Caracterización e Interpretación Automática de Descripciones Conceptuales en Dominios poco Estructurados usando Variables Numéricas

Fernando Vázquez y Karina Gibert

25 de Marzo, 2002

Resumen

La investigación que se presenta en este proyecto, tiene como objetivo fundamental: establecer una metodología para la generación automática de descripciones conceptuales, a través de un sistema de reglas difusas usando atributos cuantitativos que permitan caracterizar las diversas situaciones que se presentan en procesos enmarcados en dominios poco estructurados. La metodología se basa en una combinación de herramientas y técnicas estadísticas (ej. boxplot múltiple [1], análisis descriptivo) y métodos de inteligencia artificial (ej. aprendizaje inductivo, sistemas basados en conocimiento) de tal forma que la naturaleza de los datos se conserva, evitando transformaciones previas sobre los atributos. Así, tanto la información cualitativa como cuantitativa se induce a partir de los datos.

La metodología si inicia con un conjunto de individuos u objetos $\mathcal{I} = \{i_1, \ldots, i_n\}$, una clasificación de referencia y un conjunto de individuos nuevos de un universo de discurso, para identificar descripciones conceptuales e interpretaciones al predecir la situación típica de un nuevo individuo del dominio o proceso bajo estudio.

A partir de la partición de referencia, descrita de forma extensional, se propone un modelo conceptual que determine los atributos relevantes involucrados, describa las diversas situaciones en dicho proceso y genere automáticamente interpretaciones conceptuales de éstas. Información relevante que constituye un excelente apoyo a la toma de decisiones que involucra la gestión y/o toma de decisiones en los diversos procesos.

Para aplicar la metodología de forma iterada sobre todos los atributos cuantitativos disponibles en el dominio del caso de estudio, observar su funcionamiento y registrar los resultados obtenidos de forma automatizada, se ha implementado una herramienta que denominamos CIADEC (Caracterización e Interpretación Automática de Descripciones Conceptuales en *dominios poco estructurados* usando atributos cuantitativos), la cual ha estado evolucionando de acuerdo a los objetivos del anteproyecto de tesis.

Se ha demostrado con esta metodología que la robustez de la predicción de clases, depende de la partición de referencia que se tenga del dominio de estudio.

Se hace énfasis en la importancia de generar automáticamente la interpretación de resultados en los sistemas de KDD [2] que [FPSSU96] describe en 1996. Por lo que, parte de este trabajo está relacionado con la construcción de sistemas reales de KDD que cubran todas las fases de este proceso, incluida la generación automática de interpretaciones.

Palabras Clave: Descubrimiento del Conocimiento, Aprendizaje Automático, Sis-

[1] Herramienta gráfica estadística que distribuye objetos respecto a clases.

[2] Del Inglés Knowledge Discovery in Databases.

temas de Clasificación Basados en Reglas, Dominios Poco Estructurados, Razonamiento Inductivo Difuso.

Índice general

Parte I

PROYECTO DE TESIS

Capítulo 1

Introducción

La comprensión de la naturaleza de los métodos que utilizamos los seres humanos para clasificar datos o conocimientos, es un problema de gran interés teórico y práctico para todas las ciencias cognitivas; ya que la acción de clasificar es una de las etapas iniciales de los procesos de adquisición de conocimiento en cualquier campo científico.

Teóricamente, la comprensión del concepto "clasificación" contribuirá a entender mejor lo que implica el "aprendizaje". De hecho, es difícil concebir una forma de aprendizaje sin haber pasado antes por una forma previa de clasificación.

Por otro lado, en la práctica, el desarrollo de sistemas automáticos de clasificación es, hoy por hoy, una necesidad imperiosa de la sociedad actual ya que en muchos procesos humanos, la cantidad de datos que se generan es tan grande, que resulta muy difícil manipularlos y transmitirlos sin el auxilio de esta clase de sistemas.

La clasificación puede desarrollarse en dos grandes áreas:

- A partir de una clasificación de referencia de un universo de discurso, definir reglas para decidir la clase a la que pertenece cada elemento del universo.
- Dado un universo de discurso, construir una clasificación adecuada del mismo.

Los esfuerzos de investigación en *aprendizaje automático (machine learning)* se han centrado, principalmente, en la primera área. De hecho, la mayoría de los sistemas expertos de la primera generación (como MYCIN [Sho76], INTERNIST, PLANT/ds [MS82]) son, en la práctica, sistemas clasificadores. Esta clase de sistemas utilizan un conjunto de reglas implementadas como árboles de decisiones para determinar la clase a la que pertenece una entrada dada.

En la aproximación clásica a este problema, el experto humano es el responsable de decidir cuáles son los atributos "relevantes"para la formulación de las reglas de clasificación. Cuando se procede de esta forma, el diseñador del sistema requiere información que el experto no está preparado para proporcionar debido, fundamentalmente, a la falta de familiaridad con los términos que se utilizan en el sistema informático. Esto provoca graves problemas de comunicación al tratarse de personas que tienen una formación muy diferente, por lo que la extracción del conocimiento se hace difícil de superar y consume mucho tiempo.

Por lo tanto, la clasificación de ejemplos se presenta como una herramienta alternativa posible para la extracción del conocimiento de las descripciones que los expertos podrán dar de sus dominios.

Esta es la razón de que hayan surgido diversas metodologías que permiten el análisis de la información con vistas a crear agrupaciones de observaciones para su posterior caracterización e interpretación.

Un enfoque diferente es el de la Inteligencia Artificial ya que, para reducir el coste de la adquisición de conocimiento, se ha decidido por el uso de técnicas de aprendizaje inductivo para la automatización de procesos. De esta manera, se puede, a partir de una colección de ejemplos propuestos por el experto o extraídas directamente del dominio y, de estas técnicas, descubrir el conocimiento oculto en los datos para utilizarlo en la construcción de bases de conocimiento, disminuyendo en este sentido su coste. Este mecanismo parece más viable ya que se ha observado que los expertos tienen más facilidad para dar ejemplos de instancias de su dominio que para expresar los conceptos o reglas que les permiten identificarlas.

En el caso del dominio donde la estructura del conocimiento esté claramente asentada y exista una manera de discernir entre las diferentes categorías que lo componen, esta metodología sería clara y provechosa a la hora de construir bases de conocimiento para sistemas basados en el conocimiento, disminuyendo la interacción experto-diseñador del sistema.

Todos los problemas de adquisición del conocimiento mencionados se agravan si el dominio sobre el que se esta trabajando es un *Dominio Poco Estructurado* (Ill-Structured Domains, ISD) denominado así por [Gib94]. Estos dominios se caracterizan por:

- No existir consenso entre los expertos para la definición de todos los conceptos y objetos que los componen y las relaciones entre estos.
- Dificultad del área de conocimiento en concreto, por la falta de una metodología de investigación aceptada por todos los expertos, o por un continuo cambio en el conocimiento o en su extensión.
- Los atributos que describen los objetos pueden ser cuantitativos o cualitativos.
- Los expertos suelen disponer de grandes cantidades de conocimiento implícito, además de manejar diversos grados de especificidad, lo que hace a este conocimiento parcial y no homogéneo.

De esta forma la alternativa que parece más prometedora para resolver estas limitaciones es liberar al experto de este trabajo, mediante el desarrollo de técnicas que a partir de la evidencia empírica en forma de ejemplos, identifiquen los atributos más relevantes y formulen reglas que expresen las regularidades existentes en los datos.

En este trabajo se propone una metodología automatizada para la caracterización e interpretación de descripciones conceptuales en ISD, inspirada en el *boxplot múltiple* y la cual se describe en el capítulo §5. Esta metodología parte de un conjunto de objetos previamente clasificados por algún tipo de clasificador, toma en cuenta la distribución de los objetos en los valores de los atributos y, para cada atributo en un sistema de intervalos de longitud variable, la proporción de objetos asignados a cada clase, generando un sistema de reglas para posteriormente caracterizar e interpretar en forma automática las descripciones conceptuales de la situación a la que pertenece un nuevo objeto en el universo de discurso.

La estructura de este documento consta de dos partes:

Primera Parte: Proyecto de Tesis Se inicia, en el capítulo §2 con la motivación que sirvió de base a la investigación que aquí se propone; en el capítulo §3 se formaliza el problema de tesis y se plantean los objetivos a alcanzar en este trabajo; en el capítulo §4 se describe el estado del arte, que permitirá contextualizar el tema de tesis; en el capítulo §5 se describe la propuesta metodológica; en el capítulo §6 se presenta una aplicación a un caso de estudio; en el capítulo §7 se presentan las conclusiones importantes que hasta el momento se han obtenido en este trabajo; en el capítulo §8 se establece la agenda de tesis para el trabajo futuro y finalmente, en el capítulo §9 las publicaciones realizadas con relación al presente documento.

Segunda Parte: CIADEC En esta parte se da una visión general del sistema *CIADEC* que implementa la metodología propuesta en este proyecto de tesis. Aquí se describen, de forma general, la arquitectura, funcionalidades e implementación de *CIADEC*.

Apéndices: En el apéndice A, se explica el proceso de búsqueda bibliográfica utilizado para dar soporte a este trabajo. Finalmente, en el apéndice B, se muestran resultados complementarios y algunos gráficos obtenidos durante el desarrollo del caso de estudio descrito en la primera parte de este documento.

Capítulo 2

Motivación

En las últimas décadas, el crecimiento explosivo de los avances científicos y tecnológicos, la automatización de procesos industriales y comerciales, los avances en tecnología de almacenamiento de datos y sistemas administradores de datos han generado sistemas complejos que han rebasado nuestra capacidad para analizarlos e interpretarlos, creando la necesidad de una nueva generación de métodos, técnicas y herramientas con la capacidad para asistir inteligente y automáticamente a los seres humanos en el análisis de estas bases de datos para extraer conocimiento útil que represente los dominios del mundo real.

Descubrir la estructura o extraer conocimiento de it dominios poco estructurados (ISD) no es tarea fácil y requerimos de combinar técnicas y herramientas de diversos campos, en nuestro caso, de Estadística (análisis multivariante de datos, clustering, etc.), Inteligencia Artificial (aprendizaje automático, por ejemplo, sistemas basados en el conocimiento), Sistemas de Información (análisis, diseño, implementación, etc.), Lógica Difusa (razonamiento difuso, modelo de Mamdani, etc.), Visualización de Datos, etc. para construir *Sistemas Híbridos* que nos permitan encontrar e interpretar patrones especiales (o conceptos) en las bases de datos, para extraer conocimiento útil que represente estos dominios y que den mejor desempeño que las técnicas tradicionales o las aproximaciones clásicas de sistemas basados en conocimiento.

Por lo anterior, el interés de este trabajo es presentar una propuesta metodológica híbrida que combine herramientas y técnicas de Estadística, Inteligencia Artificial y Lógica Difusa en forma cooperativa, tal que, a partir de los atributos cuantitativos de los datos que definen los objetos de un cierto dominio (por ejemplo, un proceso industrial, medio ambiental o de cualquier otra naturaleza), podamos identificar cuales son las situaciones características (clases) que se pueden encontrar en él y enseguida analizar estas clases resultantes y estudiar su significado. Esta tarea, habitualmente es responsabilidad del técnico (en este caso del estadístico) que debe usar sus conocimientos para poner de manifiesto las principales diferencias entre clases y posteriormente en colaboración con el experto en la materia, darles interpretación.

Una vez identificadas e interpretadas estas situaciones típicas por el usuario, los conocimientos generados pueden ser usados posteriormente como herramientas de apoyo al proceso de administración y/o toma de decisiones. Incluso se ha llegado a decir que la *validación de una clasificación* (problema abierto) consiste, precisamente, en *probar* que las clases tienen *sentido* o *utilidad* [Alu96]. En este sentido, la propuesta pretende llegar un poco más lejos y establecer las bases de una metodología que facilite

la generación automática de caracterizaciones e interpretaciones conceptuales en estos dominios complejos.

En nuestro caso de estudio de una planta depuradora de la costa Catalana considerado en el capítulo §6, la metodología se ha aplicado al proceso de tratamiento de aguas residuales.

A partir de una base de datos de la planta depuradora y una partición de referencia de estos datos, se genera un sistema de reglas difusas usando los atributos cuantitativos (17) recomendados por el experto; este sistema permitirá para un nuevo objeto (día), predecir la clase (situación típica de la planta) que le corresponde y generar las caracterizaciones e interpretaciones de las descripciones conceptuales correspondientes a esa clase.

Cuando la planta depuradora no funciona bajo condiciones normales, se deben tomar decisiones para modificar algunos parámetros del proceso de depuración y re-establecer lo antes posible la normalidad. Razón por la cual, es importante contar con un sistema automatizado, que proporcione información relevante sobre la situación que la planta tiene en un momento específico. En nuestro caso, la investigación esta orientada a hacer aportaciones en ese sentido.

La propuesta metodológica tiene además, una amplia gama de aplicaciones. A continuación se mencionan sólo algunas de ellas.

- En las Instituciones de crédito, con el fin de clasificar el comportamiento de sus clientes optimizando su manejo y reduciendo al mínimo el riesgo de caer en cartera vencida. También se pueden clasificar solicitudes de crédito para decidir el otorgamiento o la negación del mismo.

- En las Universidades, con el objeto de clasificar a los alumnos. Este es un problema importante desde el punto de vista pedagógico, ya que es muy difícil determinar las causas por las cuales un estudiante es deficiente, regular o bueno. Puede utilizarse para identificar los atributos que determinan el rendimiento de los estudiantes y para generar reglas que permitan prever su comportamiento futuro.

- En Medicina, para identificar las características que ocasionan diversa enfermedades (mentales, cardiovasculares, gastrointestinales . . .), y para generar reglas que permitan determinar, *a priori*, si una persona está propensa a padecerlas.

Capítulo 3

Formulación del Problema y Objetivos de Tesis

En este capítulo se hará una descripción formal del problema y al final se mencionaran los objetivos de este proyecto de tesis.

3.1 Formulación del Problema

Sea $\mathcal{I} = \{i_1, \ldots, i_n\}$ un conjunto de individuos u objetos de un universo de discurso, denominado "conjunto de entrenamiento", que está descrito por una serie de atributos cualitativos y/o cuantitativos $X_1 \ldots X_K$, cuyos valores para cada uno de los individuos $i \in \mathcal{I}$ se representan por una matriz rectangular $\mathcal{X}$ de dimensión (n, K), como se muestra en la Tabla 3.1:

$$\mathcal{X}' = \begin{pmatrix} x_{11} & x_{12} & \ldots & x_{1k-1} & x_{1k} \\ x_{21} & x_{22} & \ldots & x_{2k-1} & x_{2k} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{n-11} & x_{n-12} & \ldots & x_{n-1k-1} & x_{n-1k} \\ x_{n1} & x_{n2} & \ldots & x_{nk-1} & x_{nk} \end{pmatrix}$$

Tabla 3.1: *Matriz de datos* $\mathcal{X}$

En la matriz $\mathcal{X}$, se tiene que x_{ik} con $1 \leq i \leq n$ y $1 \leq k \leq K$, es el valor que, el individuo i–ésimo toma para el k–ésimo atributo. Es decir, que las filas de la matriz de datos $\mathcal{X}$ contienen información relativa a las características de los individuos, la cual se puede representar como un vector de atributos de la forma:

$$x_i = \begin{pmatrix} x_{i1} & x_{i2} & \ldots & \ldots & x_{ik} \end{pmatrix}$$

y las columnas hacen referencia a los K atributos X_k.

Además, se tiene una partición de referencia de los elementos de $\mathcal{I}$, la que se denota por $\mathcal{P} = \{C01, C02, \ldots, C\xi\}$, donde la *card*$(\mathcal{P}) = \xi$ y $\mathcal{P}$ satisface las siguientes propiedades:

- $C \subseteq \mathcal{I}$

- $\cup_{C \in \mathcal{P}} C = \mathcal{I}$
- $C \cap C' = \phi,\ \forall\ C,\ C'\ \in \mathcal{P}$

Un conjunto de individuos nuevos previamente clasificados representados por P_0 y denominado "conjunto de prueba", dicho conjunto lo podemos representar en forma matricial como se muestra en la Tabla 3.2.

$$\mathcal{X} = \left(\begin{array}{ccccc|c} x^0_{11} & x^0_{12} & \dots & x^0_{1k-1} & x^0_{1k} & C_r \\ x^0_{21} & x^0_{22} & \dots & x^0_{2k-1} & x^0_{2k} & C_s \\ \vdots & \vdots & \vdots & \vdots & \vdots & \dots \\ x^0_{m-11} & x^0_{m-12} & \dots & x^0_{m-1k-1} & x^0_{m-1k} & C_t \\ x^0_{m1} & x^0_{m2} & \dots & x^0_{mk-1} & x^0_{mk} & C_u \end{array}\right)$$

Tabla 3.2: *Conjunto de prueba* P_0

donde x^0_{lk} con $1 \leq l \leq m$ y $1 \leq k \leq K$ el k–ésimo valor para el l–ésimo individuo del conjunto de prueba P_0.

Y la recomendación de un conjunto de atributos $\{X_1, X_2, \dots, X_K\}$ de estudio del experto del dominio.

Con los elementos mencionados se quiere detectar de forma eficiente las características más relevantes de las diferentes clases que permitan una fácil interpretación de resultados en un sistema enfocado a la predicción y/o al diagnóstico.

Por lo tanto, se propone obtener un modelo conceptual que nos permita:

- Un procedimiento de caracterización de clases y, que además, identifique las variables más relevantes en cada una de las clases formadas.
- La generación de un sistema de reglas que sea la base para las interpretaciones conceptuales de las clases en forma automática, usando atributos cuantitativos, como apoyo a un sistema orientado a la predicción y/o diagnóstico.
- Contribuciones al problema de validación de clases.

3.2 Objetivos de Tesis

En este trabajo se tienen objetivos a diferentes niveles:

1. Relativos al universo del discurso donde se encuentra el dominio de estudio.
   - Facilitar el estudio de este tipo de dominios poco estructurados.
   - Obtener conocimiento útil que nos permita mejorar las clasificaciones con resultados más precisos.
   - Obtener resultados fácilmente interpretables por el usuario.
2. De orden metodológico.

- Establecer una metodología híbrida formal que utilice técnicas, métodos y herramientas de Inteligencia Artificial, Estadística y Lógica Difusa que permita resolver el problema planteado en la sección §3.1.
- Obtener un modelo de conocimiento explícito para generar caracterizaciones e interpretaciones de las descripciones conceptuales de las diferentes clases en las que se ha particionado el dominio objeto de estudio.

Capítulo 4

Estado del Arte

Como se presentó en el planteamiento del problema, se tiene un conjunto de datos y el conocimiento que un experto tiene del dominio de ellos, de esto se obtiene una partición de los mismos en clases. Inspirado en el "boxplot" se realiza un tratamiento estadístico de los datos para obtener un sistema de reglas y su representación en gráficos y así poder caracterizar e interpretar un nuevo objeto. Es así que lo anterior induce a considerar la investigación dentro del contexto de los Sistemas Híbridos entre Inteligencia Artificial, Estadística, Teoría de los Conjuntos Difusos y la Lógica Difusa guiada por la construcción de sistemas reales de Descubrimiento de Conocimiento en Base de Datos (KDD) que cubra todas las fases de este proceso, incluida la generación automática de interpretaciones conceptuales; donde nuestra investigación será una aportación a dicho campo de investigación.

4.1 Estadística e Inteligencia Artificial

El término **Estadística** se deriva del latín *Status*, que se refiere a política y situación social, al Estado, empieza como una ciencia de recolección de datos económicos y demográficos. En su evolución y aún hoy en día se considera una ciencia relacionada con la colección y el análisis de datos, para extraer información y presentarla en forma comprensible y sintética.

A fines del siglo XVIII surge un periodo científico fértil en el campo de la Estadística. En este tiempo Galton (1877) presentó sus primeros trabajos sobre *Análisis de regresión*, y Pearson presentó, entre otros trabajos, en 1901, una versión preliminar del *Análisis de Componentes Principales*. Su principal discípulo Fisher (1890–1962), cuyos trabajos son considerados la base de la *Estadística moderna*, junto con Mahalanobis en 1936, presentaron los primeros trabajos acerca del *Análisis Discriminante* en el cual existe una variable respuesta, que indica la clase de todo objeto y encuentra la mejor combinación lineal de todos los atributos para distinguir la clase.

Así, desde hace mucho tiempo se utiliza la formación y distinción entre diferentes clases de objetos (clustering), tomando actualidad cuando las computadoras llegan a ser más poderosas. En 1963, Sokal y Sneath presentaron *The Numerical Taxonomy* la cual puede ser considerada como la primera formulación moderna de clustering.

La **Inteligencia Aritificial** es una disciplina formal que surge a mediados de los años 50's. Al inicio estuvo bajo el paradigma de Von Neumman y técnicas de com-

putación secuencial y su característica a través de su génesis histórica es la búsqueda para construir máquinas que "piensen".

En 1961, Minsky divide la IA en cinco tópicos: búsqueda, reconocimiento de patrones, aprendizaje, planeación e inducción. La mayoría de los trabajos serios sobre IA de acuerdo a este esquema estuvieron relacionados con búsqueda heurística.

Uno de los primeros éxitos de la aplicación en *la solución de problemas orientados al diagnóstico* fue MAYCIN en 1976 (diagnosis de infecciones), y otras técnicas como: *sistemas expertos*, *representación del conocimiento*, *aprendizaje automático*, *razonamiento*, *procesamiento de lenguaje natural*, etc. Sin embargo, las representaciones simbólicas mostraron serias limitaciones cuando hicieron frente a problemas reales y complejos, principalmente porque las mayoría de los problemas en IA son NP-complete.

En los años 70's aparece el paradigma del paralelismo (arquitectura de ordenadores en paralelo), algunas veces llamado IA micro-distribuida y denominada por algunos autores, por su metáfora implícita como: redes neuronales artificiales (ANN).

Entre el paradigma del paralelismo y del simbolismo, aparecieron la IA evolutiva y la IA macro-distribuida. La primera se caracteriza por los *algoritmos géneticos* y la segunda por los *sistemas multi-agentes* y otras técnicas.

De los campos de aplicación de estas disciplinas podemos establecer que los objetivos de la IA como de la Estadística son: la primera *desarrollar programas que "aprendan" y enriquezcan el conocimiento propio y el del usuario* y de la segunda, *presentar de forma sintética y comprensible la colección y análisis de todo tipo de información* [RGG00].

4.2 Los Sistemas Híbridos

Es claro que hoy, las nuevas tecnologías aumentan significativamente nuestra capacidad de producir, coleccionar y almacenar datos. Enormes cantidades de datos están disponibles para ser analizados y extraer conocimiento en corto tiempo.

Obtener conocimiento de conjuntos de datos grandes o pequeños–y además, mal estructurados–es una tarea muy difícil. La combinación de técnicas de análisis de datos (ej. clustering), aprendizaje inductivo (ej. sistemas basados en conocimiento), administración de base de datos y representación gráfica multidimensional, deberán producir beneficios en esta dirección y a corto plazo.

Existen diversas herramientas informáticas que tratan algunas de las situaciones mencionadas (ej. Clementine, Intelligent Manager, SPAD, SPSS, WEKA entre otras son algunas de los más famosas hoy en día), las cuales presentan principalmente una combinación de técnicas existentes, permitiendo comparación de resultados y la selección del mejor método en cada caso.

Sin embargo, en situaciones reales, es usual trabajar con dominios complejos [GC98], tales como trastornos mentales [GS97], esponjas marinas [Gib94], disfunciones tiroidales [GS99], pruebas psicofisiológicas [RGR01] y muchas más, donde las bases de datos tienen tanto atributos cualitativos como cuantitativos; y el experto tiene algún conocimiento *a priori* (en general parcial) de la estructura del dominio—el cual es difícil tomarse en cuenta por métodos de clustering—el cual es difícil de incluir en una *Base de Conocimiento*.

Durante la década pasada, en una gran variedad de dominios de aplicación, los investigadores en aprendizaje automático, teoría del aprendizaje computacional, re-

conocimiento de patrones y la estadística han hecho un esfuerzo por establecer un puente de comunicación y cooperación entre investigadores de la IA y la Estadística, Douglas H. Fisher y Bill Gale —entre otros— han establecido una línea de investigación conformada por ambas ciencias, creando la *Society for Artificial Intelligence and Statistics* [1] que tiene como objetivo *impulsar la investigación para poder combinar técnicas de estas disciplinas en la creación de* **Sistemas Híbridos** [2] *que mejoren las funciones y desempeño de los sistemas actuales en las diversas áreas tanto de la IA como la Estadística y algunas otras que estén soportadas por estas disciplinas*, dando lugar a una tercera opción que es el trabajo interdisciplinario.

"Nos parece que hay un potencial de desarrollo enorme en la intersección de la IA, la Ciencia de la Computación y la Estadística" [3]

"Cheseman y Oldfor"

4.3 El Proceso KDD

Se estima que la cantidad de información en el mundo se dobla cada 20 meses [FPSS96]; esto significa que, científicos, gobierno y sistemas de información corporativos están siendo inundados por una gran cantidad de datos que son generados y almacenados rutinariamente, los cuales aumentan las bases de datos. Estos volúmenes de datos rebasan los métodos manuales tradicionales de análisis de datos como hojas de cálculo y cuestionarios *ad-hoc*, los cuales pueden crear reportes informativos de datos, pero no pueden analizar los contenidos de estos reportes para obtener conocimiento importante. De ahí que existe una necesidad significativa para una nueva generación de técnicas y herramientas con la capacidad de asistir *inteligente* y *automáticamente* a las personas en el análisis de la gran cantidad de datos para obtener conocimiento útil. Estas técnicas y herramientas son temas de un campo emergente de descubrimiento del conocimiento en base de datos (KDD) [4] [Fay96].

El proceso KDD es interactivo e iterativo [Brachman & Anand dan un punto de vista práctico del proceso de KDD enfatizando la naturaleza interactiva del proceso], incluye varios pasos con decisiones que tienen que tomarse por el usuario. La Figura 4.1 muestra un diagrama del proceso KDD. A continuación se resume cada una de las etapas:

1. La comprensión del dominio de aplicación, el conocimiento *a priori* relevante, y las metas del usuario final.

2. La creación de un conjunto de datos destino [5]. Seleccionar un conjunto de datos, o seleccionar un subconjunto de atributos o muestra de datos, sobre los cuales se realizará el descubrimiento.

[1] Asociación para la Inteligencia Artificial y la Estadística.

[2] En general son combinación de aproximaciones de técnicas y/o métodos de diversas disciplinas como la IA, la Estadística y la Lógica principalmente.

[3] Tomada del libro Artificial Intelligence and Statistical IV, volume 89, Springer-Verlag, 1994.

[4] Del Inglés Knowledge Discovery in Datebases.

[5] Llamado conjunto de entrenamiento.

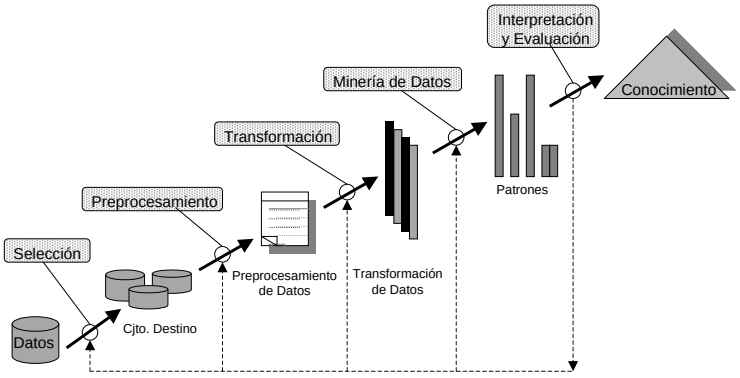

Figura 4.1: Diagrama del proceso KDD

3. La preparación y preprocesamiento de datos. Operaciones básicas, si fueran necesarias, como la eliminación de ruido, datos atípicos (outliers) o perdidos, recabar la información necesaria para modelar el ruido, decidir sobre estrategias para manejar datos perdidos, etc.

4. La reducción y proyección de datos. Encontrar características útiles para representar los datos depende de las metas del proceso. Usar reducción de la dimensionalidad o métodos de transformación para reducir el número de atributos bajo consideración o para encontrar representaciones invariantes para los datos.

