser orientadas y no orientadas, al igual que las reglas de asociación.
- **Detección de valores e instancias anómalas:** la detección de valores anómalos o atípicos (outlier detection), con el objetivo de realizar limpieza de datos. La detección de valores anómalos pueden ser muy útil para detectar precisamente comportamientos anómalos pueden seguir fraudes, fallos, intrusos o compartimientos diferenciados. La definición de instancias anómalas es más general en el sentido que no solo considera un único atributo, sino que

los considera todos. La tarea se define con el objetivo de encontrar aquellas instancias que no son similares a ninguna (o muy pocas) de las otras instancias. La manera de abordar el problema es generalmente la de agrupar los ejemplos y ver aquellas instancias que se quedan "desplazadas" de los grupos mayoritarios. Para ellos son especialmente útiles los agrupadores suaves o los estimadores de probabilidad de agrupamiento con todos los grupos se puede considerar un caso "aislado" y por lo tanto anómalo, también se utiliza otros medios no necesariamente basados en la tarea de agrupamiento, como la medición de distancias.

Algunas tareas están relacionadas, esto en cierto modo, ha hecho que la terminología de algunas de ellas sea bastante diversa y a veces, sea difícil aclararse con los nombres que utilizan algunas herramientas o textos de referencia.

Se suele utilizar el término "aprendizaje supervisado" para los métodos predictivos y el término "aprendizaje no supervisado" para los métodos no predictivos, los términos originales eran más precisos, el aprendizaje supervisado generalmente se utiliza para nombrar, de entre todos los métodos predictivos la clasificación y rara vez la regresión, en cambio , el agrupamiento se considera el problema no supervisado por excelencia mientras que el resto de métodos descriptivos no lo son en realidad.

Un caso especial de relación entre tareas ocurre cuando tenemos un problema clasificación de más de dos datos (multiclasificación) y queremos resolverlo mediante clasificadores que sólo consigan discriminar entre dos clases (clasificación binaria) ya sea porque el algoritmo que vamos a utilizar solo permite clasificación binaria.

El proceso de adaptar un problema de más de dos clases a un problema de dos clases se denomina **binarización** existen varios métodos para realizarla:

- **Uno frente al resto (one-us-all):** Se construye un clasificador binario utilizando todos los ejemplos de una clase y agrupando en una misma clase el resto de ejemplos. Esto se realiza para todas las clases. Si tenemos **n** clases, el resultado serán **n** clasificadora binaria que posteriormente habremos de combinar. Esta es la misma aproximación que se utiliza para convertir problemas de categorización, en problema de clasificación, con la diferencia de que, en categorización, luego no hay que "**fusionar**" los resultados, puesto que podemos quedarnos con varias categorías y no sólo con una sola, como en el caso de la clasificación.
- **Todos los pares (all-pairs):** Se construye un clasificador binario utilizando todos los ejemplos de dos clases e ignorando el resto. Esto se realiza para todos los pares de clases. Si tenemos **n** clases, el resultado será **n (n-1)/2** clasificadores binarios que posteriormente habremos de combinar.
- **Todas las mitades (all-halves):** Se construye un clasificador binario utilizando los ejemplos de la mitad de las clases por un lado y el resto por el otro. Si tenemos **n** clase y este número es par, tendremos **n/2** clases por un lado y **n/2** clases por el otro. Esto se realiza para todas las particiones posibles del mismo número de clases,

también se puede realizar para todas las posibles particiones, tengan o no el mismo número de clases, lo que dispara el número de variantes. La combinación en este caso es más costosa y más complicada porque la clasificación final se hace como una ponderación, ya que ninguno de los clasificadores binarios representa a clases simples.

Para la clasificación de los clasificadores binarios en el clasificador final se pueden utilizar muchas técnicas, que se basan mayoritariamente en utilizar la confianza de la predicción de cada clasificador parcial, como por ejemplo la técnica **ECOC** (**E**rror **C**orrecting **O**utput **C**ode). Por tanto para convertir un problema de multi-clasificación en clasificación binaria, los clasificadores binarios han de ser suaves.

Otra distinción importante a realizar es entre modelos (o patrones) totales y parciales. El concepto es el mismo que para las funciones; una **función total** define imagen para todos los valores del dominio, mientras que una **función parcial** sólo define imagen para algunos de los valores del dominio, siendo indefinidos para el resto.

17.2.2 Métodos: correspondencia entre tareas y métodos.

Cada una de las tareas, como cualquier problema, requiere de métodos, técnicas o algoritmos para resolverlas. Los tipos de técnicas existentes para llevar a cabo las tareas anteriormente señaladas son:

- **Técnicas algebraicas y estadísticas:** Se basan generalmente, en expresar modelos y patrones mediante fórmulas algebraicas, funciones lineales, funciones no lineales, distribuciones o valores agregados estadísticos tales como medias, varianzas, correlaciones, etc. Frecuentemente estas técnicas cuando obtienen un patrón, lo hacen a partir de un modelo ya predeterminado del cual, se estima unos coeficientes o parámetros, de ahí el nombre de técnicas paramétricas. Algunos de los algoritmos más conocidos dentro de este grupo de técnicas son la regresión lineal (global o local), regresión logarítmica y regresión lógica. Los discriminantes lineales y no lineales, basados en funciones predefinidas, es decir discriminantes paramétricos, entran dentro de esta categoría. Aunque el termino no paramétrico, se utiliza para englobar gran parte de técnicas provenientes del aprendizaje automático, como son, las redes neuronales, también existen muchas técnicas de modelización estadísticas no paramétrica.
- **Técnicas bayesianas:** Se basan, en estimar la probabilidad de pertenencia (a una clase o grupo), mediante la estimación de las probabilidades condicionales inversas o a priori, utilizando para ello el teorema de bayes, algunos algoritmo muy populares son el clasificador bayesiano naive, los métodos basados en máxima verisimilitud y el algoritmo EM. Las redes bayesianas generalizan las topologías de las interacciones probabilísticas entre variables y permiten representar gráficamente dichas interacciones.
- **Técnicas basadas en conteo de frecuencias y tablas de contingencia:** Esta técnica se basan en contar la frecuencia en la que dos o más sucesos se presentan

conjuntamente. Cuando el conjunto de sucesos posibles es muy grande, existen algoritmos que van comenzando por pares de sucesos e incrementando los conjuntos, sólo en aquellos casos que las frecuencias conjuntas superen un cierto umbral.

- **Técnicas basadas en árboles de decisión y sistemas de aprendizaje de reglas:** Son técnicas que, además de su representación en forma de reglas, se basan en dos tipos de algoritmos: los algoritmos denominado "divide y vencerás" y los algoritmos denominados "separa y vencerás".
- **Técnicas relacionales, declarativas y estructurales:** La característica principal de este conjunto de técnicas es que representan los modelos mediante lenguajes declarativos, como:
    - Los lenguajes lógicos
    - Los lenguajes funcional
    - Los lenguajes lógicos-funcionales.

  Las técnicas de **ILP** (**Programación Lógica Inductiva**) son las más representativas y las que han dado nombre a un conjunto de técnicas denominadas **Minería de Datos relacional**.

- **Técnicas basadas en redes neuronales artificiales:** Se trata de técnicas que aprenden un modelo mediante el entrenamiento de los pesos que conectan un conjunto de nodos o neuronas. La topología de la red y los pesos de las conexiones determinan el patrón aprendido. Existen innumerables variantes de organización :
    - Perceptrón simple
    - Redes multicapa
    - Redes de base radia
    - Redes de kohonen

  Con no menos algoritmos diferentes para cada organización: el más conocido es el de retro-propagación (back propagation).

- **Técnicas basadas en núcleo y máquinas de soporte vectorial:** Se trata de técnicas que intentan maximizar el margen entre los grupos o las clases formadas. Para ello se basan en unas transformaciones que pueden aumentar la dimensionalidad. Estas transformaciones se llaman núcleos (kernels). Existen muchísimas variantes, dependiendo del núcleo utilizado y de la manera de trabajar con el margen.
- **Técnicas estocásticas y difusas:** Bajo este paragua se incluyen la mayoría de las técnicas que, junto a las redes neuronales, forman lo que se denomina computación flexible (soft computing). Son técnicas en las que o bien los componentes aleatorios son fundamentales, como el **simulate annealing**, **los métodos evolutivos y genéticos**, o bien al **utilizar funciones de pertenencia difusas** (fuzzy).
- **Técnicas basadas en casos, en densidad o distancia:** Son métodos que se basan en distancias al resto de elementos, ya sea directamente, como los vecinos más próximos (los casos más similares), de una manera más sofisticada, mediante la

estimación de funciones de densidad. Además de los vecinos más próximos, algunos algoritmos muy conocidos son los jerárquicos, como two-step o COBWEB y los no jerárquicos como los K medias.

A título meramente ilustrativo, la siguiente tabla muestra algunas clases (clasificación, regresión, agrupamiento, reglas de asociación, correlaciones/factorizaciones) y algunas técnicas o algoritmos que pueden abordarlas:

Nombre	Predictivo		Descriptivo		
	Clasificación	Regresión	Agrupamiento	Reglas de asociación	Correlaciones/ factorizaciones
Redes neuronales	X	X	X		
Arboles de decisión ID3,C4.5, C5.0	X				
Arboles de decisión CART, CHART	X	X			
Otros árboles de decisión	X	X	X	X	
Redes neuronales de kohonen			X		
Regresión lineal y curvilínea		X			X
Regresión logística binaria	X			X	
K-means			X		
Apriori				X	
Naive Bayes	X				
Vecinos más próximos	X	X	X		
Análisis factorial y de Componentes Principales					X
Twostep, Cobweb			X		
Algoritmos genéticos y evolutivos	X	X	X	X	X
Máquinas de vectores soporte	X	X	X		
CN2 rules (cobertura)	X			X	
Análisis discriminante multivariante	X				

Tabla 12: Técnicas o algoritmos

Como podemos observar, la correspondencia entre tareas y técnicas es muy variada. Algunas tareas pueden ser resueltas por muy diversas técnicas y algunas técnicas pueden aplicarse para tres o incluso cuatro tareas. Esta variedad es una de las razones por la que es necesario conocer la capacidad de cada técnica, los ámbitos donde suele funcionar

mejor, la eficiencia, la robustez, etc…, en definitiva las características **funcionales** de cada técnica respecto a las demás.

17.3 Minería de Datos y aprendizaje inductivo

Al considerar tantas tareas y métodos parece que estamos tratando de problemas inconexos. Un aspecto extraño es que las técnicas pueden utilizarse para varias tareas. Es decir que hay procesos útiles para una tarea que también lo es generalmente para otras, con una ligera adaptación. La pregunta, no obstante, nos asalta: ¿Qué tiene que ver un agrupamiento con una regresión? O si hablamos de métodos, ¿Qué tiene que ver una red neuronal con un árbol de decisión? ¿Cómo puede ser qué todo esto entre dentro de la "extracción de conocimiento"?.

La respuesta a estas preguntas consiste en reconocer que todas las tareas y los métodos se centran alrededor de la idea del **aprendizaje inductivo**. Son, por decirlo de un modo más coloquial, presentaciones diferentes del mismo proceso.

¿Qué es aprendizaje inductivo?

Es un término que se utiliza en psicología, pedagogía, zoología, antropología y, más reciente en informática. Existen 4 definiciones diferentes (pero en cierto modo equivalente) del aprendizaje:

- La visión más genérica define el aprendizaje como la mejora del comportamiento a partir de la experiencia.
- La visión más externa define el aprendizaje como la capacidad de predecir observaciones futuras con plausibilidad o explicar observaciones pasadas.
- Una visión más estática define el aprendizaje inductivo como la identificación de patrones, de regularidades, existen en la evidencia.
- Una visión más teoría y matemática define el aprendizaje inductivo como eliminación de redundancia, vista como comprensión de información.

Estas 4 visiones se conjuntan perfectamente de la siguiente manera: el aprendizaje nos permite identificar regularidades en un conjunto de observaciones. Estas regularidades son en realidad redundancias que pueden ser representadas por patrones o modelos que los compriman o que los definan.

Estos patrones pueden ser utilizados para predecir observaciones futuras o explicar observaciones pasadas. Esta capacidad de predecir y explicar el entorno es fundamental para mejorar el comportamiento.

Las dos primeras definiciones son de aprendizaje, mientras que las dos últimas son de aprendizaje inductivo. El aprendizaje inductivo es un tipo especial de aprendizaje que parte de casos particulares y obtiene casos generales que generalizan o abstraen la evidencia. Existen, otros tipos de aprendizaje, como por ejemplo el aprendizaje abductivo (o explicativo,

conocido en ingles por **EBL Explanation-based learning**) y el aprendizaje por analogía, que van de casos particulares a casos particulares, este tipo de aprendizaje es menos útil para la Minería de Datos.

El aprendizaje puede ser además de varios tipos según ciertas características de la presentación de los datos y de la forma de aprender. Por ejemplo el aprendizaje puede ser **incremental** o no, dependiendo de si los datos se van presentando poco a poco, o si se tienen todos desde el principio. En el caso de la Minería de Datos se suele considerar un aprendizaje **no incremental**, ya que los datos se obtienen, de una base de datos o un almacén de datos.

