

UNIVERSIDAD CENTRAL "MARTA ABREU" DE LAS VILLAS  
FACULTAD DE MATEMÁTICA, FÍSICA Y COMPUTACIÓN  
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN



---

# Una arquitectura basada en agentes para la clasificación en entornos distribuidos

---

TESIS PRESENTADA EN OPCIÓN AL TÍTULO DE MASTER EN CIENCIA DE  
LA COMPUTACIÓN

*Autor:* INTI YULIEN GONZÁLEZ HERRERA

*Tutor:* DR. CARLOS PÉREZ RISQUET

Santa Clara, 25 de Septiembre de 2010

Algunas veces, pocas dirán, he visto maestros. He preferido siempre el respeto que no proviene de la autoridad, sino del reconocimiento.

Mi alter ego

## Resumen

En este trabajo se presenta el diseño de una arquitectura basada en agentes para la clasificación de bases de datos homogéneas en ambientes distribuidos, que utiliza como modelo de decisión los sistemas de reglas difusas, con el objetivo de obtener clasificadores interpretables. En su diseño se emplean agentes de minería que construyen conjuntos de reglas con validez en los datos locales siguiendo diferentes procedimientos. Las reglas construidas tienen un modelo común de inferencia, que permite a un agente mediador combinar los resultados y utilizar las reglas ganadoras como explicación al usuario. Los sistemas de reglas utilizados se basan en la lógica difusa clásica de Zadeh u otras lógicas multivalentes. Los resultados locales se obtienen usando diferentes procedimientos que maximizan la comprensibilidad sin sacrificar la precisión. En la propuesta, adicionalmente, pueden construirse sistemas de reglas de forma visual. Se definen también métodos de combinación que permiten conservar la interpretabilidad de los sistemas de reglas difusas en ambientes distribuidos. Se realiza una validación estadística de tres aspectos de la arquitectura propuesta a través de experimentos diferentes. El primer grupo de experimentos pretende evaluar el comportamiento, tanto en precisión como en interpretabilidad, de un clasificador construido visualmente. El segundo grupo mide el efecto de utilizar la lógica difusa compensatoria en los antecedentes de un sistema de reglas difusos en lugar de la lógica difusa clásica y el tercero mide la validez de los métodos de combinación de resultados. Los puntos claves son: el diseño de una técnica visual para la construcción de sistemas de inferencia difusos, la adaptación de un método evolutivo de generación de reglas difusas para utilizar la lógica difusa compensatoria y la integración de los resultados locales a través de estrategias de voto inspiradas en la selección de reglas ganadoras.

# Índice general

<b>Introducción</b>	<b>1</b>
<b>1. Estado del arte</b>	<b>5</b>
1.1. Minería de datos . . . . .	5
1.1.1. Clasificación . . . . .	6
1.1.2. Sistema de reglas difusas para la clasificación . . . . .	6
1.1.3. Interpretabilidad en los FRS . . . . .	7
1.1.4. Extracción de reglas difusas . . . . .	7
1.1.4.1. Diferentes enfoques . . . . .	8
1.1.4.2. Selección de reglas difusas para la clasificación siguiendo cri- terios multiobjetivos . . . . .	9
1.2. Minería visual de datos . . . . .	13
1.2.1. Construcción visual de clasificadores. . . . .	15
1.3. Minería de datos distribuida . . . . .	17
1.3.1. Sistemas multiagentes . . . . .	19
1.4. Lógica Difusa Compensatoria . . . . .	21
1.4.1. Definiciones . . . . .	21
1.4.2. Propiedades . . . . .	24
1.4.2.1. Relación de la Lógica Difusa Compensatoria con la Lógica Booleana . . . . .	24

1.4.2.2.	Compatibilidad con la Teoría de la Medición . . . . .	24
1.5.	Conclusiones . . . . .	25
<b>2.</b>	<b>Diseño de una arquitectura basada en agentes para la clasificación en entornos distribuidos conservando la interpretabilidad</b>	<b>26</b>
2.1.	Adaptación a la construcción evolutiva de sistemas de reglas difusas mediante el uso de antecedentes basados en predicados de la LDC . . . . .	27
2.2.	Propuesta para la construcción visual de un sistema de reglas difusas . . . . .	29
2.2.1.	Descripción de la técnica. . . . .	31
2.3.	Arquitectura multiagentes para el uso de reglas difusas en ambientes distribuidos	33
2.3.1.	Agentes para la integración de resultados . . . . .	35
2.3.2.	Agentes para la búsqueda visual de reglas difusas . . . . .	38
2.4.	Conclusiones . . . . .	38
<b>3.</b>	<b>Resultados Experimentales</b>	<b>40</b>
3.1.	Diseño de los experimentos . . . . .	40
3.2.	Conjuntos de datos a usar en los experimentos . . . . .	41
3.3.	Resultados experimentales de la técnica para la extracción visual de reglas difusas. . . . .	42
3.4.	Resultados experimentales en agentes para el aprendizaje de reglas basados en LDC. . . . .	45
3.5.	Validación de los métodos de combinación. . . . .	47
3.6.	Conclusiones . . . . .	49
	<b>Conclusiones</b>	<b>51</b>
	<b>Recomendaciones</b>	<b>53</b>
	<b>Anexos</b>	<b>62</b>

# Introducción

La comprensibilidad es una de las características más atractivas que se buscan en las técnicas de minería de datos (DM). En el contexto de la clasificación existen un grupo grande de modelos que actúan como cajas negras. Estos pueden resultar muy útiles para efectuar tareas de decisión pero los expertos humanos no pueden nutrirse del conocimiento extraído ni confrontar su conocimiento con la información extraída automáticamente. Los árboles de decisión, los sistemas de reglas, los sistemas de reglas difusas y otros clasificadores como el K-NN son modelos que ofrecen una explicación de sus decisiones.

En entornos distribuidos la mayoría de los trabajos utiliza como enfoque para la clasificación la combinación de los resultados mediante la fusión, la selección y otras técnicas de propósito general que son independientes de los modelos de clasificación locales. En términos de precisión este enfoque resulta efectivo, el comportamiento del meta-clasificador puede incluso sobrepasar el rendimiento de los clasificadores locales, sin embargo, de esta forma se pierde la posibilidad de mostrar la relación entre la decisión y los datos, los clasificadores globales se tornan incomprensibles.

Un enfoque alternativo en entornos distribuidos es utilizar métodos de combinación de resultados que estén definidos para técnicas locales específicas. Por ejemplo: se pueden buscar conglomerados en datos locales y luego construir resultados globales [47]. Esto no basta para definir modelos comprensibles, pero si los sistemas locales son comprensibles y los métodos de combinación se diseñan para preservar esta propiedad puede obtenerse un clasificador con esta característica. Los árboles de decisión distribuidos son ejemplo de este enfoque aplicado a la tarea de clasificación.