5. Seleccionar la tarea de minería de datos. Decidiendo si la meta del proceso KDD es clasificación, regresión, clustering, etc.

6. Seleccionar el o los algoritmo(s) de minería de datos. Seleccionar los métodos que se emplearán en la investigación para identificar patrones en los datos. Esto incluye decidir que modelos y parámetros son los apropiados y escoger un método de minería de datos compatible con el criterio del proceso de KDD.

7. La minería de datos. La investigación de patrones en una representación formal o un conjunto de representaciones como: reglas de clasificación o árboles, regresión, clustering y así sucesivamente. El usuario puede apoyar el método de minería de datos realizando correctamente los pasos previos.

8. La interpretación de los resultados obtenidos, posible retorno a cualquiera de los pasos previos del 1–7 para iteraciones posteriores.

9. La consolidación del conocimiento descubierto. Incorporación de este conocimiento en el desempeño del sistema, o simplemente documentarlo y reportarlo a las partes interesadas.

El proceso de KDD puede incluir iteraciones significativas y contener ciclos entre cualesquiera dos pasos; así en cada etapa el "minero informático" puede volver a la etapa que el requiera para continuar su trabajo. La etapa donde se descubre la información es la denominada Minería de datos.

4.4 Reconocimiento de Patrones Estadísticos

El objetivo fundamental del reconocimiento de patrones es clarificar perfiles de comportamiento de los objetos. Entre los diferentes contextos en los cuales el reconocimiento de patrones ha sido formulado, la aproximación estadística ha sido la más estudiada y usada en la práctica.

Dado un patrón, su reconocimiento/clasificación puede consistir de una de las siguientes dos tareas [Wat85]: (i) clasificación supervisada (ej., análisis discriminante) en la cual el patrón de entrada se identifica como un miembro de una clase predefinida, (ii) clasificación no supervisada (ej., clustering) en la cual el patrón se le asigna una clase desconocida hasta ese momento. Aquí el problema de reconocimiento se esta considerando como una tarea de clasificación o categorización, donde las clases están definidas por el diseñador del sistema (en clasificación supervisada) o están basadas en similitud de patrones (en clasificación no supervisada). A pesar de los poco más de cincuenta años de investigación y desarrollo en este campo, el problema general de reconocimiento de patrones con una orientación, ubicación y escalamiento no se ha resuelto, esto es, no se ha conseguido un diseño de un reconocedor de patrones automático de propósito general.

El diseño de un sistema de reconocimiento de patrones incluyen los siguientes tres aspectos: (i) adquisición de datos y preprocesamiento, (ii) representación de datos y (iii) toma de decisiones. El dominio del problema sugiere la selección de los sensores, la técnica de preprocesamiento, el esquema de representación y el modelo de toma de decisiones. Generalmente un problema de reconocimiento bien definido y suficientemente delimitado (pocas variaciones intra clases y muchas variaciones inter clases) conducen a una representación compacta de patrones y a una estrategia simple de toma de decisiones. Por lo que, ninguna aproximación por sencilla que sea será la mejor ya que se han de utilizar diferentes técnicas y métodos. En consecuencia, la combinación de éstos es una práctica de uso común en el diseño de sistemas híbridos de reconocimiento de patrones [Fu83].

Las mejores cuatro aproximaciones conocidas son: (i) Patrones de referencia ([BK89] y [Gre93]), (ii) clasificación estadística ([DL96], [DH73] y Vapnik [Vap98]), (iii) igualación sintáctica o estructural ([Fu82], [Fu83], [Pav77]y [Per98]), (iv) redes neuronales ([JDC87] y [Koh95]). La Tabla 4.1 muestra una breve descripción y comparación de estas aproximaciones.

4.4.1 Métodos Estadísticos de Clasificación

La literatura sobre el reconocimiento de patrones es vasta y dispersa encontrándose en numerosas revistas de diferentes disciplinas (ej. estadística aplicada, aprendizaje automático, redes neuronales y procesamiento de señales e imágenes). Un rápido vistazo de la tabla de contenidos de todos los temas de la IEEE *Transactions on Pattern Anal-*

Aproximación	**Representación**	**Función de Reconocimiento**	**Criterio Típico**
Patrones de referencia	Muestras, pixeles Curvas	Correlación, Medida de distancia	Error de clasificación
Estadística	Características	Función Discriminante	Error de clasificación
Sintáctica o Estructural	Primitivas	Reglas, Gramática	Error de aceptación
Redes Neuronales	Muestras, pixeles, características	Función de la Red	Error cuádratico medio

Tabla 4.1: Aproximaciones de Reconocimiento de Patrones

ysis and Machine Intelligence, desde su primera publicación en enero de 1979, revela que aproximadamente 350 artículos tratan con el reconocimiento de patrones. Aproximadamente 300 de estos artículos cubren la aproximación estadística y pueden ser categorizados en los subtemas siguientes: problema de dimensionalidad (15), reducción de la dimensionalidad (50), diseño de clasificadores (175), combinación de clasificadores (10), estimación del error (25) y clasificación no supervisada (59). Además los excelentes libros de Duda y Hart [DH73], Fukunaga [Fuk90], Devijver y Kittler [DK82], Devroye, Gyorfi y Lugosi [DL96], Bishop [Bis95], Ripley [Rip96], Schurmann [Sch92] y McLachlan [McL92], Nagy [Nag68] y Kanal [Kan94] en 1974 entre otros investigadores han contribuido notablemente al estado del arte de este tema.

La Tabla 4.2 resume los clasificadores más comúnmente usados. Muchos de ellos representan, en realidad, una familia completa de clasificadores y permiten al usuario modificar diferentes parámetros asociados y funciones de criterios. Todos (o casi todos) los clasificadores son aceptables en el sentido de que existen algunos problemas de clasificación para los cuales son la mejor opción.

4.5 Sistemas de Clasificación Basados en Reglas

Hoy los métodos de clasificación automática son utilizados en todas sus variedades para conocer la estructura de grandes conjuntos de datos, lo cual incide en los objetivos básicos de los procesos emergentes de *Minería de datos* que tan de moda ha puesto la Sociedad de la Información y las Nuevas Tecnologías.

Clustering. Es un término usado para denotar la función un gran número de técnicas que intentan determinar si existen grupos o *clusters* en un conjunto de datos y, en el caso que así sea, determinarlos.

A pesar de las diferencias en cuanto a las diferentes aplicaciones, los tipos de datos y las técnicas utilizadas, existen cinco pasos básicos [AB84] que caracterizan todo análisis de cluster [Har75]:

1. Selección de la muestra sobre la que se hará la clasificación.

Método	**Propiedad**	**Comentarios**
Árbol de decisión	Encuentra un conjunto de umbrales para una secuencia de características dependiente.	Procedimiento de entrenamiento iterativo; entrenamiento sensitivo; necesidad de poda; rápida prueba.
Discriminante lineal de Fisher	Clasificador lineal que usa optimización MSE.	Simple y rápido; similar a Bayes para las distribuciones Gaussianas con matrices de covarianzas idénticas.
Clasificador Parzen	La Regla de Bayes para la densidad de Parzen estima con desempeño el núcleo optimizado.	óptima asintóticamente; dependiente de la escala; prueba rápida.
Regla de los K-Vecinos Próximos	Asigna patrones a la clase mayoritaria entre los k vecinos próximos usando un valor optimizado para k.	óptima asintóticamente; dependiente de la escala; prueba lenta.
Clasificador Logístico	Regla de probabilidad máxima para probabilidades a *posteriori* logísticas (sigmoidales).	Clasificador lineal; procedimiento iterativo; óptimo para una familia de diversas distribuciones (Gaussianas); tipos de datos mixtos.
Clasificador de Bayes	Asigna patrones a la clase que tiene probabilidad a posteriori estimada máxima.	Pertenece a los clasificadores sencillos (lineales o cuádratica) para distribuciones gaussianas; sensitivo a la densidad de estimación de errores.
Método del Subespacio	Asigna patrones a la clase más cercana del subespacio.	En vez de normalización de invariantes, es usado el sub-espa cio de las invariantes; dependiente de la escala (métrica).
Clasificador Cercano Medio	Asigna patrones a la clase más cercana media.	Sin necesidad de entrenamiento; prueba rápida; dependiente de la escala (métrica).
Clasificador Vector de Soporte	Maximiza el margen entre las clases seleccionando un número mínimo de vectores.	Dependiente de la escala; iterativo; lento entrenamiento; no lineal e insensitivo.

Tabla 4.2: Métodos de Clasificación

2. Definición del conjunto de atributos con los que se describirá las entidades de la muestra.

3. Cálculo de las disimilitudes o distancias entre las entidades en base a dichos atributos.

4. Selección de un algoritmo de clustering y detección de grupos.

5. Validación de los resultados proporcionados por el algoritmo.

Un aspecto importante a puntualizar es que, de todas las clasificaciones posibles que se pueden hacer con un conjunto de objetos, no existe la *buena* clasificación sino que, dependiendo de los objetivos del estudio o uso que se quiera hacer, se escoge una u otra. La recomendación general es que se elija la que resulte útil en cada contexto.

Existen diferentes familias de métodos en la elección de una distancia [Gib94]:

- Métodos de particiones. Se busca la partición óptima del conjunto que se estudia en un número prefijado de classes k. Hay de dos tipos:
  - Métodos de particiones directas: Las clases que se forman serán disjuntas, y pueden ser aglomeradas o divisivas.
  - Métodos de particiones en clases solapadas: Las clases pueden solaparse, es decir, un mismo objeto puede pertenecer simultáneamente a más de una clase.

- Métodos de clasificación jerárquica. Se busca el árbol que refleja la estructura jerárquica de los datos. Según el nivel por el que se corte el árbol se obtendrá una partición más o menos precisa del conjunto objeto de estudio. Una ventaja respecto al anterior método es que no hace falta avanzar el número de clases que se quiere obtener al final.

- Otros métodos: métodos de clasificación piramidal, métodos de árboles aditivos y de clases latentes.

El principal problema para desarrollar métodos de clasificación automática es que el concepto de *cluster* no es fácil de definir. Algunas aproximaciones para definir un *cluster* pueden basarse por sus propiedades como: máxima cohesión interna y máximo aislamiento externo, propiedades propuestas por [Cor71] y [Gor80]. Además, las clases pueden presentar formas y magnitudes muy diferentes y se puede entender la dificultad de que exista una definición general de *clusters* que los incluya a todos.

El problema de fondo es que el investigador puede no conocer la estructura de los datos *a priori* y existe el peligro de interpretar la existencia de diferentes clusters cuando estos no existen realmente.

En [Alu96] se plantea hasta qué punto las clases obtenidas en un proceso de clasificación reflejan clases reales presentes en los datos, o si por el contrario, las clases obtenidas son el simple resultado de aplicar un algoritmo a los datos, es decir, una partición de una realidad contínua.

También se afirma que la experiencia prueba que, aunque nos encontremos en este último caso, la tipología obtenida puede ser igualmente útil, ya que aunque no se pueda

hablar de clases realmente diferenciadas entre ellas, la partición obtenida suele facilitar la comprensión de los datos y por tanto su operatividad. En este caso hablamos de *clases instrumentales* en oposición a *clases reales*.

Algoritmo genérico de clasificación ascendente jerárquica. Una clasificación jerárquica es una secuencia de clasificaciones en la que los *clusters* más grandes se forman a través de la fusión consecutiva de clusters más pequeños.

Existen muchos algoritmos de clasificación ascendente jerárquica cada uno con sus propias variantes y que conducen a diferentes clasificaciones. Sin embargo, si se quisiera presentar un algoritmo genérico para los métodos de clasificación ascendente jerárquica, este podría ser el que plantea en [DM84].

Uno de los algoritmos que se enmarca en este esquema de clasificación es el conocido como de los vecinos recíprocos encadenados que se describe a continuación.

Vecinos recíprocos. El algoritmo de los *vecinos recíprocos* utiliza un concepto propio para determinar cuáles son los individuos que se agregan:

Son vecinos recíprocos los individuos i, i' si i es el objeto más próximo a i' en la muestra, y i' es a su vez el más próximo a i. De este modo, en la clasificación por vecinos recíprocos, siempre se agregarán parejas de vecinos recíprocos.

La principal propiedad de este método es que el resultado no depende del orden como se procesan los datos (ni del orden como se producen las agregaciones) porque se está trabajando con un criterio global sobre todos los datos.

La Figura 4.2 ilustra cómo en este algoritmo se produce un encadenamiento de objetos que lleva del objeto más cercano al siguiente más cercano hasta que se bucla en un lazo. El lazo es precisamente la expresión gráfica de las parejas de vecinos recíprocos. Cuando se halla uno, se produce una agregación con la consecuente creación de una nueva clase. Es frecuente representar en forma de árbol la secuencia de agregaciones de un proceso así. Estos árboles reciben el nombre de *dendogramas*.

En estos contextos identificar cuáles son las parejas de elementos más próximos (o de vecinos recíprocos en este último caso) en cada iteración requiere la definición de una métrica sobre el espacio de los atributos que permita calcular la distancia entre dos individuos.

Clasificación aprovechando el conocimiento declarativo de los expertos. Los métodos clásicos de clasificación automática aplicada a *dominios poco estructurados* [Gib94], muchas veces presentan resultados que no se pueden interpretar. En muchas ocasiones el experto tiene suficiente conocimiento para organizar parte del dominio en entidades que tengan sentido. Sin embargo, los métodos estadísticos clásicos prácticamente ignoran esta información. $Klass^{+}$ implementa la metodología de *clasificación basada en reglas* cuya idea fundamental es recoger este conocimiento en forma de reglas que subdividan el espacio de clasificación en entornos coherentes y respetar esta primera estructuración sugerida directamente por el experto. Con esto se pretende cubrir tres objetivos: incorporación a la clasificación de información antes ignorada (como relaciones entre atributos o restricciones), recogida de los objetivos de la clasificación que se pretende obtener y garantizar la interpretabilidad de la clasificación obtenida [GC94].

Representación del conocimiento del experto e interpretación. Introducir un nivel semántico en el proceso de clasificación ha de permitir una interpretación más clara de las clases finales. Incluir relaciones entre atributos, condiciones de pertenencia a una clase o restricciones de incompatibilidad de grupos de objetos en un único

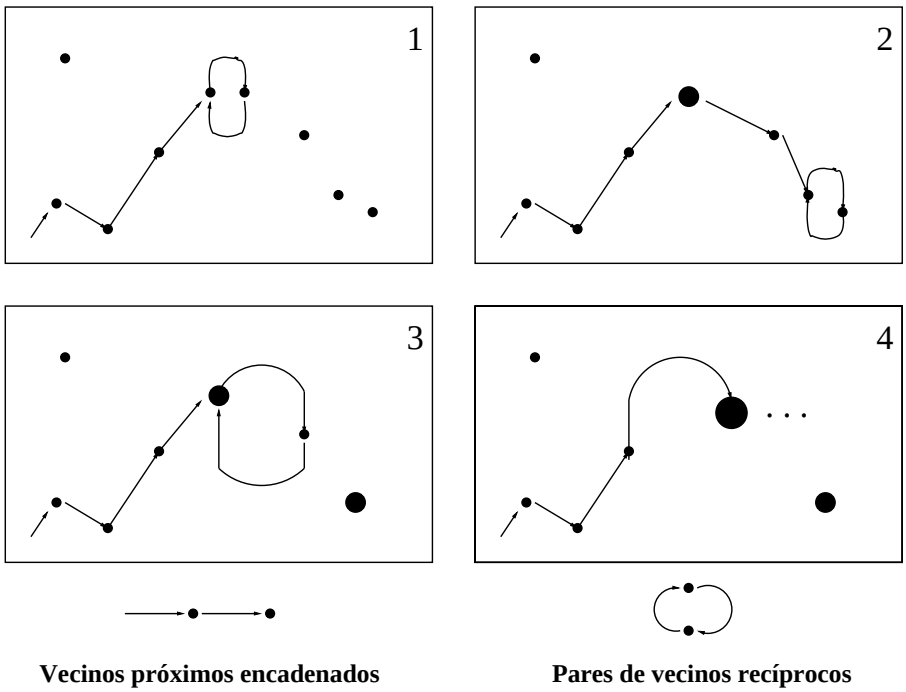

Figura 4.2: El proceso de los vecinos recíprocos encadenados

formalismo conduce a buscar un modelo de representación muy genérico con suficiente potencia para tratar todo esto. Ésta es la razón por la que el conocimiento adicional que proporcione el experto se representa a través de reglas lógicas de primer orden.

La estructura de las reglas que contempla el método que vamos a usar es sencilla desde el punto de vista sintáctico y muy potente. Una regla está compuesta de una parte derecha que indica el nombre de alguna clase C y una parte izquierda con la condición A que ha de satisfacer un objeto i para formar parte de la clase C. En resumen, diremos que un objeto i es seleccionado por una regla del tipo:

$$\mathrm{r} = (\mathrm{A} \rightarrow \mathrm{C})$$

si A se evalúa como cierto para el objeto i.

En general, los objetos pueden satisfacer una, ninguna o más de una regla. Aquéllos que no cumplan ninguna regla no son motivo de preocupación, ya que se ha dicho que el experto proporciona un conocimiento parcial sobre el dominio.

Metodología de clasificación basada en reglas. Una vez construída la Base de Reglas, con ayuda del experto, se puede evaluar qué objetos satisfacen cada una de las reglas. Algunos no satisfacen ninguna. El conjunto de objetos que están en esta situación forma parte de lo que se denota como *clase residual* y se integran a la jerarquía global en la última etapa del proceso de clasificacón con reglas.

El resultado de evaluar las reglas sobre los individuos es una partición de la muestra en k clases más la *clase residual*, donde k es el número de partes derechas distintas en las reglas.

Con la finalidad de respetar la estructura de la clasificación jerárquica hace falta que las clases inducidas por las reglas se constituyan en forma de árbol. En primer lugar se realiza una clasificación local a cada una de las clases inducidas por las reglas. Eso genera los primeros nodos internos del árbol final. Por último, los centros de

dichas clases se clasifican junto a los elementos de la clase residual para integrar todos los elementos en un único árbol ascendente jerárquico que es el que dará lugar a la clasificación final. Sobre las ventajas de trabajar con este tipo de metodología, véase [Gib96], [GS97] y [GC92].

4.6 La Lógica Difusa y el Razonamiento Difuso

Introducción. Una gran variedad de ciencias aplican métodos de Inteligencia Artificial principalmente para modelar el razonamiento del experto. Para el diseño de tales sistemas inteligentes, la importancia de la Lógica Difusa ha ganando gran aceptación [Zad93]. Publicaciones recientes han mostrado también que los sistemas híbridos en IA han conseguido buenos resultados, combinando Lógica Difusa e Inteligencia Artificial para la diagnosis médica en la prevención de enfermedades, redes neuronales para el reconocimiento de patrones, sistemas de inferencia difusos para incorporar conocimiento humano, realizar inferencia y tomar decisiones, etc. Es importante considerar que los problemas complejos del mundo real requieren sistemas inteligentes que combinen conocimiento, técnicas y metodologías de diferentes fuentes. Estos sistemas inteligentes deberán poseen experiencia como la del humano dentro de un dominio específico, adaptándose y aprendiendo a hacer lo mejor en ambientes dinámicos y explicando como toman decisiones o acciones. De cara a los problemas de cálculo, es más ventajoso usar diferentes técnicas de cálculo sinérgicas que exclusivas, obteniendo como resultado la construcción de *sistemas híbridos* inteligentes.

Lógica Difusa. El cerebro humano interpreta la imprecisión y la información sensorial incompleta proporcionada por los sentidos perceptivos. La teoría de los conjuntos difusos y la lógica difusa proporcionan un método sistemático para tratar con tal información lingüísticamente, y realizar cálculo numérico usando etiquetas lingüísticas definidas por funciones de pertenencia. Más aún, una selección de reglas difusas de la forma Si–Entonces forma la componente clave de un sistema de inferencia difuso que puede modelar en forma efectiva la experiencia humana en una aplicación específica.

La lógica como base para el razonamiento puede distinguirse por sus tres componentes principales (independientes del contexto): valores de verdad, vocabulario (operadores) y razonamiento (tautologías, silogismos). En la lógica de Boole, los valores de verdad son 0 (falso) o 1 (verdadero) y por medio de estos valores de verdad, se define el vocabulario vía las tablas de verdad.

Una distinción entre la verdad material y la lógica [MG81] se hace en las llamadas lógicas extendidas: La lógica modal [HC68] distingue entre verdad necesaria y posible, y la lógica temporal [McD82] entre enunciados que fueron verdaderos en el pasado y aquellos que serán verdaderos en el futuro. La lógica epistémica [BA96] trata del conocimiento y las creencias, la lógica déontica [Ris71] con lo que debe hacerse y que permite ser verdadero. La lógica modal, en particular, podría ser una buena base para aplicar diferentes medidas y teorías de la incertidumbre.

Otra extensión de la lógica de Boole es el cálculo de predicados, el cual es un conjunto lógico teórico que usa cuantificadores y predicados para los operadores de la lógica de Boole.

La *lógica difusa* [Zad65] hace una extensión del conjunto teórico de la lógica multivaluada en la cual los valores de verdad son atributos lingüísticos (términos de verdad

de atributos lingüísticos).

Lo mismo que en la lógica clásica, los operadores se definen en la lógica difusa a través de tablas de verdad, usando el Principio de Extensión para obtener las definiciones de estos operadores. Hasta ahora la teoría de la posibilidad ha empezado a ser usada para definir operadores en lógica difusa, aunque hay otros operadores que también han sido investigados [MZ82] y que podrían usarse.

Además, podemos considerar conectivos mixtos como funciones para calcular el grado de pertenencia conjunta vía las t-normas en problemas de clasificación [Piera87].

Un punto importante es la relación y diferencia entre los conceptos de probabilidad y posibilidad, con este último concepto se tiene una estrecha relación con el grado de pertenencia a un conjunto difuso. El concepto de posibilidad juega un importante papel particularmente en la representación del significado, en la gestión o manejo de la incertidumbre en sistemas de clasificación, sistemas inteligentes y en algunas otras aplicaciones.

Razonamiento Difuso. En realidad, cuando las personas hablamos acerca de un sistema del mundo real, lo hacemos en tres etapas:

- Seleccionamos un conjunto de atributos que podrían ser entendidas como un conjunto de entidades bien diferenciadas. Tales atributos pueden estar directamente vinculados a la experiencia sensorial–y entonces expresadas en una manera informal–o pueden estar determinadas por medio de procedimientos de mediciones más precisas.

- Establecemos las relaciones entre los atributos, ligando sus estados particulares. Esto en realidad se hace dando reglas como *Si (hecho A) entonces (hecho B)*, donde cada hecho describe un estado o un valor preciso de algún atributo particular.

- Finalmente, hay una tercera etapa donde los conjuntos de reglas se organizan para construir una teoría o un modelo que describe el sistema del mundo real bajo estudio.

El sistema está bien comprendido cuando su teoría no conduce a conclusiones contradictorias o a enunciados experimentalmente falsos acerca del sistema.

En este contexto el término *inferencia* se aplica a cualquier algoritmo que se use para derivar consecuencias de hechos conocidos dentro del modelo. La inferencia en un amplio sentido puede aparecer en diferentes formas dependiendo del contexto considerado, desde la manipulación simbólica en una base de datos lógica hasta la evaluación de una función numérica o vectorial. En el caso anterior, las reglas aparecen bajo la forma: *Si* $X = x$ *entonces* $Y = f(x), x \in X$, con hechos conocidos como $X = x_0$ ó $X \in A$, siendo A un subconjunto de X. En Dubois y Prade (1996) leemos: "las reglas Si-Entonces son una herramienta clave para expresar piezas de conocimiento en lógica difusa".

Sin embargo, cuando los atributos considerados vienen de conceptos graduales como altura, temperatura, cantidad y algunas otras, las descripciones de sus estados están algunas veces dadas también por enunciados graduales e implícitamente vagos. Ejemplos de estos enunciados son *la temperatura es alta*, *el color es azul*, etc. Más aún, en este caso el conocimiento acerca del sistema puede presentarse en forma de enunciados

condicionales ligando estos estados vagos de los atributos, tales como *Si la temperatura es baja, entonces el color es verde.*

Cuando los estados vagos de los atributos están representados por *conjuntos difusos* del universo del discurso donde los atributos toman sus posibles valores, el problema surge naturalmente de cómo determinar los hechos y las reglas que se han de combinar para derivar nuevos hechos. Esta es la esencia de la *inferencia difusa.* Estos hechos vagos en el contexto de los conjuntos difusos les llamaremos *enunciados difusos* ó *proposiciones difusas*, y las reglas relacionadas con estos hechos como *reglas difusas* ó *enunciados condicionales difusos.*

Lo que es evidente desde el punto de vista de la lógica es que la inferencia lógica tiene lugar a nivel semántico. A diferencia de los procedimientos de la lógica clásica, que derivan conclusiones por manipulación simbólica, en la lógica difusa los enunciados difusos están siempre relacionados a los conjuntos difusos que los representan, y el proceso de inferencia total se realiza por manipulación numérica de sus funciones de pertenencia. En esta forma los hechos inferidos se construyen a partir de sus funciones de pertenencia, y no en forma inversa.

En los diferentes significados que puede tener un enunciado difuso, hay una característica común que todas las reglas comparten, es decir su capacidad para ser aplicadas a situaciones no lejanas de aquellas para las cuales han sido originalmente concebidas. La inferencia difusa tiene la ventaja de su *versatilidad* para derivar consecuencias cuando los hechos conocidos no coinciden exactamente con cualquiera de los antecedentes de las reglas que describen el conocimiento acerca del sistema. Tales procesos de inferencia son referidos en la literatura como *razonamiento aproximado*, y están obviamente más cercanos a la forma humana de pensar que a los procedimientos clásicos de inferencia. Este aspecto de la lógica difusa es *relevante* e importante para la Inteligencia Artificial.

La idea original de realizar inferencia difusa por medio de relaciones difusas compuestas (regla de composición difusa) fue introducida por Zadeh [Zad73]. Esta aproximación naturalmente conduce a un patrón de inferencia que se extiende al *modus ponens* y que puede ser fácilmente generalizado a situaciones más complejas donde se consideren varios atributos (principio de proyección–combinatoria). En este contexto nos preguntamos: ¿cómo deberá interpretarse una regla difusa dada *Si X es A entonces Y es B* en términos de una relación difusa sobre el producto Cartesiano de los universos de discurso X x Y?. Las dos respuestas más aceptables a esta pregunta se encuentran en la literatura y vienen de dos trabajos pioneros, uno de Zadeh y el otro de Mamdani. La aproximación de Zadeh toma a $R(x, y) = I(A(x), B(y))$ donde I significa una función de *implicación multivaluada*, mientras la aproximación de Mamdani toma a $R(x, y) = A(x) \otimes B(y)$ con $\otimes \stackrel{\text{def}}{=} min$ (o más generalmente cualquier función multivaluada). Esta segunda forma de realizar inferencia es la más usual en el campo del control difuso. La selección entre aproximaciones basadas en implicación y en conjunciones depende sobre el significado deseado de la regla y la forma condicional de combinar hechos inferidos de las diferentes reglas.

La interpretación de procesos de inferencia difusa como procesos de razonamiento aproximado nos permite comparar que tan lejanos son los hechos conocidos de los antecedentes y hechos inferidos de los consecuentes.

4.7 Sistemas de Clasificación Basados en Reglas Difusas

Hoy en día, las aplicaciones más importantes de la teoría de los conjuntos difusos desarrollada por Zadeh en 1965 [Zad65] son los Sistemas Basados en Reglas Difusas (SBRD). Esta clase de sistemas constituye una extensión de los Sistemas Clásicos Basados en Reglas, debido a que tratan con reglas difusas en vez de reglas lógicas clásicas. Gracias a esto, han sido aplicados éxitosamente a una amplia gama de problemas de diferentes áreas que presentan diferentes formas de incertidumbre y vaguedad.

Sistemas Basados en Reglas Difusas (SBRD). Un Sistema Basado en Reglas Difusas (SBRD) presenta dos componentes principales: 1) el sistema de inferencia, que ejecuta el proceso de inferencia difuso necesario para obtener una salida del cuando ha sido especificada una entrada, y 2) la Base de Reglas Difusas (BRD) representa el conocimiento que se tiene acerca del problema ha resolver, formando un conjunto de reglas.