Dependiendo de si se puede actuar sobre las observaciones, el aprendizaje puede ser **interactivo** o no. En el caso de la Minera de Datos tenemos generalmente datos históricos que no podemos afectar, con lo que no podemos actuar sobre las observaciones. El aprendizaje interactivo es útil cuando se puede interaccionar con un entorno para generar nuevas observaciones y así facilitar la tarea de aprendizaje. Un tipo especial de aprendizaje interactivo es aquel en el que podemos preguntar a un oráculo sobre cualquier ejemplo. Este tipo de aprendizaje puede empezar a ser relevante para la Minería de Datos web o Minería de Datos en entorno software, donde podamos considerar a los usuarios como oráculos, a los que podemos preguntar por sus preferencias. También podemos considerar a la base de datos como un oráculo, entonces tenemos en realidad lo que se conoce como **Query Learning** (aprendizaje por consulta).

Un tipo especial de aprendizaje interactivo es el aprendizaje por refuerzo, en el que un agente va realizando una serie de tareas, que si son acertadas son recompensadas positivamente (premio) y si son desacertadas se penalizan (castigo). Aunque no vamos a tratar este tipo de aprendizaje, puede ser de gran utilidad en ámbitos donde se desea personalizar portales web, aplicaciones o asistentes software a la preferencia del usuario. Actuando este de manera que recompense o penalice según la aplicación se comporte o no apropiadamente.

17.3.1 Los patrones son hipótesis. Evaluación

Una de las características presentes en cualquier tipo de aprendizaje y en cualquier tipo de técnica de Minería de Datos es su carácter **hipotético**, es decir, lo aprendido puede ser refutado por evidencia futura.

En muchos casos, los modelos no aspiran a ser modelos perfectos, sino modelo **aproximados**. En cualquier caso, al estar trabajando con hipótesis, es necesario realizar una evaluación de los patrones obtenidos, con el objetivo de estimar su validez y poder compararlos con otros.

Según las dos primeras definiciones de aprendizaje, la manera de evaluar un patrón o modelo es ver cómo se comporta con observaciones futuras.

Según las otras dos definiciones de aprendizaje, un modelo es mejor cuanta más regularidad encuentra, es decir, cuanto más comprima la evidencia.

Además de la evaluación de los modelos o patrones extraídos por cada método, es interesante evaluar y comparar métodos. Puede parecer evidente que un método será mejor que otro si genera mejores modelos. La curiosidad es que dado cualquier método de aprendizaje, siempre existen problemas que son resueltos mejor por otros métodos. Esta característica se denomina lacónicamente "no hay almuerzo gratis" (no free lunch theorem).

Un aspecto importante en la evaluación de métodos es su variabilidad. Puede haber métodos de aprendizaje que tengan una precisión media muy alta, pero que tengan muchos altibajos. Esto se puede medir para distintos problemas, se dice que el método tiene un amplio rango o abanico de aplicación. Podemos hablar de si el método funciona de manera regular para distintas muestras de datos del mismo problema, es decir, si el método es muy susceptible al orden, cantidad o subconjunto de datos presentado (del mismo problema). En este caso hablamos de la estabilidad de un algoritmo de aprendizaje. Otro término relacionado es la robustez al ruido, en el que se intenta medir si el método trata bien conjuntos de datos con valores erróneos o faltantes.

17.3.2 Métodos retardados y anticipativos. Comprensibilidad

Un aspecto muy relevante en una extracción de conocimiento es que los modelos extraídos sean comprensibles. La primera condición para que un modelo sea comprensible es que tengamos un modelo. Esto puede parecer una sandez, pero existen técnicas de Minería de Datos que resuelven tareas sin construir modelos, al menos explícitamente.

Los métodos sin modelo y con modelo reciben generalmente el nombre de métodos retardados o perezosos (lazy) y métodos anticipativos o impacientes (eager):

- **Métodos retardados o perezosos:** El método actúa para cada pregunta o petición requerida. No se construye modelo. Optimización local. Los ejemplos deben de preservarse porque son necesarios para realizar cada predicción. El tiemplo de respuesta empieza a degradarse cuando el número de ejemplos es muy grande porque hay que consultar mucho de ellos. En cambio, la ventaja es que no hay que entrenar el modelo.
- **Método anticipativos o impacientes:** El método obtiene un modelo a partir de todos los ejemplos. Los ejemplos, por tanto, pueden ignorarse. optimización global. Se requiere de un tiempo de entrenamiento, que suele ser grande, pero una vez entrenando el modelo, su aplicación suele ser instantánea.

En general, los métodos retardados, para ser eficientes, actúan generalmente teniendo en cuenta sólo los ejemplos cercanos. En este sentido suelen ser locales. En cambio los métodos impacientes construyen un modelo global.

En la siguiente Ilustración se muestra una clasificación de algunos métodos donde se indica si generan modelo y de cuales generan un modelo comprensible.

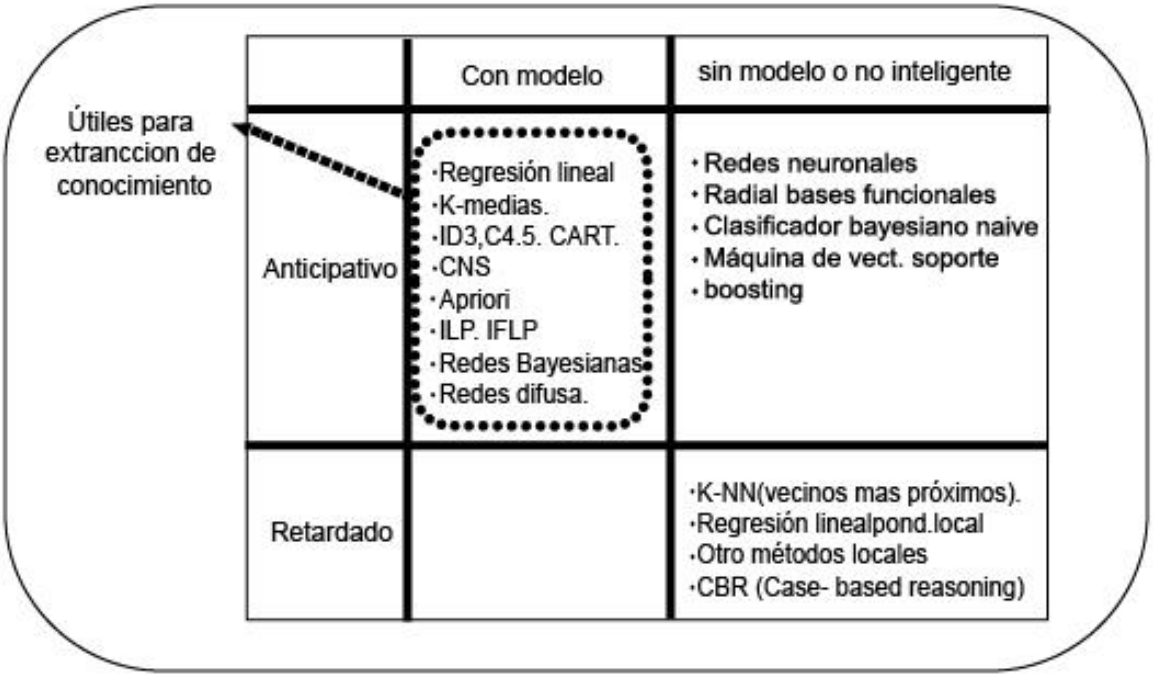


Ilustración 51: Clasificación de algunos métodos con modelo / sin modelo

La eficiencia del aprendizaje

Un aspecto fundamental a considerar en el aprendizaje es el esfuerzo computacional que se requiere. Lógicamente, entre dos métodos, preferiremos aquel que obtenga patrones de una manera más rápida. La eficiencia del aprendizaje, depende de muchos aspectos. Depende generalmente del:

- Número de ejemplos.
- Número de atributos.
- Complejidad de los ejemplos.
- Espacio de hipótesis que se está considerando.
- Conocimiento previo existente.
- Nivel de error permitido.
- Lo sorprendente o innovador que se busquen los patrones.

Existe una rama del aprendizaje automático, denominada teoría del aprendizaje computacional (**COLT**, **C**omputational **L**earning **T**heory), que se encarga de analizar esta complejidad para distintas tareas, dependiendo de todos estos factores.

Uno de los factores que más influyen en la eficiencia del aprendizaje es el tamaño de los datos, que suele ser función del número de ejemplos, el número de atributos y la complejidad de los valores existentes. Uno de los aspectos que más influyen es precisamente el número de atributos, en particular si estos son no significativos. En este caso tenemos un espacio de

alta dimensionalidad y los métodos que se basan en distancias se pierden en un espacio tan poco denso. Este problema se denomina como la maldición de la dimensionalidad (the course of dimensionality).

Otro aspecto que influye en el aprendizaje es la calidad de datos y la existencia de conocimiento previo, es decir, concepto de apoyo que facilite el aprendizaje. Dentro de la teoría del aprendizaje computacional, se sabe muy bien que el problema del aprendizaje para lenguajes de expresividad universales es intratable. Incluso se sabe que es intratable para lenguajes bastantes restringidos.

Existe un concepto el del aprendizaje **PAC** (probabilistic approximate correct), que flexibiliza y parametriza el error permitido, mostrando que para algunos lenguajes se puede aprender de una manera eficiente.

Finalmente, otro aspecto que afecta a la complejidad del aprendizaje son las aspiraciones de novedad, interés o sorpresa que uno se marque a los patrones extraídos. Aquellos que está explícito en los datos no suele ser interesante, pero tampoco cuesta. Es precisamente aquello que está implícito en los datos de una manera poco evidente, lo que generalmente responde y aporta conocimientos novedosos.

17.4 El lenguaje de los patrones. Expresividad

La característica más diferenciadora de los métodos de aprendizaje es la manera en la que se expresan los patrones aprendidos.

Cada método permite expresar mejor ciertos tipos de patrones. De ahí el hecho de que existan tantos métodos; la variedad de métodos permite capturar distintos tipos de patrones, si uno falla podemos probar con otro.

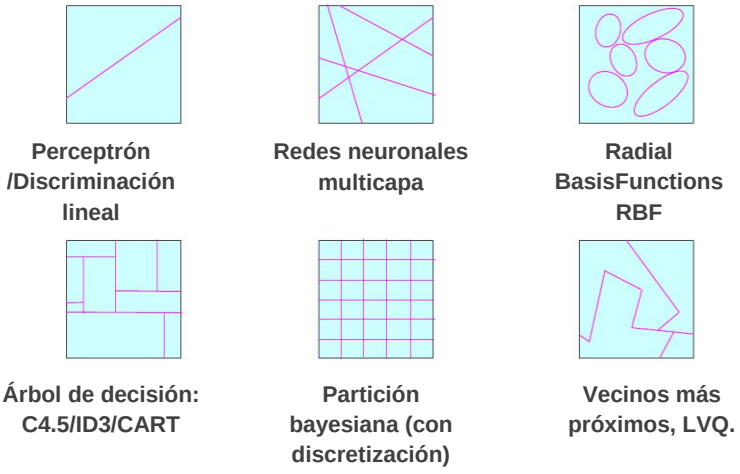


Ilustración 52: Representación de los tipos de patrones que son capaces de capturar distintos métodos.

En la Ilustración se muestra que tipos de patrones son capaces de expresar algunos de los métodos. Muchos de estos métodos se basan en colocar fronteras, linderos o confines entre zonas. Cada una de las zonas es capaz de aglutinar una clase o un grupo.

Otros métodos no establecen fronteras explícitamente, sino que se basan en el concepto de centros y clasifican / agrupan por la distancia a estos centros o zonas más densas.

17.4.1 ¿Qué expresividad es necesaria? Subajuste y sobreajuste.

A tenor de la variedad de métodos no debe existir ninguna Ilustración geométrica que no pueda capturarse con buena aproximación por alguno o varios de los métodos.

El problema es que existen patrones que no dependen de la proximidad o similitud espacial o geométrica entre los individuos. Este tipo de patrones no pueden ser aprendidos por los métodos de “cerca y rellena”. Esto puede parecer poco natural, porque, normalmente, es de esperar que los elementos se rodean de elementos se rodeen de elementos de su condición. Sin embargo, el patrón existe y las herramientas de Minería de Datos deberían de ser capaces de capturarlo.

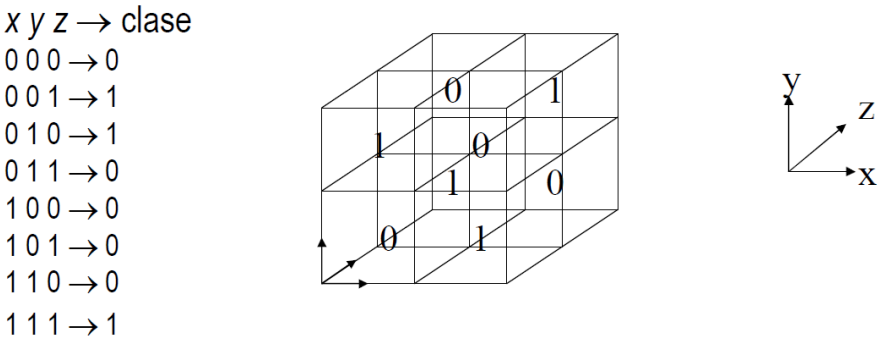


Ilustración 53: problema de la paridad. Problema relacional sin zonas geométricas distinguibles.