Problemáticas adicionales surgen en configuraciones distribuidas extremas. Entre estas puede estar la imposibilidad de mover los datos entre sitios producto de su volumen y las restricciones de privacidad que impidan compartir datos o incluso modelos construidos sobre los datos. Bajo estas condiciones el enfoque tradicional de almacén de datos central se vuelve

impráctico. Los retos que aparecen en la construcción de modelos de clasificación distribuidos con condiciones restrictivas han sido abordados mediante el uso de agentes [32, 11]. Las características propias que presentan, cooperar, competir, adaptarse y ser autónomos, resultan útiles para arquitecturas de sistemas de minería de datos distribuida.

La modelación difusa del conocimiento es un enfoque muy exitoso producto de su expresividad y capacidad para representar la incertidumbre y la vaguedad. Los modelos usados se basan en diferentes definiciones de los operadores. Entre los modelos para la clasificación se encuentran los árboles de decisión difusa y los sistemas de reglas difusas (FRS). Estos últimos son útiles para la construcción de sistemas expertos, donde las reglas pueden construirse manualmente o mediante el aprendizaje basado en casos, ofrecen además elevada precisión. La característica más notable de los FRS es, sin embargo, que representan la información extraída de forma muy comprensible.

Una arquitectura para la clasificación de bases de datos homogéneas basada en agentes y que utilice como modelos de decisión los sistemas de reglas difusas es un enfoque de interés. Por eso en este trabajo se presenta un diseño de dicha naturaleza. En la misma, agentes de minería construyen conjuntos de reglas con validez en los datos locales siguiendo diferentes procedimientos. Las reglas construidas deben tener un modelo común de inferencia. Agentes mediadores combinan los resultados y utilizan las reglas ganadoras como explicación al usuario.

Los modelos difusos utilizados se basan en la lógica difusa clásica de Zadeh u otras lógicas multivalentes. La lógica difusa compensatoria (LDC) es un modelo lógico multivalente que renuncia a varios axiomas clásicos para lograr un sistema idempotente y sensible que asimila virtudes de las escuelas normativas y descriptivas.

Los resultados locales se obtienen usando diferentes procedimientos que maximizan la comprensibilidad pero no a costa de la precisión. Adicionalmente, pueden construirse sistemas de reglas de forma visual. Un enfoque como este permite a los expertos una mayor comprensión de las relaciones entre las variables.

Los sistemas multiagentes han sido utilizados para realizar tareas de minería de datos respetando las restricciones de seguridad [47]. Por otro lado se han presentado trabajos para la construcción de FRS a partir de datos centralizados que se distribuyen para compartir el esfuerzo computacional [52]. La combinación de las decisiones de las reglas en FRS ha sido abordada por varios estudios.

Un área de investigación activa en la minería visual de datos es la construcción de sistemas

de clasificación a partir de la visualización de los ejemplos de entrenamiento. La precisión de los modelos publicados resulta competitiva y ofrece ventajas adicionales. Los trabajos encontrados en la literatura se basan en la construcción de árboles de decisión, ninguno en la construcción de otros modelos.

La LDC ha sido aplicada en varios problemas [24, 25] pero ha sido poco evaluada en entornos donde los predicados - en el caso de este trabajo se trata de antecedentes - sean construidos a partir de datos. No se han encontrado adaptaciones de métodos de clasificación que la utilicen.

## Preguntas de investigación

- ¿La lógica difusa compensatoria puede introducir variaciones positivas al usarla en sistemas de reglas difusas?
- ¿Puede construirse un sistema de reglas difusas para la clasificación a partir de una técnica de minería de datos visual?
- ¿Los sistemas construidos visualmente tendrán un comportamiento comparable a otros modelos en cuanto a precisión del clasificador y comprensibilidad del sistema?
- ¿La combinación de resultados locales a partir de características específicas de los sistemas de reglas borrosos puede redundar en meta-clasificadores con buen rendimiento para entornos distribuidos con datos no compartidos?

## Hipótesis de investigación.

- La introducción de la lógica difusa compensatoria puede incrementar la precisión de los clasificadores basados en sistemas de reglas difusas.
- Una técnica para la construcción visual de clasificadores puede proveer sistemas de inferencia con un rendimiento similar a otros modelos de clasificación.
- La integración de resultados definida específicamente para los sistemas de reglas difusas brinda rendimientos similares a otros meta-clasificadores con el valor adicional de mantener la comprensibilidad del conocimiento.

## Objetivo General

Proponer una arquitectura basada en agentes para la obtención de clasificadores interpretables en ambientes distribuidos.

## Objetivos Específicos

1. Definir los componentes de la arquitectura y su modo de interacción.
2. Evaluar el efecto de usar la lógica difusa compensatoria en sistemas de clasificación mediante reglas difusas.
3. Desarrollar un enfoque para la construcción visual de sistemas de reglas difusas.
4. Definir métodos de combinación que permitan mantener la interpretabilidad de los sistemas de reglas difusas en ambientes distribuidos.

# Capítulo 1

## Estado del arte

La minería de datos es un campo del conocimiento con aplicaciones en las ciencias, la industria y los negocios. Constituye además un área de activa investigación y desarrollo producto de los retos que subsisten y de los nuevos que van apareciendo al integrarse en nuevos entornos tecnológicos. Producto de la gran cantidad de enfoques con que han sido tratadas las problemáticas que aborda es necesario restringir el análisis a un subconjunto limitado de ellos.

En este capítulo se tratan un grupo de temas de interés general. En la sección 1.1 se presenta una definición de la minería de datos y se aborda el problema de la clasificación mediante sistemas de reglas difusas. La minería visual de datos y algunos enfoques novedosos para su aplicación a la construcción de modelos para la clasificación, son descritos en la sección 1.2. La sección 1.3 presenta la necesidad de replantear las técnicas de DM para adaptarse a ambientes de datos distribuidos y muestra enfoques basados en agentes inteligentes. La lógica difusa compensatoria, una lógica multivalente, es brevemente descrita en la sección 1.4. Finalmente, se realiza un resumen analítico de estos campos y su modo de integración.

### 1.1. Minería de datos

La tecnología actual brinda la posibilidad de obtener gran cantidad de datos en diversos campos de aplicación. Los datos contienen información, pero el carácter complejo de los mismos, junto a su volumen, impiden identificarla con facilidad. La minería de datos está definida como el proceso de descubrir patrones en los datos. El proceso debe ser automático

o (más frecuentemente) semiautomático. Los patrones descubiertos han de ser relevantes porque brinden alguna ventaja, usualmente una ventaja económica. Los datos se encuentran invariablemente en cantidades sustanciales [67].