En el diseño de un sistema inteligente de esta clase se deberán de realizar dos tareas principales para una aplicación concreta: i) seleccionar los operadores difusos involucrados en el sistema de inferencia, esto es definir la forma en la cual el proceso de inferencia difusa se realizará, y ii) obtener una adecuada BRD acerca del problema ha resolver. La exactitud de los SBRD para resolver un problema específico depende, directamente de ambas componentes.

La primera tarea ha de ser ampliamente analizada en la literatura especializada, y se ha de realizar una gran cantidad de estudios teóricos y comparativos para tratar con el problema de seleccionar los mejores posibles operadores difusos en el sistema de inferencia ([ATV83], [Yag88] y [Yag93]).

En relación a la segunda tarea del diseño, parece ser más difícil la decisión porque la composición de la BRD depende directamente del problema ha resolver. Debido a la complejidad de la derivación de la BRD, se han propuesto una gran cantidad de técnicas automáticas para tal efecto.

Los Sistemas Basados en Reglas Difusas (SBRD) combinan la precisión de la predicción con un alto nivel de interpretabilidad, lo cual los hace muy adecuados para el diseño de *Sistemas de Clasificación* en problemas reales.

Sistemas de Clasificación Basados en Reglas Difusas. En un Sistema de Clasificación Basado en Reglas Difusas (SClBRD), se distinguen dos componentes: 1) La Base de Conocimiento (BC), compuesta de una Base de Reglas (BR) y una Base de Datos (BD), la cual es específica para un problema dado de clasificación, y 2) un Modelo de Razonamiento Difuso (MRD).

El diseño de un SClBRD implica encontrar ambas componentes, y este proceso se lleva a cabo a través de un proceso de aprendizaje supervisado, que inicia con un conjunto de objetos clasificados correctamente (conjunto de entrenamiento) y cuyo objetivo es diseñar un Sistema de Clasificación, asignando etiquetas de clase a nuevos objetos con un mínimo de error. Finalmente, se calcula el desempeño del sistema sobre los datos de prueba para obtener una estimación acerca del error de predicción del SClBRD. El proceso se ilustra en la Figura 4.3.

Base de Conocimiento. La Base de Conocimiento (BC) esta compuesta de la BR y la BD. En la literatura especializada, se han usado diferentes tipos de reglas y

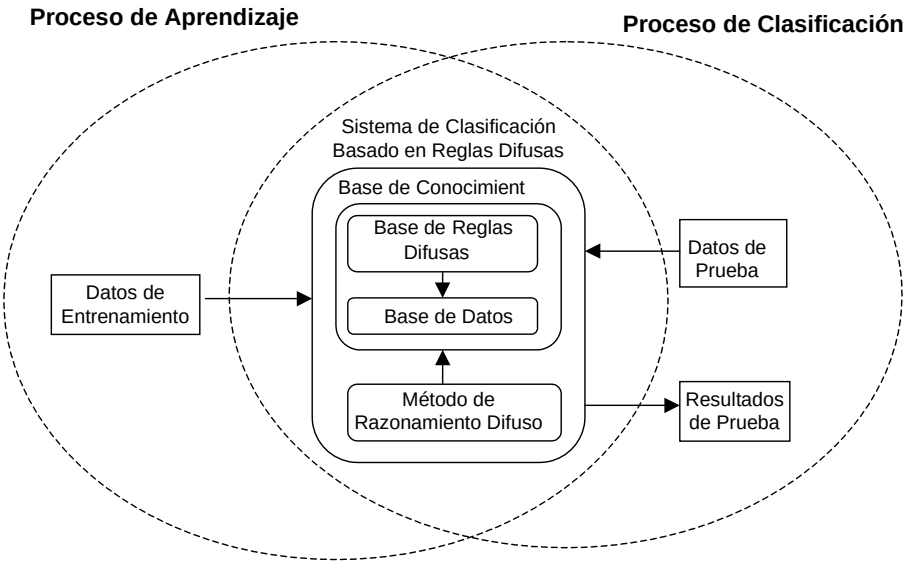

Figura 4.3: Diseño de un SClBRD (Aprendizaje/Clasificación)

su diferencia consiste en la composición del consecuente: una clase ([AT97] [CDJHb]), una clase y un grado de certeza asociado a la clasificación de esa clase [INT], y el grado de certeza asociado a la clasificación de cada una de las clases posibles [MMP].

Un SClBRD esta compuesto de una BR del siguiente tipo de reglas:

$$R_k: \ Si \ x_1 \in A_1^k \ \wedge \ldots\ldots \ \wedge \ x_n \in A_n^k \ \text{ entonces } \ Y \in C_j \text{ con } \ r^k$$

donde:

- $x_1,\ldots\ldots x_n$ son los atributos seleccionados para el problema de clasificación.
- $A_1^k,\ldots\ldots A_n^k$ son etiquetas lingüísticas usadas para discretizar los dominios de los atributos cuantitativas o cualitativas.
- Y es la clase $C_j \ \in \{C_1,\ldots\ldots C_\xi \ \}$ a la que pertenece el objeto.
- y, r_k es el grado de certeza de la clasificación en la clase C_j para un objeto que pertenece al subespacio difuso definido por el antecedente de la regla.

Base de Datos. La Base de Datos (BD) contiene la definición de los conjuntos difusos asociados a los términos lingüísticos usados en la BR. Esta transformación es común para todas las reglas en la BR para mantener la naturaleza lingüística de los SClBRD.

Método de Razonamiento Difuso. Un Método de Razonamiento Difuso (MRD) es un procedimiento de inferencia, que deriva conclusiones a partir de un conjunto de reglas difusas y un objeto. El uso de un método de razonamiento que combine la información de la reglas disparadas por el objeto ha ser clasificado, puede mejorar la capacidad de generalización del Sistema de Clasificación.

Un modelo de razonamiento general lo podemos describir en la siguiente forma:

En la clasificación de un objeto $E_t = (x_{t1}, x_{t2}, \ldots\ldots, x_{tK})$, la base de reglas $R = \{R_1, \ldots\ldots, R_L\}$ esta dividido en ξ subconjuntos de acuerdo a la clase indicada por su consecuente,

$$R = R_{C_1} \cup R_{C_2} \cup \ldots \cup R_{C_\xi}$$

y siguiendo el esquema siguiente:

1. **Grado de Compatibilidad**. El grado de compatibilidad del antecedente con el objeto se calcula para todas las reglas en la BR, aplicando una t-norma [[ATV83], [DP85]] sobre el grado de pertenencia de los valores del individuo (e_{ti}) a los correspondientes subconjuntos difusos.

$$R_k(E_t) = T(\mu_{A_1^k}(e_{t1}), \ldots, \mu_{A_n^k}(e_{tn})), \quad \text{k} = 1, \ldots, \text{L}$$

2. **Grado de Asociación**. El grado de asociación del objeto E_t con las ξ clases se calcula de acuerdo a cada regla en la BR.

$$b_i^k = h(R_k(E_t), r_k), \ \text{k} = 1, \ldots, \mid R_{C_i} \mid \quad \text{i} = 1, \ldots, \xi$$

3. **Función de Ponderación**. Los valores obtenidos son ponderados por medio de una función g. Una expresión que promueve los valores altos y penaliza los pequeños parece ser la selección más adecuada para esta función.

$$B_i^k = g(b_i^k), \quad \text{k} = 1, \ldots, \mid R_{C_i} \mid \ \text{i} = 1, \ldots, \xi$$

4. **Grado de validez de la clasificación para todas las clases**. Para calcular este valor, se usa un operador de agregación que combine, para cada clase, el grado de asociación positivo calculado en el paso anterior.

$$Y_i = \text{f}(B_i^k, \text{k} = 1, \ldots, \mid R_{C_i} \mid \ \text{i} = 1, \ldots, \xi \quad \text{y} \quad B_i^k > 0)$$
$$\text{i} = 1, \ldots, \xi \ \text{ con } \ f \text{ un operador de agregación.}$$

   El operador f regresa un valor entre el mínimo y el máximo. Si se selecciona f como el operador máximo tenemos el Modelo de Razonamiento Difuso Clásico.

5. **Clasificación**. Se aplica una función de decisión F a los grados de clasificación del individuo. Esta función regresa la etiqueta de clase que corresponde al valor máximo.

$$C_l = F(Y_1, \ldots, Y_\xi) \ \text{ tal que } Y_l = max_{j=1,\ldots,\xi} Y_j$$

Así, en los Sistemas de Clasificación Basados en Reglas Difusas (SClBRD), el Método Clásico de Razonamiento Difuso (MCRD), grado máximo de asociación, clasifica un nuevo objeto del dominio con el consecuente de la regla con el grado más alto de asociación ([AT97], [CYT96], [CDJHb], [INT], [Kun], [MMP]). Usando este método de inferencia, se pierde información proporcionada por las otras reglas difusas con diferentes etiquetas lingüísticas que representan también el valor en el atributo patrón (clase), aunque probablemente con menor grado.

Por otro lado, es bien conocido que en otros SBRD como los controladores lógicos difusos el mejor desempeño se obtiene cuando se usan métodos de defuzificación que operan sobre subconjuntos difusos obtenidos de las reglas difusas satisfechas (aquellas cuyos datos de entrada satisfacen sus antecedentes), tomando en consideración todas ellas para obtener el valor de la salida vía el método de defuzificación [CHP].

4.8 Descripción del proyecto marco

4.8.1 Introducción

El presente trabajo de tesis se ubica dentro de los métodos híbridos y se integrá en un proyecto de investigación marco que dirige la Dra. Karina Gibert. La línea de investigación inicia en 1995 con el objetivo principal de estudiar los *dominios poco estructurados* [GC94].

La primera propuesta constituye la tesina [Gib91] y después la tesis doctoral de Karina Gibert [Gib94] que cristalizó en la formulación de la *metodología de clasificación basada en reglas* y una primera versión del sistema informático que la implementa, denominado KLASS [Gib94] y que se ha utilizado en diversas aplicaciones [GC93a, GC93b, GHC96, GC98, GS99, GSM00].

Desde su inicio se han producido diversas ampliaciones del sistema que le han permitido evolucionar y abrir nuevas posibilidades de investigación. En la subsección §4.4 se aporta una breve cronología que destaca las etapas más relevantes del proyecto marco.

El objetivo del proyecto marco es construir una plataforma integrada de soporte al análisis inteligente de *dominios poco estructurados*, incluyendo todo tipo de herramientas, desde las más básicas de análisis descriptivo hasta las más sofisticadas como la *clasificación basada en reglas* y herramientas de *apoyo a la interpretación de resultados*, relacionadas con la minería de datos y el proceso KDD [GAC98].

Considerando las características especiales de este tipo de dominios, se han desarrollado métodos mixtos de análisis que combinan técnicas estadísticas con técnicas de inteligencia artificial para resolver los problemas que se plantean en este contexto [GA98].

Todo el *software* que se desarrolla en el seno del proyecto marco se acaba integrando a lo que podríamos llamar *herramienta master*, que actualmente es $joc.KLASS+$, y que aglutina herramientas de muy distinta naturaleza ofreciendo la interfaz necesaria para que puedan comunicarse entre ellas y transferir la información necesaria en cada momento del análisis.

Esta herramienta informática ha venido evolucionando de forma continua desde su origen en la medida en que se ha avanzado en la investigación y experimentación de la línea de investigación antes mencionada.

En el seno de este proyecto marco se han desarrollado distintos proyectos de fin de carrera (PFCs) tanto de la Diplomatura en Estadística como de la Ingeniería en Informática en todos sus niveles (superior y técnicas).

Actualmente existe un grupo de personas investigando y trabajando en equipo, entre los que se están desarrollando dos proyectos de tesis doctoral del programa de doctorado en Inteligencia Artificial de la UPC.

4.8.2 Evolución del proyecto marco

- **KLASS v0.** Tesina de Ingeniería informática de Karina Gibert. "Klass. Estudi d'un sistema d'ajuda al tractament estadístic de grans bases de dades". Clasifica matrices de datos heterogéneas usando una distancia mixta definida especialmente para ello [Gib91, GC97] (febrero 1991).

- **KLASS v1.** Tesis doctoral en Informática de Karina Gibert. "L'us de la Informació Simbòlica en l'Automatització del Tractament Estadístic de Dominis poc Estructurats."Es una ampliación de **KLASS v0.** Incorpora la clasificación basada en reglas [Gib94, GC94] (noviembre 1994).

  Herramienta informática, orientada a la clasificación automática de *dominios poco estructurados*, implementada en LISP y lenguaje C. Ha sido desarrollada en el departamento de EIO de la UPC e implementa la metodología de *clasificación basada en reglas*, que en pocas palabras, es una *estrategia mixta de clasificación automática* que usa una combinación de métodos basados en el conocimiento (Inteligencia Artificial) y clasificación ascendente jerárquica (tradicionalmente de la Estadística).

- **xcn.KLASS** PFC de Ingeniería Informática de Xavier Castillejo. Incorpora a **KLASS v1** una interfaz de ventanas independientes, implementado en C, que comunica con el núcleo LISP. Existe una versión PC de la interfaz que facilita el uso de KLASS (sobre SUN) desde PCs a usuarios que desconocen LISP y UNIX [Cas96] (julio 1997).

- **jj.KLASS** PFC la Diplomatura en Estadística de Juan José Márquez y Juan Carlos Martín. Incorpora a la versión **KLASS v1** nuevas opciones para el tratamiento de datos faltantes, la posibilidad de trabajar con objetos ponderados e implementa un test no paramétrico de comparación de clasificaciones [MM97] que se aplicó al análisis de disfunciones de tiroides [GS97, GS99] (octubre 1997).

- **xt.KLASS** PFC de Ingeniería Informática de Xavier Tubau. Incorpora a la versión **xcn.KLASS** cuatro métricas mixtas más y el módulo nuevo de comparación de clasificaciones [GAC98] de **jj.KLASS** [Tub99] (septiembre 1999). Sobre esta versión Angela Twore desarrolló un PFC de la DE, diseñando un experimento para estudiar el comportamiento de las distintas métricas, así como el análisis estádistico de los resultados.

- **KLASS+** PFC de Ingeniería Técnia en Informática de Sistemas Sílvia Bayona. Fusión definitiva de las versiones **xt.KLASS** y **jj.KLASS**. Además incorpora un módulo nuevo de análisis descriptivo y de ayuda a la interpretación [GA00, GSM00, CDG$^+$01] de datos y de clases resultantes, con el propósito de reorientar KLASS, haciéndola más general [Bay00] (2000). Sobre esta versión, el proyecto de la diplomatura estadística de Begoña Gómez, consistió en el desarrollo de herramientas de muestreo y de clasificación basada en bootstrap, las cuales aún están por integrarse.

Situación actual del proyecto marco Al momento de iniciar el desarrollo de *CIADEC*, se estaba consolidando la versión *joc.KLASS+* paralelamente y de forma independiente. Esta situación, muestra que el proyecto marco es dinámico y constantemente se alimenta de nuevas investigaciones y experimentos. Actualmente existe un equipo de 8 personas trabajando de forma coordinada en las siguientes tareas:

- PFC de Diplomatura de Estadística de Miguel Angel Nieto. Compilación de técnicas de Minería de Datos y de Descubrimiento de Conocimiento.

- **joc.KLASS+** PFC de Ingeniería en Informática de la UIB de Josep Oliveras. Incorpora a la versión *sbh.KLASS+* tres métricas mixtas más: Gower [Gow71], Diday-Gowda [DG92], la métrica generalizada de Minkowski [IY94] propuesta por Ichino y Yaguchi.

- PFC de Licenciatura en Estadística FME de Angela Twore. Diseña el experimento de comparación de resultados para las nuevas métricas.

- PFC de Licenciatura en Estadística FME de Juan Carlos Martín. Analiza el impacto de categorizar previamente las variables numéricas sobre el clustering en el contexto de las disfunciones tiroideas.

- COLUMBUS Tesis doctoral programa de IA de la UPC de Jorge Rodas. Diseño de la metodología para el descubrimiento de conocimiento en medidas seriadas muy cortas y repetidas con factor de bloque (KDSM). Construcción de un primer satélite de *joc.KLASS+*, denominado COLUMBUS que implementa la metodología KDSM.

- **CIADEC** Tesis doctoral programa de IA de la UPC de Fernando Vázquez. Diseño de la metodología para la caracterización e interpretación automática de descripciones conceptuales, en *dominios poco estructurados*, con variables numéricas (*AUGERISD*)[VG01a, VG01b, VG02b]. Construcción en Java, de un satélite de *joc.KLASS+* denominado *CIADEC* (segunda parte de este documento) que implementa la metodología (*AUGERISD*). Actualmente existe ya un prototipo de *CIADEC* que se encuentra en fase de experimentación y pruebas.

- **java.KLASS** PFC de Ingeniería en Informática de Gema Gómez. Consiste en traducir al lenguaje de programación Java el núcleo LISP de la versión más reciente de KLASS, se integrarán todos los módulos y satélites (*COLUMBUS* y CIADEC) C y Java en una interfaz general con total transparencia para el usuario.

- PFC de Ingeniería en Informática de Mar Colillas. Programación en Java del módulo de análisis descriptivo de datos y clases.

En la Figura 4.4 se aprecia el panorama general de la evolución del programa master del proyecto marco.

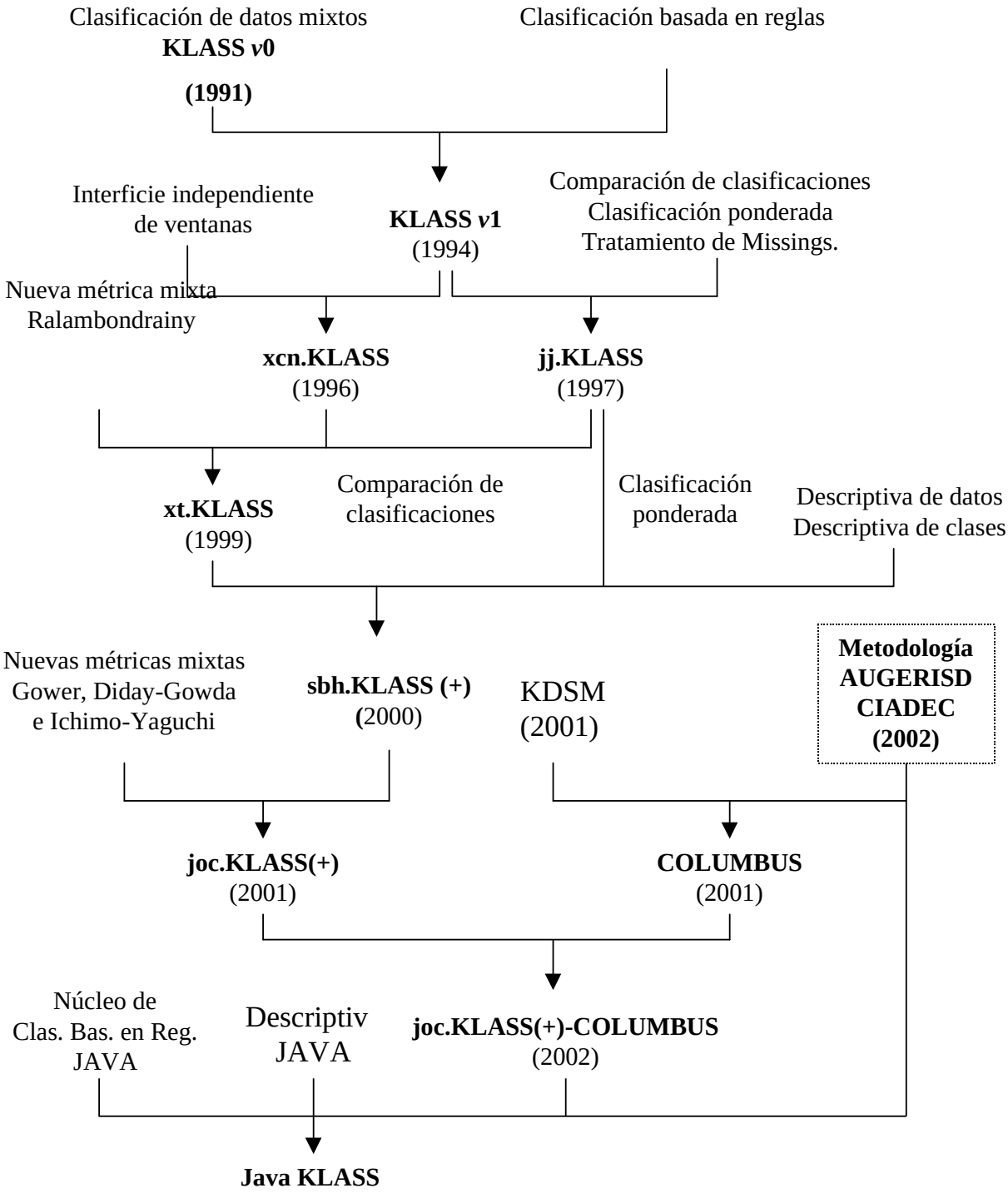

Figura 4.4: Cronología del proyecto marco.

Capítulo 5

Propuesta Metodológica

5.1 Introducción

El objetivo fundamental de este trabajo es la descripción conceptual de una partición viable obtenida sobre el dominio por cualquier medio (por ejemplo, a través de $KLASS^+$ o $LINNEO^+$). Partiendo de los trabajos previos sobre la interpretación a partir de atributos cualitativos [Gib94], en donde se analizó la caracterización de clases a partir de conceptos fundamentales como: la variable caracterizadora, el sistema caracterizador y la variable ε–caracterizadora; además, en [GS00, Rod99] se ha encontrado que el *box-plot múltiple* es una herramienta ágil y potente con atributos numéricas para identificar elementos útiles, considerándolo como la base de la propuesta metodológica para la detección de las variables características en atributos cuantitativos que se propone en este proyecto de tesis.

El punto de partida nos sitúa de pleno en el empleo de técnicas de clasificación automática que particionen los datos del dominio de estudio en un conjunto de clases realizando una clasificación utilizando el método de clasificación basada en reglas, propuesto por [Gib94, GC94], donde se introduce el conocimiento adicional, parcial y no homogéneo que posee el experto del dominio a través de un conjunto de reglas (predicados lógicos de CP_1) para que actúe como un sesgo semántico durante el proceso de clasificación, mejorando la comprensión de las clases obtenidas.

Nuestra aproximación a lo que sería un proceso automático de interpretación de clases tiene su origen en la idea del boxplot múltiple. Así, la metodología aunque inspirada en esta herramienta gráfica estadística ha sido automatizada usando algoritmos no gráficos, calculando los valores máximo y mínimo de cada clase, procediendo a una ordenación ascendente del total de estos valores a las distintas clases. Los extremos de los intervalos de longitud variable a generar serán los valores contiguos dos a dos. Con ello se construye la tabla de contingencia entre los intervalos y las clases, lo que dará cuantas observaciones hay en cada clase para cada intervalo. A partir de esta tabla se obtienen las distribuciones condicionadas a cada intervalo y que resulta en el porcentaje de elementos de cierto intervalo en cada clase. Así, podemos asociar a un objeto cualquiera su grado de pertenencia a cada clase. Esta idea da lugar a un gráfico de grados de pertenencia difusos para cada clase y cada variable. A partir de aquí es fácil conectar la metodología con un modelo de creación de etiquetas lingüísticas que generen automáticamente las interpretaciones de las descripciones conceptuales de las clases.

La propuesta metodológica aporta un sistema de caracterización de clases, basado en predicados de lógica de primer orden ($CP1$), que permiten máxima potencia y flexibilidad para detectar atributos cuantitativos caracterizadores en algunas clases, permitiendo un procedimiento de generación automático de reglas, que formarán parte de la base de conocimiento de un sistema orientado a la predicción y/o diagnóstico. Además, la automatización de esta metodología ofrecerá un conjunto de herramientas de apoyo a la interpretación como: la construcción de un sistema de reglas, visualización de las funciones de pertenencia de una variable X_k a las distintas clases C, evaluación de objetos nuevos de acuerdo a las reglas generadas y validación de la calidad de la predicción teniendo como base un conjunto de nuevos objetos.

5.2 Metodología

Los pasos que conforman la metodología son los siguientes:

1. **Descripción estadística de las variables**

   En esta primera etapa, se utilizan algunas técnicas descriptivas clásicas que permiten identificar el comportamiento y naturaleza de los datos sobre la matriz $\mathcal{X}$. Esta etapa sirve para obtener información preliminar acerca de la variabilidad de las mediciones y para representar los *box-plots múltiples*, que permiten observar la relación entre las variables y las clases y, en especial es útil para representar las diferencias entre grupos.

2. **Uso del *box-plot múltiple* como herramienta gráfica, para la detección de variables caracterizadoras**

   En esta parte, el *box-plot* múltiple se usa como una herramienta para visualizar y comparar la distribución de una variable a través de todas las clases. En esta parte representación, podemos identificar lo que se denominan *variables caracterizadoras* de la clase C, concepto que descansa a su vez en el concepto de *valor propio* de una clase C. Así, definimos los siguientes conceptos [Gib94]:

   - Un valor $c_s^k \in D_k$ de la variable X_k es *propio* de la clase C, si cumple:

     $$(\exists i \in C : x_{ik} = c_s^k) \wedge (\forall i \notin C : x_{ik} \neq c_s^k)$$

     Estos valores, cuando ocurren, identifican una clase con toda seguridad, por lo que, los llamaremos *valores caracterizadores* de C y los denotamos por λ_{sc}^k, siendo C la clase y k la variable.

   - Una variable X_k es *parcialmente caracterizadora* de la clase $C \in \mathcal{P}$ si tiene al menos un valor propio de la clase C, aunque puede compartir alguno con otras clases; llamemos V_C^k al conjunto de valores *parcialmente caracterizadora de* C:

     $$V_C^k = \{c_j^k : c_j^k \text{ es } \textit{valor propio} \text{ de } X_k \text{ para la clase } C\}$$

- Una variable X_k es *totalmente caracterizadora*[1] de la clase $C \in \mathcal{P}$, si todos los valores que tiene X_k en la clase C son *propios* de C. En este caso, denotamos por Λ_C^k el conjunto de estos valores, los cuales caracterizan totalmente a la clase C:

$$\Lambda_c^k = \{c_j^k : c_j^k \in V_c^k \wedge \forall C' \neq C, c_j^k \notin V_{c'}^k\}$$

Es muy fácil observar si el box-plot de cierta clase no interseca con el de las demás; en un caso así, la variable es *totalmente caracterizadora* [2]. A veces, sólo es una parte del box-plot la que no interseca; en ese caso se trata de una variable *parcialmente caracterizadora*.

Para identificar estas variables, estudiaremos los valores propios que toma una variable X_k en una clase C, *en relación* a las otras y poder ver si son de la clase o no; para ello hay que analizar cómo son las interacciones entre clases.

3. **Estudio de interacciones entre clases**

En este proceso, es de nuestro interés considerar las variables, en su estado natural, evitando cualquier transformación arbitraria sobre su naturaleza, que pudieran alterar el sentido de la iteracción.

Esta etapa consiste en identificar todas las intersecciones que se dan entre los valores de las variables y las distintas clases, determinando en qué puntos del rango de las variables están cambiando estas intersecciones; así podemos identificar las distintas combinaciones de clases donde se puede dar un mismo valor de cierta variable y como consecuencia hacer emerger los valores propios (caracterizadores) de una clase; éstos nos identificarán variables total o parcialmente caracterizadoras.

Sin embargo, en la práctica no se puede basar un proceso automático en la interpretación de una representación gráfica, por lo que en los siguientes apartados se propone una alternativa equivalente, pero automatizable.

4. **Sistema de intervalos o ventanas de longitud variable**

Estas intersecciones se pueden encontrar de forma exacta con un coste computacional mínimo, solamente calculando los valores mínimos y máximos por variable y clase y ordenándolos en forma conveniente. A partir de dicha ordenación, se define una discretización de la variable en un conjunto de intervalos, sobre los que se podrá identificar los valores propios de una variable en todas las clases.