Si aprendemos la función de paridad con una red neuronal tenemos un patrón artificial, ya que la función de paridad no queda definida a partir del número de atributos de un determinado valor sino como una combinación “arbitraria” de pesos. Además, la red que calcula la paridad de un número de n dígitos (entradas) no es la misma que la que lo calcula para n+1 dígitos (entradas).

La razón es que el patrón del problema de la paridad es “relacional”. El patrón depende de las relaciones entre varios atributos. En este caso el patrón está definido de tal manera que la distancia no puede ser aprovechada.

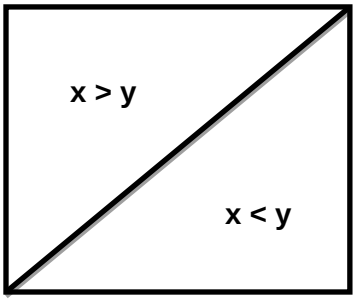


Ilustración 54: problema relacional con zonas geométricas sencillas.

17.5 Breve comparación de métodos

Breve comparación de algunos de los métodos más señalados, destacando sus rasgos más importantes:

- **Técnicas de modelización estadística:** Las técnicas paramétricas son muy eficientes, disponibles en multitud de herramientas y en muchos casos comprensibles. Las técnicas paramétricas son mucho más expresivas, aunque suelen perder en comprensibilidad debido a la utilización de núcleos y aproximaciones locales, las técnicas no paramétricas son bastantes más ineficientes para grandes volúmenes de datos. Tanto técnicas paramétricas y no paramétricas trabajan más cómodamente con datos numéricos.

- **Técnicas bayesianas:** Son fáciles de usar, muy eficientes, pueden tratar muchos atributos, son muy robustos al ruido, la expresividad es limitada y depende de la discretización, son estables a la muestra.

- **Técnicas basadas en árboles de decisión y sistemas de aprendizaje de reglas:** Son una de las estrellas de la Minería de Datos. Son fáciles de usar, admiten atributos discretos y continuos, tratan bien los atributos no significativos, los valores faltantes y el ruido. Son bastantes eficientes y obtienen resultados para clasificación bastantes buenos, aunque la mayor ventaja de estos métodos es su inteligibilidad.

- **Técnicas relacionales y declarativas:** Son técnicas muy expresivas que permiten tratar datos con estructuras y capturas patrones relaciones y recursivos, así como expresar el conocimiento previo en forma de reglas. El mayor inconveniente de esta técnica es la dificultad de manejo y la poca eficiencia.

- **Técnicas basadas en redes neuronales artificiales:** Aunque existen redes con valores por defecto y donde el número de capas y de nodos internos se calcula automáticamente, requieren de cierta experiencia para poderles sacar el máximo

partido, su ventaja principal es que, cuando están bien ajustadas, obtienen precisiones muy altas.

- **Técnicas basadas en núcleo y máquinas de soporte vectorial:** Son técnicas muy eficientes que permiten trabajar con datos con alta dimensionalidad. Proporcionan modelos muy precisos. El inconveniente más importante es que a veces es necesario elegir una buena función de núcleo para obtener buenos resultados.

- **Técnicas estocásticas y difusas:** Este tipo de técnica son demasiados diversas para intentar dar unas características conjuntas a todas ellas. Una de la característica de esta técnica sea su costo computacional.

- **Técnicas basadas en casos, en densidad o distancia:** Son fáciles de usar en general, eficientes si el número de ejemplos no es excesivamente grande. Al ser técnicas generalmente locales tienen bastantes expresividad. Al estar basadas en distancias, no digieren bien los atributos no significativos.

GUÍA DE APOYO PARA EL ESTUDIANTE

Tema 6. Técnicas de Minería de Datos

I. Mencione:

a. La extracción de conocimiento a partir de datos tiene como objetivos descubrir patrones que entre otras cosas, deben ser:
b. Mencione las tareas Predictivas:
c. Mencione las tareas Descriptivas:

II. Complete:

a. El aprendizaje puede ser _____ o _____.
b. El _____ de adaptar un problema de más de dos clases a un problema de dos clases se denomina _____.
c. El _____ es un _____ que se utiliza en psicología, pedagogía, zoología, antropología y, más reciente en informática.
d. Una de las _____ presentes en cualquier tipo de _____ y en cualquier tipo de _____ de Minería de Datos es su carácter _____.

III. Escriba **V** si es Verdadero y **F** si es Falso según convenga:

a. El Método Retardado o Perezoso actúa para cada pregunta o petición requerida construyendo un modelo: ___.
b. La visión más genérica define el aprendizaje como la mejora del comportamiento a partir de la experiencia: ___.
c. Una de las características presentes en cualquier tipo de aprendizaje y en cualquier tipo de técnica de Minería de Datos es su carácter **hipotético**: ___.
d. La característica más diferenciadora de los métodos de aprendizaje es la manera en la que se expresan las tareas aprendidas: ___.
e. Las **Técnicas bayesianas** Son difíciles de usar, muy eficientes, pueden tratar muchos atributos, son menos robustos al ruido, la expresividad es ilimitada y depende de la discretización, son estables a la muestra: ___.

SOLUCIÓN GUÍA DE APOYO PARA EL ESTUDIANTE

Tema 6. Técnicas de Minería de Datos

I. Mencione:

a. La extracción de conocimiento a partir de datos tiene como objetivos descubrir patrones que entre otras cosas, deben ser: **Válidos, Novedosos, Interesantes y Entendibles.**
b. Mencione las tareas Predictivas: **Clasificación, Clasificación suave, Categorización, Preferencia o priorización, Regresión.**
c. Mencione las tareas Descriptivas: **Agrupamiento o Clustering, Correlación y Factorizaciones, Reglas de Asociación, Dependencias Funcionales y Detección de valores e Instancias Anómalas.**

II. Complete:

a. El aprendizaje puede ser **interactivo** o no.
b. El **proceso** de adaptar un problema de más de dos clases a un problema de dos clases se denomina **Binarización**.
c. El **aprendizaje inductivo** es un **término** que se utiliza en psicología, pedagogía, zoología, antropología y, más reciente en informática.
d. Una de las **características** presentes en cualquier tipo de **aprendizaje** y en cualquier tipo de **técnica** de Minería de Datos es su carácter **hipotético**.

III. Escriba **V** si es Verdadero y **F** si es Falso según convenga:

a. El Método Retardado o Perezoso actúa para cada pregunta o petición requerida construyendo un modelo: **F**.
b. La visión más genérica define el aprendizaje como la mejora del comportamiento a partir de la experiencia. **V**.
c. Una de las características presentes en cualquier tipo de aprendizaje y en cualquier tipo de técnica de Minería de Datos es su carácter **hipotético**: **V**.
d. La característica más diferenciadora de los métodos de aprendizaje es la manera en la que se expresan las tareas aprendidas: **F**.
e. Las **Técnicas bayesianas** Son difíciles de usar, muy eficientes, pueden tratar muchos atributos, son menos robustos al ruido, la expresividad es ilimitada y depende de la discretización, son estables a la muestra: **F**.

PARTE III: Enunciados de las prácticas de laboratorio

GUIA PRÁCTICA Nº 0

INSTALACIÓN, AMBIENTE Y ENTORNO R: EXPORTAR E IMPORTAR Y PROCESAR CUADROS DE DATOS

Objetivo general

- Estimular en los estudiantes interesados en Minería de Datos el uso de los procedimientos matemáticos y métodos estadísticos para la realización de Minería de Datos, a través de un software potente de distribución libre y muy utilizada alrededor del mundo por investigadores y profesionales de todas las ciencias.

Requerimientos

Para la realización de la práctica se necesitan los siguientes requerimientos:

- Hardware:
  - Procesador con velocidad de 2.1 GHz
  - Memoria RAM de 1 GB.
  - Una PC con Windows o con Linux.
- Software
  - Programa R versión 3.2.0 o superiores

Contenido:

- Instalación en del Programa R en Windows.
- La Ventana Principal R-Gui, sus Paneles y Componentes.
- La Barra de Herramientas y la Barra de Íconos.
- Instalación de paquetes adicionales y Ayuda en R
- Programación en R, Importar y Exportar Bases de Datos.
- Definir Objetos, Funciones de Usuario y Demostraciones en R.
- Creación y Manipulación de Objetos en R
- Funciones Matrices y Gráficos en R.

Introducción a R

R es un sistema para análisis estadísticos y gráficos creado por Ross Ihaka y Robert Gentleman. R tiene una naturaleza doble de programa y lenguaje de programación y es considerado como un dialecto del lenguaje S creado por los Laboratorios AT&T Bell.

R se distribuye gratuitamente bajo los términos de la GNU General Public Licence; su desarrollo y distribución son llevados a cabo por varios estadísticos conocidos como el Grupo Nuclear de Desarrollo de R.

R está disponible en varias formas: el código fuente escrito principalmente en C (y algunas rutinas en Fortran), esencialmente para maquinas Unix y Linux, o como archivos binarios precompilados para Windows, Linux (Debian, Mandrake, RedHat, SuSe), Macintosh y Alpha Unix.

Los archivos necesarios para instalar R, ya sea desde las fuentes o binarios pre-compilados, se distribuyen desde el sitio de internet Comprehensive R Archive Network (CRAN) junto con las instrucciones de instalación. Para las diferentes distribuciones de Linux (Debian,...), los binarios están disponibles generalmente para las versiones más actualizadas de ´estas y de R; visite el sitio CRAN si es necesario.

R posee muchas funciones para análisis estadísticos y gráficos; estos últimos pueden ser visualizados de manera inmediata en su propia ventana y ser guardados en varios formatos (jpg,png, bmp, ps, pdf, emf, pictex, xfig; los formatos disponibles dependen del sistema operativo).

Los resultados de análisis estadísticos se muestran en la pantalla, y algunos resultados intermedios (como valores P-, coeficientes de regresión, residuales,...) se pueden guardar, exportar a un archivo, o ser utilizados en análisis posteriores.

- **Instalación en GNU/Linux**

Para la instalación, distribuciones derivadas de debian (Ubuntu, Guadalinex,. . .), en una consola se introduce en una sola línea: sudo apt-get install r-base-html r-cran-rcmdr r-cran-rodbcr-doc-html r-recommended

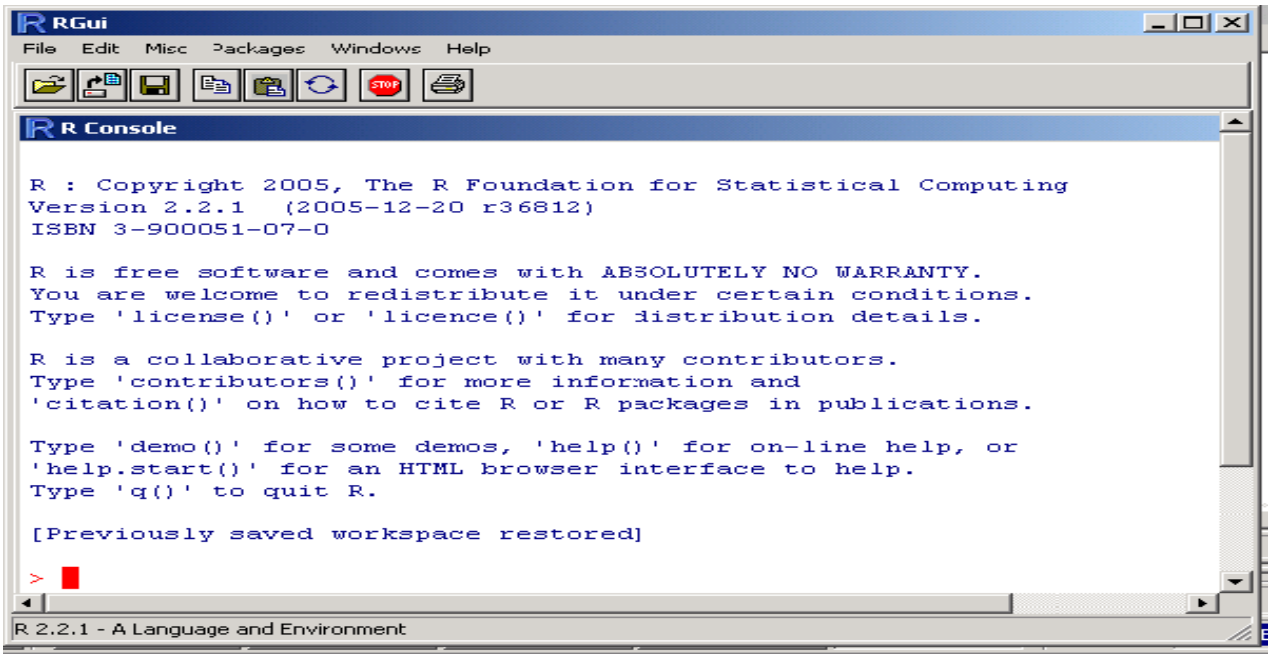
Otra opción es utilizar el gestor de paquetes de la propia distribución e instalar los paquetes r-base-html, r-cran-rcmdr, r-cran-rodbc,r-doc-html y r-recommended.