La minería de datos trata cuatro problemas fundamentales: búsqueda de conglomerados, búsqueda de reglas de asociación, la clasificación y la regresión. Los problemas de búsqueda de reglas de asociación y de conglomerados son de aprendizaje no supervisado y los de clasificación y regresión; que están estrechamente relacionados, son de aprendizaje supervisado [67, 70].

### 1.1.1. Clasificación

Sean  $U_1, U_2, \dots, U_n$  y  $C = \{v_1, \dots, v_c\} \subset \mathbb{N}$  conjuntos, donde  $|C| = c$ . Un conjunto de datos de entrenamiento es de la forma  $T = \{ \langle X, Y \rangle : X \in U_1 \times U_2 \times \dots \times U_n, Y \in C \}$  con  $|T| = m$  y a cada par de  $T$  se le conoce como ejemplo de entrenamiento. Una instancia o patrón es un vector  $x \in U_1 \times U_2 \times \dots \times U_n$  y el problema de clasificación consiste en asignar a una instancia un elemento del conjunto  $C$ . En el contexto de la clasificación a los elementos de  $C$  se les conoce como clases y la entidad que es capaz de asignar las clases es un clasificador. Un conjunto de entrenamiento  $T$  es utilizado para construir un clasificador mediante, por ejemplo, el aprendizaje automático.

Existe una amplia variedad de modelos de clasificación. Pueden citarse los modelos basados en reglas, los árboles de decisión, los modelos estadísticos, las redes neuronales, los sistemas de reglas difusas y otros [67, 70, 33].

### 1.1.2. Sistema de reglas difusas para la clasificación

Sea  $T$  un conjunto de entrenamiento, se puede asumir entonces que existen conjuntos difusos y sus funciones de membresía para cada atributo. Se utilizan reglas del siguiente tipo para la clasificación de nuevos patrones:

$$\begin{aligned} \text{Rule } R_q &: \text{if } x_1 \text{ is } A_{q1} \text{ and } \dots \text{ and } x_n \text{ is } A_{qn} & (1.1) \\ &\text{then Class } C_q \text{ with } CF_q, \quad q = 1, 2, \dots, N_{rules} \end{aligned}$$

donde  $R_q$  es la etiqueta de la q-esima regla difusa,  $x = (x_1; \dots; x_n)$  es un vector n-dimensional,  $A_q = (A_{q1}; \dots; A_{qn})$  es el vector de antecedentes difusos con etiquetas lingüísticas (i.e.,  $A_{qi}$

es un termino lingüístico como “pequeño” y “grande”),  $C_q$  es una clase que resulta el consecuente,  $CF_q$  es el peso de la regla, y  $N_{rule}$  es el número de reglas difusas. El peso de la regla  $CF_q$ , que puede verse como el grado de certeza de la regla difusa  $R_q$ , es usado como la fortaleza de  $R_q$  en el razonamiento difuso.

Sea  $S$  un conjunto de reglas difusas de la forma descrita en 1.1. Un nuevo patrón  $x_p$  es clasificado por una regla ganadora única  $R_w$ , que es elegida del conjunto de reglas  $S$  según:

$$\mu_{A_w}(x_p) \cdot CF_w = \max\{\mu_{A_q}(x_p) \cdot CF_q \mid R_q \in S\} \quad (1.2)$$

Si múltiples reglas difusas tienen el mismo producto máximo pero diferentes clases en el consecuente, al clasificar un nuevo patrón  $x_p$ , la clasificación de  $x_p$  es rechazada. La clasificación puede ser rechazada también si ninguna regla de  $S$  es compatible con el nuevo patrón  $x_p$  (esto es,  $\mu_{A_q}(x_p) = 0, \forall R_q \in S$ ).

### 1.1.3. Interpretabilidad en los FRS

En la construcción de sistemas de reglas difusas se tienen dos objetivos fundamentales: maximizar la precisión de los clasificadores y minimizar la complejidad de los sistemas. La complejidad de los sistemas de reglas difusas puede medirse por el número de reglas que tiene, la cantidad de términos lingüísticos que utiliza y la longitud promedio de las reglas que lo componen. Cuando un FRS tiene una complejidad baja se dice que es interpretable.

El problema es que estos dos objetivos son aparentemente incompatibles, pues una forma de incrementar la precisión es aumentar el tamaño de los sistemas. Los métodos de generación de FRS deben establecer un compromiso para mejorar el rendimiento [39]. En varios estudios publicados se muestra como es posible obtener una alta precisión a partir de pocas reglas simples.

### 1.1.4. Extracción de reglas difusas

En esta sección se realiza una breve revisión de diferentes enfoques desarrollados para la generación de sistemas de reglas difusas. Se hace hincapié en la búsqueda con métodos evolutivos siguiendo criterios multi-objetivos.

#### 1.1.4.1. Diferentes enfoques

La obtención de reglas difusas para la clasificación es un problema abordado desde varias perspectivas. Las herramientas utilizadas incluyen técnicas neuro-difusas, métodos basados en conglomerados y algoritmos evolutivos. Entre las soluciones descritas están las basadas en colonias de hormigas [13, 29], las basadas en optimización mediante algoritmos genéticos [42, 53, 15, 39] y otros [53, 1].

La construcción de sistemas de reglas difusas utilizando la meta-heurística de optimización mediante colonias de hormigas (ACO) ha sido tratada por varios autores. Los resultados muestran una precisión de los clasificadores equivalente a otros métodos bien conocidos.

En el contexto de los métodos basados en algoritmos genéticos se distinguen tres corrientes principales:

1. En los algoritmos estilo Michigan cada regla representa un individuo de la población. El sistema de inferencia difuso es representado por toda la población. Puesto que diversas reglas participan en el proceso de inferencia, estas se encuentran en un estado constante de competencia y a la vez cooperan para formar un sistema de inferencia eficiente [12, 17, 35].
2. En el estilo Pittsburgh cada individuo representa todo un sistema de inferencia con varias reglas. Una población está compuesta por varios sistemas de inferencia. Los operadores genéticos son aplicados para producir nuevas generaciones de sistemas difusos. Este enfoque permite incluir criterios adicionales en la función de calidad. La desventaja fundamental es el costo computacional, puesto que varios sistemas difusos deben ser evaluados en cada generación [59].
3. En los algoritmos iterativos, como en el enfoque Michigan, cada individuo codifica una sola regla. Un algoritmo evolutivo es utilizado para encontrar una sola regla que provee una solución parcial. El algoritmo evolutivo es usado iterativamente para descubrir nuevas reglas, hasta que se construye una nueva regla base. Este enfoque combina la velocidad del enfoque Michigan con la simplicidad de evaluación de calidad del enfoque Pittsburgh. Como en otros métodos de construcción incremental de reglas este puede crear particiones no óptimas del espacio de antecedentes [18, 31].
4. Además de estos se encuentran las soluciones híbridas que combinan los enfoques [42].