Formalizando estos conceptos tenemos que, si m_C^k y M_C^k son los mínimos y los máximos de la variable X_k en la clase $C \in \mathcal{P}$, observados de la descriptiva o del *box-plot múltiple*, donde $m_C^k = min_{i \in C}\{x_{ik}\}$ y $M_C^k = max_{i \in C}\{x_{ik}\}$. Ahora se procede a ordenarlos en forma ascendente, este proceso consiste en:

- Definir $\mathcal{M}^k$ como el conjunto de todos los mínimos y máximos correspondientes a la variable X_k, en todas las clases de $\mathcal{P}$, esto es:

[1] Son aquellas variables que toman valores en individuos de la clase C que no son tomados por ningún otro objeto de las otras clases y viceversa.

[2] En lo sucesivo le llamaremos *variable caracterizadora* de esa clase.

$$\mathcal{M}^k = \{m^k_{c_1}, \ldots, m^k_{c_\xi}, M^k_{c_1}, \ldots, M^k_{c_\xi}\}$$

siendo la $card(\mathcal{M}^k) = 2\xi$

- Ordenando $\mathcal{M}^k$ de menor a mayor, se construye un conjunto $\mathcal{Z}^k$ de forma:

$$\mathcal{Z}^k = \{z^k_i \ ; \ i = 1 : 2\xi\}$$

tal que:
i) $z^k_1 = min\mathcal{M}^k$
ii) $z^k_i = min(\mathcal{M}^k \setminus \{z^k_j \ ; \ j < i\}), \ i = \{2, \ldots, 2\xi\}$
Dado que $\mathcal{Z}^k = \{z^k_i\}$ es un conjunto ordenado, sus elementos tienen la siguiente propiedad:

$$\mathcal{Z}^k = \{z^k_j | z^k_{j-1} < z^k_j; \ 1 < j \leq 2\xi\}$$

A este conjunto se le denomina *puntos de corte*.

- A partir de este conjunto ordenado, construimos el sistema de intervalos de longitud variable I^k en la siguiente forma:

$$I^k = \{I^k_s : 1 \leq s \leq 2\xi - 1\}$$

donde
i) $I^k_1 = [z^k_1, z^k_2]$
ii) $I^k_s = (z^k_s, z^k_{s+1}], \quad (s = 2 : 2\xi - 1)$

De ahí se define una nueva variable categórica I^k cuyo conjunto de valores es $\mathcal{D}^k = \{I^k_1, \ldots, I^k_{2\xi-1}\}$. I^k identifica todas las intersecciones entre clases que define X_k, representando un sistema de intervalos de longitud variable asociado a dicha variable.

Así, si tenemos 2ξ puntos de corte diferentes se generán a lo más $2\xi - 1$ intervalos y la $card(\mathcal{D}^k) = 2\xi - 1$, recordando que ξ es el número de clases de la partición de referencia que se quiere caracterizar.

Además, siendo D^k el dominio de X_k, $\mathcal{D}^k$ representa una categorización del mismo, pero no es arbitraria en absoluto, y además se calcula de forma inmediata. Por último, hay que observar que para construir I^k ya no hace falta realizar el *box-plot múltiple*, aunque éste sigue siendo una excelente representación de lo que se esta haciendo.

5. **Construcción de la tabla de contingencia de clases vs intervalos**

En esta etapa se realiza la construcción de la tabla de contingencia para una variable X_k, como una matriz de números A, en la cual los renglones están representados por los intervalos I^k encontrados en la etapa anterior y las columnas, por las clases de la partición $\mathcal{P}$ de referencia; así, una cierta casilla de la matriz A, indica el número de elementos de $\mathcal{I}$, cuyos valores de X_k se encuentran en el intervalo representado por I^k_s. En general, para un cierto valor de la variable X_k se tienen objetos en distintas clases. De esta forma definimos la tabla de contingencia como $A = I^k \times \mathcal{P} = (n_{sc}(s = 1 : 2\xi - 1), (c \in \mathcal{P}))$, donde n_{sc} es la

$card\{i \in C \wedge x_{ik} \in I_s^k\}$, es decir, n_{sc} es el *número* de elementos de C cuyo valor de X_k *está* en I_s^k, teniendo la matriz A dimensión constante $(2\xi - 1, \xi)$ porque ésta sólo depende de ξ.

Usaremos I^k para caracterizar las clases de $\mathcal{P}$, para ello buscaremos si I^k tiene algún valor *propio* o *parcialmente caracterizador* en alguna clase. Intuitivamente, los valores propios son valores exclusivos de la clase C y gráficamente son muy fáciles de identificar en un *box-plot múltiple*, quedando la misma información reflejada en la tabla A.

La característica de un valor *propio* o *parcialmente caracterizador* de I^k en la clase C sobre una tabla de contingencia A es tal que cumple:

> I_s^k es valor *propio* o *parcialmente caracterizador* de la clase C si
>
> - $n_{sc} \neq 0$, y
> - $\forall\ C' \neq C,\ n_{sc'} = 0$

Si además

> - $\forall\ s' \neq s,\ \ n_{s'c} = 0$
>
> entonces I_s^k es un valor *totalmente caracterizador* de C.

Como en lo habitual se encuentran pocos valores *totalmente caracterizadores*, en sentido estricto, lo común, son los valores *propios* o *parcialmente caracterizadores*. Es decir, valores que determinan parte de una clase, la cual tiene que cuantificarse para poder determinar el poder de caracterización de dichos valores.

Sea $(1-\varepsilon),\ \varepsilon \in [0,1]$ el grado de caracterización de una clase C, para un valor. Ya en [Gib94] aparece la idea de $(1-\varepsilon)$—*caracterización* y se maneja en todos los trabajos posteriores a nivel de variable. Ello conduce a situaciones en apariencia complejas como el hecho de que X_k sea $(1-\varepsilon_1)$—*caracterizadora* de C y también $(1-\varepsilon_2)$—*caracterizadora* de C' con $\varepsilon_1 \neq \varepsilon_2$.

En realidad esto sucede porque lo que determina el poder de caracterización no es la variable en sí, sino los valores que toma y su distribución a lo largo de las clases. Así, de ahora en adelante, trasladaremos a nivel de valores este análisis.

Diremos que, dada una variable X_k

> Un valor *$(1-\varepsilon)$—caracterizador* de C es aquel *valor propio* de C que sólo identifica $(1-\varepsilon)\,\%$ de C.

Existe aún una tercera situación, que corresponde al patrón que llamaremos *valor caracterizador no propio*, el cual satisface la siguiente propiedad:

> - I_s^k es un *valor no propio* de la clase C si cumple:
> $n_{sc} \neq 0\ \wedge\ \forall s' \neq s\ \ n_{s'c} = 0$

Para analizar los valores concretos de ε en la partición $\mathcal{P}$ será necesario un análisis previo que pasará por la tabla de contingencia $I_s^k \times \mathcal{P}$, entre otras cosas.

6. **Construcción de la tabla de distribuciones condicionadas a los intervalos**

Es fácil construir ahora la tabla de distribuciones condicionadas a cada intervalo, de modo que las casillas representan una estimación de la probabilidad de que un elemento x_{ik} de un cierto intervalo I_s^k, pertenezca a una clase específica C.

Podemos representar la tabla de distribuciones condicionadas como una matriz de la forma $B = I^k \times \mathcal{P}$, cuyos valores toman la forma:

$$B = (p_{sc}(s = 1 : 2\xi - 1), (c = 1 : \xi))$$

siendo ξ la cardinalidad de $\mathcal{P}$ ($card(\mathcal{P})$), p_{sc} la frecuencia relativa de los individuos de valor $X_k \in I_s^k$ que se encuentran en la clase $C \in \mathcal{P}$ y cuyo valor $p_{sc} = n_{sc}/n_{I_s^k}$, donde n_{sc} es el número de individuos que pertenecen al intervalo I_s^k y a la clase C, y $n_{I_s^k} = \Sigma_{c=1}^{\xi} n_{sc}$ es el número total de objetos que se encuentran en el mismo intervalo I_s^k.

De acuerdo a la construcción de la tabla de distribuciones condicionada B, la podemos caracterizar por las siguientes propiedades:

- Para los valores de la variable I^k (renglones) en cada uno de los intervalos I_s^k se tienen probabilidades p_{sc} en el sentido frecuentista de que un elemento x_{ik} le sea asignada la clase C, cumpliendo con:
  i) $p_{sc} \in [0, 1]$
  ii) $\Sigma_{i=1}^{\xi} p_{sc_i} = 1$

En la tabla de frecuencias condicionadas B, los valores *caracterizadores*, de la clase C son todavía más fácil de identificar, porque se detectan observando una sola casilla de la clase y pueden ser *parcialmente caracterizadores* o *totalmente caracterizadores* dependiendo si existe o no interacción entre clases. Así, tenemos que:

- Un valor I_s^k de la clase C es un valor *propio* o *parcialmente caracterizador* si su frecuencia $p_{sc} = 1$.
- Un valor I_s^k de la clase C es un valor *totalmente caracterizador* si su frecuencia $p_{sc} = 1{,}0,\ \ p_{s'c} = 0,\ \ \forall\ s' \neq s$.
- I_s^k es un valor *caracterizador no propio* si $p_{sc} \in (0, 1)$.

Visto cómo se identifican los valores *caracterizadores*, vamos ahora a cuantificar el *grado de caracterización* tal y como ya se definió.

> El valor I_s^k de la variable X_k será $(1 - \varepsilon)$—caracterizador de C si $n_{sc} = (1 - \varepsilon) \cdot n_c$

El *grado de caracterización* en este contexto, se interpreta como la parte proporcional (porcentaje) de individuos de $\mathcal{C}$, cuyos valores de la variable X_k se encuentran en el intervalo I_s^k.

REGLA	CONJUNTO ANTECEDENTE		
	$I_s^k \subset C$	$I_s^k = C$	Probabilidad
$x_{ik} \in I_s^k \longrightarrow C$	propio parcial caract.	propio total caract.	$p_{sc} = 1$
$x_{ik} \in I_s^k \xrightarrow{p_{sc}} C$		total caract. no propio	$p_{sc} \in (0,1)$

Tabla 5.1: Relación entre reglas de asociación y valores propios

7. **Generación del sistema de reglas** $\mathcal{R}(X_k, P)$

   Así, para cada valor propio (total o parcial) de la clase C, se puede extrae una regla que *identifica* la clase con el mínimo de información, de la forma:

   $$(X_k \in \Lambda_c^k) \longrightarrow C$$

   donde X_k es la k—ésima variable, Λ_c^k es el conjunto de valores *propios* de la clase C.

   Ahora bien, si un valor es *caracterizador no propio* entonces, cuando se da ese valor, la clase de asignación puede ser una u otra con distintos grados de certeza. De ahí que, la regla

   $$(X_k \in I_s^k) \longrightarrow C$$

   deje de ser segura.

   Podemos definir p_{sc} como el grado de certeza de esa regla, entendiendo que p_{sc} (frecuencia relativa sobre una muestra) constituye una buena estimación puntual de la probabilidad de que un objeto que toma valores en ese intervalo I_s^k, pertenezca realmente a la clase C.

   Así, si I_s^k es un caracterizador no propio de C, podemos generar una regla:

   $$x_{ik} \in I_s^k \xrightarrow{p_{sc}} \ i \in C$$

   donde p_{sc} lo podemos definir en forma equivalente como una probabilidad condicional $P(C|I^k = I_s^k)$ en la siguiente forma:

   $$p_{sc} = P(C|I^k = I_s^k) = card\{i \text{ tal que } x_{ik} \in I_s^k \wedge i \ \in C\}/\ n_{I_s^k}$$

   De hecho, p_{sc} está indicando con qué probabilidad el objeto i pertenece a la clase correcta C a partir del valor de X_k, considerando que existen otros individuos que toman valores en I_s^k y se dispersan en las demás clases.

   El esquema en la Tabla 5.1 establece la relación entre el conjunto antecedente I_s^k donde se encuentra el valor de la variable I^k, la forma de la regla de asociación y el valor de su probabilidad de asignación p_{sc} a la clase C.

   De ahí se observa que los valores propios siempre generan reglas seguras, pero el poder de caracterización depende del cardinal del conjunto antecedente. Si este coincide con toda la clase, entonces hay una caracterización completa de la misma. Sino, es parcial. Se observa que la Tabla 5.1 tiene una casilla vacía.

Esta casilla identifica un cuarto caso que responde a un cuarto patrón; se trata de la situación más general que le llamaremos *valor genérico* y que permitirá generar caracterizadores parciales y no seguros, representando este el caso más débil de todos. Se define así

- Un valor I_s^k de la variable I^k es un *valor genérico* de la clase C si
  i) $p_{sc} \in (0,1) \wedge$
  ii) $\exists\ s'$ tal que $p_{s'c} \neq 0, \quad s' \neq s, \ \wedge$
  iii) $\exists\ c'$ tal que $p_{sc'} \neq 0, \quad c' \neq c.$

Estos valores los podemos interpretar como el subconjunto de individuos i de la clase C que comparten su valor I_s^k tanto con las demás clases, existiendo a su vez en la misma clase C algunos otros elementos que pertenecen a otros intervalos.

A partir de los conceptos anteriores, se puede realizar la siguiente identificación, en relación a los valores *caracterizadores*.

- Si I_s^k es el valor de la variable I^k (un intervalo de X_k), y $p_{sc} \in (0,1]$ es su frecuencia condicionada para la clase C entonces podemos generar para cada elemento de la Tabla B reglas de la forma:

$$Si\ x_{ik}\ \in\ I_s^k\ para\ el\ objeto\ i\ \xrightarrow{p_{sc}}\ i\ \in\ C$$

  donde: x_{ik} es el valor de la k–ésima variable para el i–ésimo objeto, I_s^k el intervalo al que pertenece dicho valor y C es la clase caracterizada a partir de I_s^k con probabilidad p_{sc}.

Esta definición es general y cubre como casos particulares las reglas resultantes de los valores propios de C, que incluye los valores a $p_{sc} = 1$, que corresponden a las reglas seguras.

Así, para cada tabla de distribución condicionada a intervalos B se puede derivar el siguiente sistema de reglas asociado a X_k para identificar cierta partición $\mathcal{P}$.

$$\begin{aligned}\Re(X_k, \mathcal{P}) = \{\ r_l\ :\ x_{ik} \in I_s^k \xrightarrow{p_{sc}} i \in C \quad con\ \ p_{sc} > 0,\ \ p_{sc} \in B,\\ l = \{\ 1, \ldots, (2\xi - 1)\xi\},\ s = \{1, \ldots, (2\xi - 1)\}\}\end{aligned}$$

Este sistema ha de permitir identificar las distintas clases a partir de X_k.

Fijada una sola clase C que se quiere caracterizar, las probabilidades de todas las reglas que presentan a C como parte derecha pueden verse como una *distribución de posibilidades* [DPB99] y [LdM90] que asigna a cada valor de la variable I^k su grado de pertenencia a la clase C y que se representa como un gráfico (ver Figura 5.1) con cada una de las funciones horizontales. Cabe mencionar, que el área bajo estas funciones ya no es 1, puesto que se componen de probabilidades que provienen de distintas distribuciones condicionadas (las de $C|\mathcal{I} = I_s^k,\ \forall s$). Así, definimos

$$\pi_k^C(x_{ik}) \stackrel{\text{def}}{=} p_{sc}, \quad x_{ik} \in I_s^k$$

Para cada elemento de la partición $\mathcal{P}$ (columnas de las matrices A y B) que son las distintas clases, se tiene una distribución de posibilidad π_k^C, que indica el *grado de compatibilidad* del valor de X_k con la asignación a C. En esta distribución se tiene un número finito de *niveles de posibilidad* de C, distinguiendo valores entre lo "imposible" (codificado por 0) y lo "completamente posible" (codificado por 1).

Parafraseando lo anterior, se tiene que para toda $x_{ik} \in I_s^k$, $\pi_k^C(x)$ representa hasta qué punto es posible que cierto valor de X_k implique la pertenencia a C. La función π_k^C representa una restricción flexible de los valores de la variable X_k con las siguientes convenciones:

- $\pi_k^C(x_{ik}) = 0$, significa que la pertenencia a la clase C es imposible;
- $\pi_k^C(x_{ok}) > 0$, significa que la pertenencia a la clase C es posible a distintos grados (pj., débil, fuerte, muy fuerte etc.), tanto más intenso cuanto más se acerque a 1, que representa la pertenencia segura.

Finalmente, se obtiene un sistema global que contiene reglas difusas o posibilistas, a partir del cual, para cierto valor del atributo X_k se da con mayor o menor grado de pertenencia a cada clase de cierta partición de referencia $\mathcal{P}$.

A partir de aquí, veremos como (apartado §9 de esta sección) la representación gráfica de este sistema permite generar interpretaciones automáticas de las clases.

8. **Validación del Sistema Global de Reglas**

En la metodología propuesta, los *box-plot múltiples* se han usado para la determinación de los valores característicos, considerando como base un sistema de intervalos de longitud variable. Esto permite identificar cual es la estructura natural que subyace en la base de datos del dominio de estudio variable por variable. Esto, ha permitido desarrollar un método rápido para construir un sistema de reglas difusas asociadas a cada variable, el cual queda reflejado en la tabla de distribuciones condicionadas a intervalos $B = \mathcal{P}|I^k$. Un primer propósito, aún en fase de desarrollo, es reducir la ambigüedad inherente al sistema de reglas $\Re(X_k, \mathcal{P})$ considerando el criterio de grado más grande de asociación (con el consecuente de la regla con la probabilidad máxima), el cual nos conduce a un sistema reducido $\Re'(X_k, \mathcal{P})$ mucho más pequeño en número de reglas, sin ambigüedad pero conservando incertidumbre.

Como una aplicación práctica, la evaluación del sistema de reglas consiste en tomar cada uno de los elementos del conjunto de prueba y evaluarlos en el correspondiente sistema de reglas de la partición de referencia. Así, considerando la variable X_k y una partición de referencia, tomamos cada valor $x_{ik} \ \ \forall i$ en el conjunto de prueba y los evaluamos en el sistema de reglas reducido $\Re'(X_k, \mathcal{P})$, en cada caso, para cada valor x_{ik} se localizan los intervalos I_s^k, la clase C y la probabilidad correspondiente. Es decir, si existe una regla

$$\mathrm{r} : x_{ik} \in I_s^k \xrightarrow{p_{sc}} C,$$

la clase C se asigna al individuo i con un grado de pertenencia p_{sc} considerando únicamente la variable X_k. El resto de las variables se evalúan de igual forma.

Este proceso continua hasta agotar todas las variables de todos los individuos en el conjunto de prueba.

El siguiente paso es considerar de acuerdo al un criterio de agregación de información elegido, la combinación de todas las variables por individuo del conjunto de entrenamiento y determinar el número de individuos mal clasificados para calcular el error de predicción del sistema de reglas como un parámetro de validación del sistema de reglas generado.

9. **Interpretación de Clases**

   La interpretación de las clases resultantes es siempre de gran importancia para usar los conocimientos generados como herramientas de apoyo a la posterior toma de decisiones. Incluso se ha llegado a decir que la validación de una clasificación, se ha considerado como el grado de interpretabilidad y/o utilidad de éstas, sin ningún otro criterio que el de un especialista que mira las clases resultantes.

   Teniendo, como base la tabla de distribuciones condicionadas a los intervalos analizada en el apartado §6 de esta sección, se puede asociar a un individuo cualquiera i su *grado de pertenencia* a cada clase. Esto hemos dicho da lugar a un gráfico de grados de pertenencia difusos para cada clase y para cada variable como se muestra en la Figura 5.1. En el gráfico el eje horizontal es común y representa el rango de X_k; para cada clase se representa el grado de pertenencia de los valores de X_k según las reglas. La forma escalonada de dichas funciones de pertenencia se debe a la categorización de X_k en I^k. Así, dado un valor de X_k se visualiza fácilmente su relación con las otras clases.

   Se observa que a partir de esta representación gráfica, el paradigma difuso [VG02a] constituye un excelente soporte al proceso de interpretación.

   Esto es, porque el sistema $\Re(X_k, \mathcal{P})$ contendrá reglas con el mismo antecedente (I_s^k) y partes derechas diferentes (clases) en distintos grados de pertenencia. Por otro lado, una clase C se reconoce por muchas reglas, lo que trae consigo problemas de imprecisión e incertidumbre en el modelo de razonamiento asociado a la caracterización de la clase. Esto es claramente visible en la representación gráfica de la Figura 5.1 y evidencia que se presenta una situación compleja que por sus características se presta a su contextualización en el paradigma de los conjuntos difusos [AG91] y [AGR93], su extensión la lógica difusa y la teoría de la posibilidad; los que constituyen un excelente soporte para representar y manejar piezas de información que contienen tanto la imprecisión como la incertidumbre, como es el caso en la determinación de la clase C de un objeto $i \in \mathcal{I}$.

   A partir de aquí, debemos soportar el proceso con un método de creación de etiquetas lingüísticas que genere descripciones conceptuales de las clases del estilo:

   > Si la variable X_k toma valores muy altos entonces ese objeto i se asocia a $C03$

   donde, el grado de pertenencia de una variable específica X_k al concepto "muy altos" vendría determinado precisamente por un gráfico de $C03$ como el de la Figura 5.1. Así, una vez que se ha asignado la clase C a un nuevo individuo,

podemos analizar los gráficos de distribución variable por variable para obtener conocimiento útil y comprensible en la interpretación conceptual de la clase identificada y su relación con otras clases.

5.3 Estado actual

Hoy, la metodología AUGERISD se está implementando en un sistema denominado CIADEC, cuya descripción es objeto en la parte II de este proyecto (capítulo §10).

El estado actual del sistema CIADEC desarrolla las siguientes tareas: a partir de una base de datos descrita con atributos cuantitativos y una partición de referencia viable, genera los sistemas de intervalos de longitud variable para cada atributo seleccionado, la tabla de distribución condicionada a intervalos por atributo, el sistema global y reducido de reglas por atributo, los gráficos de las funciones de pertenencia para cada atributo seleccionado y clase, el gráfico de asignación de clase por atributo para cada objeto nuevo.

Como trabajo futuro consultar el cronograma de actividades de la tesis, Tabla 8.1.

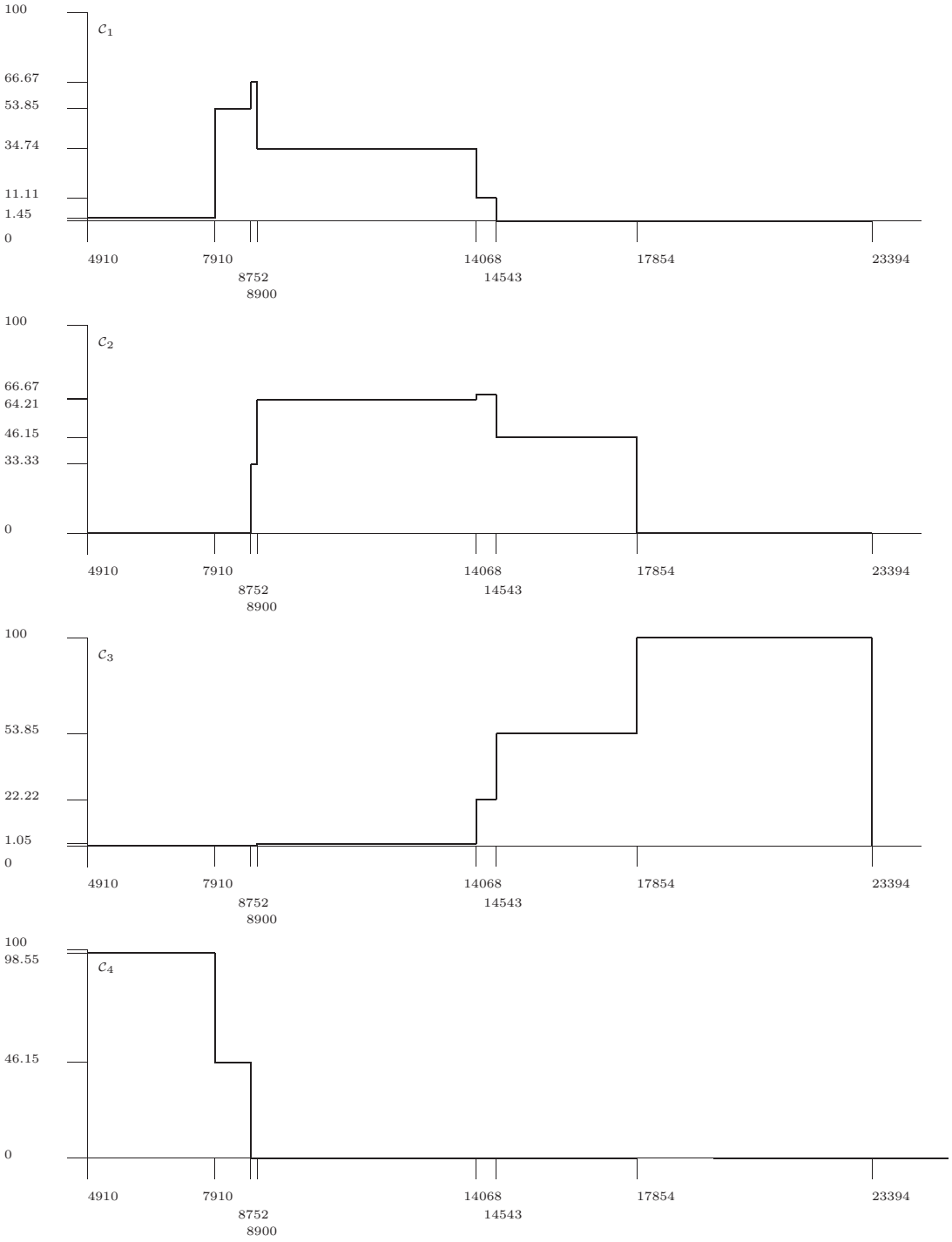

Figura 5.1: Diagrama de grados de pertenencia a las clases de la variable Q-AB

Capítulo 6

Caso de Estudio

6.1 Introducción

En este capítulo realizaremos la aplicación de la metodología descrita a los datos de la planta depuradora de aguas residuales.

Las grandes áreas urbanas producen gran cantidad de aguas residuales[1],y cuando el medio ambiente está contaminado y la calidad del agua empeora debido a que el proceso residual llega a superar el desempeño de la auto-regulación de las aguas recibidas. En este caso, se deben tomar ciertas medidas previsorias para restaurar el equilibrio del medio ambiente [RPSM].

Las plantas depuradoras de aguas residuales proporcionan un importante equilibrio entre el medio ambiente y las aguas residuales concentradas de las áreas urbanas. Si estas últimas se liberan de forma descontrolada, se degradaría el medio ambiente, elemento esencial para el bienestar de los seres humanos.

Para tratar adecuadamente las aguas residuales son necesarias distintas operaciones y procesos unitarios. El diagrama del proceso de una estación depuradora incluye diferentes combinaciones de agentes físicos, químicos y biológicos (ver proceso global en la Figura 10.1). Esta última representa un esquema típico, así como la secuencia lógica de tratamiento, dividida en diferentes fases, las que son resumidas brevemente a continuación (para mayor detalle referirse a [SM95] y [Rod99]).

El **pretratamiento** es la primera etapa para la depuración de aguas residuales. En esta fase, se realiza una primera separación de los sólidos, arrastrados por el agua residual cuando llega al recolector. Con ello se pretende evitar obstrucciones posteriores y otros problemas sobre las bombas o válvulas utilizadas a lo largo de todo el proceso. Esta operación física se realiza mediante una secuencia de rejas, que abren y cierran automáticamente.

El **tratamiento primario** corresponde a la segunda etapa del proceso. En esta fase, se deja reposar el agua en un tanque de sedimentación primaria, para que decante la materia orgánica sedimentable, el resto de la arena o partículas inorgánicas, que no se han retenido en el *pretratamiento*.

Posteriormente, se lleva a cabo, la etapa más importante del proceso, conocido co

[1] "toda combinación de líquidos o aguas que transportan residuos procedentes de residencias, instalaciones públicas y centros comerciales e industrias, a las cuales, de manera eventual,se pueden agregar aguas subterráneas, superficiales y pluviales-[Rod99].