- **Instalación en Windows**

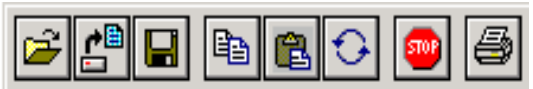
La descarga de R en el equipo se efectúa desde:
http://cran.es.r-project.org/bin/windows/base/release.htm
Luego se procede con la ejecución, siguiendo las instrucciones para la instalación

- **La Ventana Principal R-Gui, sus Paneles y Componentes.**

Una vez instalado en el computador, accedemos al sistema desde el menú Inicio en Programas, buscamos la carpeta R y activamos la opción R i386 3.1.2, luego de lo cual se abre esta Ventana de Inicio del Sistema, activándose el panel R Console por defecto.



- **Barra de Herramientas y Barra de Iconos**



La Barra de Herramientas activa para la R-Console dispone de los siguientes Menús y algunas utilidades, para otros Paneles los Menús varían:

[File]: Manipular archivos generados por el sistema. Cambiar de directorio de trabajo, por defecto es [C:\Archivos de programa\R\R-2.2.1].

[Edit]: Edición de datos y matrices. Limpiar Panel de Trabajo. Editar y cambiar con Ilustración de formato en Rgui preferidas por el usuario.

[Misc]: Detener ejecución de cálculo en procesos [Escape]. Enlistar o Eliminar los objetos actualmente creados por el usuario en el Sistema.

[Package]: Cargar paquete o Biblioteca [Library] desde el sistema instalado en el computador. Instalar o Actualizar paquetes desde Internet.

[Windows]: Organizar los distintos paneles activos en Cascada o Mosaico. Pasar a cualquiera de los paneles activos.

[Help]: Combinación de Teclas útiles en la R-Console. Respuestas a preguntas más frecuentes en R (FAQ). Diversos Manuales en PDF. Referencia sobre funciones construidas en R. Ayuda Html instalada en el computador. Ayuda en la Búsqueda de algún ítem en el sistema instalado. A propósito de ítem alguno. Sitios en Internet del proyecto CRAN.

- **Ayuda dentro del programa**

?rnorm
!Cuidado¡
?if # mal
?help("if")

- **Usar R desde un editor**

¿Por qué usar R desde un editor? El uso de scripts y el mantenimiento del código ordenado y comentado es una "buena práctica estadística" (ver también load history, save history).

- **Importar y Exportar Bases de Datos**:

Existen varias formas para Importar / Exportar bases de datos de otros sistemas, información detallada se encuentra desde el menú Ayuda **[Help]** en **[Manuals (in PDF) – R Data Import/Export]**. Para importar se tiene **scan**(), **read.table**() y **read.csv**(), para exportar se tiene **write.table**() y **write.csv**, entre otras; la extensión **csv** en Microsoft-Excel delimita valores por coma.

Previamente debe cambiar de directorio en Menú [File] [Change dir…] al de Practica0 (previamente creada) ubicada en Escritorio. Así todo archivo creado en R se ubicará en esa carpeta y todo archivo creado en otro sistema será buscado desde allí.

Ejemplo: Exportar la BD **women** hacia Excel al archivo llamado **mujeres.csv** en la carpeta Practica0 ubicada en el escritorio. Abrir desde Excel barco.csv y guardarla como Libro de Excel pero con el nombre buque. Desde la R-Console importar el archivo buque desde Excel y almacenarlo en el objeto llamado barco, ver la estructura y el contenido de barco

write.csv(women, file = "mujeres.csv") # Se exporta la BD a un archivo Excel
Desde Excel se abre autos (Abrir Tipo de Archivo: Todos los archivos),
recordar que la carpeta de ubicación es Practica0 en Escritorio
En Excel se Guarda la BD como mujeres y de tipo: CSV (delimitado por comas))
Volver al ambiente R a la R-Console y ejecutar lo siguiente
medida = read.csv("mujeres.csv ") # Se crea el objeto medida
str(medida) # Se muestra la estructura del objeto medida

- **Guardando datos**

La función write.table guarda el contenido de un objeto en un archivo. El objeto es típicamente un marco de datos ('data.frame'), pero puede ser cualquier otro tipo de objeto (vector, matriz,. . .). Los argumentos y opciones son:

write.table(x,file ="", append = FALSE, quote = TRUE,sep =" ",eol = "\n", na = "NA", dec = ".", row.names = TRUE,col.names = TRUE, qmethod = c("escape", "double"))

- **Instalación de paquetes adicionales:**

Desde el menu Packages->Install package(s)...
Primero nos pide seleccionar el "CRAN mirror".
Desde R, con install.packages()

- **R como calculadora**

```
r    <- 5
area <- 2 * pi + r
area
[1] 11.28319
```

- **Creación y Manipulación de Objetos en R**

Objetos en R

- Casi todo en R es un objeto, incluyendo funciones y estructuras de datos.
- En R Los valores perdidos (*simplemente, hemos perdido algún dato y no sabemos qué valor toma*) por el código: NA (*Not available*).

Tipos de objetos

✓ **Dataframes**

Un **dataframe** (a veces se traduce como **'marco de datos'**) es una generalización de las matrices donde cada columna puede contener tipos de datos distintos al resto de columnas, manteniendo la misma longitud. Es lo que más se parece a una tabla de datos de SPSS o SAS, o de cualquier paquete estadístico estándar. Se crean con la función data.frame().

```
usuario <- c( 1:5 )
fecha    <- c( "15/11/08", "15/12/08", "15/13/08", "15/14/08", "10/15/08" )
#aa/dd/mm
pais <- c( "CR", "NIC", "GUA", "NIC", "HON" )
sexo  <- c( "M", "F", "F", NA, "F" )
edad  <- c ( NA, 45, 25, 39, 99 )
q1  <- c( 5, 3, 3, 3, 2 )
q2  <- c( 5, 5, 5, NA, 2 )
q3  <- c( 5, 5, 2, NA, 1 )
df  <- data.frame(usuario, fecha, pais, sexo, edad, q1, q2, q3, stringsAsFactors = FALSE )
df
```

✓ **Matrices**

Una matriz es un vector con un atributo adicional (dim), que a su vez es un vector numérico de longitud 2 que define el número de filas y columnas. Se crean con la función matrix().

```
matrix( data = 1:4, nrow = 2, ncol = 2, byrow = F, dimnames = NULL )
##      [,1] [,2]
## [1,]    1    3
## [2,]    2    4
```

✓ **Vectores**

Son matrices de una dimensión que solamente pueden contener valores homogéneos, ya sean numéricos, alfanuméricos o valores lógicos. Emplearemos la función c() para construir vectores (función “combine”)

```
#vectores
x <- c( 1, 2, 3, 4, 5, 6, 7, 8 )
y <- c( "gato", "perro", "leon", "zebra" ,"jirafa", "jaguar", caballo", "elefante" )
```

Podemos acceder a los elementos un vector usando los corchetes, que indican subíndice, así

```
y[ 6 ])
[1] "jaguar"
```

- **Aplicando funciones a objetos**

Una funcionalidad que resulta muy útil al trabajar con R es que podemos aplicar funciones a una gran cantidad de objetos: vectores, arrays, matrices, dataframes…

```
# aplicar funciones a objetos 'complejos'
y <- c (1.23, 4.56, 7.89)
round( y )
[1] 1 5 8
```

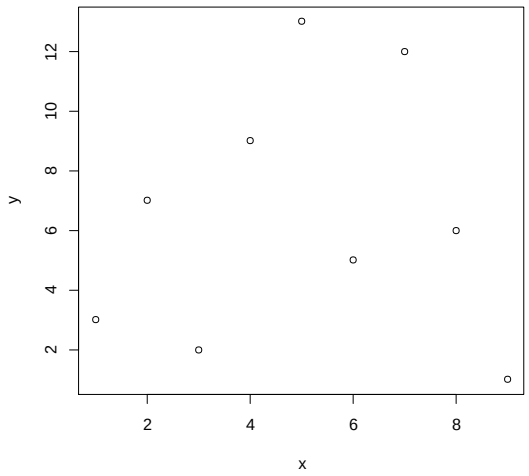
- **Gráficos**

✓ R incluye muchas y variadas funciones para hacer gráficos.
✓ El sistema permite desde gráficos muy simples a figuras de calidad para incluir en artículos y libros.
✓ También podemos ver un buen conjunto de ejemplos con demo(graphics).

- **Gráficos de dispersión**

plot(x): dibuja los valores de x (en el eje y) ordenados en el eje x.

```
#Ejemplo 1
x <- 1:9
y <- c (3, 7, 2, 9, 13, 5, 12, 6, 1) # creando algunos datos
plot(x, y)
```

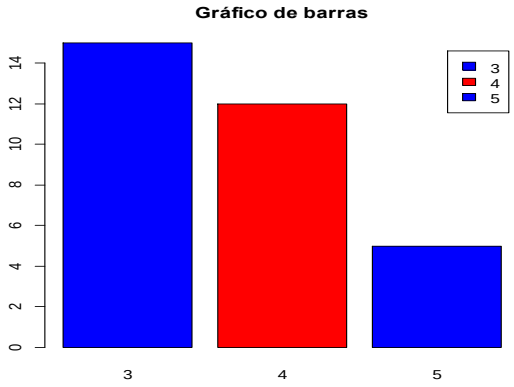


```
plot(x, y, type = "b", main = "punto-linea", xlab = "eje abcisas", ylab = "eje ordenadas")
```

Gráficos de barras

Creamos gráficos de barras con la función barplot(x), donde x es un vector o una matriz. Si x es un vector, los valores determinan las alturas de las barras y si es una matriz y beside = FALSE (por defecto) entonces cada barra corresponde a una columna de altura, con los valores de la columna dando las alturas de "sub-barras" apiladas. Si x es una matriz y beside = TRUE, entonces los valores de cada columna se yuxtaponen en lugar de apilarse

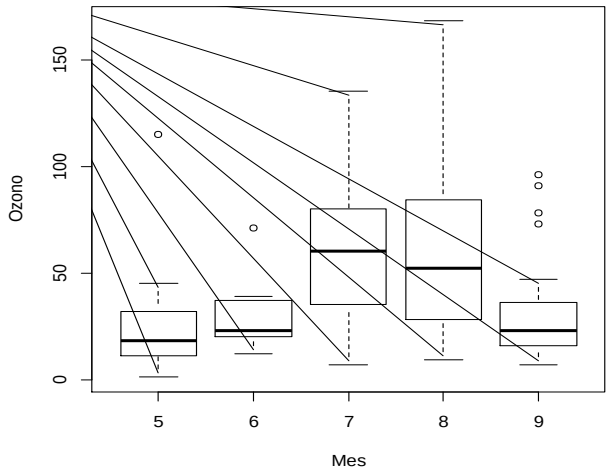
```
x <- table(mtcars$gear)
barplot(x, main = "Gráfico de barras", names.arg = c("3 Gears", "4 Gears",
   "5 Gears"))

barplot(x, main = "Gráfico de barras", beside = TRUE, col = c("blue", "red"),
   legend = rownames(x))
```



Boxplots

Los diagramas de caja son muy útiles para ver rápidamente las principales características de una variable cuantitativa, o comparar entre variables.

```
with(cars, boxplot(speed))
airquality$Month <- factor(airquality$Month)
boxplot(Ozone ~ Month, data = airquality, xlab = "Mes", ylab = "Ozono")
```

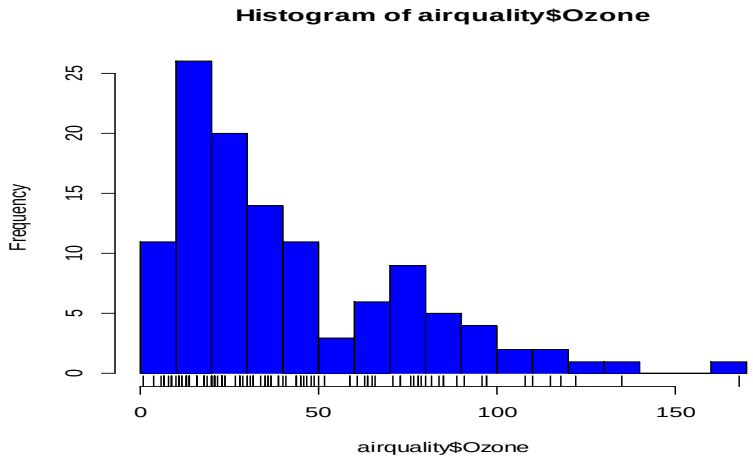


Histogramas

Podemos crear histogramas con la función hist(x), donde x es un vector numérico. Con la opción frec = FALSE se representan las densidades en lugar de las frecuencias. La opción breaks = n controla el número de barras.

airquality

```
hist(airquality$Ozone, breaks = 13, col = "blue")
rug(airquality$Ozone)
```



GUIA PRÁCTICA Nº 1

FUNCIONES GENÉRICAS, OPERADORES ARITMÉTICOS (suma resta, multiplicación, potencia, división, división entera, residuo), FUNCIONES ESPECIALES (sqrt, abs, factorial, gamma, choose, exp, log, log10, etc.), OPERADORES LÓGICOS (! , & , && , | , || , xor, isTRUE) Y OPERADORES RELACIONALES (==, != , < , <= , > , >=).