Los enfoques propuestos a partir de modelos evolutivos permiten construir sistemas difusos en que son aprendidos varios de sus parámetros. Entre los parámetros que pueden aprenderse están las funciones de pertenencia a incluir en cada variable, los consecuentes de cada regla, los pesos de las reglas, la longitud de cada regla junto a su estructura y el número de reglas óptimo del sistema [65, 12, 13, 15, 18].

#### 1.1.4.2. Selección de reglas difusas para la clasificación siguiendo criterios multiobjetivos

Un problema común que deben enfrentar los enfoques evolutivos cuando se aplican sobre conjuntos de datos con un número alto de dimensiones es la complejidad del sistema de reglas generado. Esto significa que se pueden generar un número grande de reglas, cada una con muchos antecedentes. Esta problemática, aunque no afecta la precisión del clasificador tiene un impacto mayor en la interpretabilidad de las reglas por parte de los usuarios [39].

En [41] se presenta un enfoque que usa dos fases para extraer el FRS. En la primera un conjunto de reglas candidatas es generado a partir de la evaluación de las mismas en los datos. En la segunda fase se realiza una selección de las reglas que formarán el sistema de inferencia utilizando criterios multiobjetivo.

La generación de las reglas candidatas se realiza a partir del cálculo de dos medidas: la *confianza* y el *soporte*. Estas medidas han sido utilizadas tradicionalmente para medir la fortaleza de una regla de asociación [2]. Las reglas *if – then* pueden verse como reglas de asociación. La confianza de una regla mide cuál es el grado de validez de la regla, el soporte mide qué parte del conjunto de datos es cubierto por la regla.

La confianza de una regla puede calcularse como en [51]:

$$c(A_q \Rightarrow Class_q) = \frac{\sum_{x_p \in Class_q} \mu_{A_q}(x_p)}{\sum_{p=1}^m \mu_{A_q}(x_p)} \quad (1.3)$$

Mientras el soporte puede calcularse como:

$$s(A_q \Rightarrow Class_q) = \frac{\sum_{x_p \in Class_q} \mu_{A_q}(x_p)}{m} \quad (1.4)$$

Ambas expresiones utilizan el grado de compatibilidad. Este mide la compatibilidad de un patrón de  $x_p$  con el antecedente de una regla. Se define el grado de compatibilidad del patrón

$x_p$  con el antecedente  $A_q$  usando el operador producto

$$\mu_{A_q}(x_p) = \mu_{A_{q1}}(x_1) \times \mu_{A_{q2}}(x_2) \times \dots \times \mu_{A_{qn}}(x_n) \quad (1.5)$$

donde  $\mu_{A_q}$  es la función de membresía del conjunto  $A_{qi}$  de antecedentes difusos. Para la generación de reglas candidatas se determina el consecuente y el grado de certeza de cada regla. Estos valores pueden especificarse con facilidad a partir de la compatibilidad de los patrones de entrenamiento con el antecedente de la regla.

El consecuente de la clase  $C_q$  es especificado a partir de la confianza identificando la clase con el mayor valor.

$$c(A_q \Rightarrow C_q) = \max_{h=1, \dots, C} \{c(A_q \Rightarrow Class_h)\} \quad (1.6)$$

El grado de certeza de la regla se especifica, por ejemplo, para el caso de problemas con varias clases como

$$CF_q = c(A_q \Rightarrow C_q) - \frac{1}{C-1} \sum_{\substack{h=1 \\ h \neq C_q}}^C c(A_q \Rightarrow C_h) \quad (1.7)$$

En los casos anteriores en lugar de utilizar la confianza como medida se puede seleccionar el soporte, o el producto de ambos. En cualquiera de estos casos la selección del consecuente es igual. Si en la ecuación 1.6 existen varios argumentos que maximicen el valor de confianza entonces no puede especificarse el consecuente de la regla y por tanto, esta no es generada.

En este enfoque la transformación de cada variable se realiza definiendo términos lingüísticos con diferente granularidad. En primer lugar las variables son normalizadas y sobre ellas se pueden identificar, por ejemplo, los siguientes conjuntos difusos.