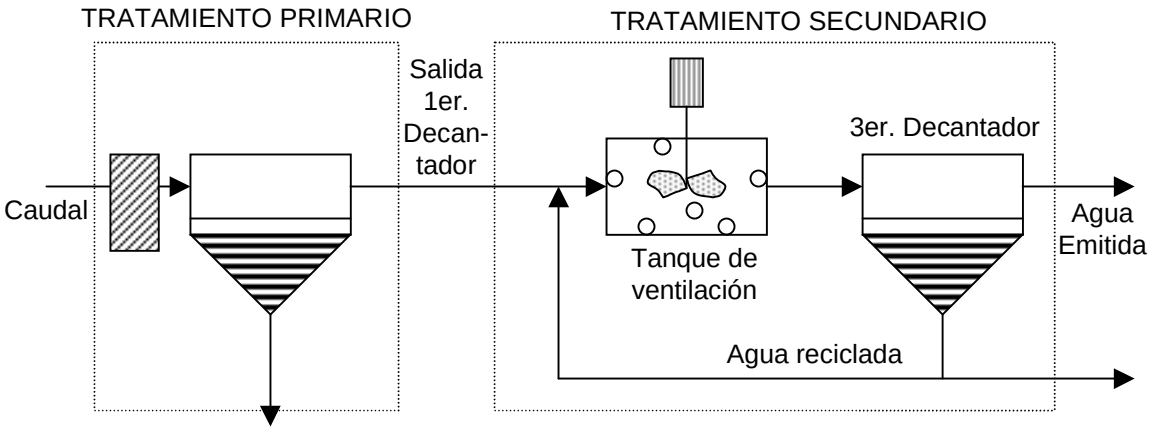

Figura 6.1: Diagrama típico del proceso de tratamiento de aguas residuales

biológico del agua, es decir, la *degradación* de la materia orgánica disuelta en el agua residual. Lo anterior ocurre por la acción de una población multiespecífica de microorganismos, conocida como *biomasa*. La reacción que se produce, tiene lugar en uno o varios reactores biológicos, dependiendo del número de reactores que tenga la planta. Finalmente, una nueva decantación se lleva a cabo en los sedimentadores secundarios, para emitir posteriormente el agua. El objetivo del proceso antes descrito, es conseguir una buena separación, entre el agua ya tratada y la *biomasa*.

Los sólidos sedimentados de ambas fases de decantación son enviados (purga) hacia una línea de tratamiento específico, conocida como *"línea de barros"* y eventualmente, realimentan la *biomasa* del reactor biológico.

Cuando la planta depuradora no funciona bajo condiciones normales, se deben tomar decisiones para modificar algunos parámetros del proceso de depuración y restablecer lo antes posible la normalidad. Por esto, es importante contar con un sistema automatizado, que nos proporcione la información relevante sobre la situación que la planta tiene en un momento especifíco. En nuestro caso, la investigación esta orientada a hacer aportaciones en ese sentido.

Como se mencionó anteriormente, un buen conocimiento sobre la situación de la planta en tiempo real constituye un excelente apoyo a la gestión de la misma. Por ello, el objetivo de esta aplicación es presentar la metodología expuesta en el capítulo §5 para la generación automática de descripciones conceptuales, que caractericen las distintas situaciones que se pueden presentan en un cierto día (registro) en la planta depuradora.

Se partirá de dos clasificaciones de referencia para identificar situaciones típicas. A partir de estas clasificaciones, se propone un modelo conceptual que determine los atributos relevantes involucradas en el proceso y describa e interprete las diversas situaciones que se presentan en un cierto día en cada una de ellas y mostrar que la predicción de clases depende de la partición de referencia.

6.2 Presentación de los datos

Los datos analizados en esta aplicación provienen de una planta depuradora de la Costa Catalana (España), y están formados por un total de 218 observaciones, obtenidos consecutivamente el mismo número de días. Cada observación corresponde a la media diaria de repetidas mediciones sobre un conjunto de 63 atributos.

El conjunto de datos cuantitativos y cualitativos que se recogen en la planta depuradora, describe el estado de la planta a través de un conjunto de 63 atributos, algunas de las cuales se midieron en distintos puntos de la planta (AB: a la entrada de la planta, SP1: después del primer decantador, B: en el reactor biológico, SP3: después del tercer decantador, AT: a la salida de la planta) y otras se obtuvieron por cálculos a partir de las primeras [SMCLP97].

Los expertos recomiendan trabajar con un subconjunto de 19 atributos, 17 de las cuales son numéricas y son: Q-AB (caudal a la entrada de la planta), DQO-AB (materia orgánica química a la entrada), COND-AB (conductividad eléctrica a la entrada de la planta), DQO-SP1 (materia orgánica química en el primer decantador), Q-SP3 (caudal a la salida del tercer decatador), DQO-AT (materia orgánica química en el agua tratada, a la salida), SST-AT (total de sólidos en suspensión), NH4-AT (amonio sobre el agua tratada), NO3-AT (nitrato sobre el agua tratada), IVF (índice volumétrico de fangos), CM (carga másica), ESC-B (presencia de espuma en el tanque de ventilación), ASP-AT (calidad del agua tratada), ZOO (*Zooglea*, NFILAM (número de bacterias filamentosas diferentes), BIODIV-MIC (biodiversidad de la microfauna en el fango activo), P-FLAG (*Flagelados* $> 20\mu m$) y 2 atributos categóricas, FILAM (bacteria filamentosa dominante) y FLOC (copos de fango activado). En este trabajo, hemos tomado como referencia estos atributos, las cuales se representan como X_k, tanto los atributos cuantitativos como las cualitativos.

Este conjunto de datos $\mathcal{I}$ de 218 días han sido previamente clasificados por la herramienta $Linneo^+$ y el sistema híbrido denominado $Klass^+$ y lo consideraremos como el conjunto de entrenamiento T_0 y otro conjunto de 25 nuevos individuos también previamente clasificados serán usados para validación de nuestro sistema de reglas y le denominaremos conjunto de prueba P_0.

6.3 Particiones de referencia: $Linneo^+$ y $Klass^+$

Partición de $Linneo^+$ $\mathcal{P}_L$. El estado de la planta (el atributo clase) fue previamente identificado por medio de un proceso de clasificación semi-automático usando la herramienta $Linneo^+$ y el criterio del experto.

$Linneo^+$, que es una herramienta de adquisición de conocimiento semi-automática utilizada en la construcción de clasificaciones para dominios poco estructurados, fue el software utilizado para particionar los datos.

Después de un proceso iterativo de clasificación supervisada por el experto, los 218 fueron clasificados en 20 situaciones típicas que ocurren en la planta. Estas 20 clases corresponden a los clusters obtenidos con la clasificación de $Linneo^+$ usando un radio igual a 10 excepto para dos clases no detectadas con este radio. Otras clasificaciones con diferentes radios descubrieron otros dos nuevos clusters que corresponden a dos estados de la planta.

Aunque la clasificación de $Linneo^+$ es una clasificación que los expertos han dado una interpretación válida; también se ha observado que como en un proceso de clasificación automática, los propios expertos reconocen que no es la única y que incluso podría ser mejorada. La validación de esta clasificación no ha sido contrastada previamente por medio objetivos, razón por la cual se propone una segunda partición obtenida por $Klass^+$.

Partición de $Klass^+$ $\mathcal{P}_K$. El sistema $Klass^+$ presenta diferencias importantes con respecto a otros clasificadores: el procesamiento de información simbólica y una metodología específica con restricciones declarativas, de ahí que, $Klass^+$ se considera como una herramienta de ayuda a la adquisición de conocimiento, cubriendo un doble propósito:

- Implementar un método de clasificación con restricciones basada en el conocimiento.

- Una herramienta de ayuda a la adquisición de conocimiento basada en métodos estadísticos, orientada a la gerenación de reglas para un sistema de diagnóstico y/o predicción.

La base de la metodología de $Klass^+$ es un método de clasificación ascendente jerárquico, que utiliza el algoritmo de vecinos recíprocos encadenados. La estrategia de clasificación consiste, en detectar los pares de vecinos recíprocos que han sido fusionados y construir el árbol de agregaciones.

En esta aplicación la métrica mixta y el criterio de Ward se han usado para clasificar el conjunto de entrenamiento T_0 usando además la metodología basada en reglas. Básicamente, se realizan dos procesos de agrupamiento: uno por las reglas del experto y el otro para los objetos que no satisficieron las reglas del experto llamada clase residual. Ambas clasificaciones jerárquicas son integradas en una sola partición para el conjunto total T_0. La Figura 6.2 representa el dendrograma final.

La clasificación de $Klass^+$ para los 218 individuos del conjunto de entrenamiento se hizo tomando en cuenta las reglas dadas por el experto y haciendo un corte del árbol igual a 20. Las clases que se obtuvieron son las siguientes:

$$
\begin{array}{rr}
Clase\hat{C}01 = Classer164, & Clase\hat{C}02 = Classer179, \\
Clase\hat{C}03 = Classer118, & Clase\hat{C}04 = Classer176, \\
Clase\hat{C}05 = Classer197, & Clase\hat{C}06 = Classer140, \\
Clase\hat{C}07 = Classer165, & Clase\hat{C}08 = Classer196, \\
Clase\hat{C}09 = ClasserCp1, & Clase\hat{C}10 = Classer194, \\
Clase\hat{C}11 = Classer191, & Clase\hat{C}12 = D50, \\
Clase\hat{C}13 = Classer204, & Clase\hat{C}14 = Cs0, \\
Clase\hat{C}15 = Classer170, & Clase\hat{C}16 = Classer198, \\
Clase\hat{C}17 = Classer202, & Clase\hat{C}18 = D07, \\
Clase\hat{C}19 = Classer201, & Clase\hat{C}20 = Classer173
\end{array}
$$

Comparación entre las particiones $Linneo^+$ **y** $Klass^+$. Analizando los elementos que contienen cada una de las 20 clases en la clasificación de $Klass^+$ y comparándola con la obtenida con $Linneo^+$ se han obtenido los siguientes resultados :

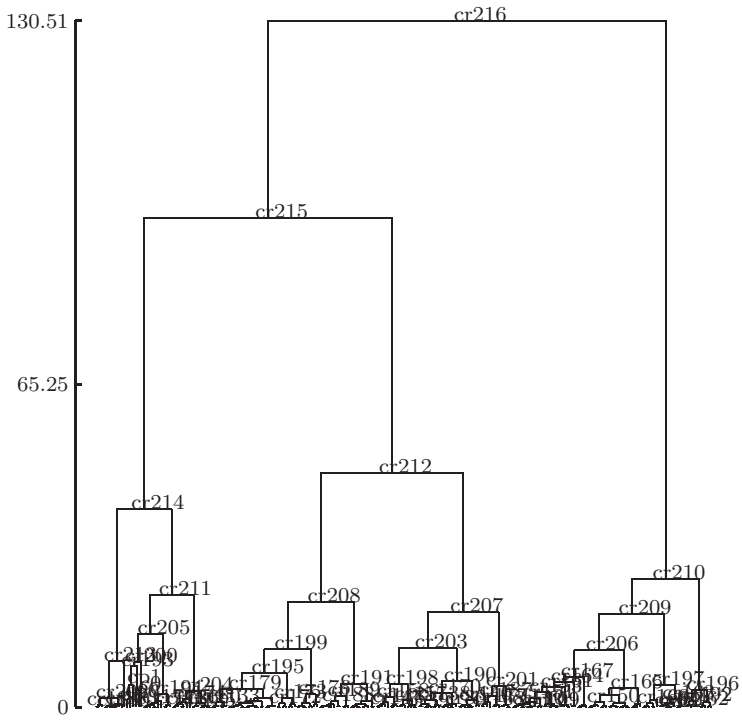

Figura 6.2: Árbol general del clustering basado en reglas para un corte en 20 clases

- De las 20 clases entre ambas clasificaciones, siete clases se identifican como muy similares, las relaciones de estas clases entre ambas clasificaciones son:
- Haciendo una tabla cruzada (ver Tabla 6.1) entre las dos clasificaciones, obtenemos una matriz donde los elementos de la diagonal representan los elementos comunes entre diferentes clases de ambas clasificaciones, teniendo un total de 49 objetos en clases similares representando un 43.11 % de elementos coincidentes y el resto de los objetos se dispersan en las otras clases, formando clases diferentes con características diferentes.

6.4 Análisis por atributo

El sistema CIADEC §10 (Caracterización e Interpretación Automática de Descripciones Conceptuales en Dominios Poco Estructurados usando Variables Numéricas) [VG02b] es un sistema que implementa la metodología AUGERISD [VG01a] "Generación Automática de Reglas Difusas en Dominios Poco Estructurados con Variables Numéricas", la cual permite la caracterización e interpretación automática de descripciones conceptuales en dominios poco estructurados previamente clasificados, combinando conceptos y técnicas de estadística e inteligencia artificial y lógica difusa.

Además, la automatización de esta metodología ofrecerá un conjunto de herramientas que permitan:

- Construir un sistema de reglas para la predicción de clases, diagnóstico...
- Visualizar las funciones de pertenencia de un atributo X_k a las distintas clases.

CLASES	$\hat{C}01$	$\hat{C}02$	$\hat{C}03$	$\hat{C}04$	$\hat{C}05$	$\hat{C}06$	$\hat{C}07$	$\hat{C}08$	$\hat{C}09$	$\hat{C}10$	$\hat{C}11$	$\hat{C}12$	$\hat{C}13$	$\hat{C}14$	$\hat{C}15$	$\hat{C}16$	$\hat{C}17$	$\hat{C}18$	$\hat{C}19$	$\hat{C}20$	$Klass^+$
C01	*19*				6		18	4	1	1			8		5	5		1	4		72
C02		*23*		3		6					5		1		1	2			4	4	49
C03			*1*	2																	3
C04				*2*																	2
C05	1		3		*4*								2								10
C06																	1				1
C07					1			1													2
C08								*2*					1				2				5
C09									*2*												2
C10	2					1				*8*				1		4					16
C11				1							*15*				6						22
C12												*1*									1
C13	1					1							*2*		1				1		6
C14														*1*						1	2
C15							1	1							*1*	1					4
C16								1								*3*			1		5
C17										1							*7*				8
C18								1													1
C19					4														*1*		5
C20																				*2*	2
$Linneo^+$	23	23	4	8	15	8	19	10	3	10	20	1	14	2	14	15	10	1	11	7	218

Tabla 6.1: Comparación entre las particiones de $Linneo^+$ and $Klass^+$

- Evaluar un conjunto de objetos nuevos de acuerdo a las reglas generadas.
- Validar la calidad de la asignación teniendo un conjunto de prueba P_0.

Como una estrategia de trabajo, se aplicará la metodología haciendo el análisis para el atributo DQO-AT (materia orgánica química en el agua tratada) y la partición de referencia dada por $Klass^+$, de tal forma que el método pueda seguirse de cerca. Esto se hace con el fin de ilustrar la ejecución de la metodología y permitir comentarios específicos en cada paso y posteriormente dar los resultados para el resto de los atributos.

1. **Descripción estadística de las variables**

   De acuerdo con el conjunto de datos obtenidos, lo primero que se realizó fue una descripción estadística, la cual permitió obtener información preliminar como: número de objetos pertenecientes a cada clase de la clasificación de referencia; la media, la mediana, la desviación estándar, los valores mínimos, máximos y atípicos (outliers) para cada atributo en cada una de las clases; un grupo de atributos con un 35 % aproximado de valores pérdidos (NH4-AT, NO3-AT, IVF, CM y BIODIV-M), cabe aclarar que estas son las que se miden con poca frecuencia en la clase; en el resto de los atributos se observó menos del 4 %.

2. **Uso del *boxplot múltiple* como herramienta gráfica, para la detección de variables caracterizadoras**

   Con el boxplot múltiple, se visualiza la distribución de los valores de cada una de los atributos por clases. En nuestro caso, el primer boxplot múltiple fue para el atributo DQO-AT (ver Figura 6.3) el cual consiste de una representación gráfica que muestra

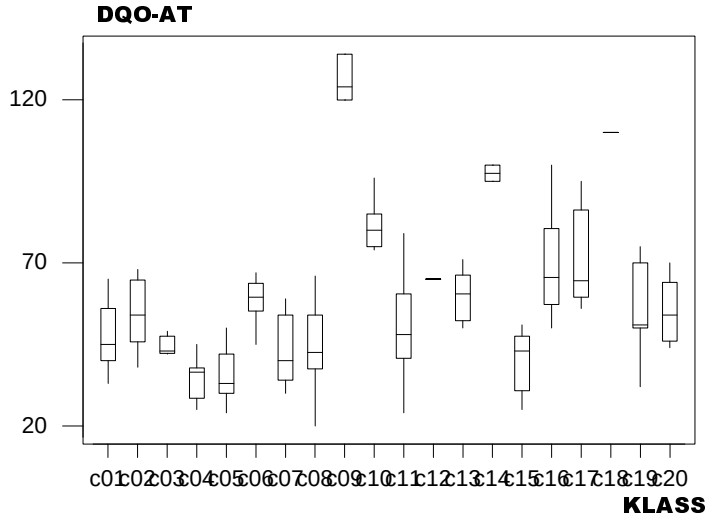

Figura 6.3: boxplot del atributo DQO-AT.

como los valores del atributo por clases se distribuyen. En cada boxplot por clase, los valores outliers o valores atípicos son marcados por "*", se despliega una caja desde Q1 (primer quartil) hasta el Q3 (tercer quartil) e incluye un 50 % de elementos de la clase, la mediana de los valores esta marcada dentro de la clase con un signo horizontal y los bigotes se extienden hasta el mínimo y el máximo por clase.

A partir de la observación de los boxplots se determinan los *valores caracterizadores, en sentido estricto* [2] de las clases. Gráficamente, se puede apreciar si la proyección horizontal del *boxplot* de cierta clase no interseca con la de las demás; en un caso así, el atributo, es totalmente caracterizador de esa clase.

Por ejemplo, si observamos el *boxplot* del atributo DQO-AT (materia orgánica química en el agua tratada) en la Figura 6.3, cualquier valor en (120, 134]mg/l es *totalmente caracterizador* para la clase $\hat{C}09$. Esto se debe a que en ese intervalo ningún otro boxplot interseca con el boxplot de la clase $\hat{C}09$ y todos sus valores están comprendidos en dicho intervalo. Lo anterior significa que, cualquier día en donde DQO-AT (materia orgánica química) tomara un valor en el intervalo (120, 134]mg/l estará en la clase $\hat{C}09$ y viceversa todos los días que hay en la clase $\hat{C}09$ contienen un valor de DQO-AT entre (120, 134]mg/l. Otro valor en el intervalo (100, 110] es *totalmente caracterizador* para la clase $\hat{C}18$ del mismo atributo.

3. **Determinación de la interacción entre clases**

   Se observa en la Figura 6.3 que existen intervalos del atributo DQO-AT donde pueden coincidir $\hat{C}02$, $\hat{C}06$, $\hat{C}07$, $\hat{C}11$, $\hat{C}13$, $\hat{C}16$, y $\hat{C}17$, como es el valor de $DQO - AT =$ 57mg/l, el cual se encuentra en el intervalo I_{19}^{DQO-AT}; u otros donde intersecan las clases $\hat{C}05$, $\hat{C}07$, $\hat{C}15$ y $\hat{C}19$, como es el caso para el valor de 31 mg/l localizado en el intervalo $I_6^{DQO-AT} = (30, 32]mg/l$.

   Queda claro que, el poder informativo asociado al valor del atributo DQO-AT depende directamente del cardinal del conjunto de clases que interseca. Por ello, estudiar dónde cambia ese cardinal es extremadamente adecuado.

[2] También puede calificarse de las siguientes maneras: totalmente caracterizadores o seguros.

4. **Sistema de intervalos o ventanas de longitud variable**

   La generación de los intervalos o ventanas de diferentes longitudes, se realiza tomando los puntos de corte contiguos dos a dos del conjunto $\mathcal{Z}^k$. Esto dio como resultado un sistema de intervalos abiertos por la izquierda y cerrados por la derecha, excepto el primer intervalo en cada atributo, el que se considera cerrado por ambos lados. Esta forma de presentar los intervalos se debe a las características propias de la herramienta utilizada. Con ello se dispone de un atributo categórico $\mathcal{I}^{DQO-AT}$ asociada a DQO-AT, que indica todas las intersecciones entre clases. Para el atributo DQO-AT, tenemos el siguiente sistema de intervalos:

$$\begin{array}{rrr} I^{DQO-AT} = & \{I_1^k = [20,24), & I_2^k = [24,24], \\ I_3^k = \ (24,25], & I_4^k = (25,25], & I_5^k = (25,30], \\ \ldots, & \ldots, & \ldots, \\ \ldots, & \ldots, & \ldots, \\ I_{34}^k = (96,100], & I_{35}^k = (100,100], & I_{36}^k = (100,110], \\ I_{37}^k = (110,110], & I_{38}^k = (110,120], & I_{39}^k = (120,134]\} \end{array}$$

5. **Construcción de la tabla de contingencia de clases vs intervalos**

   Una vez obtenido el sistema de intervalos, se construye la tabla de contingencia entre los intervalos y las clases. En los renglones marcamos los intervalos y en las columnas las clases; ya se dijo que las intersecciones de renglón con columna contienen el número de observaciones que hay en cada clase para cada intervalo.

   Ya hemos comentado que I^{DQO-AT} representa una categorización *no arbitraria* de DQO-AT, que hace emerger *todos* los puntos donde cambian las intersecciones entre clases.

   Así, mientras el valor de I_1^{QDO-AT}, indica el intervalo [20, 24]ml/l de DQO-AT, también está indicando la *zona en la que se da la clase* $\hat{C}08$. Por el modo como se ha construído I^{DQO-AT} estamos seguros de que precisamente a partir de 24 mg/l de materia orgánica química serán otras las clases que se puedan dar simultáneamente.

6. **Construcción de la tabla de distribuciones condicionadas a los intervalos**

   Una vez obtenida la tabla de contingencia, se construye la tabla de distribuciones condicionadas a intervalos para el atributo de estudio DQO-AT. A partir de esta tabla para cada valor $x_{iDQO-AT}$ se le asigna una probabilidad p_{sc}, $s = 1, \ldots, 2\xi - 1$, que representa el grado de pertenencia del individuo i a la clase C, de acuerdo a este atributo. Y las probabilidades asociadas a la clase C pueden verse como una distribución de posibilidades que asigna a cada valor del atributo I^k su grado de pertenencia a la clase C y dicha distribución puede representarse por medio de un gráfico (ver Figura 5.1).

   Para el atributo de estudio DQO-AT se pueden reconocer los cuatro tipos de valores característicos:

   - Valores propios *totalmente caracterizadores*:
     $I_{36}^{DQO-AT} = (100, 110]$ de $\hat{C}18$ y
     $I_{39}^{DQO-AT} = (120, 134]$ de $\hat{C}09$
   - Valores propios *parcialmente caracterizadores* :
     $I_{28}^{DQO-AT} = (71, 74]$ de $\hat{C}10$
     $I_{33}^{DQO-AT} = (95, 96]$ de $\hat{C}10$

- Valores *No propios* son:
  $I_{20}^{DQO-AT} = (59, 65]$ de $\hat{C}12$
- Y, los valores *Genéricos*: El resto de los valores no nulos en la tabla de distribución son valores genéricos.

7. **Generación del sistema de reglas $\mathcal{R}(X_k, P)$**

   Interpretando p_{sc} como una estimación de la probabilidad de que $i \in C$ dado que $i \in I^k$ $(C|I^k)$ podemos obtener un sistema de reglas que represente los grados de pertenencia de un día a cada clase de acuerdo a un atributo dado.

   Considerando las distribuciones condicionadas a los intervalos, como distribuciones de posibilidad (grado de imprecisión), podemos asociar a un objeto (día) cualquiera, su(s) grado(s) de pertenencia a la(s) clase(s) y obtener un sistema de reglas, que representen el grado de pertenencia a algunas de las clases. Esto da lugar a un gráfico de grados de pertenencia para cada clase y para cada atributo, que representa una buena ilustración de lo que ocurre en una situación real (ver Figura 6.4). Así, si tomamos el valor de $x_{iDQO-AT} = 93mg/l$ y trazamos una línea vertical sobre él, se obtienen los grados de pertenencia de este valor a cada una de las clases. Esta forma de representar gráficamente este sistema permite obtener conocimiento útil y comprensible para la interpretación conceptual de las clases identificadas.

   Con respecto al atributo DQO-AT, esta etapa del proceso genera un sistema total $\Re(DQO - AT, \mathcal{P})$ de 123 reglas, una por cada celda no nula en la matriz de distribuciones condicionadas a intervalos para dicho atributo. Como la mayoría de los intervalos presenta diferentes grados de pertenencia a diferentes clases, esto genera un número de reglas con diferentes consecuentes dentro del mismo intervalo. Por ejemplo, si el atributo DQO-AT toma el valor de 92.6 mg/l, éste se localiza en el intervalo $I_{31}^{DQO-AT} = (79, 95]$ y satisface cuatro reglas en el sistema global de reglas con diferentes grados de pertenencia. En este caso particular, el grado de pertenencia a la clase $\hat{C}10$ es 0.40, a la clase $\hat{C}14$ es 0.10, a la clase $\hat{C}16$ es 0.20, a la clase $\hat{C}17$ es 0.30. Para el resto de las clases es 0.0, por lo tanto, hay cuatro reglas para asignar clases en este día, de acuerdo al nivel de DQO-AT. En notación de cálculo de predicados de primer orden, las expresamos de la siguiente manera:

$$\text{Si } x_{ik} \in (79, 95] \xrightarrow{0,40} i \in \hat{C}10, \qquad \text{Si } x_{ik} \in (79, 95] \xrightarrow{0,10} i \in \hat{C}14$$
$$\text{Si } x_{ik} \in (79, 95] \xrightarrow{0,20} i \in \hat{C}16, \qquad \text{Si} x_{ik} \in (79, 95] \xrightarrow{0,30} i \in \hat{C}17$$

   Esto presenta una situación ambigua; la decisión puede ser problemática. Como una primera aproximación al proceso de tomar una decisión, proponemos reducir el conjunto de reglas de cada intervalo I_s^k a sólo una regla. Siguiendo el criterio del modelo clásico de razonamiento aproximado para sistemas de clasificación difusa respecto a seleccionar la regla que presente probabilidad máxima en cada intervalo [CDJHa]. Esto corresponde a un criterio de agregación de información muy fuerte, que elimina la modelación difusa que tanto hemos defendido, con su consiguiente pérdida de información. Sobre esta decisión convendrá hacer un análisis a profundidad más adelante, pero de momento permite reducir la ambigüedad del sistema de reglas resultante, que llamaremos *Sistema Reducido de Reglas* $\Re'(DQO - AT, \mathcal{P})$. Evidentemente en este sistema de reglas hay como máximo una regla por intervalo, con lo que un conjunto de $card(\Re(X_k, \mathcal{P})) = (2\xi - 1)\xi$, llega a ser de $card(\Re'(X_k, \mathcal{P})) = 2\xi - 1$.