Objetivo general

- Aplicar funciones matemáticas, estructurando procedimientos algorítmicos y de programación utilizando el lenguaje y entorno R, a casos de aplicación práctica.

Objetivos específicos

- Escudriñar ejemplos de aplicación práctica que permitan aglutinar funciones matemáticas para que el alumno se apropie de las técnicas de programación en el uso de funciones.

Requerimientos

Para la realización de la práctica se necesitan los siguientes requerimientos:

- Hardware:
  - Procesador con velocidad de 2.1 ghz
  - Memoria RAM de 1 GB.
  - Una PC con Windows o con Linux.
- Software
  - Programa R versión 3.2.0

Operadores en R.

Operador	Descripción	Ejemplo
Asignación <– ó –>	Puede ser de derecha a izquierda ← ó de izquierda a derecha → usando combinación de teclas, con el signo menos y el símbolo “>” o “<”.	n <- 15 n [1] 15
		5 -> n n [1] 5
Adición, Sustracción, Multiplicación, División **+–* /** Potencia, división entera, residuo ^ ó **,%%,%/%	Operadores Aritméticos.	x=5-1 y=8 (v <- (2*x + y + 1)/2) [1] 8.5
		2 ^ 3 = 8 # Potencia
		9%%2 = 1 # División residual
		9 %/% 2 = 4 # División entera
FUNCIONES ESPECIALES Sqrt(), abs(),	sqrt () Extrae la Raíz cuadrada de x.	sqrt(25) [1] 5

factorial, gamma, choose, exp, log, log10, etc. Funciones matemáticas especiales relacionadas con las funciones beta y gamma:		
	abs() devuelve el valor absoluto de un valor x	abs(-9:9) [1] 9 8 7 6 5 4 3 2 1 0 1 2 3 4 5 6 7 8 9
	factorial () calcula el factorial de un número:	factorial(c(4,3,2)) [1] 24 6 2
	gamma (x): vectores numéricos	gamma(c(4,5,6)) [1] 6 24 120
	choose() calcula la combinacion $_nC_r$.	choose(n,r) n: n elements r: r subset elements nCr = n!/(r! * (n-r)!) choose(5,2) [1] 10
	exp() Calcula el valor exponencial de un número o un número de vector	x <- 5 exp(x) [1] 148.4132
	función log ( ) calcula los logaritmos naturales (ln) para un número o vector.	log(5) #ln5 [1] 1.609438
	log10 calcula logaritmos comunes (Lg)	log10(5) #lg5 [1] 0.69897
OPERADORES LÓGICOS (! , & , && , | , || , isTRUE	Operador negación !, X no es igual a Y	x <- (0:2) !(x > 0) [1] TRUE FALSE FALSE
	# operador lógicos and & Las más cortas son vectorizados , es decir que pueden devolver un vector	((-2:2) >= 0) & ((-2:2) <= 0) [1] FALSE FALSE TRUE FALSE FALSE
	# && La forma más larga evalúa de izquierda a derecha examinar sólo el primer elemento de cada vector.	((-2:2) >= 0) && ((-2:2) <= 0) [1] FALSE
	isTRUE esta function prueba si el valor es verdad	isTRUE(1>0) [1] TRUE
	Muestra los resultados de X o de Y	x <- c(1:10) x[(x>8) | (x<5)] [1] 1 2 3 4 9 10
	XOR || es un tipo de disyunción lógica de dos operandos que es verdad si solo un operando es verdad pero no ambos.	x <- c(1:10) x[(x>8) ||(x<5)] [1] 1 2 3 4 5 6 7 8 9 10
OPERADORES RELACIONALES (==, != , < , <= , > , >=).	Desigualdad X es diferente de Y	x=8 y=6 b<-(x!=y) b [1] TRUE
	Compara la igualdad de dos objetos	x <- 1:3; y <- 1:3 x == y [1] TRUE TRUE TRUE
	Evalúa si un objeto X es mayor que un objeto Y; X>Y En X< Y evalúa si un objeto X es menor que un objeto Y	3 > 2 # TRUE (¿tres es mayor que dos?) 3 < 2 # FALSE
	Compara cada elemento de 'x' con el primer elemento del vector 'y'.	x=9 y=13

		x >= y [1] FALSE
	x >= y evalúa si un valor X es mayor o igual que un valor Y en el caso x <= y evalúa si un valor X es menor o igual que un valor Y	x <- (1:3) y<- (4:6) x >= y [1] FALSE FALSE FALSE x <= y x <= y [1] TRUE TRUE TRUE

Ejercicio propuesto 1:

Enunciado: Dada una fecha, en día, mes y año, calcular el día de la semana.

GUIA PRÁCTICA Nº 2

FUNCIONES PARA CONTROL DE FLUJO (if, if-else, for, while, repeat, break, next)
(ESTUDIO DE CASO: MÁXIMO COMÚN DIVISOR DE DOS NÚMEROS)

Objetivo general

- Aplicar funciones para control de flujo estructurando procedimientos algorítmicos y de programación utilizando el lenguaje y entorno R, a casos de aplicación práctica.

Objetivos específicos

- Implementar estructuras de programación para desarrollar el caso práctico de encontrar el MCD (Máximo común divisor) de números enteros.

Requerimientos

Para la realización de la práctica se necesitan los siguientes requerimientos:

- Hardware:
  - Procesador con velocidad de 2.1 GHz
  - Memoria RAM de 1 GB.
  - Una PC con Windows.
- Software
  - Programa R versión 3.2.0

Operadores en R.

Operador	Descripción	Ejemplo
Controladores de flujo (if, if – else, for, while, repeat, break next)	while(){} :: Comienza comprobando una condición y continua siempre que se cumpla	x <- 1 ## valor inicial... while(x < 11) { print(x) x <- x + 1 ## Aumento el valor de x de a 1 por iteración } ## imprime los enteros del 1 al 10

	for(){} :: Repite una acción un nº determinado de veces.	mat <- matrix(1:20, 4, 5)
		for(i in 1:4) { # i será el índice de las filas de mat
		for(j in 1:5) { # j será el índice de las columnas de mat
		print(mat[i, j]) # Nótese que se usan letras distintas, para evitar
		} # Confusiones También se ajusta la indexación }

	If Se ejecuta el comando si la condición es TRUE	if(3 > 2) { print('Es verdadero!') print(' :-D ') }
	If –else Cuando se cumple una condición dada, se ejecuta el comando1, en caso contrario, se ejecuta el	x <- 1:10 clasif <- ifelse(x > 5, 'grande', 'chico') clasif <- paste(x, clasif)

	comando2	clasif
	repeat{} :: Inicia un bucle infinito del que sólo se sale con ***break***	x0 <- 1 tol <- 1e-8 repeat{ x1<-computeEstimate() if (abs(x1-x0)<tol){ break }else{ x0<-x1 } }

	next Interrumpe la iteración del loop y pasa a la siguiente	for(i in 1:10) {
		if(i == 4) next
		print(i)
		}# imprime del 1 al 10 excepto el 4

data.frame	Esta función crea cuadro de datos , colecciones de variables que comparten muchas de las propiedades de las matrices y de las listas , utilizados como la estructura de datos fundamental de la mayoría de software de modelado de R	L3 <- LETTERS[1:3] fac<- sample(L3, 10, replace = TRUE) (d <- data.frame(x = 1, y = 1:10, fac = fac)) ## lo "mismo" con nombre de columnas automáticas data.frame(1, 1:10, sample(L3, 10, replace = TRUE))

function()	Para crear una función en la R-Console se utiliza la palabra reservada **function()**, dentro del paréntesis se ubican las variables y parámetros a evaluar (separados por comas)	# Se crea la función f
		f = function(x, y) x**2 + y**2
		# Se crea el objeto sol
		sol = f(3,4)
		# Se muestra el objeto
		sol
		# Se genera un gráfico 3D.
		persp(x <- y <- -10:10, y, outer(x,y,f))

max/ min	Devuelve los máximos (en paralelo) y mínimos de los valores de entrada	*min(5:1, pi) # -> one number* *pmin(5:1, pi) # -> 5 numbers*

Ejercicio resuelto: Identificar el año según sea regular o bisiesto

```
TipoAnio = function(anio) {       # La funciones DIV y MOD en R son "%/%" y "%%"
a0 <- "Anio_Regular" ; a4 <- "Anio_Bisiesto"    # Crear Etiqueta del anio
anioS <- ifelse(anio %% 100 == 0, anio/100, anio)        # Tomar el siglo o el anio
anioT <- ifelse(anioS %% 4  == 0, a4, a0)                # Asignar la etiqueta condicional
 data.frame(Anio = anio, Tipo = anioT) }                  # Mostrar cuadro de resultado
# Por ejemplo evaluar c(1600,1700,1800,1900,1970,1972,2000,2016)
TipoAnio( c(1600,1700,1800,1900,1970,1972,2000,2016) )
anyos = TipoAnio( 1500:2400 )     # Ejemplo de varios siglos consecutivos
str(anyos)                        # Estructura del objeto llamado anyos
table(anyos$Tipo)                 # Tabla de resumen para la variable binaria anyos$Tipo
barplot(table(anyos$Tipo))        # Generar un Diagrama de Barras para la variable binaria.
```

Ejercicio propuesto Nº 1

Enunciado: Se dispone de un lote de terreno rectangular de 15 metros de largo por 12 metros de ancho. Determine: ¿Cuántas parcelas cuadradas y de que dimensiones debe ser?

Tarea: Ejemplos de aplicación de MCD y del mcm

- ✓ ¿Relación entre el mínimo común múltiplo (mcm) y el MCD?

Tarea: Crear una función que extraiga el mcm de dos números

- ✓ Modificar la función MCD y agregarle el mcm

GUIA PRÁCTICA Nº 3

REGRESIÓN LINEAL SIMPLE, MÚLTIPLE Y NO LINEAL (ESTUDIO DE CASO: PESO Y TALLA DE MUJERES)

Objetivo general

- Analizar desde un conjunto de datos, el comportamiento de una variable dependiente en relación a una o más variables independientes, utilizando modelos de regresión lineal o polinómico en el entorno R.

Objetivos específicos

- Describir el tipo de correlación de cada par de variables, en forma numérica o de forma gráfica.
- Validar y comparar el modelo que tenga el mejor ajuste a partir de resultados numéricos y de gráficos de diagnóstico, que permitan apoyar a la toma de decisiones o proponer normativas.

Requerimientos

Para la realización de la práctica se necesitaron los siguientes requerimientos:

- Hardware:
  - Procesador con velocidad de 2.1 GHz
  - Memoria RAM de 1 GB.
  - Una PC con Windows o con Linux.
- Software
  - Programa R versión 3.2.0, o posterior.

Introducción

Regresión lineal es una técnica estadística para investigar y modelar la relación entre variables.

La regresión lineal se adapta a una amplia variedad de situaciones:

- En el área de investigación social, el análisis de regresión se utiliza para predecir un amplio rango de fenómenos, desde medidas económicas hasta diferentes aspectos del comportamiento humano.
- En el contexto de la investigación de mercados puede utilizarse para determinar en cual de diferentes medios de comunicación puede resultar más eficaz invertir; o para predecir el número de ventas de un determinado producto.
- En el área de la física se utiliza para caracterizar la relación entre variables o para calibrar medidas. Etc.

En el caso de dos variables (**regresión simple**) como en el de más de dos variables (**regresión múltiple**), el análisis de regresión lineal puede utilizarse para explorar y cuantificar la relación entre una variable llamada dependiente o criterio (Y) y una o más variables llamadas independientes o predictoras (X_1, X_2,…, X_k), así como para desarrollar una ecuación lineal con fines predictivos.

La diferencia entre variable predictora y respuesta no siempre es completamente clara y depende algunas veces de nuestros objetivos. Algunos nombres conocidos para las variables predictoras y respuestas son:

Variables predictoras equivale a: entradas, regresoras, independientes, exógenas, etc.

Variable de respuesta equivale a: salidas, dependiente, endógena, etc.

Se pueden resolver muchos problemas por medio de la regresión lineal, y puede conseguirse todavía más aplicando las transformaciones a las variables para que un problema no lineal pueda convertirse a uno lineal, que puede resolverse entonces por el método de mínimos cuadrados.

¿Cómo se analiza un modelo de regresión?

Para analizar un modelo de regresión se pueden establecer básicamente dos pasos.

Paso 1. Estimar los parámetros del modelo de regresión. Este proceso es llamado **ajuste del modelo a los datos**.

Paso 2. El siguiente paso de un análisis de regresión es chequear que tan bueno es el modelo ajustado. El resultado de este chequeo puede indicar si el modelo es razonable o si el ajuste original debe ser modificado (a la Minería de Datos generalmente omite este paso y se queda con cualquier modelo de ajuste).

Recomendaciones:

Profundizar en la generalización de los modelos lineales y no lineales; aquellos polinómicos y los lineales múltiples (en varias variables regresoras), además de las superficies de respuesta.

Operadores en R.