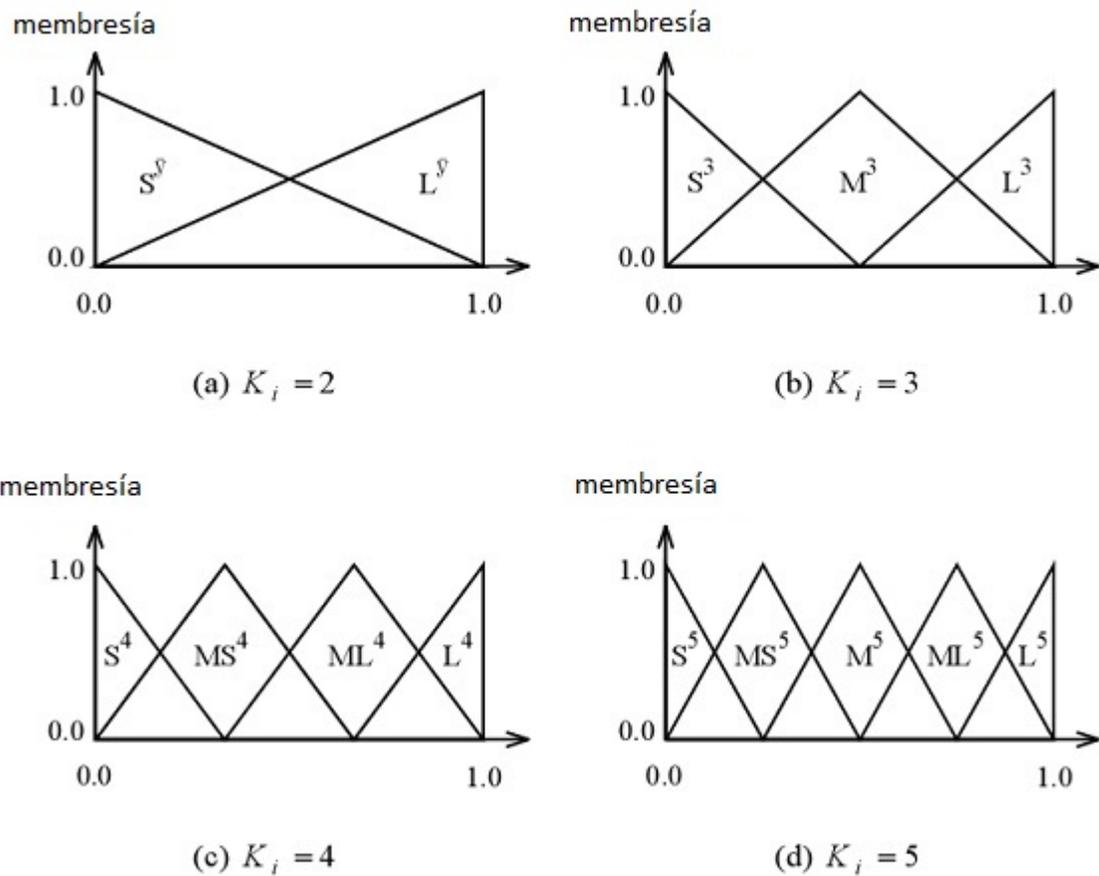


Figura 1.1: Partición difusa del intervalo  $[0, 1]$  que se utiliza. Se muestran diferentes grados de granularidad.

De esta forma se codifica cada regla difusa como un arreglo de los conjuntos difusos en el antecedente. Se utilizan números naturales para denotar cada conjunto difuso (e.g según la definición de la figura 1.1 se usarían los números del 1 al 14 puesto que hay 14 conjuntos). Un valor especial es utilizado para codificar la carencia de interés en el atributo correspondiente.

Las reglas generadas son aquellas cuyo número de antecedentes es menor o igual que un  $K$ . Este conjunto de reglas son divididas en  $C$  grupos según su consecuente y ordenadas descendientemente por el valor de confianza (soporte o su producto). Si se desean seleccionar  $N$  reglas se toman  $N/C$  reglas de cada grupo. El conjunto de  $N$  reglas es el resultado que provee la primera fase del método. La segunda fase del método utiliza una búsqueda evolutiva para determinar un subconjunto de reglas que conformen el sistema de inferencia difuso.

Los cromosomas son representados por una cadena binaria de longitud  $N$  que indica si una

regla pertenece o no al sistema de inferencia. Con esta representación es fácil definir el operador de cruzamiento y el de mutación [41]. La evaluación de un cromosoma  $S$  se realiza según la función

$$fitness(S) = w_1 \cdot f_1(S) - w_2 \cdot f_2(S) - w_3 \cdot f_3(S) \quad (1.8)$$

donde  $w_1 + w_2 + w_3 = 1$ ,  $f_1(S)$  es el número de patrones clasificados correctamente por las reglas en  $S$ ,  $f_2(S)$  es el número de reglas en  $S$  y  $f_3(S)$  es la longitud total de las reglas en  $S$ . El algoritmo citado usa además una población secundaria que constituye la élite y de la que se selecciona una parte para incluirla en la población principal en cada una de las generaciones.

A continuación se muestra el algoritmo general para la construcción del sistema de reglas difusas:

---

**Algorithm 1.1** Construcción del sistema de reglas difusas [41].

---

```
RuleSystem BuildClassifier(TrainingData data, int k, int n)
// DATA : CONJUNTO DE EJEMPLOS DE ENTRENAMIENTO
// K : ES LA CANTIDAD MÁXIMA DE CONSECUENTES DE UNA REGLA
// N : ES LA CANTIDAD DE REGLAS EN LA POBLACIÓN INICIAL
{
    // FASE 1: GENERACIÓN DE REGLAS
    rules = GenerateRules(k)
    GenerateConsequentAndCF(rules, data)
    rules = ObtainingRules(n, rules)
    // FASE 2: SELECCIÓN DE REGLAS
    population = GenerateRandomSubsets(n, popsize)
    secondaryPopulation = GenerateSecondary(population)
    CalculateFitness(population, rules, data) // ECUACIÓN 1.8
    while (!end)
    {
        better = ObtainingSystemWithHightFitness(population)
        // UTILIZANDO LA SELECCIÓN POR TORNEO BINARIO,
        // EL OPERADOR DE CROSSOVER
        // Y LA MUTACIÓN CON INCLINACIÓN
        population = GenerateNewPopulation(population)
        UpdateSecondaryPopulation(secondaryPopulation, population)
        CalculateFitness(population, rules, data)
    }
    return Subset(population, rules)
}
```

---

## 1.2. Minería visual de datos

La visualización de datos es crucial en la minería de datos y el descubrimiento de conocimiento. La visualización científica ha sido reconocida como una herramienta útil para la exploración de datos, la prueba de hipótesis y el análisis de resultados [34]. En el área específica de visualización de la información han sido desarrolladas gran cantidad de técnicas para enfrentar las diferentes problemáticas: análisis de conglomerados, identificación de valores fuera de rango, identificación de patrones y otros. El conjunto de técnicas y metodologías descritos para enfrentar estos problemas conforman la base de la minería visual de datos.

La visualización científica es un área que trata de explotar la capacidad humana para percibir patrones visuales. Con este propósito se crean imágenes, a partir de los datos, que son presen-

tados a los usuarios. La creación de una representación visual en cada técnica está definida de forma clara con el propósito de resaltar características de los datos. Una visualización puede ser una representación geométrica, matemática o física de un fenómeno o por el contrario puede ser alguna clase de metáfora visual de los datos. Las metáforas visuales son utilizadas en problemas específicos y pueden tener un gran impacto; desafortunadamente no pueden establecerse metáforas visuales para un problema arbitrario de minería de datos.

Toda técnica de visualización científica recibe como entrada un conjunto de datos y un conjunto de parámetros de configuración. Se ha probado mediante innumerables estudios que la configuración de parámetros a usar influye de manera notable en la expresividad visual. No existe, para la mayoría de las técnicas, una vía automática para la selección óptima de parámetros. Aunque en este sentido los investigadores han tenido poco éxito, algunos enfoques han sido publicados [54, 28, 57]. Hasta el momento, por lo tanto, la visualización se realiza asignando valores arbitrarios a los parámetros. Con el propósito de hacer frente a esta desventaja, las técnicas han sido modificadas para incluir la posibilidad de interacción.

La posibilidad de interacción visual es una de las características deseables de una técnica de visualización científica. Toda técnica de visualización al ser utilizada con datos reales debe enfrentar el problema de la pérdida de información y la interacción es un método efectivo para ello. Un usuario experto puede intervenir en el resultado visual modificando los parámetros para cambiar completamente la imagen aumentando la expresividad visual instantáneamente.