8. **Validación del Sistema Global de Reglas**

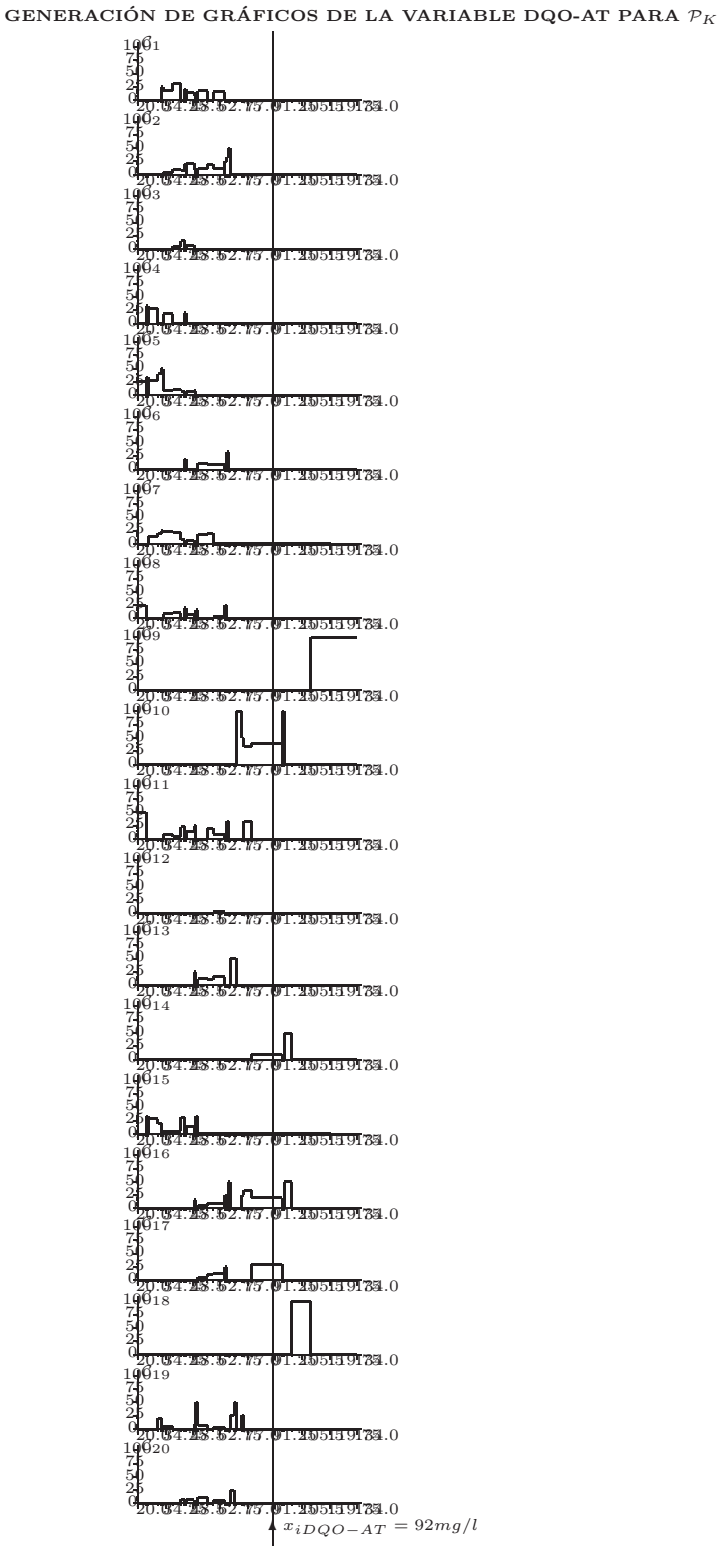

Figura 6.4: Gráfico de las funciones de pertenencia para el atributo DQO -AT

REF	$\hat{P}01$	$\hat{P}02$	$\hat{P}04$	$\hat{P}05$	$\hat{P}07$	$\hat{P}09$	$\hat{P}10$	$\hat{P}11$	$\hat{P}13$	$\hat{P}15$	$\hat{P}18$	$\hat{P}19$
$\hat{C}01$	*15*	2		1	4			1				
$\hat{C}02$	9	*18*			1			1		1		
$\hat{C}03$	1	1								2		
$\hat{C}04$	1		*2*		4							
$\hat{C}05$	2	1	3	*4*	2					1		
$\hat{C}06$	6											
$\hat{C}07$	7	3	1	2	*5*					1		
$\hat{C}08$	4				2			1				1
$\hat{C}09$						*3*						
$\hat{C}10$							*9*					
$\hat{C}11$	4	3			2			*10*		3		
$\hat{C}12$	1											
$\hat{C}13$	7	1						3	*3*			
$\hat{C}14$							1					
$\hat{C}15$	1		3	1	1					*5*		2
$\hat{C}16$	4	3					4					
$\hat{C}17$	5						3					
$\hat{C}18$						1					*1*	
$\hat{C}19$	2			1	1		1		2			*3*
$\hat{C}20$	4	1							1	1		

Tabla 6.2: Incidencia de los objetos (días) en la clase de referencia y la de predicción para el atributo DQO-AT

En esta parte de la metodología consideraremos el conjunto de entrenamiento que han sido previamente clasificados tanto por *Linneo+* como por $Klass^{+}$ para obtener el desempeño del sistema de reglas obtenido para el atributo DQO-AT. El proceso se ha descrito en el apartado §8 de la Sección §5.2.

La Tabla 6.2 compara la clase real de cada elemento con la asignada por las reglas, resumiendo el número de objetos que coinciden en ambas particiones. Los elementos asignados correctamente se ubican en la diagonal de esta tabla enmarcados en cursiva y el resto, representa errores de clasificación.

En este caso, para el atributo DQO-AT, se produce un error del 63.77 %. El proceso de validación se aplica al resto de los atributos.

Se observa que los errores de predicción son considerables, incluso a veces al que supondría una asignación aleatoria. Aducimos esta situación básicamente a dos razones:

- Estamos tratando únicamente un atributo con independencia de las demás.
- La desambigüización es muy fuerte (grado máximo) y no toma en cuenta la probabilidad de las casillas "error".

9. **Interpretación de Clases Resultantes**

Hemos mencionado que un método de apoyo a la generación de interpretaciones conceptuales es el uso de etiquetas lingüísticas que nos permita dar el significado de las clases en forma natural. Según el gráfico generado para el atributo DQO-AT, si por

ejemplo tomamos el valor de $x_{ik}^{DQO-AT} = 92{,}6\ mg/l$ y trazamos una línea vertical sobre él (ver Figura 6.4), obtendremos el grado de pertenencia de este valor a cada clase. Para el ejemplo obtenemos que el grado de pertenencia a la clase $\hat{C}10$ es 0.40 %, el grado de pertenencia a la clase $\hat{C}14$ es 0.10 %, el grado de pertenencia a la clase $\hat{C}16$ es 0.20 %, el grado de pertenencia a la $\hat{C}17$ es de 0.30 %, ya determinadas con el sistema de reglas del inciso 7), para el resto de las clases es cero.

Además, observando el gráfico tenemos que los valores altos para la materia química orgánica (DQO-AT) a la salida de la planta se da en las clases $\hat{C}09$, y $\hat{C}18$, valores intermedios en las clases $\hat{C}02$, $\hat{C}05$, $\hat{C}10$, $\hat{C}13$, $\hat{C}16$ $\hat{C}17$ y $\hat{C}19$, valores bajos en el resto de las clases.

Así, de esta forma podemos generar descripciones conceptuales de las clases:

> Si el atributo materia química orgánica de salida toma valores altos entonces ese día se asocia a $\hat{C}09$

donde, el grado de pertenencia de la materia química orgánica concreto al concepto valores "*altos*"vendría dado por la función de pertenencia de la clase, digamos $\hat{C}09$ en la Figura 6.4.

Como podremos darnos cuenta la interpretación por atributo es un conocimiento parcial que poco no ayuda, siendo más importante considerar la contribución de todas los demás atributos para la caracterización e interpretación de clases para nuevos objetos.

De esta forma terminamos la aplicación de la metodología por atributo para considerar el análisis multivariante.

6.5 Análisis multivariante

En esta fase de la aplicación al caso de estudio consideraremos la contribución de información que cada una de los atributos en consideración tiene en la asignación de la clase para un objeto nuevo del conjunto de prueba.

Consideremos el análisis de todos los atributos en forma conjunta. Por ejemplo, tomemos el objeto $i = 23$ (del conjunto de prueba) que de acuerdo a la partición de $Klass^+$ se le asignó la clase $\hat{C}01$; por el análisis por atributo y tomando como criterio de agregación el de probabilidad máxima se tiene, para el atributo DQO-AT, su valor es $x_{iDQO-AT} = 55$, localizado en el intervalo I_{17}^{DQO-AT}, la clase de predicción en el consecuente es $\hat{C}01$ con una probabilidad de 0.18. Con respecto al atributo SST-AT, su valor es $x_{iSST-AT} = 5{,}6$, el cual se localiza en el intervalo I_{11}^{SST-AT}, y de acuerdo al sistema de reglas se le asigna la clase $\hat{C}11$ con una probabilidad de 0.15. Con respecto al atributo Q-AB, su valor es $x_{iQ-AB} = 9732$, en el intervalo I_{17}^{Q-AB}, y la parte derecha de la correspondiente regla al criterio de agregación ya establecido es la clase $\hat{C}01$ con una probabilidad de 0.25. Para el atributo Q-SP1, su valor es $x_{iQ-SP1} = 9732$, en el intervalo I_{27}^{Q-SP1}, la clase asignada es $\hat{C}02$ con una probabilidad de 0.60, para el resto de los atributos se procede de forma similar.

En el ejemplo se observa que diferentes reglas (las de probabilidad máxima) se disparan con consecuentes diferentes, nuevamente presentando el problema de ambigüedad de asignación de clases a nivel de atributos. Como se ha mencionado anteriormente, en este proceso, sólo una regla por atributo se satisface (la de grado máximo de pertenencia) usando el criterio para desambiguar la confusión. La Figura 6.5 muestra la representación gráfica de la asignación clase|atributo para el objeto $i = 23$ del conjunto de prueba P_0; estas representaciones gráficas de las reglas disparadas en todos los atributos por objeto, permiten seleccionar

criterios de agregación más adecuados.

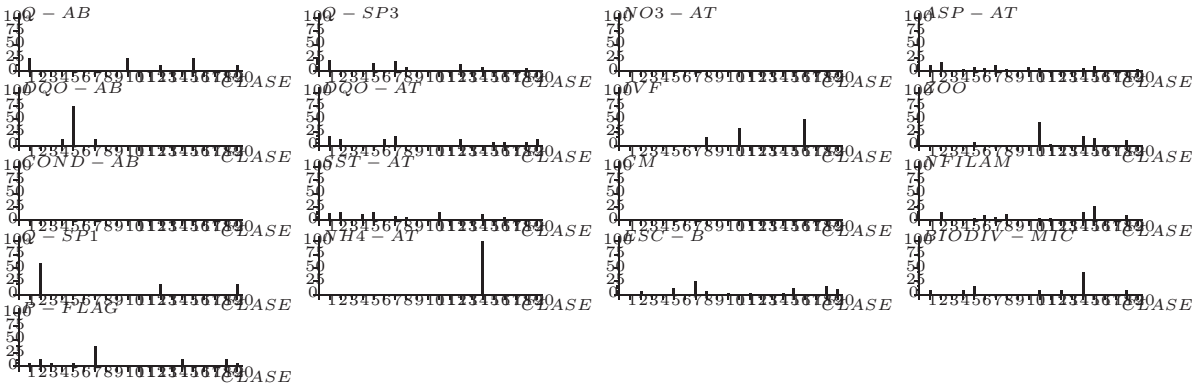

Figura 6.5: Gráfico de asignación de clases|atributo para el objeto $i = 23$ del conjunto de prueba P_0

Este proceso puede realizarse sobre cada una de los atributos. Sin embargo, genera un diagnóstico aparentemente inconsistente ya que esta primera aproximación no considera los grados de certeza de los demás atributos. Consideremos el ejemplo anteriormente expuesto, la Tabla 6.3 resume lo que ocurre sobre el objeto (día) $i = 23$ del conjunto de prueba P_0 y cuatro atributos de acuerdo a $\Re'(X_k, \mathcal{P})$. La clase de referencia para $i = 23$ es $\hat{C}01$, la cual es reconocida por tres de los cuatro atributos considerados.

		$\Re'(X_k, \mathcal{P})$		$\Re(X_k, \mathcal{P})$							
Atributo	**Valor**	$\hat{C}$	P	$\hat{C}$	P	$\hat{C}$	P	$\hat{C}$	P	$\hat{C}$	P
DQO-AT	55	$\hat{C}01$	0,18	$\hat{C}02$	0,12	$\hat{C}06$	0,12	$\hat{C}07$	0,18	$\hat{C}13$	0,12 ...
SST-AT	5.6	$\hat{C}11$	0,15	$\hat{C}01$	0,13	$\hat{C}02$	0,15	$\hat{C}03$	0,02	$\hat{C}04$	0,11 ...
Q-AB	9732	$\hat{C}01$	0,25	$\hat{C}10$	0,25	$\hat{C}13$	0,13	$\hat{C}16$	0,25	$\hat{C}20$	0,12
Q-SP1	9732	$\hat{C}02$	0,60	$\hat{C}13$	0,20	$\hat{C}20$	0,20				

$\hat{C}$ clase de la $\mathcal{P}_K$ P Probabilidad

Tabla 6.3: Asignación de clases para diferentes atributos del objeto $i = 23$ para P_0

Sin embargo, si trabajamos directamente con el sistema global de reglas $\Re(X_k, \mathcal{P})$, en contra de lo esperado, mejora la asignación de la clase. Note como se obtiene $\hat{C}01$ en algunas de las reglas de los atributos considerados y estas reglas presentan grados de pertenencia razonables y muy cercanos a la regla que resulta en $\mathcal{R}'(X_k, P)$ (en el caso de DQO-AT).

Así, el proceso de validación del sistema total de reglas consiste en: a partir de un conjunto de prueba P_0 previamente clasificado, medir la calidad de asignación de clases (porciento de objetos clasificados correctamente sobre el total de ellos) a los objetos nuevos, considerando un análisis en el cual un criterio de agregación de información de los atributos se tome en forma conjunta. Esta forma es para predecir la clase de cada uno de los objetos nuevos del conjunto de prueba y estimar la calidad de esta predicción. Del análisis de la Tabla 6.4, observamos que no todos los atributos conducen a la misma clase de predicción. Por ejemplo, el objeto 2 tiene asignada la clase $\hat{C}07$ con una probabilidad de 0.30 para el atributo Q-AB y, también tiene asignada la clase $\hat{C}02$ con una probabilidad de 0.20 para el atributo DQO-AT, y la clase $\hat{C}07$ con una probabilidad de 0.161,.... Nuevamente, tomaremos el criterio de probabilidad máxima para resolver el conflicto de asignación de clase para los objetos de P_0.

P_0	Q-AB		DQO-AB		COND-AB		...		...		BIODIV		P-FLAG		Classes	
i	$\hat{C}$	P	$\hat{C}$	P	$\hat{C}$	P	$\hat{C}$	P	$\hat{C}$	P	$\hat{C}$	P	$\hat{C}$	P	$\hat{C}$K	$\hat{P}$PM
1	$\hat{C}$01	.286	$\hat{C}$5	.20	$\hat{C}$01	.20	..	..	..	..	*	*	$\hat{C}$01	.129	$\hat{C}$01	$\hat{P}$15
2	$\hat{C}$07	.30	$\hat{C}$02	.20	$\hat{C}$07	.167	..	..	..	..	$\hat{C}$20	.154	$\hat{C}$01	.129	$\hat{C}$07	$\hat{P}$07
3	$\hat{C}$16	.325	$\hat{C}$01	.25	$\hat{C}$08	.20	..	..	..	..	$\hat{C}$16	.265	$\hat{C}$16	.27	$\hat{C}$07	$\hat{P}$11
..	..	..	..	..	..	..	..	..	..	..	..	..	..	..	..	..
..	..	..	..	..	..	..	..	..	..	..	..	..	..	..	..	..
24	$\hat{C}$01	.286	$\hat{C}$05	.75	$\hat{C}$01	.667	..	..	..	..	$\hat{C}$15	.417	$\hat{C}$01	.129	$\hat{C}$01	$\hat{P}$05
25	$\hat{C}$06	0.50	$\hat{C}$04	.50	$\hat{C}$02	.50	..	..	..	..	$\hat{C}$02	.182	$\hat{C}$01	.129	$\hat{C}$02	$\hat{P}$15

$\hat{C}$K : clase de Klass $\hat{P}$MP : Clase de predicción de probabilidad máxima

Tabla 6.4: Asignación de clase y probabilidad para cada atributo e individuo del conjunto de prueba P_0

Comparando las clase real y la asignada por las reglas para cada uno de los elementos del conjunto de prueba P_0, se observa que un 55 % de días fueron bien clasificados.

De esta forma podemos hacer una estimación de la calidad de la predicción. Así, tenemos que siete objetos de P_0: 1, 3, 8, 9, 10, 11, 14, 15, 17, 24 y 25 fueron mal clasificados con respecto a la partición de $Klass^+$, teniendo un error del 45 % y una estimación de la calidad de la predicción del 55 %.

Para la partición de referencia de $Linneo^+$ se hizo un análisis similar, obteniendo un error global del 40 % y una estimación de la calidad de la predicción del 60 %.

6.6 Criterios de Agregación

Consideramos que uno de los factores que inciden directamente en la asignación de clases es el criterio de agregación que se toma al hacer el análisis multivariable. Por lo tanto, analizaremos dos criterios que consideramos nos darán mejores resultados, dado que estos criterios en principio "toman en cuenta"la contribución de todos los atributos, estos son: criterio de *Votación* y criterio de *Suma máxima*.

Al igual que el criterio de agregación de probabilidad máxima estos últimos criterios tienen como entrada el conjunto de entrenamiento T_0, la partición de referencia, en nuestro caso seguimos trabajando con $Klass^+$, el conjunto de prueba P_0 y su partición correspondiente.

Los criterios de agregación de votación y suma máxima consisten en:

Votación. Para cada individuo i del conjunto de prueba, leemos el valor x_{ik} del atributo X_k, lo ubicamos en el intervalo correspondiente, digamos I_s^k, de la tabla de distribuciones, inicializamos un contador por atributo para llevar el récord de cántos atributos con probabilidades distintas de cero se le asignan a $C01$, cuántas a $C02$, ... y así sucesivamente hasta determinar el número de atributos que se le asignan a $C20$. Nos fijamos en el número máximo de votos y al individuo i le asignamos la clase correspondiente que tiene ese número.

Suma máxima. Para cada individuo i del conjunto de prueba, leemos el valor x_{ik} del atributo X_k, lo ubicamos en el intervalo correspondiente, digamos I_s^k, de la tabla de distribuciones, inicializamos un sumador por atributo para llevar la suma de las probabilidades de los atributos que se les asigna la clase $C01$, la suma de probabilidades de los atributos que se les asigna la clase $C02$, ... y así sucesivamente hasta obtener la suma de

probabilidades de los atributos que se les asigna la clase $C20$. Nos fijamos en la suma máxima y al objeto i le asignamos la clase correspondiente a esa suma máxima.

6.7 Resultados

Aplicando los criterios de agregación antes descritos a nuestros datos para obtener la clase de predicción y, utilizando la herramienta CIADEC se tienen los siguientes resultados:

i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
K	1	7	7	10	1	1	1	7	2	2	2	11	2	2	1	2	11	1	1	1	1	1	1	1	2
PM	15	7	11	10	1	1	1	5	1	13	1	11	2	19	2	2	2	1	1	1	1	1	1	5	15
Vot	4	7	5	10	1	1	1	2	2	2	2	11	2	2	10	2	1	2	1	1	2	1	1	2	7
Sum	4	7	2	10	1	1	2	2	2	2	2	11	1	2	10	2	11	2	10	1	13	1	1	2	2

K : Clase de Klass, PM : Clase por Probabilidad Máxima, Vot : Clase por Votación y Sum : Clase por Suma Máxima

Tabla 6.5: Asignación de clases de predicción considerando los criterios de probabilidad máxima, votación y suma máxima

Observando la Tabla 6.5 podemos determinar los errores de predicción para cada uno de los criterios aplicados a los datos de la planta depuradora y con la partición de referencia $Klass^+$, ellos son: 45 % para el criterio de probabilidad máxima, 36 % para el de votación y 40 % para el de suma máxima. Para la partición de referencia de $Linneo^+$ con los mismos datos, obtuvimos los siguientes errores de predicción: 48 % para el criterio de probabilidad máxima, 38 % para el de votación y 40 % para el de suma máxima.

6.8 Comparación de métodos

En esta parte se presenta el estudio comparativo de diferentes métodos (estadísticos y de aprendizaje automático) para el reconocimiento de patrones de conocimiento del conjunto de datos provenientes de una planta depuradora de aguas residuales, que se discute en [CDG$^+$01]. En el artículo se analiza el desempeño cuantitativo, en términos de la exactitud de la predicción sobre ejemplos no vistos, número de atributos, ejemplos usados y el desempeño cualitativo en términos de la interpretación del significado para los expertos del dominio. Los métodos usados fueron: inducción de árboles de inducción (C4.5), dos técnicas diferentes de inducción de reglas (CN2 y BPRI) y dos métodos de aprendizaje basados en memoria (IBL Y CBL). La mayoría de los patrones de conocimiento extraídos por los métodos son explicítos, pero en algunos otros, como en las técnicas de aprendizaje basados en memoria, el conocimiento extraído es implícito.

La evaluación comparativa fue publicada en la revista AI Communications en marzo del año 2001 con los resultados que se muestran en la Tabla 6.6.

Con la formalización, mejoras al método (análisis multivariante y criterios de agregación) y automatización de la metodología BPRI (Box-Plot Rule Induction) los resultados han sido alentadores y prometedores. La Tabla 6.7 muestra los resultados recientemente y apostamos en que podemos seguir mejorando la metodología.

Tabla 6.6: Comparación de métodos inductivos para predicción. Agosto, 2001

Método	Número de atributos	Número de ejemplos	Exactitud de predicción sobre P_0 (%)	Interpretación del significado	Exactitud de predicción conjunto total (%)
C4.5	24	243	63.51	Parcial	89.7
CN2	44	243	63.98	Parcial	98.8
BPRI	63	243	58.9	Parcial	–
k-NN	63	243	76.38	No	100
J48	–	243	64.4	Parcial	–
J48, 10 i bagging	–	243	70.7	No	–
J48, 10 i AdaBoost	–	243	73.6	No	–
C4.5	11	243	65.11	General	87.2
CN2	19	243	65.45	General	95.9
k-NN	19	243	71.22	No	100
Opencase1	19	243	68.73	No	100
Opencase2	19	220	62.50	Si	97.1
Opencase3	19	243	64.20	Si	98.8
Opencase4	63	243	70.40	No	100

donde: opencase1 $\stackrel{\text{def}}{=}$ opencase (plain memory 19 att.), opencase2 $\stackrel{\text{def}}{=}$ opencase (hierarchical, relevant cases), opencase3 $\stackrel{\text{def}}{=}$ opencase (hierarchical, all cases 63 att.)

Tabla 6.7: Comparación de métodos inductivos para predicción. Marzo, 2002

Método	Número de atributos	Número de ejemplos	Exactitud de predicción sobre P_0 (%)	Interpretación del significado	Exactitud de predicción conjunto total (%)
C4.5	24	243	63.51	Parcial	89.7
CN2	44	243	63.98	Parcial	98.8
BPRI	63	243	64.5	Parcial	97.24
k-NN	63	243	76.38	No	100
J48	–	243	64.4	Parcial	–
J48, 10 i bagging	–	243	70.7	No	–
J48, 10 i AdaBoost	–	243	73.6	No	–
C4.5	11	243	65.11	General	87.2
CN2	19	243	65.45	General	95.9
k-NN	19	243	71.22	No	100
Opencase1	19	243	68.73	No	100
Opencase2	19	220	62.50	Si	97.1
Opencase3	19	243	64.20	Si	98.8
Opencase4	63	243	70.40	No	100

donde: opencase1 $\stackrel{\text{def}}{=}$ opencase (plain memory 19 att.), opencase2 $\stackrel{\text{def}}{=}$ opencase (hierarchical, relevant cases), opencase3 $\stackrel{\text{def}}{=}$ opencase (hierarchical, all cases 63 att.)

Capítulo 7

Conclusiones

Las conclusiones a las que hemos llegado en este trabajo son las siguientes.

- Se ha comprobado que la predicción de clases depende de la partición de referencia.
- Por razones económicas, la metodología se aplicó a un solo atributo (DQO-AT) la que tuvo una exactitud de predicción del 63.77 % en un conjunto de prueba de 25 individuos.
- En cuanto a la generación del sistema de reglas global persisten las inconsistencias al precipitar la desambiguación sobre la fase de análisis por atributos aislados, los cuales inciden en los altos errores relativos de asignación de clase por atributo.
- Es más probable que se consiga una reducción mayor del error de asignación retardando la reducción del sistema de reglas hasta el último paso, para realizar un análisis global de los nuevos días y mejorar la calidad del diagnóstico.
- El coste computacional de la metodología es bajo en relación a la información que proporciona, puesto que se resuelve un análisis de intersecciones de grado ξ (en nuestro caso 20), calculando solamente ξ máximos y ξ mínimos y ordenándolos.
- Este trabajo constituye un punto importante en la construcción de un sistema de diagnóstico en plantas depuradoras de aguas residuales. Se ha presentado un método que genera un sistema de reglas difusas a partir de atributos cuantitativos medidos en diferentes puntos de la planta y con el objetivo de identificar situaciones características.
- La aplicación de esta primera aproximación de la propuesta metodológica a la planta depuradora nos permite obtener conocimiento relacionado a las diferentes situaciones que se presentan en el proceso de caracterización de clases. En este proceso hemos determinado la existencia de cuatro tipos de valores: propios o no, parcialmente y totalmente caracterizadores, que juegan importantes papeles en el sistema de reglas.
- Los beneficios a largo plazo de la metodología son:
  - Como ayuda en la interpretación de situaciones características en dominios poco estructurados para la predicción de nuevos objetos.
  - Obtención de conocimiento de dominios poco estructurados caracterizando las diferentes situaciones que pueden ocurrir en un proceso dado.
  - Proporcionar soporte a la toma de decisiones en procesos que utilicen atributos cuantitativos.

Capítulo 8

Trabajo Futuro y Agenda de la Tesis

8.1 Trabajo Futuro

En este proyecto de tesis, se plantea un desarrollo más amplio sobre algunos temas que hay necesidad de estudiar con más profundidad, para consolidar esta metodología, el cual se ha identificado como trabajo futuro.

1. Estudiar formalmente las implicaciones de tomar el sistema de intervalos en tal forma que el primer intervalo $\mathcal{I}_1^k$ sea cerrado por la izquierda y abierto por la derecha e $\mathcal{I}_2^k$ sea cerrado por ambos lados, esto por el hecho de que la regla en $\mathcal{I}_1^k$ tenga sentido o de considerar los min_{ci} y max_{ci} como puntos aislados.

2. Estudiar la conveniencia de optimizar los $\mathcal{I}^k$ para que no contengan modalidades vacías, fruto de combinar extremos de clases distintas de igual valor.

3. El método repetido como hasta ahora para todos los atributos produce poca cobertura (los ϵ son pequeños), se trata de introducir las modificaciones necesarias para cubrir la mayor parte de casos con reglas.

4. Establecer si es posible, una relación entre p_{sc} (frecuencia relativa de individuos en $\mathcal{I}_1^k$ y pertenecen a la clase $C \in \mathcal{P}$) y ϵ grado de caracterización a una clase C).

5. Suavizar las funciones de pertenencia de los gráficos por atributo.

6. Implementar un modelo difuso que nos permita la creación de etiquetas lingüísticas para generar automáticamente descripciones conceptuales.

7. Continuar la búsqueda de un criterio de agregación que nos permita aumentar la eficiencia del sistema de reglas.

8. Desarrollar una interfície de navegación para el sistema *CIADEC* o integrarlo como un nuevo módulo del software $KLASS^+$ (orientado a la clasificación de dominios poco estructurados) como una herramienta de ayuda a la interpretación de resultados.

8.2 Agenda de la Tesis

En esta sección se propone un plan de actividades para poder realizar la propuesta planteada en este proyecto y desarrollar la tesis doctoral y su cronograma correspondiente (Tabla 8.1).

	2002			2003		
Etapa	**Jul-Ago**	**Sep-Oct**	**Nov-Dic**	**Ene-Feb**	**Mar-Abr**	**May-Jun**
1	•	•				
2	•	•	•			
3		•	•	•		
4		•	•	•	•	
5				•	•	
6					•	•
7				•	•	
	2003			**2004**		
Etapa	**Jul-Ago**	**Sep-Oct**	**Nov-Dic**	**Ene-Feb**	**Mar-Abr**	**May-Jun**
8	•	•				
9		•	•			
10		•	•			
11	•	•	•			
12				•	•	
13					•	
14						•

Tabla 8.1: Cronograma de las etapas para el desarrollo de la tesis.

- **Etapa 1**. Continuar con la revisión bibliográfica del tema.
- **Etapa 2**. Revisión e incorporación de mejoras a la metodología (trabajo futuro).
- **Etapa 3**. Estudio y selección de un criterio de agregación que permita mayor eficiencia al sistema *CIADEC*.
- **Etapa 4**. Diseño e implementación de un modelo de etiquetas lingüísticas para la generación automática de descripciones conceptuales.
- **Etapa 5**. Revisión e incorporación de mejoras al módulo de gráficos del sistema *CIADEC*.
- **Etapa 6**. Pruebas, validaciones y ajustes de la implementación de la metodología.
- **Etapa 7**. Desarrollo y presentación de un artículo para un congreso sobre Inteligencia Artificial, Estadística o Lógica Difusa.
- **Etapa 8**. Pruebas en algún otro dominio similar.
- **Etapa 9**. Desarrollo y presentación de un artículo para la revista Computación y Sistemas.
- **Etapa 10**. Comparación con otros métodos.
- **Etapa 11**. Redacción de la tesis.
- **Etapa 12**. Revisión por parte de los referis.
- **Etapa 12**. Corrección de la tesis.
- **Etapa 14**. Lectura de la tesis.