Operador	Descripción	Ejemplo
abline	Esta función añade una o máslíneas rectas a través de la trama actual.	sale5 <- c(6, 4, 9, 7, 6, 12, 8, 10, 9, 13) plot(sale5) abline(lsfit(1:10, sale5))
lm	Se utiliza para ajustar los modelos lineales. Se puede utilizar para llevar a cabo la regresión, el análisis de estrato único de varianza y	ctl<- c(4.17,5.58,5.18,6.11,4.50,4.61,5.17,4.53,5.33,5.14) trt<- c(4.81,4.17,4.41,3.59,5.87,3.83,6.03,4.89,4.32,4.69) group <- gl(2, 10, 20, labels = c("Ctl","Trt")) weight <- c(ctl, trt) lm.D9 <- lm(weight ~ group)

	análisis de covarianza.	lm.D90 <- lm(weight ~ group - 1) # omitiendo intercepto
curve	Dibuja una curva que corresponde a una función en el intervalo [From, to] (de, hasta).	plot(cos, -pi, 3*pi) curve(cos, xlim = c(-pi, 3*pi), n = 1001, col = "blue", add = TRUE)
par(mfrow = c (nfilas , ncols))	R hace que sea fácil de combinar múltiples parcelas en un gráfico en general, utilizando el par (). La función par (), puede incluir la opción mfrow = c (nfilas , ncols) para crear una matriz de nfilasxncolmunas parcelas que son llenadas de por fila. mfcol = c (nrows , ncols) rellena la matriz por columnas.	attach(mtcars) par(mfrow=c(2,2)) plot(wt,mpg, main="Scatterplot of wt vs. mpg") plot(wt,disp, main="Scatterplot of wtvsdisp") hist(wt, main="Histogram of wt") boxplot(wt, main="Boxplot of wt")
sapply	Se aplica una función a los elementos de una lista y devuelve los resultados en un vector, matriz o una lista.	mat1 <- matrix(rep(seq(4), 4), ncol = 4) mat1.df <- data.frame(mat1) y2 <- sapply(mat1.df, function(x, y) sum(x) + y, y = 5) y2 X1 X2 X3 X4 15 15 15 15
xlab/ ylab	Etiqueta del eje x/ Etiqueta del eje y	x <- c(1:10); y <- x; z <- 10/x plot(x,y, title("Un ejemplo de crear etiquetas", xlab="Valores de X ",ylab="Valores de Y"))

Ejercicio propuesto 1.

Dado los datos de promedios de pesos y la altura de las mujeres americanas entre las edades de 30 – 39. Se pide realizar.

a) Efectué el modelo lineal y modelo cuadrático con conjunto de datos women.R
b) Graficar el ajuste en cada uno de los modelos.
c) Validar el error y grados de libertad

Tarea: Otros conjuntos de datos en R de interés al tema de Regresión

GUIA PRÁCTICA Nº 4

SERIES DE TIEMPO PARA FASES EXPLORATORIAS (ESTUDIO DE CASO: PASAJEROS PROCEDENTES DE VUELOS INTERNACIONALES (AIRPASSENGERS)

Objetivo general

- Estudiar y describir los cambios de una variable con respeto al tiempo a través de un modelo matemático no paramétrico.

Objetivos específicos

- Aplicar sobre una base de datos univariada, herramientas estadísticas y gráficas para el análisis de las series de tiempo y predecir valores futuros.

Requerimientos

Para la realización de esta práctica se necesitaron los siguientes requerimientos:

- Hardware:
  - Procesador con velocidad de 2.1 GHz
  - Memoria RAM de 1 GB.
  - Una PC con Windows o con Linux
- Software
  - Programa R versión 3.2.0

Los contenidos a desarrollar en este tema son los siguientes:

- Gráficos descriptivos simples y agrupados una Serie de Tiempo.
- Componentes a identificar en una Serie de Tiempo: Tendencia y Estacionalidad
- Ajuste y Predicción a corto plazo de una Serie de Tiempo..

Introducción

Definición: Una serie temporal es una sucesión de observaciones de una variable tomadas en varios instantes de tiempo.

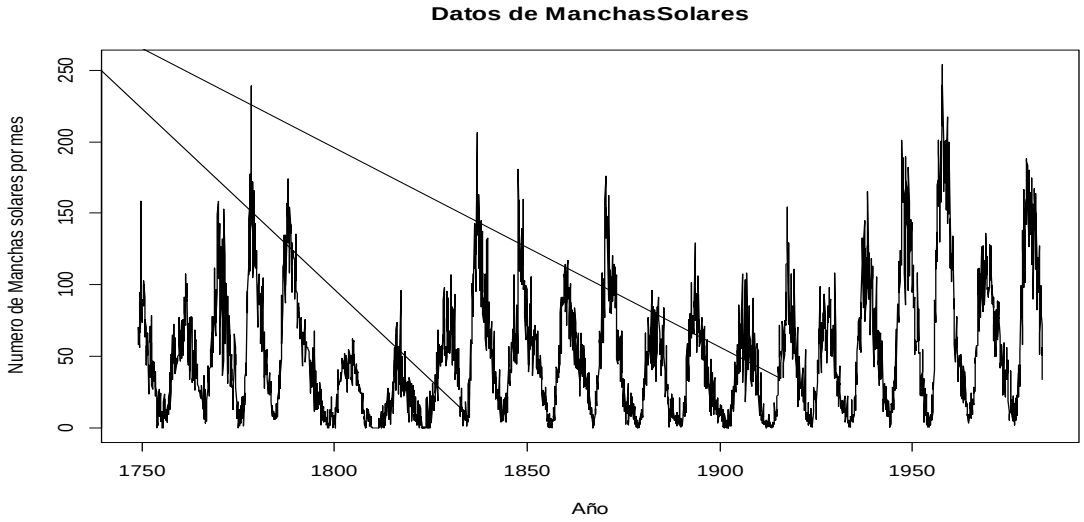
Ejemplos de series temporales podemos encontrarlos en muchos campos de conocimiento:

- Economía: producto interno bruto anual, tasa de inflación, tasa de desempleo, nivel de exportaciones e importaciones, fluctuaciones bursátil, etc.
- Demografía: nacimientos, muertes o morbilidad por fechas, número de turistas ingresando a la región, migraciones, etc.
- Meteorología: temperaturas máximas, medias o mínimas, precipitaciones diarias, velocidad del viento, etc.
- Medio ambiente: concentración media mensual de nitratos en agua, alcalinidad media anual del suelo, emisiones anuales de CO_2, etc.

Representación gráfica de una serie temporal

A menudo, se representa la serie en un gráfico temporal, con el valor de la serie en el eje de ordenadas y los tiempos en el eje de abscisas.

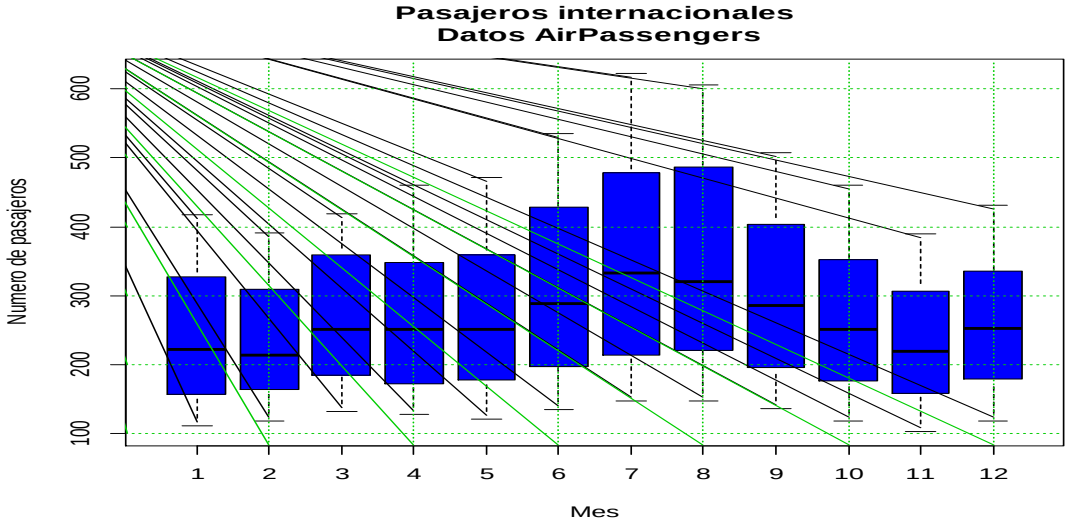
Ejemplo 1. El siguiente gráfico temporal muestra el número de manchas solares por mes para los años 1749 a 2013, observe del mismo en qué años se alcanzan los picos (datos sunspot.R).

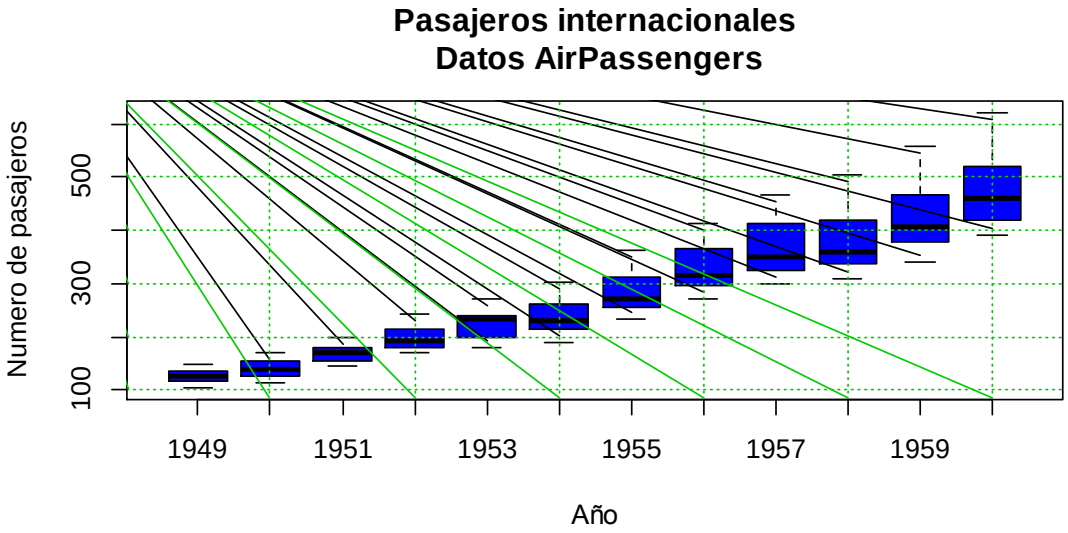


Otros tipos de gráficos temporales

Gráficos por periodos de observación: ***Boxplot;*** sirven para agrupar por período o particiones del tiempo anual (mes, trimestre, etc.) en varios años o por mes.

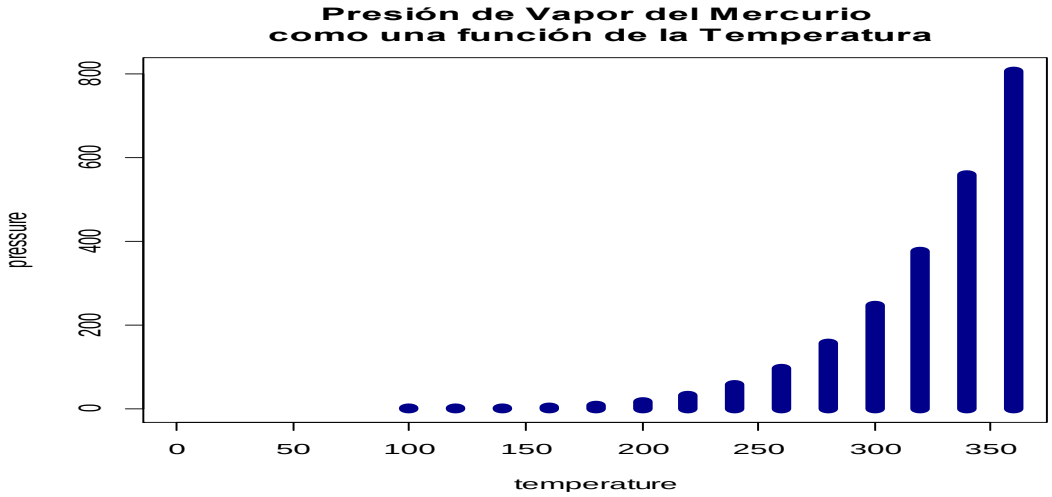
Ejemplo 2. Agrupaciones de 12 años por cada mes para analizar la componente Estacional, y se observa que para julio y agosto se dan las máximas en los ingresos de pasajeros procedentes de vuelos internacionales.





Del gráfico de cajas previo se observa la tendencia positiva a largo plazo del comportamiento de los ingresos de pasajeros al país. También la falta de Estacionariedad en Media y en Varianza, pues esta incrementa con los años.