La minería visual de datos está basada en la visualización de datos multiparamétricos por lo que debe enfrentar sus mismas problemáticas. Los problemas que se observan al visualizar datos multiparamétricos han sido bien descrito en [60] y los trabajos allí citados. El problema fundamental es que no existe una proyección capaz de transmitir toda la información, por lo que los desarrolladores deben escoger qué tipo de información desean preservar. Adicionalmente debe considerarse que cada dimensión de un conjunto de datos puede tener diferente nivel de medición.

Una gran cantidad de investigaciones han sido desarrolladas en el área de la minería visual de datos. Algunas de las técnicas fundamentales son las matrices de diagramas de dispersión [60], las coordenadas paralelas [37] y las técnicas basadas en iconos. Existen además trabajos que presentan nuevos enfoques y metáforas visuales como los segmentos de círculo [7, 45], los patrones recursivos [46], las coordenadas en forma de estrellas [44] y otras. Todas estas técnicas han sido clasificadas siguiendo diferentes criterios. Un enfoque de clasificación muy

difundido está basado en el tipo de entidad visual que se utiliza. Siguiendo este criterio se tienen las técnicas *geométricas, basadas en pixel, basadas en iconos y las proyecciones*.

A continuación se realiza una breve descripción de algunas técnicas que tienen influencia sobre este trabajo:

***Coordenadas paralelas***: Esta técnica utiliza un eje de coordenada por cada dimensión del conjunto de datos. Los ejes de coordenadas están dispuestos de forma paralela entre sí. Cada muestra es representada a través de una polilínea que recorre todos los ejes cortándolos en base al valor de la muestra para el atributo correspondiente. Entre los parámetros con los que se puede interactuar se encuentran el color de cada muestra, el orden en que son representados los atributos y la forma en que se representan las variables categóricas. La técnica es una de las más empleadas producto a su fácil implementación y alta expresividad. Esta es una técnica que ha sido blanco de modificaciones con diferentes objetivos. Una descripción detallada y algunas de sus ventajas y desventajas puede encontrarse en [58].

***Matrices de diagramas de dispersión***: Esta es una técnica que utiliza una matriz de visualización, cada celda de la matriz contiene un diagrama de dispersión entre un par de variables. La técnica muestra en cada celda una proyección del conjunto de datos. La diagonal principal de la matriz es subutilizada y la técnica es sensible al orden de las dimensiones. Algunas aplicaciones de esta técnica pueden verse en [60].

**Segmentos de círculo**: Se utiliza un círculo como imagen base. El círculo es dividido en segmentos y se asocia una variable a cada uno. Las muestras son ordenadas según una de las variables y se presentan según este orden desde el interior del círculo hacia fuera, cada variable de la muestra en el segmento que se le ha asignado y con el color que le corresponde en la escala de esa variable. A cada par  $\langle \text{muestra}, \text{variable} \rangle$  le corresponde un pixel. La técnica tiene la capacidad de mostrar un gran volumen de datos y permite revelar patrones y conglomerados [7].

### 1.2.1. Construcción visual de clasificadores.

Las técnicas de visualización son una herramienta para la minería de datos. La mayoría ha sido diseñada como apoyo a la obtención de conocimiento por parte de los usuarios, más que como herramienta de extracción y “digitalización” de ese conocimiento. Esto es, en muchas visualizaciones los usuarios pueden identificar con facilidad reglas de asociación, conglomerados y sus características e incluso reglas de clasificación, sin embargo no pueden

crear un modelo a partir de la imagen para clasificar nuevas instancias. El enfoque tradicional de la minería de datos ha tratado la visualización como una herramienta para identificar hipótesis que luego son probadas estadísticamente o usadas para mostrar resultados. Aunque este enfoque es sumamente útil, el uso de la visualización en otras áreas puede incrementar la efectividad en la minería de datos.

En [63, 6, 69, 62] se presentan resultados en este sentido, todos los enfoques permiten, a través de la interacción con los datos mediante la visualización, crear modelos de clasificación. Estos trabajos usan diferentes técnicas bien conocidas como base para su desarrollo.

En [6] se presenta una aplicación de la técnica de segmentos de círculo para la construcción de árboles de decisión. El modelo mostrado permite construir árboles con varios intervalos por dimensión y es muy útil para conjuntos de datos de gran volumen. El rendimiento del clasificador es similar a varios métodos bien conocidos para la construcción automática de árboles de decisión y el número de reglas es menor en muchos casos. Las condiciones de decisión en este modelo están basadas en la división de las variables en intervalos.

En [63] se muestran los resultados de la evaluación de un modelo para la construcción visual de árboles de decisión. En este caso la interacción visual utiliza como imagen base la técnica de coordenadas en estrella. Las condiciones de decisión en cada nodo están basadas en la pertenencia de la instancia a una región del espacio de proyección. La técnica resulta intuitiva incluso para usuarios sin experticidad. La precisión alcanzada por el clasificador en los experimentos es equivalente a los de otros clasificadores bien conocidos.

En [62] se presenta una extensión al enfoque anterior. En este caso la modificación involucra la posibilidad de elegir entre varios métodos de proyectar el conjunto de datos hacia la imagen. Las técnicas de proyección elegidas son las coordenadas en forma de estrella y la técnica de coordenadas paralelas. En el trabajo se presentan ventajas adicionales de este enfoque, en específico la posibilidad de extraer información adicional de los datos. Se realizan experimentos que muestran una precisión competitiva de los clasificadores construidos.

Un modelo de interacción visual para la clasificación es propuesto en [69]. El modelo utiliza, en específico, la técnica de coordenadas paralelas. El primer paso del modelo es proyectar los datos hacia la imagen según esta técnica. A continuación es usado un optimizador basado en el análisis de discriminante lineal para adecuar la visualización a la tarea de clasificación. Esta visualización optimizada puede ser utilizada para construir un árbol de decisión y obtener un sistema de reglas. El enfoque tiene el mérito de imbuir al usuario en el proceso de construcción del clasificador y de hacerlo comprender la naturaleza de los datos.

### 1.3. Minería de datos distribuida

La minería de datos y extracción de conocimientos, desde fuentes de datos de gran tamaño, es un importante área de investigación. Las oportunidades de mayor impacto están en su aplicación sobre almacenes de datos para negocios, control de procesos y atención personalizada a clientes en línea [56, 14]. La mayoría de las técnicas de minería de datos fueron desarrolladas para el caso centralizado y necesitan ser replanteadas para adaptarse al caso distribuido. La demanda creciente de pasar a usar fuentes de datos masivas, naturalmente distribuidas en una red de computadoras con limitados ancho de banda y recursos computacionales, ha motivado el desarrollo de métodos paralelos y distribuidos de extracción de conocimiento.