Capítulo 9

Publicaciones

A continuación se enlistan las publicaciones que dan soporte al presente trabajo.

1. Autores: Vázquez, F., Gibert, K.
   Título: Generación Automática de Reglas Difusas en Dominios Poco Estructurados con Variables numéricas.
   Institución: Universidad Politécnica de Cataluña.
   Departamento: Lenguajes y Sistemas Informáticos.
   Fecha: noviembre 2001.
   Tipo: Reporte de Investigación.
   Referencia: LSI-01-51-R.
   Dirección: Barcelona, España.

2. Autores: Vázquez, F., Gibert, K.
   Título: Generación Automática de Reglas Difusas en Dominios Poco Estructurados con Variables Numéricas.
   Publicación: Actas de la Conferencia de la Asociación Española para la Inteligencia Artificial (CAEPIA).
   Fecha: noviembre, 2001.
   Lugar: Gijón, España. Pages: 143–152

3. Autores: Vázquez, F., Gibert, K.
   Título: Implementation of the methodology "Automatic Characterization and Interpretation of Conceptual Descriptions in ill-Structured Domains using Numerical Variables.
   Institución: Universidad Politécnica de Cataluña.
   Departamento: Lenguajes y Sistemas Informáticos.
   Fecha: marzo 2002.
   Tipo: Reporte de Investigación.
   Referencia: LSI-02-28-R.
   Dirección: Barcelona, España.

4. Autores: Vázquez, F., Gibert, K.
   Título: Fundamentos de la Teoría de los Conjuntos Borrosos y Lógica Borrosa.
   Institución: Universidad Politécnica de Cataluña.
   Departamento: Lenguajes y Sistemas Informáticos.
   Fecha: mayo 2002.
   Tipo: Reporte de Investigación.
   Referencia: LSI-02-3-T.
   Dirección: Barcelona, España.

Bibliografía

[AB84] M.S. Alderfer and R.K. Blashfield. Cluster Analysis. *Sage Publication*, 1984. California, USA.

[AG91] J. Aguilar and K. a Gibert. Sobre variables lingüísticas, difusas, paradigmas parmenidianos y lǵicas multivaluadas. *ESTYLF*, 1:185–192., 1991.

[AGR93] J. Aguilar, K. Gibert, and Rodríguez. Fuzzy semantic in expert process control. *LNAI*, 1993.

[Alu96] T. Aluja. *Análisis Factoriales Descriptivos con SPAD-N.* 1996.

[AT97] S. Abe and R. Thawonmas. A fuzzy classifier with ellipsoidal regions.. *IEEE Trans. on Fuzzy Systems,*, pages 358–368, 1997.

[ATV83] C. Alsina, E. Trillas, and L. Valverde. On some logical connectives for fuzzy set theory. *Math. Anal. Appl.*, 93.:149–163, 1983.

[BA96] R. Brachman and T. Anand. The Process of Knowledge Discovery in Databases: A Human-Centered Approach. *In Advances in Knowledge Discovery and Data Mining*, pages 65–78, 1996. Ed. U.Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, AAAI/MIT Press.

[Bay00] S. Bayona. Descriptiva de dades y de classes. PFC Facultat d'Informàtica, UPC, jul 2000.

[Bis95] C.M. Bishop. *Neural Networks for Pattern Recognition.* Clarendon Press, Oxford, first edition, 1995.

[BK89] R. Bajcsy and S. Kovacic. Multiresolution elastic matching. *Computation Vision Graphics Image Process.*, 46:1–21, 1989.

[Cas96] X. Castillejo. Un entorn de treball per a Klass. PFC Facultat d'Informàtica, UPC, jul 1996.

[CDG+01] J. Comas, S. Dzeroski, K. Gibert, I. Roda, and M. Sànchez-Marrè. Knowledge discovery by means of inductive methods in wastewater treatment plant data. *AI Communications. The european journal on artificial intelligence*, 14(1):45–62, march 2001.

[CDJHa] O. Cordón, M.J. Del Jesús, and F. Herrera. A proposal on reasoning methods in fuzzy rule-based classification system.

[CDJHb] O. Cordón, M.J. Del Jesús, and F. Herrera. Completeness and consistency conditions for learning fuzzy rules.

[CHP] Cordón, F. Herrera, and A. Peregrín. Applicability of the fuzzy operators in the design of fuzzy logic controllers.

[Cor71] R. Cormack. A review of clasification. In *Journal of the Royal Statistical Society (Series A)*, pages 134: 321–367, 1971.

[CYT96] Z. Chi, H. Yan, and Pham T. Fuzzy algorithms with applications to image processing and pattern recognition. *World Scientific,*, pages 101–105, 1996.

[DG92] E. Diday and K.C. Gowda. Symbolic clustering using a new similaritiy measure. In *IEEE Trans. on systems, man.,and cib.*, volume 22, pages 368–378, 1992.

[DH73] R.O. Duda and P.E. Hart. *Pattern Classification and Scene Analysis.* Wiley and Sons., New York, 1973.

[DK82] P.A. Devijver and J. Kittler. *Pattern Recognition: A Statistical Approach.* Prentice Hall., London, first edition, 1982.

[DL96] L. Devroye, L. Gyorfi and G. Lugosi. *A Probabilistic Theory of Pattern Recognition.* Springer Verlang, Berlin, first edition, 1996.

[DM84] E. Diday and J.V. Moreau. Learning hierarchical clustering from examples. In N 289 Centre de Rocquencourt, Rapports de Recherche, editor, *INRIA*, 1984.

[DP85] D. Dubois and H. Prade. A review of fuzzy set aggregation connectives. *Information Sciences*, 36:85–121, 1985.

[DPB99] D. Dubois, H. Prade, and J. Bezdek. *Fuzzy sets in approximate reasoning and information system*, volume 1. Kluwer Academic Publishers, 1999.

[Fay96] U. Fayyad. *From Data Mining to Knowledge Discovery: An overview.* 1996. ISBN 0-262-56097-6.

[FPSS96] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. From Data Mining to Knowledge Discovery in Databases (a survey). *AI Magazine.*, 3(17):37–54., 1996.

[FPSSU96] U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthursamy. Advances in Knowledge Discovery and Data Mining. *AAAI Press.*, 1996.

[Fu82] K.S. Fu. *Syntactic Pattern Recognition and Applications.* Prentice-Hall, Englewwod Cliffs., 1982.

[Fu83] K.S. Fu. A step toward unification of syntactic and statistical pattern recognition. *IEEE Trans. Pattern Analysis and Machine Intelligence.*, 5(2):200–205, 1983.

[Fuk90] K. Fukunaga. *Introduction to Statistical Pattern Recognition.* Academic Press., 1990.

[GA98] K. Gibert and T. Aluja. A computational technique for comparing classifications and its relationship with knowledge discovery. In *International Seminar on New Techniques and Technologies for Statistics*, pages 193–198, Italy, nov 1998.

[GA00] K. Gibert and Salvador A. Aproximación difusa a la identificación de situaciones Características en el tratamiento de aguas residuales. *Congreso español sobre tecnologías y lógica fuzzy.*, 1, 2000. Sevilla, Esp.

[GAC98] K. Gibert, T. Aluja, and U. Cortés. Knowledge Discovery with Clustering Based on Rules. In Quafafou Eds., editor, *Principles of Data Mining and Knowledge Discovery*, volume 1510 of *Lecture Notes in Artificial Intelligence*, pages 83–92, Nantes, 1998. Springer-Verlag. Interpreting Results.

[GC92] K. Gibert and U. Cortés. KLASS: Una herramienta estadística para la creación de prototipos en dominios poco estructurados. *proc. IBERAMIA-92.*, pages 483–497, 1992. Noriega Eds. México.

[GC93a] K. Gibert and U Cortés. Combining a knowledge based system with a clustering method for an inductive construction of models. In *Proc. 4th Int Work. on AI and Stats.*, 1993. Florida, USA.

[GC93b] K. Gibert and U Cortés. On the uses of the expert knowledge for automatic biasing of a clustering method. In *ITI 93. Proceedings of the International Conference on Information Technology Interfaces*, pages 219–224, Croatia, 1993. issn 1330-1012.

[GC94] K. Gibert and U. Cortés. Combining a knowledge-based system and a clustering method for a construction of models in ill-structured domains. In *Artificial Intelligence and Statistics IV*, volume 89 of *Lecture Notes in Statistics*, pages 351–360, New York, N.Y. US., 1994. Springer-Verlag.

[GC97] K. Gibert and U. Cortés. Weighing quantitative and qualitative variables in clustering methods. *Mathware and Soft Computing*, 4(3):251–266, 1997.

[GC98] K. Gibert and U. Cortés. Clustering based on rules and knowledge discovery in ill-structured domains. *Computación y Sistemas.*, 1(4):213–227, 1998. ISSN 1405 - 5546. Impreso en México.

[GHC96] K. Gibert, M. Hernández, and U. Cortés. Classification based on rules: an application to Astronomy. In Ed. U. Tokio. Kobe. Japón, editor, *Proceedings of 5. Conference of International Federation of Classification Societies*, pages 69–72, Mar 1996.

[Gib91] K. Gibert. Klass. Estudi d'un sistema d'ajuda al tractament estadístic de grans bases de dades. Master's thesis, UPC, 1991.

[Gib94] K. Gibert. *L'us de la Informació Simbòlica en l'Automatització del Tractament Estadístic de Dominis Poc Estructurats.* In the statistics and operations research phd. thesis., Universitat Politecnica de Catalunya, Barcelona, Spain, 1994.

[Gib96] K. Gibert. On the uses and costs of rules-based classification. In A. Prat. Physica-Verlag, editor, *Proceedings of Computational Statistics*, pages 265–270, march 1996.

[Gor80] A.D. Gordon. *Clasification.* Chapman & Hall, London, 1980.

[Gow71] J.C. Gower. A General coefficient if similarity and some of its properties. *Biometrics*, 27:857–874, 1971.

[Gre93] U. Grenander. *General Pattern Theory.* Oxford University Press., 1993. First Edition.

[GS97] K. Gibert and Z. Sonicki. Classification Based on Rules and Medical Research. In Rocco Curto, editor, *VIII International Symposium on Applied Stochastic Models and Data Analysis*, pages 181–186, Italy, 1997. ASMDA 97.

[GS99] K. Gibert and Z. Sonicki. Classification Based on Rules and Thyroids Dysfunctions. *Applied Stochastic Models in Business and Industry*, 15(4):319–324, octuber 1999.

[GS00] K. Gibert and A. Salvador. Aproximación difusa a la identificación de situaciones características en el tratamiento de aguas residuales. In *X Congreso Español sobre tecnologías y lógica fuzzy*, pages 497–502, España, sep 2000. ESTYLF 2000.

[GSM00] Roda-I. Cortés U. Gibert, K. and Sànchez-Marrè. Identifying characteristic situations in wastewater treatment plants. *Workshop in Binding Environmental Sciences and Artificial Intelligence*, 1:1–9, 2000. ECAI.

[Har75] J.A. Hartigan. *Clustering Algorithms.* John Wiley & Sons, London (England), 1975.

[HC68] G.E. Hughes and M.J. Cresswell. *An Introduction to MOdal Logic.* London, eds., 1968.

[INT] H. Ishibuchi, K. Nozaki, and H. Tanaka. Distributed representation of fuzzy rules and its applications to pattern classification.

[IY94] M. Ichino and H. Yaguchi. Generalized Minkowski Metrics for Mixed feature-type data analysis. *IEEE Transaction on systems, man and cybernetics*, 22(2):146–153, 1994. April.

[JDC87] A.K. Jain, R.C. Dubes, and C.C. Chen. Bootstrap Techniques for error estimation. *IEEE Trans. Pttern Analysis and Machine Intelligence.*, 9:628–633, 1987.

[Kan94] M. Kantrowitz. Milestones in the Development of Artificial Intelligence 1994. web, 1994.

[Koh95] T. Kohonen. Self-Organizing Maps. *Springer Series in Information Sciences.*, 30, 1995.

[Kun] L.I. Kuncheva. On the equivalence between fuzzy and statistical classifiers.

[LdM90] R. López de Mántaras. *Approximate reasoning models.* Ellis Horwood series in AI, 1990.

[McD82] J. McDermott. R1: A rule-based configurer of computer systems. 1982.

[McL92] G. McLachlan. *Discriminant Analysis and Statistical Pattern Recognition.* Wiley and Sons, New York, first edition, 1992.

[MG81] E.H. Mamdani and G.R. Gaines. *Fuzzy reasoning and its Applications.* Mamdani-Gains eds., 1981.

[MM97] J. Márquez and J.C. Martín. La clasificación automática en las ciencias de la salud. PFC, octubre 1997. Facultat de Matemàtiques i Estadística, UPC.

[MMP] D.P. Mandal, C.A. Murthy, and S.K. Pal. Formulation of a multivalued recognition system.

[MS82] R.S. Michalski and R. Stepp. Automated construction of classifications: Conceptual clustering versus mumerical taxonomy. *IEEE Trans.*, 5:396–410, 1982.

[MZ82] M. Mizumoto and H.J. Zimmermann. *Comparison of fuzzy reasoning methods.* Great Britain, first edition, 1982.

[Nag68] G. Nagy. State of the art in pattern reconition. *Proc. IEEE.*, 56:836–862, 1968.

[Pav77] T. Pavlidis. *Structural Pattern Recognition.* Springer-Verlag., New York., 1977.

[Per98] L.I. Perlovsy. Conundrum of combinatorial complexity. *IEEE Trans. Pattern Analysis and Machine Intelligence.*, 20:666–670, 1998.

[RGG00] J. Rodas, J. Gramajo, and K. Gibert. AI versus Statistics : Some Common Topics. Research DR 2000-13, Technical University of Catalonia, Barcelona, Spain, May 2000. http://www.lsi.upc.es/dept/techreps/html/R01-6.html.

[RGR01] J. Rodas, K. Gibert, and J. Rojo. Electroshock Effects Identification Using Clasification Techniques. *Springer's Lecture Notes of Computer Science Series*, Crespo, Maojo and Martin (Eds.):238–244, 2001. Second International Symposium, ISMDA 2001.

[Rip96] B. Ripley. *Pattern Recognition and Neural Networks.* Cambridge University Press., Cambridge, first edition, 1996.

[Rod99] D. Rodríguez. Análisis de los datos de una planta depuradora de aguas utilizando la clasificación basada en reglas, 1999.

[RPSM] R. Roda, M. Poch, and U. Sànchez-Marrè, M. Cortés.

[Sch92] G. Schuhfried. *Wiener Testsystem. Vienna Reaction Unit, Basic Program*, 1992. Development and production of scientific equipment. Mödling, Austria.

[Sho76] E.H. Shortlife. *MYCIN: A rule-based computer program for advising physicians regarding antimicrobial therapy selection.* PhD thesis, Stanford University, USA, USA, 1976.

[SM95] M. Sànchez-Marrè. *An Integrated Supervisory Multi-level Architecture for WasteWater Treatment Plants.* PhD thesis, UPC, 1995.

[SMCLP97] M. Sànchez-Marrè, U. Cortés, J. Lafuente, and M. Poch. Concept formation in WWTP by means of classification techniques: A compared study. *Applied Intelligence.*, 7:147–166., 1997.

[Tub99] X. Tubau. Sobre el comportament de les mètriques mixtes en algorismes de Clustering. PFC, octubre 1999. Facultat d'Informàtica, UPC.

[Vap98] V.N. Vapnik. *Statistical Learning Theory.* Wiley and Sons, 1998.

[VG01a] F. Vázquez and K Gibert. Automatic generation of fuzzy rules in ill structures domains with numerical variables. Research LSI-01-51-R, Technical University of Catalonia, Barcelona, Spain, December 2001. http://www.lsi.upc.es/dept/techreps/html/R01-51.html.

[VG01b] F. Vázquez and K Gibert. Generación Automática de Reglas Difusas en Dominios Poco Estructurados con Variables Numéricas. In *Actas de la Conferencia de la Asociación Española para la Inteligencia Artificial*, volume 1, pages 143–152, España, nov 2001. CAEPIA 01.

[VG02a] F. Vázquez and K Gibert. Fundamentos de la Teoría de los Conjuntos Borrosos y la Lógica Borrosa. Research LSI-02-3-T, Technical University of Catalonia, Barcelona, Spain, March 2002. http://www.lsi.upc.es/dept/techreps/html/R02-28.html.

[VG02b] F. Vázquez and K Gibert. Implementation of the methodology "Automatic Characterization and Interpretation of Conceptual Descriptions in ill-Structured Domains. Research LSI-02-28-R, Technical University of Catalonia, Barcelona, Spain, January 2002. http://www.lsi.upc.es/dept/techreps/html/R02-28.html.

[Wat85] S. Watanabe. *Pattern Recognition: Human and Mechanical.* Wiley, 1985.

[Yag88] R.R. Yager. On ordered weighted averaging aggregation operators in multicriteria decision making. *IEEE Trans. on Systems, Man and Cibernetics.*, 18:183–190, 1988.

[Yag93] R.R. Yager. Familes of OWA operators. *Fuzzy Sets and Systems.*, 59:125–148, 1993.

[Zad65] L.A. Zadeh. Fuzzy Sets. *Information and Control,*, pages 338–353, 1965.

[Zad73] L.A. Zadeh. Outline of a new approach to the analysis of complex systems and decision processes. *IEEE Trans. Syst. Man Cybernet,*, pages 28–44, 1973.

[Zad93] L.A. Zadeh. The role of fuzzy logic and soft computing in the conception and design of intelligent systems. *8th Austrian Artificial Intelligence Conference, LNAI 695.*, 695:1–5, 1993.

Parte II

CIADEC

Capítulo 10

CIADEC

En esta segunda parte del proyecto de tesis, se hace una breve introducción al sistema CIADEC [1], se describe su estructura y sus funcionalidades.

10.1 Introducción

El sistema CIADEC implementa la metodología que se definió en [VG02b] cuyo título es "Automatic Generation of Fuzzy Rules in ill-Structured Domains-[VG01a], la que permite caracterizar las diferentes clases a partir de una clasificación previamente establecida, en dominios poco estructurados y obtener automáticamente interpretaciones conceptuales de éstas, con respecto a atributos cuantitativos.

10.2 Diseño modular del sistema CIADEC

El sistema CIADEC, surge de la necesidad de automatizar la caracterización e interpretación de clases en dominios poco estructurados previamente particionados combinando conceptos, técnicas de inteligencia artificial y estadística. Mediante la automatización se persigue reducir el tiempo necesario para llevar a cabo esta tarea, agilizando tanto las actividades asociadas al análisis de datos como a la obtención de información relevante que posteriormente sea útil en la gestión y/o toma de decisiones en esos dominios.

10.2.1 Arquitectura del sistema CIADEC

La entrada del sistema es la matriz de datos X y la partición de referencia P, teniendo como salidas, según la opción del usuario:

- La asignación de clases a un conjunto de objetos nuevos.
- La calidad de asignación del sistema de reglas.
- La representación gráfica de las funciones de pertenencia por atributo. Dichos gráficos son generados en código LaTeX y se pueden exportar a cualquier documento o bien ser visualizados en pantalla conectando con el visualizador de LaTeX. Este tratamiento se adecúa a la filosofía de otras herramientas que se comparten en el mismo equipo de trabajo y que, en un futuro, se han de integrar en una plataforma común.

[1] de las primeras letras del nombre Caracterización e Interpretación Automática de Descripciones Conceptuales.

A nivel conceptual el sistema CIADEC esta formado por los siguientes cinco módulos:

- Módulo I. Generador de Intervalos de Longitud Variable (GILOVA)
- Módulo II. Generador de Tablas de Distribuciones (Funciones de Pertenencia) Condicionadas a Intervalos (GETADI)
- Módulo III. Generador de Sistemas de Reglas (GESIRE)
- Módulo IV. Generador de Gráficos de funciones de pertenencia de $X_k|C$ (GEGRALA)
- Módulo V. Validación (VALIDA)

10.2.2 Estructuras de datos

En este apartado se explica la representación de datos y la estructura de ficheros que el sistema CIADEC necesita para que funcione.

Representación de datos

Los individuos que forman el conjunto T_0 están descritos por una serie de atributos o características y pueden ser de dos tipos:

- Atributos cualitativos o categóricos: Corresponden a un tipo de característica de los individuos que se expresan mediante adjetivos. Estos atributos cualitativos se dividen en ordinales [2] y nominales [3].
- Atributos cuantitativos: Son características medibles y se expresan en forma numérica.

Si se dispone de n individuos y de k atributos que los describen, los valores de todas estas variables para el conjunto de individuos se representan mediante una matriz rectangular X de dimensiones (n, k). Las filas de la matriz contendrán la información de los individuos, mientras que las columnas hacen referencia a los atributos. Si los individuos son caracterizados simultáneamente con atributos cuantitativos y cualitativos, la matriz de datos se considera heterogénea.

A las observaciones no presentes en la matriz de datos X se les denomina valores faltantes. En caso de valores faltantes les asignamos un "$*$" con valor NaN (Not a Number) para que sea tratable desde el punto de vista del algoritmo.

La Figura 10.1 muestra la arquitectura modular del sistema CIADEC.

10.2.3 Estructuras de ficheros

En este apartado se la estructura de ficheros que el sistema CIADEC necesita para que funcione.

Las estructuras de los ficheros que describen el flujo de datos en el sistema CIADEC son de dos tipos de entrada y salida y extensiones: dat, par, iks, dci, tex, srg, srr, tcp, vsg, vsm y coi.

[2] Son aquellas para las que existe una relación de orden

[3] Son aquellas que no presentan una ordenación de sus valores

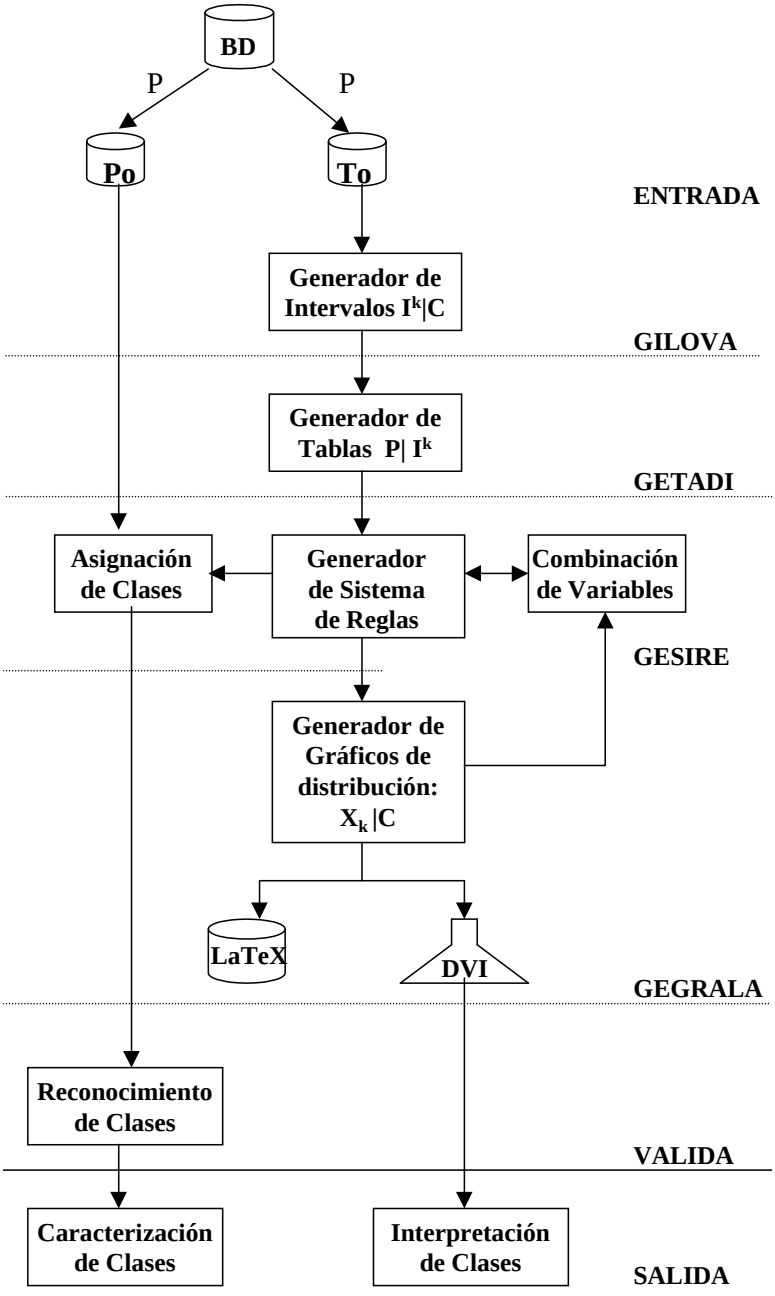

Figura 10.1:
Diagrama Conceptual del Proceso CIADEC

- $< nombre\_fichero.dat >$ Contiene la matriz de datos X por renglones. Para cada individuo u objeto i hay una lista con las coordenadas que le definen en cada atributo y su formato es el estándar de este tipo de fichero: Los elementos de una línea estarán separados por al menos un espacio. En la primera línea van los nombres de los atributos, en caso de que no estén los nombres se asignarán a los atributos los nombres por defecto NONAMEk, donde k es el número del atributo en consideración. El Cuadro 10.1 muestra el formato de un fichero con extensión dat.

x_1	x_2	...	...	x_n	XX
v_{11}	v_{12}	...	...	v_{1n}	id_1
v_{21}	v_{22}	...	...	v_{2n}	id_2
...	...	...	...	...	...
...	...	...	...	...	...
v_{m1}	v_{m2}	...	...	v_{mn}	id_m

Tabla 10.1: Estructura de un fichero extensión dat

- $< nombre\_fichero.par >$ Cuando la partición de referencia no está incluída en la matriz de datos X, este fichero contiene información referida a dicha partición, su estructura es una columna que conserva el orden de asignación de la clase de referencia con respecto al orden de los individuos en el conjunto de datos de la matriz X. El Cuadro 10.2 muestra el formato de un fichero con extensión par.

nom_Obj	Clase
nom_1	id_1
nom_2	id_2
...	...
...	...
...	...
nom_n	id_n

Tabla 10.2: Estructura de un fichero extensión par

- $< nombre\_fichero.iks >$ Un fichero con extensión iks contiene información sobre el sistema de intervalos de longitud variable correspondiente a el atributo X_k, su estructura consiste de un renglón donde se encuentran los 2ξ valores límites del sistema de intervalos separados por al menos un espacio. El Cuadro 10.3 muestra el formato de un fichero con extensión iks.

Z_1	Z_2	Z_3	...	...	$Z_{2\xi-1}$	$Z_{2\xi}$

Tabla 10.3: Estructura de un fichero extensión iks

- $< nombre\_fichero.dci >$ Un fichero con extensión dci contiene información sobre la tabla de distribuciones condicionadas a intervalos para un cierto atributo X_k cuya estructura es la siguiente: en la primera línea van los límites de los intervalos y en el resto de las casillas los valores de la función de pertenencia p_{sc} por clase C, todos sus

elementos están separados por al menos un espacio. El Cuadro 10.4 muestra el formato de un fichero con extensión dci.

$$\begin{array}{cccccc} p_{11} & p_{21} & p_{31} & \cdots & \cdots & p_{(2\xi-1)1} \\ p_{12} & p_{22} & p_{32} & \cdots & \cdots & p_{(2\xi-1)2} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ p_{1\xi} & p_{2\xi} & p_{3\xi} & \cdots & \cdots & p_{(2\xi-1)\xi} \end{array}$$

Tabla 10.4: Estructura de un fichero extensión dci

- $< nombre\_fichero.tex >$ Fichero que contiene la estructura de las instrucciones en código LaTeX de los gráficos de las funciones de pertenencia condicionadas a intervalos por atributo X_k y por clase C . El Cuadro 10.5 muestra el formato de un fichero con extensión tex.