Ejemplo de Diagrama de Barras. Distribución mensual de la precipitación media en



Clasificación de series temporales

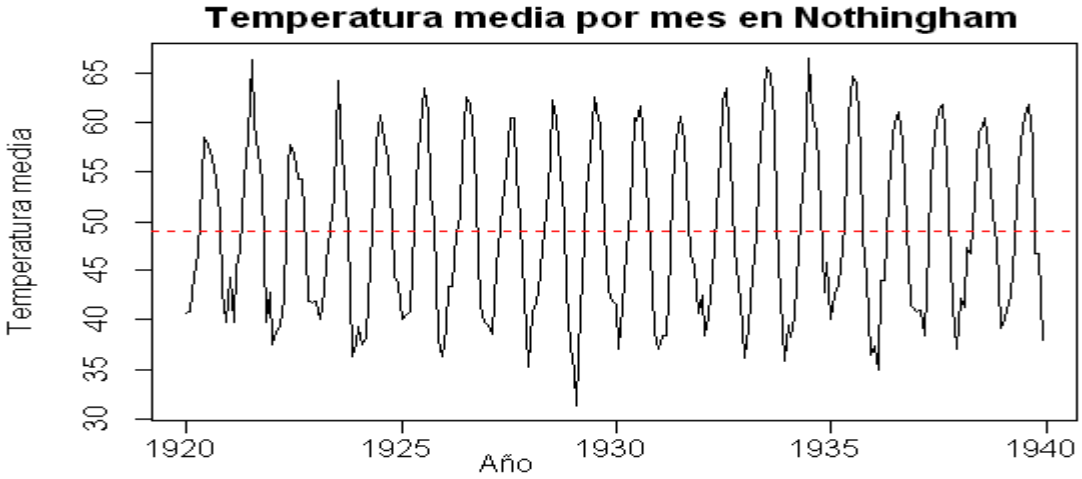
- Una serie es **estacionaria** si la media y la variabilidad se mantienen constantes a lo largo del tiempo.
- Una serie es **no estacionaria si** la media y/o la variabilidad cambian a lo largo del tiempo.

Series no estacionarias pueden mostrar cambios de varianza. Además Series no estacionarias pueden mostrar una **tendencia** (positiva, negativa o a tramos), es decir que la

media crece o baja a lo largo del tiempo. Incluso pueden presentar **efectos estacionales,** es decir que el comportamiento de la serie es parecido en ciertos tiempos periódicos en el tiempo.

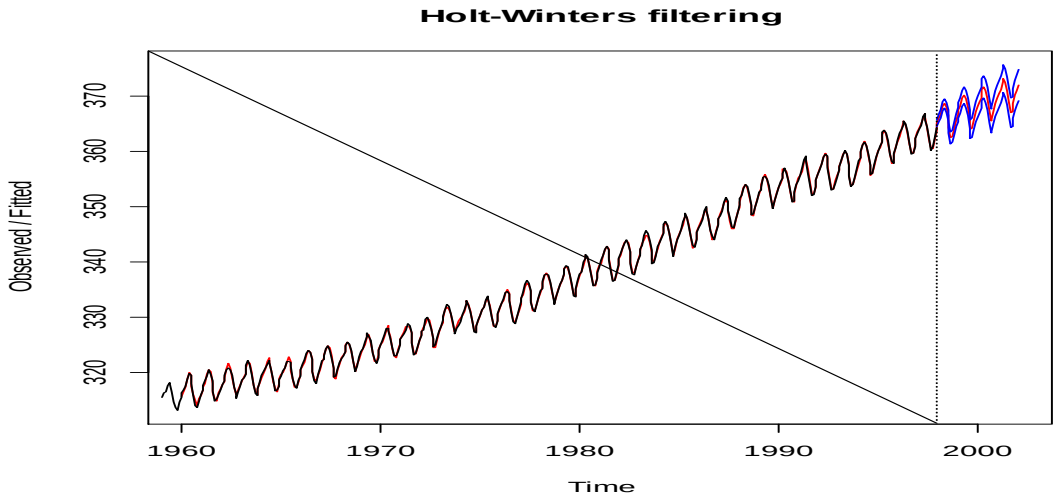
Téngase el cuidado de no confundir Estacional con Estacionaria; refiriéndose la primera a los cambios periódicos anuales de una serie de tiempo y la segunda a la propiedad de estabilidad en media y en varianza de la Serie de Tiempo. Claramente la Serie AirPassengers es No estacionaria ni en media ni en varianza.

Serie estacionaria: Variaciones anuales de la Temperatura media por Mes en Nothingham para el período Enero/1920 a Diciembre/1939.



Serie no estacionaria: El gráfico siguiente de la Serie temporal Emisiones de CO2, muestra además de las predicciones una Tendencia positiva por lo cual No es Estacionaria

Predicción: Predicción a 50 meses de la emisión de gas CO2 (datos co2.R)



Operadores en R.

Operador	Descripción	Ejemplo
Airpassengers	*Muestra número de pasajeros mensuales de una aerolínea de 1949-1960*	AirPassengers
data ()	*Función que muestra la lista de las bases de datos que contiene R*	data(AirPassengers)
frequency	*El número de observaciones por unidad de tiempo.*	ts(1:10, frequency = 4, start = c(1959, 2)) # Segundo trimestre de 1959
HoltWinters	*Produce un gráfico de la serie temporal original junto con los valores ajustados. Opcionalmente, los valores predichos (y sus límites de confianza) también se pueden trazar.*	m <- HoltWinters(co2) p <- predict(m, 50, prediction.interval = TRUE) plot(m, p) ## estacional Holt-Winters (m <- HoltWinters(co2)) plot(m) plot(fitted(m)) (m <- HoltWinters(AirPassengers, seasonal = "mult")) plot(m) ## no estacional Holt-Winters x <- uspop + rnorm(uspop, sd = 5) m <- HoltWinters(x, gamma = FALSE) plot(m)
predict	*Es una funcióngenérica para predicciones de resultados para varias funcionepredictions . La función invoca en particular métodos los cuales dependen en la clase de el primer argumento*	m <- HoltWinters(co2) p <- predict(m, 50, prediction.interval = TRUE) plot(m, p)
ts	*Es una serie temporal y como tal contiene atributos adicionales tales como frecuencia y fechas.* *Series de tiempo. La función ts crea un objeto de clase "ts" (serie de tiempo) a partir*	ts(1:47, frequency = 12, start = c(1959, 2)) Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec 1959 1 2 3 4 5 6 7 8 9 10 11 1960 12 13 14 15 16 17 18 19 20 21 22 23 1961 24 25 26 27 28 29 30 31 32 33 34 35 1962 36 37 38 39 40 41 42 43 44 45 46 47

	de un *vector (serie de tiempo única) o una matriz (serie multivariada).*	

Ejercicio propuesto 1.

En este apartado *AirPassenger*ses una de las bases de datos contenidas en R, la cual describe el ***número de pasajeros mensuales de una aerolínea en los años de 1949 hasta el año 1960.*** Utilizando la BD mencionada se pide realizar lo siguiente:

a) ¿Qué tipo de serie se refleja?
b) Hacer una predicción para los próximos 2 años y 5 años

GUIA PRÁCTICA Nº 5

SERIES DE TIEMPO CON MODELIZACIÓN (ESTUDIO DE CASO: PRODUCTOS DE LA CANASTA BÁSICA NICARAGÜENSE DEL AÑO 2012 AL AÑO 2014)

Objetivo general

- Analizar el comportamiento de un conjunto de datos longitudinales, sobre tres o más variables a través gráficos y modelos matemáticos que permitan las proyecciones de una variable a corto plazo.

Objetivos específicos

- Implementar modelos y técnicas descriptivas sobre conjuntos de datos multivariados utilizando el lenguaje y entorno R, que permita apoyar a la toma de decisiones desde los resultados obtenidos.

Requerimientos

Para la realización de la práctica se necesitaron los siguientes requerimientos:

- Hardware:
  - Procesador con velocidad de 2.1 GHz
  - Memoria RAM de 1 GB.
  - Una PC con Windows o con Linux.
- Software
  - Programa R versión 3.2.0, o posterior.

Operadores en R.

Operador	Descripción	Ejemplo
library(MTS)	(MTS) es un paquete general para el análisis de series de tiempo lineal multivariable y la estimación de modelos de volatilidad multivariante. También se ocupa de los modelos de factores, modelos de factores restringidos, análisis de componentes principales asintóticas de uso común en las finanzas y econometría, y el análisis de componentes principales volatilidad.	*data("mts-examples", package="MTS");* *# La siguiente función* *# carga el paquete MTS* *# en la consola de R.* *library(MTS)*
paste	Concatena vectores luego de convertirlos a caracteres (este ejemplo lo usaremos para nombrar las dimensiones de la siguiente matriz)	*filas <- paste("r", 1:2, sep="")* *columnas <- paste("c", 1:3, sep="")*
matrix	Crea una matriz a partir del conjunto dado de valores	*datos <- c(1:3,11:13)* *mdat <- matrix(datos, nrow = 2, ncol = 3, byrow = TRUE, dimnames = list(filas,columnas))* *mdat*
MTSplot()	Proporciona gráficos de secuencia (sobre el tiempo) de vectores de serie de tiempo: MTSplot(datos, caltime = NULL) *# caltime define la fecha calendario; campo por*	*datos = rnorm(1500)* *xt = matrix(datos,500,3)* *MTSplot(xt)* *zt = log(qgdp[,3:5]) ; xt =*

	defecto es Nulo, con índice en el tiempo.	*diffM(zt); MTSplot(xt)*

Ejercicio propuesto 1.

Dado como datos los nombres y los precios de cada uno de los 36 productos de la canasta básica de Nicaragua, desde el año 2012 al año 2014. Se pide realizar.

a) Prepare los datos longitudinales en un conjunto multivariado de datos.
b) Realice la predicción a un año de 3 productos de la canasta básica de Nicaragua.
c) De los 3 productos, los precios ¿son estacionarios o no estacionarios?

Fuente: http://www.inide.gob.ni/

Tarea 1: Construya gráficos de cajas que permita comparar los precios de tres productos relacionados en su venta y consumo, de la canasta básica de Nicaragua.

Tarea 2: Busque bases de datos sobre ventas, consumo, tasas e índices que permitan practicar las técnicas aquí descritas (recargas telefónicas, ventas de supermercados, ventas de restaurantes, etc.).

GUIA PRÁCTICA Nº 6

ANÁLISIS DE CONGLOMERADOS, CLUSTER ANALYSIS O CLUSTERING (ESTUDIO DE CASO: CAUSAS DE MORTALIDAD EN EL DPTO. DE LEÓN, NICARAGUA DEL AÑO 2010 AL I SEMESTRE DEL AÑO 2015)

Objetivo general

- Determinar en conjunto de datos multivariados, si distintos ítems, instrumentos, métodos o personas pueden agruparse para su consecuente análisis y relación con otras variables en grupos más reducidos o compactos.

Objetivos específicos

- Determinar si existe asociación entre variables, utilizando el método K-means y herramientas gráficas para el análisis de clusters o asociación.

Requerimientos

Para la realización de la práctica se necesitan los siguientes requerimientos:

- Hardware:
  - Procesador con velocidad de 2.1 GHz.
  - Memoria RAM de 1 GB.
  - Una PC con Windows o con Linux

- Software
  - Programa R versión 3.2.0 o posterior

- Bibliografía:
  - http://www.ecured.cu/Clustering

Los contenidos a desarrollar en este tema son los siguientes:

- Definición
- Algoritmo de Clustering: Simple K-Means
- Gráficas

Introducción

Definición: Clustering. También conocido como agrupamiento, es una de las técnicas de Minería de Datos, el proceso consiste en la división de los datos en grupos de objetos similares. Cuando se representan la información obtenida a través de clusters se pierden algunos detalles de los datos, pero a la vez se simplifica dicha información. Es una técnica en la que el aprendizaje realizado es no supervisado.

Desde un punto de vista práctico. El clustering juega un papel muy importante en aplicaciones de Minería de Datos, tales como exploración de datos científicos, recuperación

de la información y minería de texto, aplicaciones sobre bases de datos espaciales (tales como GIS o datos procedentes de astronomía), aplicaciones Web, marketing, diagnóstico médico, análisis de ADN en biología computacional y muchas otras.

De forma general, las técnicas de Clustering son las que utilizando algoritmos matemáticos se encargan de agrupar objetos. Usando la información que brindan las variables que pertenecen a cada objeto se mide la similitud entre los mismos, y una vez hecho esto se colocan en clases que son muy similares internamente (entre los miembros de la misma clase) y a la vez diferente entre los miembros de las diferentes clases.

Algoritmos de Clustering

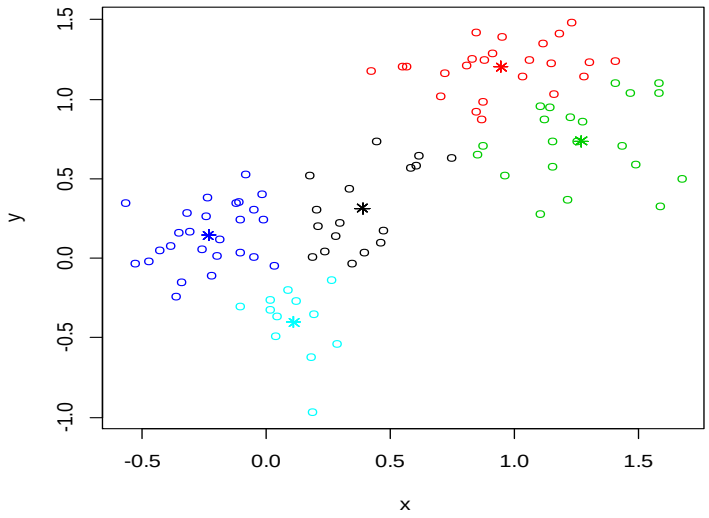
Simple K-Means

Este algoritmo debe definir el número de clusters que se desean obtener, así se convierte en un algoritmo voraz para particionar. Los pasos básicos para aplicar el algoritmo son muy simples. Primeramente se determina la cantidad de clusters en los que se quiere agrupar la información, en este caso las simulaciones. Luego se asume de forma aleatoria los centros por cada clusters. Una vez encontrados los primeros centroides el algoritmo hará los tres pasos siguientes:

1. Determina las coordenadas del centroide.
2. Determina la distancia de cada objeto a los centroides.
3. Agrupa los objetos basados en la menor distancia.