El problema de obtención de patrones, asociado a la extracción de conocimiento distribuido se conoce como minería de datos distribuida (DDM). Los métodos de DDM deben ser capaces de obtener resultados con validez parcial, analizando los datos individuales de cada sitio, que al ser enviados a una localización y combinados representen resultados globales. Un número considerable de técnicas de DDM están disponibles para diversas problemáticas, como las reglas de asociación distribuidas, los conglomerados distribuidos y la clasificación/regresión [2, 3, 20].

Uno de los enfoques más usados en aplicaciones de negocios para DDM es aplicar técnicas de DM en datos que provienen de fuentes variadas y que han sido recopilados en un almacén de datos central. A pesar del éxito comercial de este enfoque, la aplicación de DDM mediante la centralización de los datos puede ser impráctica o imposible para algunas configuraciones de negocios en ambientes distribuidos. Esto es producto de que entre los retos que las técnicas de DDM deben enfrentar están la autonomía, privacidad y escalabilidad. Por ejemplo, cuando los datos pueden ser vistos en el almacén de datos desde muchas perspectivas diferentes y a diferentes niveles de abstracción, se puede amenazar el objetivo de proteger los datos individuales de invasiones de privacidad. Los requerimientos de respetar un cierto grado de autonomía y privacidad sobre los datos locales pueden hacer la minería de datos monolíticas no factibles.

Problemas adicionales deben abordarse por la necesidad de aumentar la capacidad de analizar conjuntos de datos de gran volumen que están distribuidos sobre un número grande de sitios. Cualquier minería en línea de estos conjuntos de datos en un almacén de datos central puede ser prohibitivo en términos de costo de comunicación y procesamiento. En la actualidad, la mayoría de los trabajos en DDM usan procesamiento distribuido y la descomposición de los

problemas de minería de datos para dar capacidad a conjuntos de datos mas grandes. Una lección de las investigaciones recientes en DDM es que la cooperación entre procesos de DM distribuidos pueden redundar en resultados valiosos incluso sin un control centralizado.

Esto lleva a la pregunta de si existe algún valor real adicionado en usar los conceptos de la tecnología de agentes para el desarrollo de sistemas avanzados de DDM. En general, las características inherentes al software basado en agentes, de ser autónomos, capaces de adaptarse y de realizar razonamientos en grupo, parecen acoplarse muy bien a los requerimientos mencionados en los problemas de DDM. Agentes autónomos de minería de datos, como tipos especiales de agentes de información, pueden realizar varias operaciones de minería en nombre del usuario y en colaboración con otros agentes. Los sistemas de agentes de información cooperativos, para la minería de datos sobre fuentes de datos distribuidas, heterogéneas y masivas, parecen ser un enfoque natural a aplicar.

A continuación se presenta una breve descripción de algunos sistemas basados en agentes y que resultan representativos en el área de DDM. Tres características de estos sistemas son analizadas:

- Tipo de datos y medios de aplicar seguridad a los datos procesados.
- Técnicas de minería de datos aplicadas, implementación del sistema y agentes.
- Arquitectura con respecto a la coordinación y control central, ejecución del procesamiento de datos, y sobre la transmisión de agentes, de datos y modelos en el transcurso de las tareas de minería de datos.

**BODHI** ha sido diseñado como un marco de trabajo para tareas de DDM como aprendizaje inductivo supervisado y regresión, sobre localizaciones de datos heterogéneas. El sistema garantiza modelos de datos locales y globales correctos con un nivel bajo de comunicación en la red. *BODHI* ofrece intercambio de mensajes y estaciones de agentes para la ejecución de agentes móviles en cada sitio local. El proceso de minería es distribuido a las estaciones de agentes locales, los agentes se mueven bajo demanda y llevan sus estados, datos y conocimientos. Un agente central facilitador es responsable de iniciar y coordinar las tareas de DM.

**PADMA** trata el problema de DDM para sitios con datos homogéneos. Modelos de conglomerados parciales son primero obtenidos por agentes estacionarios locales en los

diferentes sitios. Todos los modelos locales son recolectados por un agente central que realiza un análisis de conglomerados de segundo nivel para obtener un resultado global. Los agentes individuales efectúan la búsqueda jerárquica de conglomerados para la clasificación de documentos de texto y la visualización de la información web.

**JAM** es un sistema multiagente escrito en Java y utilizado para el meta-aprendizaje en DDM. Diferentes clasificadores como Ripper, CART, ID3, C4.5, Bayes y WEPBLS pueden ser ejecutados sobre bases de datos heterogéneas por agentes que residen en un sitio o que son importados desde otro sitio de la red. Cada agente local construye un modelo de clasificación y diferentes agentes crean sus clasificadores usando diferentes técnicas. JAM provee un conjunto de agentes meta-clasificadores para combinar múltiples modelos que en muchos casos mejoran la precisión en las predicciones. Una vez que el clasificador combinado es obtenido, el agente central de JAM controla la ejecución de estos módulos para clasificar conjuntos de datos de interés.

**Papyrus** es un sistema escrito en Java para enfrentar tareas de DM sobre conglomerados de sitios heterogéneos en extensas redes. Soporta diferentes tareas y modelos predictivos incluyendo C4.5. Agentes de comunicación mueven datos, resultados intermedios y modelos entre conglomerados para realizar todo el procesamiento localmente y disminuir el tráfico en la red. Cada conglomerado de sitios tiene un nodo distinguido como el punto de acceso y control para los agentes. La coordinación del trabajo de todos los conglomerados es realizada por un nodo central o distribuido entre un grupo de nodos distinguidos. Papyrus soporta varios métodos para combinar e intercambiar los modelos predictivos obtenidos localmente mediante el uso de un lenguaje de marcas especial.

Un punto común en estas herramientas es que tratan de integrar el conocimiento extraído de los datos en localizaciones de redes distribuidas geográficamente con un tráfico mínimo en la red y maximizando el procesamiento local. Las soluciones de DDM basadas en agentes han sido aplicados en estos casos a bases de datos homogéneas y también a bases de datos heterogéneas.

### 1.3.1. Sistemas multiagentes

Observando el estado del arte de los sistemas DDM basados en agentes presentados en la sección anterior, resaltan los siguientes argumentos a favor o en contra del uso de agentes.

*Autonomía de las fuentes de datos.* Un agente de DM puede considerarse una extensión modular a un sistema de gestión de datos que manipula el acceso a las fuentes de datos subyacentes de acuerdo con las restricciones de autonomía del sistema, de los datos y el modelo.