$\%$ Contenido de la *figura grande* para el atributo X_k
$\backslash begin\{figure\}$
$\{\backslash setlength\ \{\backslash unitlength\{1pt\}\}$
$\backslash begin\{picture\}(540, 700)(50, -175)$
$\%$ Para cada clase Ci el origen de cada grafo se coloca en la posición
$\%$ $(0,\ 140 \cdot c)$, donde c varía de 0 a 3 en cada hoja tamaño
a4, utilizando la instrucción:
$\backslash put(0, 140 \cdot c)\{G_C\}$
donde G_C es el código LaTeX del grafo para la clase C, con $1 \leq$C$\leq \xi$.
$\backslash end\{picture\}$
$\backslash end\{figure\}$

Tabla 10.5: Estructura de un fichero extensión tex

- $< nombre\_fichero.srg >$ Fichero que contiene la estructura del sistema de reglas globales $\Re(X_k, \mathcal{P})$ para el atributo seleccionado X_k, el contenido de este fichero es recomendable sea tipo latex. Es un fichero en forma de columna de $(2\xi-1)(\xi)$ elementos, el sub-índice de la regla nos marca la posición en que se dispara la regla. El Cuadro 10.6 muestra el formato de un fichero con extensión srg.

$$\begin{array}{c} r_{11}: \ x_{ik} \in I_1^k \xrightarrow{p_{sc}} C1 \\ r_{12}: \ x_{ik} \in I_1^k \xrightarrow{p_{sc}} C1 \\ \cdots \\ \cdots \\ r_{2\xi-1,\xi}: \ x_{ik} \in I_{2\xi-1}^k \xrightarrow{p_{sc}} C\xi \end{array}$$

Tabla 10.6: Estructura de un fichero extensión srg

- $< nombre\_fichero.srr >$ Fichero que contiene la estructura de un sistema reducido de reglas $\Re^*(X_k, \mathcal{P})$ para el atributo seleccionada X_k cuando se ha optado por escoger algún criterio de agregación. Como una primera aproximación en este trabajo hemos

$$
\begin{aligned}
r_1 &: \; x_{ik} \in I_1^k \xrightarrow{p_{max}} C \\
r_2 &: \; x_{ik} \in I_2^k \xrightarrow{p_{max}} C \\
&\cdots \\
&\cdots \\
r_{2\xi-1} &: \; x_{ik} \in I_{2\xi-1}^k \xrightarrow{p_{max}} C
\end{aligned}
$$

Tabla 10.7: Estructura de un fichero extensión srr

elegido el criterio de máxima probabilidad, así que tenemos una regla por cada intervalo I_s^k del sistema $\mathcal{I}^k$. El Cuadro 10.7 muestra el formato de un fichero con extensión srr.

- $< nombre\_fichero.tcp >$ Fichero que contiene información sobre la comparación entre la clasificación de referencia (C) y la obtenida por el sistema de reglas ($\hat{C}$) en cada atributo, donde c_{ij} es el número de coincidencias entre ambas clasificaciones para el atributo X_k. El Cuadro 10.8 muestra el formato de un fichero con extensión tcp.

	C_1	...	...	C_ξ
$\hat{C}1$	c_{11}	...	...	$c_{1\xi}$
$\hat{C}2$	c_{21}	...	...	$c_{2\xi}$
...	...	...	...	...
...	...	...	...	...
$\hat{C}\xi$	$c_{\xi 1}$	...	...	$c_{\xi\xi}$

Tabla 10.8: Estructura de un fichero extensión tcp

- $< nombre\_fichero.vsg >$ Fichero que contiene la información de las probabilidades y consecuentes de las reglas que se disparán para los individuos del conjunto de prueba P_0. El Cuadro 10.9 muestra el formato de este tipo de fichero.

No.	C	P	...	C	P
1	C_{11}	p_{11}	...	$C_{1\xi}$	$p_{1\xi}$
2	C_{21}	p_{21}	...	$C_{2\xi}$	$p_{2\xi}$
...	...	...	...	...	...
...	...	...	...	...	...
n	C_{n1}	p_{n1}	...	$c_{n\xi}$	$p_{n\xi}$

Tabla 10.9: Estructura de un fichero extensión vsg

- $< nombre\_fichero.vsm >$ Fichero que contiene la información de las probabilidades máximas y consecuentes de las reglas que se disparán para los individuos del conjunto de prueba P_0. El Cuadro 10.10 muestra el formato de este tipo de fichero.

- $< nombre\_fichero.coi >$ Fichero que contiene la información sobre la coincidencias entre las clases de predicción ($\hat{C}$) y la referencia (C) para cada uno de los individuos del conjunto de prueba. El Cuadro 10.11 muestra la estructura de este tipo de fichero con extención coi.

No.	C	P_{max}
1	C_1	p_{1max}
2	C_2	p_{2max}
...	...	...
...	...	...
n	C_n	p_{nmax}

Tabla 10.10: Estructura de un fichero extensión vsm

No.	$\hat{C}$	C
1	$\hat{C}_1$	C_1
2	$\hat{C}_2$	C_2
...	...	...
...	...	...
n	$\hat{C}_n$	C_n
Número	de	coincidencias:

Tabla 10.11: Estructura de un fichero extensión coi

10.2.4 Descripción de los módulos del sistema CIADEC

La descripción de los módulos que forman la aquitectura de CIADEC se enfoca sobre el objetivo principal, la entrada, salida y expectativas de funcionamiento de cada uno de ellos.

Módulo I: Generación de Intervalos de Longitud Variable (GILOVA)

Dada el atributo de estudio X_k el módulo I tiene como principal objetivo el de generar un sistema de intervalos de longitud variable calculando los valores de $\mathcal{I}^k$. La llamada a la función principal del módulo es:

TablaInterv(X, X_k, P, int nclases)

Entrada: Los parámetros de entrada en este módulo son: el atributo a seleccionar X_k, la matriz de datos X y en su caso la partición P del conjunto de datos y el número de clases nclases de la partición.

Salida: Para cada atributo seleccionado X_k, como salida un vector que representa el sistema de intervalos de longitud variable y cuya estructura es la de un fichero con extensión iks, digamos el nombre del atributo en estudio y representado por $< X_k.iks >$.

Descripción: En este módulo se realizan las siguientes funciones al hacer la llamada a la función principal y son: Construir un vector con el mínimo y máximo de X_k en cada clase, ordenar los valores de ese vector de menor a mayor y generar el fichero .iks.

Módulo II: Generación de la Tabla de Distribuciones Condicionadas a Intervalos (GETADI)

El módulo II GETADI tiene como principal objetivo la generación de la tabla de distribuciones condicionadas a intervalos (o funciones de pertenencia). La llamada a la función principal del

módulo es:

TablaFrec(X, X_k, P, short nClases, short nInterv)

Entrada: Este módulo tiene como entrada el conjunto de datos X, el atributo seleccionado X_k, la partición P, el número de clases nClases y el número de intervalos nInterv.

Salida: La salida en este módulo es una tabla de distribuciones condicionada a intervalos de la forma $\mathcal{P}|I^k$ representada por medio de un fichero el nombre del atributo y con extensión dci ($< X_k.dci >$).

Descripción: En esta parte se lee nuevamente la columna X_k de la matriz de datos X, ubicando cada valor x_{ik} en el intervalo $\mathcal{I}_s^k$ y clase C correspondiente, contando el número de elementos en cada una de las casillas de la tabla $\mathcal{P}|I^k$ desde $k = 1 : 2\xi - 1$ y posteriormente sumando columnas obtenemos los $n_{I_s^k}$ para dividir cada casilla por su $n_{I_s^k}$ y determinar las probabilidades p_{sc} correspondientes.

Módulo III: Generación de Sistemas de Reglas (GESIRE)

El módulo III GESIRE genera primero, un sistema de reglas difusas de inducción basado en la matriz de distribuciones condicionada a intervalos por atributo seleccionado X_k, $\Re(X_k, \mathcal{P})$ en un fichero de tipo .srg para el sistema de reglas completo y una vez que hemos elegido un cierto criterio de agregación para reducir la ambiguedad inherente al sistema de reglas obtenemos un sistema de reglas reducido $\Re^*(X_k, \mathcal{P})$ que se guarda en un fichero tipo .srr. La llamada a la función principal del módulo es:

GeneradorReglas($< X_k.dci >$, X_k, short nClases, short nInterv)

Entrada: La entrada es un fichero con extensión .dci que representan la tabla $\mathcal{P}|I^k$ de distribuciones condicionada a intervalos del atributo X_k en estudio y sus dimensiones nClases y nInterv.

Salida: La salida es a dos niveles dependiendo del interés del usuario: si se desea el conjunto global de reglas es un fichero con extensión src, por otro lado si se desea elegir algún criterio de agregación la salida será un fichero con extensión srr.

Descripción: En este módulo se construye el sistema global de reglas $\Re(X_k, \mathcal{P})$ por atributo seleccionado X_k a partir de la tabla de distribuciones condicionada a intervalos y considerando un criterio de agregación (pj., el de probabilidad máxima) obtenemos un sistema reducido de reglas $\Re^*(X_k, \mathcal{P})$.

Módulo IV: Generación de Gráficos en LaTeX (GEGRALA)

El módulo IV GEGRALA tiene como objetivo generar gráficos para las funciones de pertenencia $f(X_k|C)$. La llamada a la función principal del módulo es:

GeneradorGrafico($< X_k.dci >, X_k.tex, X_k$)

Entrada: El fichero de entrada para la generación de gráficos LaTeX es la tabla $\mathcal{P}|I^k$ representadas por un fichero con extensión dci ($< X_k.dci >$).

Salida: La salida es de dos tipos:

- Como fichero con extensión .tex, con el nombre del atributo de estudio ($< nombre\_Atributo.tex >$).
- Y visualización en pantalla del gráfico.

La estructura de los ficheros tex es la que sigue un formato de fichero en LaTeX (ver Cuadro 10.5) que dibuje el gráfico.

Descripción: En este módulo la generación de gráficos de las tablas $\mathcal{P}|I^k$ de distribuciones se hace a partir de la tabla de distribuciones generada en el módulo II y considerando las operaciones de transfomación en 10.2.5.

Módulo V: Validación del Sistema de Reglas (VALIDA)

Este módulo representa la etapa final del proceso y es donde se realiza la validación del sistema de reglas que hemos obtenido cuando existe un conjunto de validación P_0, primero para cada una de los atributos X_k y luego una vez hecha la asignación de la clase correspondiente C a cada uno de los individuos i del conjunto de prueba P_0, comparándola con la clase de referencia asociada a cada uno de éstos obtener la calidad de asignación del sistema. La llamada a la función principal del módulo es:

ValidaReglas(FileDat $P_0.dat$, short nInterv)

Entrada: Recibe como entrada el conjunto de prueba P_0 donde cada individuo tiene asignada la clase (C) que le corresponde de acuerdo a la clasificación de referencia y por otro lado, la tabla $\mathcal{I}^k|P$ ($< X_k.dci >$). Con esta tabla y el valor x_{ik} se puede calcular la clase de predicción ($\hat{C}$) de cada individuo y aplicando el criterio de probabilidad máxima obtener el sistema reducido de reglas.

Salida: La salida es un fichero con extensión tcp que determina el número de coincidencias entre las clases de referencia (C) y de predicción ($\hat{C}$) para los elementos de P_0.

Descripción: En este módulo se hace una comparación cruzada entre la clase asignada (la de referencia) y la de predicción (la asignada debido al sistema de reglas reducido) para determinar el grado de confiabilidad de nuestro sistema de reglas. Se calcula en % de error.

10.2.5 Sobre la Generación de Gráficos en LaTeX

En esta parte se propone una forma de representar gráficamente este sistema de reglas que permite obtener conocimiento útil y comprensible para la interpretación conceptual de las clases identificadas.

A nivel diseño este módulo se hizo en forma imbricada a dos niveles para la reutilización del código LaTeX de estos gráficos en documentos posteriores.

1. **A nivel de *figura grande***. La generación del paquete de *grafos de interpretación* para la partición $\mathcal{P}$ de un cierto dominio. Para hacer esto, se genera una *figura grande* en la que están imbricados todos los grafos $G_c, 1 \leq c \leq \xi$, donde ξ es el número de clases de la partición $\mathcal{P}$ (20 en nuestro caso). En este nivel generamos una *figura grande* con las siguientes convenciones e instrucciones de LaTeX.

Definimos: $w = 540$ el ancho de la figura grande, $h = 700$ la altura de la figura grande (paquete), $(x_0, y_0) = (50, -175)$ el origen de la figura (esquina inferior izquierda).

Las instrucciones en LaTeX a este nivel por página son:

% Contenido de la *figura grande* para la variable X_k.

$\backslash begin\{figure\}$ $\{\backslash setlength$ $\{\backslash unitlength\{1pt\}\}$

$\backslash begin\{picture\}(540, 700)(50, -175)$

Para cada clase Ci el origen de cada grafo se coloca en la posición $(0, 140 \cdot c)$, donde c varía de 0 a 3 en cada hoja tamaño a4, utilizando la instrucción:

$\backslash put(0, 140 \cdot c)\{G_{Ci}\}$

donde G_{Ci} es el código LaTeX del grafo para la clase Ci, con $1 \leq$Ci$\leq \xi$.

$\backslash end\{picture\}$

$\backslash end\{figure\}$

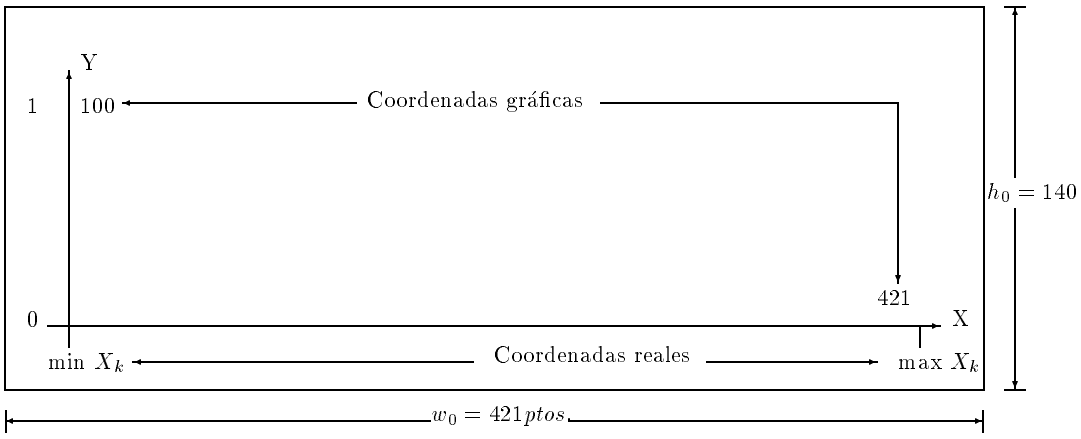

Figura 10.2: Diagrama de la figura grande

2. **A nivel de grafos**. La generación para cada una de las clases de la partición $\mathcal{P}$ de los grafos $\{G_{Ci}\}$, se hace tomando en cuenta los siguientes elementos.

   (a) Marco de cada grafo $\{G_{Ci}\}$.

   (b) Etiqueta del grafo correspondiente a la clase Ci.

   (c) Trazo, graduación y etiquetas del eje de las X's, así como las marcas de límites de los intervalos sobre este eje.

   (d) Trazo, graduación y etiquetas del eje de las Y's, así como las marcas de las probabilidades sobre este eje..

   (e) Función de pertenencia correspondiente al grafo G_{Ci}.

   - El Marco de cada grafo $\{G_{Ci}\}$. A este nivel de grafo para cada clase Ci se deberán tener los siguientes datos: Siendo, $w_0 = 421$ el ancho, $h_0 = 140$ la altura del grafo, $(x_0, y_0) = (50, -175)$ la esquina inferior izquierda del marco de cada grafo, Ci el índice de las clases, con $C1 \leq Ci \leq C\xi$, donde en este caso ξ es el número de

clases (en nuestro caso 20),este marco se obtiene con las siguientes instrucciones en LaTeX:

$\backslash begin\{picture\}(421,130)(0,0)$
{Elementos del grafo}
$\backslash end\{picture\}$

Los elementos del grafo son:

(a) Etiqueta del grafo correspondiente a la clase Ci.
$\backslash put(5,100)\{\$\{\backslash cal\ C\}\_\{c\}\$\}$

(b) Con respecto al eje de las X's

- Trazo del eje de las X's.
  $\backslash put(0,0)\{\backslash line(1,0)\{421\}\}$
- Graduación sobre el eje X. Representar las marcas graduales de longitud 10 sobre el eje X, dividimos la longitud l_x del eje X entre 8 y utilizando la siguiente instrucción:
  Para $i \leftarrow i+1,\ \ 0 : \mathrm{i} : 8$
  $\backslash put((l_x/8)\cdot i, 0)\{\backslash line(0,-1)\{10\}\}$
- Etiquetas sobre el eje X. Representar las etiquetas $\{e_i\}$ de las marcas sobre el eje X cada $\frac{M^k-m^k}{8}$ unidades a partir de la primera marca vertical, que indiquen la abcisa de la variable X_k que se está representando. La marca 0 coincide con el mínimo de la variable X_k, en general $e_i = m^k + \frac{M^k-m^k}{8}\cdot i$, con $0 : \mathrm{i} : 8$.
  Estas etiquetas se situarán exactamente debajo de cada marca con lo que sus coordenadas en X seran las mismas que para las marcas y las de Y seran constantes a -20 ptos., considerando que 10 ptos. por debajo del eje X estan ocupados por la propia marca y reservamos 10 ptos. para la etiqueta.

  Para $i \leftarrow i,\ \ 0 : \mathrm{i} : 8$
  $\backslash put((\frac{l_x}{8})\cdot i, -20)\backslash mbox[c]\{e_i\}$

- Marcas de límites de los intervalos sobre el eje X. Son marcas de longitud 5 sobre el eje X, que representan los límites de los intervalos I_s de la variable I^k sobre el eje X y convenientemente reescalada sobre el gráfico, con un factor de escala T_x entre el rango de la variable X_k y la longitud l_x del eje X. Así, si el rango de la variable X_k es $[m^k, M^k]$, donde: M^k es el máximo y m^k es el mínimo para la variable X_k y la longitud del eje X es l_x puntos entonces un valor x cualquiera estará posicionado en el grafo en la posición dada por la siguiente transformación:

$$x' = \frac{x-m^k}{M^k-m^k}\cdot l_x = (x-m^k)\cdot\frac{l_x}{M^k-m^k} = (x-m^k)\cdot T_x$$

  Las coordenadas de los intervalos están distanciados $\mid I_1 \mid, \mid I_2 \mid, \ldots, \mid I_{2\xi-1} \mid$ respectivamente. Sus posiciones sobre el eje de X son: $m^k + \mid I_1 \mid$, $m^k + \mid I_1 \mid + \mid I_2 \mid, \ldots, m^k + \mid I_1 \mid + \mid I_2 \mid + \ldots + \mid I_{2\xi-1} \mid$.

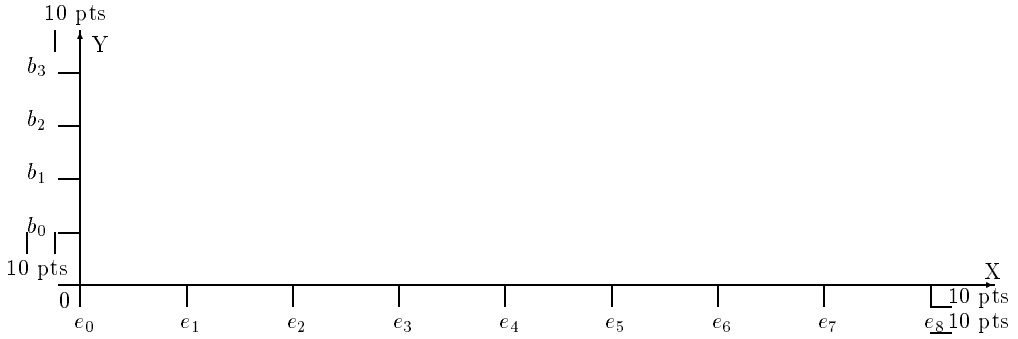

Figura 10.3: Posiciones de las etiquetas

Con lo cual los límites del intervalo I_s serán las posiciones

$$[min\ I_s = max\ I_{s-1}, max\ I_s]$$

y en términos de las magnitud de los intervalos I^k igual a:

$$[m^k + \mid I_1 \mid + \ldots + \mid I_{s-1} \mid,\ m^k + \mid I_1 \mid + \ldots + \mid I_s \mid]$$

Así transformando la marca del límite superior del intervalo I_j sobre el eje X está dada por la siguiente instrucción:
Para $j \leftarrow j+1,\ \ 0 : \mathrm{j} : 2\xi - 2$

$\backslash put(\sum_{s=1}^{j} |I_s| \cdot T_x, 0)\{\backslash line(0,-1)\{5\}\}$

donde, $|I_s| = \mid lim\ sup\ I_s - lim\ inf\ I_s \mid$ representa la longitud del intervalo I_s.

(c) Con respecto al eje de las Y's

- Eje de las Y's.
  $\backslash put(0,0)\{\backslash line(0,1)\{110\}\}$

- Graduación sobre el eje Y. Representar las marcas graduales de longitud 10 sobre el eje Y, dividimos la longitud l_y del eje Y entre 4 y utilizando la siguiente instrucción:
  Para $i \leftarrow i+1,\ \ 0 : \mathrm{i} : 3$
  $\backslash put(0, (\frac{l_y}{4} \cdot i))\{\backslash line(-1,0)\{10\}\}$

- Etiquetas sobre el eje Y. Representar las etiquetas $\{b_i\}$ de las marcas sobre el eje Y cada $\frac{l_y}{4}$ unidades a partir de la primera marca horizontal, que indiquen la probabilidad que se está representando. La marca 0 coincide con la probabilidad 0 y en general se tiene que:
  Para $i \leftarrow i+1,\ \ 0 : \mathrm{i} : 3,\ \ b_i = \frac{l_y}{4} \cdot i$ y su ubicación utilizando la siguiente instrucción:

  $\backslash put(-25, (\frac{l_y}{4} \cdot i))\backslash mbox[c]\{b_i\}$

- Marcas de las probabilidades sobre el eje Y. Son marcas de longitud 5 sobre el eje Y, que representen los valores de la distribución de probabilidad de X_k condiconada a los intervalos I^k de la clase Ci. Considerando la longitud del eje Y, $l_y = 100$ ptos, tenemos que el factor de escalamiento $R = 100$ y la localización de estas probabilidades es directa. Esto es, y' $= 100 \cdot y$, quedando las marcas para la clase Ci de la siguiente forma:
  Para la clase Ci se tiene:
  $j \leftarrow j + 1,\ \ 0 : \mathrm{j} : 2\xi - 2$

  $\backslash put(0, p_{jc} \cdot 100)\{\backslash line(-1, 0)\{5\}\}$

A nivel de función de pertenencia.

A este nivel se tienen dos pasos importantes:

- Marcar los límites de los intervalos I_s sobre el eje X.
- Dibujar la función escalonada que sobre cada I_s vale p_{sc}.

Las coordenadas del grafo G_{Ci} de la función de pertenencia para la clase Ci, en el intervalo I_s son

$$((m^k + \textstyle\sum_{j=1}^{s-1} |I_j|),\ p_{sc})$$

donde: I_s es el s-ésimo intervalo de I^k, p_{sc} es la probabilidad sobre el s-ésimo intervalo y $\mid I_j \mid$ es la longitud del j-ésimo intervalo de I_k.

Transformando estas coordenadas para ubicarlas en el marco del grafo se tiene:

$$(\sum_{j=1}^{s-1=39} |I_j|),\ p_{sc} \cdot 100)$$

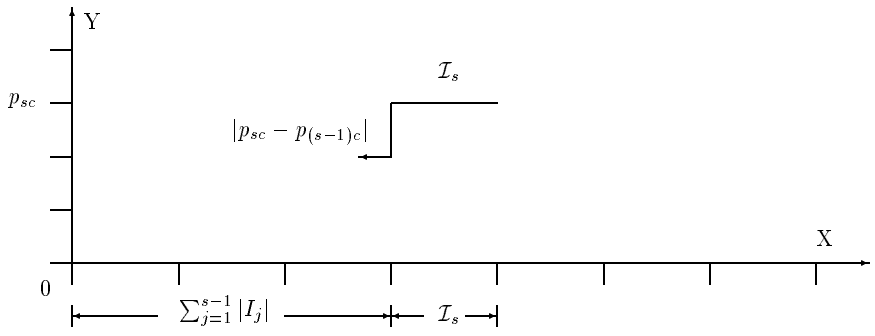

Figura 10.4: Función de pertenencia para la clase Ci

Sobre este intervalo, la función de pertenencia toma valores de línea horizontal de longitud $|I_s|$, a fin de tener una función continua, uniendo los segmentos horizontales con otros verticales que salvará el salto entre $p_{(s-1)c}$ y $p_{(s)}$.

Situados en el origen de cooordenadas trazaremos una contínua vertical hasta la primera probabilidad y una horizontal sobre el primer intervalo, luego subiremos o bajaremos

hasta el valor de la segunda probabilidad y a continuación una segunda horizontal sobre el segundo intervalo y así sucesivamente hasta el trazo de una última horizontal sobre el último intervalo $I_{2\xi-1}$. Definimos ahora: $I_0 = 0$ y $p_{0c} = 0$ y utilizamos las siguientes instrucciones:

Calculamos los segmentos verticales sobre el eje Y correspondientes a los saltos de la función de pertenencia y vamos dibujando el valor de dicha función sobre cada I_s.

Para $s \leftarrow s+1,\ \ 0 : \mathrm{s} : 2\xi - 2 = 38$

Si $p_{sc} - p_{(s-1)c} \geq 0$ entonces:

$\backslash put((\sum_1^s |I_{s-1}|) \cdot T_x,\ p_{sc} \cdot T_y)\{\backslash line(0,1)\{p_{sc} - p_{(s-1)c}\}\}$

sino:

$\backslash put((\sum_{j=1}^{s-1} |I_j|) \cdot T_x,\ p_{sc} \cdot T_y)\{\backslash line(0,-1)\{|\ p_{sc} - p_{(s-1)c}\ |\}\}$

después:

$\backslash put((\sum_{j=1}^{s} |I_j|) \cdot T_x,\ p_{sc} \cdot T_y)\{\backslash line(1,0)\{|\ I_s\ |\ \cdot T_x\}\}$

Utilizando la misma forma imbricada de figuras, el otro tipo de gráfico en este módulo, el de gráficos que representan la combinación de atributos se hizo de la siguiente forma: Considerando como entrada el conjunto de prueba P_0, se lee para el primer individuo $i = 1$ el valor de la primer atributo X_1, se localiza el intervalo que le corresponde en la matriz de distribuciones condicionadas a intervalos de ese atributo y se toma ese renglón con sus clases y probabilidades correspondientes y se construye el primer gráfico ; luego se lee el segundo atributo X_2 se localiza el intervalo a que corresponde en la matriz de distribuciones de ese atributo y se construye este gráfico, que representará todas las clases asociadas con sus correspondientes probabilidades y así sucesivamente hasta el atributo X_{17}, de todo esto obtenemos 17 gráficos para el primer individuo. En seguida se considera el segundo individuo $i = 2$ y se repite el proceso anterior y así sucesivamente hasta agotar todos los individuos del conjunto de prueba (30 elementos), de tal forma que obtengamos 17 gráficos que representan las clases y sus correspondientes probabilidades de los 17 atributos seleccionadas por individuos en el conjunto de prueba P_0.

A nivel algoritmo se definen dos constructores: Class GeneradorGrafico y Class GeneradorLatex.

10.3 Implementación del sistema CIADEC

CIADEC es un Sistema orientado a la caracterización e interpretación automática de clases en dominios poco estructurados implementado en el lenguaje de programación JAVA 2 SDK, versión 1,3,1_01 en la plataforma Windows 95/ 98/ 2000/ NT4.0. El sistema deberá operar en un entorno de PC´ s y se ha desarrollado a partir de la metodología formal descrita en el ya citado reporte [VG01a].