Finalmente quedarán agrupados por clusters, los grupos de simulaciones según la cantidad de clusters que el investigador definió en el momento de ejecutar el algoritmo

Ejemplo de gráfico bidimensional utilizando Cluster



Operadores en R.

Operador	Descripción	Ejemplo
iris	Muestra en conjunto de datos Iris, ejemplo biológico multivariado clásico que muestra el número de especies y las mediciones longitud y ancho de sépalo o pétalo de cada planta. Edgar Anderson's Iris Data: Este famoso conjunto de datos del iris da las medidas en centímetros de las variables longitud sépalo y pétalo de anchura y longitud y anchura, respectivamente, por 50 flores de cada una de las 3 especies de iris. Las especies son Iris setosa, versicolor, y virginica.	str(iris)
data ()	Función que muestra la lista de las bases de datos que contiene R	data(iris)

kmeans ()	Realiza conglomerados kmeans dentro de una matriz. Parámetros de entrada básicos son Matriz de datos y un número entero para el número de medias (centroides).	# un ejemplobidimensional
		x <- rbind(matrix(rnorm(100, sd = 0.3), ncol = 2), matrix(rnorm(100, mean = 1, sd = 0.3), ncol = 2))colnames(x) <- c("x", "y")
		(cl <- kmeans(x, 2))
		plot(x, col = cl$cluster)
		points(cl$centers, col = 1:2, pch = 8, cex = 2)

Plot()	Traza una gráfica. X-Y	plot(cars) lines(lowess(cars))

points()	Es una función genérica para dibujar una secuencia de puntos en las coordenadas especificadas.	# para 200 datos normales
		# generados (m=0, s=1)
		plot (-4:4, -4:4, type = "n") # setting up coord. System
		points(rnorm(200), rnorm(200), col = "red")
		points(rnorm(100)/2, rnorm(100)/2, col = "blue", cex = 1.5)

table	Utiliza los factores de clasificación cruzada para construir una tabla de contingencia de los recuentos en cada combinación de niveles de los factores.	table(rpois(100, 5))

Ejercicio propuesto 1:

En este apartado de conglomerados k-means de datos, **iris** es una de las bases de datos contenida en R, la cual describe un conjunto de datos del iris ,da las medidas en centímetros de las variables longitud sépalo y pétalo de anchura y longitud y anchura , respectivamente, por 50 flores de cada una de las 3 especies de iris. Las especies son Iris setosa, versicolor , y virginica .

.A) ¿Cuántos conjuntos de conglomerados refleja?

Ejercicio propuesto 2:

Dado los datos de: Las causas de mortalidad en el departamento de León, Nicaragua de los años 2010 al primer semestre del año 2015.

Se pide:

a) Importar la base de datos llamada "rmortalidadfecha.csv"
b) Genere tablas de frecuencia y gráficos para cada variable categórica.
c) ¿Quiénes mueren más en el departamento de León Hombres o Mujeres?
d) ¿Qué día muere más gente?
e) ¿Qué edades tiene el mayor índice de muertes en el Dpto. de León?
f) ¿Cuál es el mayor índice de causas de muerte en el dpto. de León?
g) ¿Qué Municipios del Dpto. de León tienen mayor cantidad de defunciones?
h) Analizar los resultados.

 # Tarea:

- ✓ Genere un diagrama de barras combinado que muestre Sexo con Municipio.
- ✓ Genere una pirámide poblacional de Sexo por Edad.

18) CONCLUSIONES

Con este trabajo podemos concluir que:

La Minería de Datos, es la aplicación de unos algoritmos provenientes de diferentes ramas del conocimiento, a través de los cuales se pretende obtener un modelo que permita por medio de su aplicación al conjunto total de los datos, la obtención de los resultados en forma de reglas, patrones o asociaciones, para luego ser interpretados y aplicarlos al estudio, obteniendo así beneficios referente a la toma de decisiones.

La Minería de Datos se relaciona con varias áreas de estudio comprendidas en la formación de un ingeniero en sistemas, tales como: estadística, bases de datos, algoritmos, procesos aleatorios y sistemas de información, entre otros; pero que resulta hasta la fecha un término poco explorado en el contenido de diversas materias que hacen parte de la formación en el Dpto. de computación. Por esto, esta tesis puede considerarse como una aproximación para realizar un estudio de la teoría y compararlo con la práctica, para así poder comenzar a trabajar un poco más en este tema, pensando en que este trabajo sea el punto de partida de futuras investigaciones.

La realización de esta asignatura vendría a beneficiar al departamento de computación en especial a los estudiantes de la carrera de Ingeniería de Sistemas de Información en la aplicación de Minería de Datos con manejos de Base de Datos en donde el Ingeniero pueda poner en práctica sus habilidades y conocimientos, como un aporte al descubrimiento de nuevos conocimientos o información y como apoyo a la toma de decisiones.

19) RECOMENDACIONES

- Al docente que imparta esta asignatura, realizar los ajustes estadísticos que crea convenientes a los modelos utilizados en las guías prácticas.

- Tanto los temas teóricos como los enunciados de las prácticas pueden someterse a revisiones con el paso del tiempo, para mantener actualizada esta asignatura.

- Realizar prácticas en las que se tomen en cuenta otras funcionalidades de la Minería de Datos que no se lograron abarcar en este documento como; minería de texto, Almacenes de Datos, OLTP y OLAP e inteligencia de negocios (Business Intelligence).

20) GLOSARIO

- **Análisis exploratorio de datos:** Uso de técnicas estadísticas tanto gráficas como descriptivas para aprender acerca de la estructura de un conjunto de datos.
- **Atributos**: es una especificación que define una propiedad de un Objeto, elemento o archivo. También puede referirse o establecer el valor específico para una instancia determinada de los mismos.
- **Árbol de decisión:** Estructura en forma de árbol que representa un conjunto de decisiones. Estas decisiones generan reglas para la clasificación de un conjunto de datos.
- **Base de datos multidimensional:** Base de datos diseñada para procesamiento analítico on-line (OLAP). Estructurada como un hipercubo con un eje por dimensión.
- **CART:** (Árboles de clasificación y regresión) Una técnica de árbol de decisión usada para la clasificación de un conjunto da datos. Provee un conjunto de reglas que se pueden aplicar a un nuevo (sin clasificar) conjunto de datos para predecir cuáles registros darán un cierto resultado. Segmenta un conjunto de datos creando 2 divisiones. Requiere menos preparación de datos que CHAID.
- **CHAID** Detección de interacción automática de Chi cuadrado: Una técnica de árbol de decisión usada para la clasificación de un conjunto da datos. Provee un conjunto de reglas que se pueden aplicar a un nuevo (sin clasificar) conjunto de datos para predecir cuáles registros darán un cierto resultado. Segmenta un conjunto de datos utilizando tests de chi cuadrado para crear múltiples divisiones. Antecede, y requiere más preparación de datos, que CART.
- **Clasificación:** Proceso de dividir un conjunto de datos en grupos mutuamente excluyentes de tal manera que cada miembro de un grupo esté lo "más cercano "posible a otro, y grupos diferentes estén lo "más lejos" posible uno del otro, donde la distancia está medida con respecto a variable(s) específica(s) las cuales se están tratando de predecir. Por ejemplo, un problema típico de clasificación es el de dividir una base de datos de compañías en grupos que son lo más homogéneos posibles con respecto a variables como "posibilidades de crédito "con valores tales como "Bueno" y "Malo".
- **Clustering (agrupamiento):** Proceso de dividir un conjunto de datos en grupos mutuamente excluyentes de tal manera que cada miembro de un grupo esté lo" más cercano" posible a otro, y grupos diferentes estén lo "más lejos" posible uno del otro, donde la distancia está medida con respecto a todas las variables disponibles.
- **Cuartiles:** son los tres valores que dividen al conjunto de datos ordenados en cuatro partes porcentualmente iguales
- **Data cleansing:** Proceso de asegurar que todos los valores en un conjunto de datos sean consistentes y correctamente registrados.
- **Data Mining:** La extracción de información predecible escondida en grandes bases de datos.

- **Data Warehouse:** Sistema para el almacenamiento y distribución de cantidades masivas de datos.
- **Datos anormales:** Datos que resultan de errores (por ej.: errores en el tipeado durante la carga) o que representan eventos inusuales.
- **Dimensión:** En una base de datos relacional o plana, cada campo en un registro representa una dimensión. En una base de datos multidimensional, una dimensión es un conjunto de entidades similares; por ej.: una base de datos multidimensional de ventas podría incluir las dimensiones Producto, Tiempo y Ciudad.
- **Histogramas:** En estadística, un **histograma** es una representación gráfica de una variable en forma de barras, donde la superficie de cada barra es proporcional a la frecuencia de los valores representados, ya sea en forma diferencial o acumulada. Sirven para obtener una "primera vista" general, o panorama, de la distribución de la población, o la muestra, respecto a una característica, cuantitativa y continua, de la misma y que es de interés para el observador (como la longitud o la masa).
- **KDD:** (Knowledge Discovery in Databases) es un campo de las ciencias de la computación referido al proceso que intenta descubrir patrones en grandes volúmenes de conjuntos de datos.
- **Mediana:** representa el valor de la variable de posición central en un conjunto de datos ordenados
- **Modelo analítico:** Una estructura y proceso para analizar un conjunto de datos. Por ejemplo, un árbol de decisión es un modelo para la clasificación de un conjunto de datos.
- **Modelo lineal:** Un modelo analítico que asume relaciones lineales entre una variable seleccionada (dependiente) y sus predictores (variables independientes).
- **Modelo no lineal:** Un modelo analítico que no asume una relación lineal en los coeficientes de las variables que son estudiadas.
- **Modelo predictivo:** Estructura y proceso para predecir valores de variables especificadas en un conjunto de datos.
- **Navegación de datos:** Proceso de visualizar diferentes dimensiones, "fetas" y niveles de una base de datos multidimensional.
- **OLAP Procesamiento analítico on-line (On Line Analitic prossesing):** Se refiere a aplicaciones de bases de datos orientadas a array que permite a los usuarios ver, navegar, manipular y analizar bases de datos multidimensionales.
- **Outlier:** Un item de datos cuyo valor cae fuera de los límites que encierran a la mayoría del resto de los valores correspondientes de la muestra. Puede indicar datos anormales. Deberían ser examinados detenidamente; pueden dar importante información.
- **Rango intercuartil:** IQR es una estimación estadística de la dispersión de una distribución de datos. Consiste en la diferencia entre el *tercer* y el *primer* cuartil.
- **Reconocimiento de patrones (Pattern Recognition):** Se trata de un grupo de técnicas orientadas a evaluar la similitud y las diferencias entre señales. Se involucran en esto a varios tipos de pre-procesamiento tales como la transformada de Fourier.
- **Redes Neuronales (Neural Networks):** Grupo de unidades no-lineales interconectadas y organizadas por capas. Estas pueden ser funciones matemáticas y números almacenados en computadoras digitales, pero pueden ser elaboradas también mediante

dispositivos analógicos como los transistores a efecto de campo (FET).A pesar del incremento en velocidad y de la escala de integración en los semiconductores, la mejor contribución de las redes neuronales tendrá que esperar por computadoras más rápidas, masivas y paralelas.

- **Regresión lineal:** Técnica estadística utilizada para encontrar la mejor relación lineal que encaja entre una variable seleccionada (dependiente) y sus predicados (variables independientes).
- **Regresión logística:** Una regresión lineal que predice las proporciones de una variable seleccionada categórica, tal como Tipo de Consumidor, en una población.

21) BIBLIOGRAFÍA

Libros consultados:

- Gráficos Estadísticos con R, Juan Carlos Correa y Nelfi González , Posgrado en Estadística-Universidad Nacional-Sede Medellín
- Introducción a Minería de Datos. José Hernández Orallo, Mª José Ramírez Quintana, Cesar Ferri Ramírez.
- R para Principiantes, traducido por Jorge A. Ahumada, RCUH/ University of Hawaii & USGS/ National Wildlife Health Center (Emmanuel Paradis Institut des Sciences de l'E´volution Universit Montpellier II, F-34095 Montpellier cdex 05. France)

Páginas web consultadas:

- http://www.dataprix.com/531-selecci-n-exploraci-n-destino-dep-sito
- www.sinnexus.com/business_intelligence/datamining.aspx
- www.wikipedia com
- www.wikilibros.com