*DDM Interactiva.* Agentes de asistencia pueden limitar drásticamente la necesidad de intervención del usuario en el proceso de minería de datos. Por ejemplo, agentes de DM pueden anticipar los límites de espacios de búsqueda especialmente grandes y adecuar los resultados intermedios de acuerdo a las preferencias del usuario.

*Selección dinámica de fuentes de datos y recolección de datos.* Uno de los retos más importantes en ambientes de datos distribuidos, donde la disponibilidad de las localizaciones y su contenido puede variar, es seleccionar las fuentes de datos más valiosas. Bajo las condiciones descritas, agentes de DM pueden ser aplicados para seleccionar de forma adaptativa las fuentes de datos según criterios dados, como la cantidad, tipo y calidad de los datos, así como la carga de tráfico en la red y la carga del servidor.

*Escalabilidad de DM a fuentes de datos masivas distribuidas.* Una opción para reducir el tráfico en la red y la carga de procesamiento en el servidor es permitir a los agentes de DM migrar a las localizaciones existentes en un sistema DDM donde se desee realizar tareas de minería localmente. Estos agentes pueden regresar al servidor o enviar datos pre-seleccionados para su procesamiento.

*DDM multi-estrategia.* Para algunas aplicaciones complejas puede resultar más efectivo aplicar una combinación de técnicas de minería de datos [49], que aplicar solo una en particular. Agentes de DM pueden aprender, durante el curso de sus acciones, qué técnicas deben aplicar para datos de una localización en específico dada una tarea de minería de datos.

*DM colaborativa.* Los agentes de DM pueden operar independientemente en los datos que han recolectado en sitios locales y combinar sus respectivos modelos. Otra forma de enfocar el proceso es hacer que los agentes compartan el conocimiento a medida que este es descubierto para beneficiarse de la opinión de otros agentes. Las técnicas de minería de datos pueden ser aplicadas para realizar minería de datos homogéneos en entornos distribuidos.

*Seguridad y confiabilidad.* Este es un argumento contra el uso de agentes para DDM. Todo sistema basado en agentes para sistemas de DDM tiene que lidiar con el problema de garantizar la seguridad y privacidad de los datos. Sin embargo, cualquier fallo en la implementación del acceso a los datos para un agente puede brindarle a éste, acceso no previsto a datos sensibles. Aun más, cualquier operación que pueda realizar un agente de DDM en un sistema con

una arquitectura segura puede redundar por ejemplo en ataques de denegación de servicio. Un problema adicional con los agentes móviles es que un huésped malicioso puede analizar la información portada por los agentes móviles. Este es conocido como el problema de la inferencia y ocurre con más frecuencia en configuraciones donde los agentes pueden cruzar las fronteras de zonas confiables.

## 1.4. Lógica Difusa Compensatoria

La lógica difusa compensatoria (LDC) es una lógica multivalente en la que los operadores no cumplen las reglas de asociatividad y se han definido de forma tal, que faciliten la compensación de los valores de verdad de unos predicados básicos con otros [5].

### 1.4.1. Definiciones

Sea  $x = (x_1, x_2, \dots, x_n)$  un elemento cualquiera del producto cartesiano  $[0, 1]^n$ . Una cuarteta de operadores continuos  $(c, d, o, n)$ ,  $c, d : [0, 1]^n \rightarrow [0, 1]$ ,  $o : [0, 1]^2 \rightarrow [0, 1]$  y  $n : [0, 1] \rightarrow [0, 1]$  constituyen una Lógica Compensatoria, si se satisface el siguiente grupo de axiomas:

1. Axioma de compensación:

$$\text{mín}(x_1, x_2, \dots, x_n) \leq c(x_1, x_2, \dots, x_n) \leq \text{máx}(x_1, x_2, \dots, x_n)$$

2. Axioma de conmutatividad o simetría:

$$c(x_1, \dots, x_i, \dots, x_j, \dots, x_n) = c(x_1, \dots, x_j, \dots, x_i, \dots, x_n)$$

3. Axioma de crecimiento estricto: Si  $x_1 = y_1, x_2 = y_2, \dots, x_{i-1} = y_{i-1}, x_{i+1} = y_{i+1}, \dots, x_n = y_n$  son desiguales de cero, y  $x_i > y_i$  entonces  $c(x_1, x_2, \dots, x_n) > c(y_1, y_2, \dots, y_n)$

4. Axioma de veto: Si  $x_i = 0$  para algún  $i$  entonces  $c(x) = 0$

5. Axioma de reciprocidad difusa:

$$o(x, y) = n[o(y, x)]$$

6. Axioma de transitividad difusa: Si  $o(x, y) \geq 0,5$  y  $o(y, z) \geq 0,5$  entonces  $o(x, z) \geq \text{máx}(o(x, y), o(y, z))$

7. Leyes de Morgan:

$$n [c(x_1, x_2, \dots, x_n)] = d [n(x_1), n(x_2), \dots, n(x_n)]$$

$$n [d(x_1, x_2, \dots, x_n)] = c [n(x_1), n(x_2), \dots, n(x_n)]$$

Los operadores  $c$  y  $d$  reciben el nombre de conjunción y disyunción, respectivamente. El operador  $o$  recibe el nombre de orden estricto difuso, y  $n$  el de negación. El Axioma de Compensación, que da nombre a la estructura propuesta, es el más importante; la propiedad que refleja suele ser utilizada en la literatura de operadores difusos para definir el concepto de operador compensatorio [22]. Obsérvese que para el caso particular de dos componentes, el hecho de que el valor del operador se encuentre entre el mínimo y el máximo, puede interpretarse como que el segundo valor compensa el valor del primero en la veracidad de la conjunción. La idea se generaliza al caso de  $n$  componentes.

El Axioma de Conmutatividad o Simetría es deseable porque es natural que su resultado sea independiente del orden en que se tomen los predicados básicos.

La introducción del Axioma de Crecimiento Estricto dota al sistema de una sensibilidad que hace que cualquier variación en los valores de los predicados básicos modifique el valor de verdad del predicado compuesto, siempre que ninguno de los predicados básicos tenga valor cero. Como consecuencia de este axioma se tiene además la deseada propiedad de no asociatividad porque no existen operadores compensatorios asociativos, estrictamente crecientes [23].

El nombre de Axioma de Veto, alude a su interpretación en el marco de los problemas de decisión; esta propiedad otorga a cualquier predicado básico de una conjunción la capacidad de vetar, es decir la capacidad de impedir cualquier forma de compensación cuando su valor es igual a cero.

En la literatura se aborda el concepto de orden estricto difuso de diferentes maneras. [19, 16, 61]

Se pueden definir entonces los operadores como:

1. Operador de orden:

$$o(x, y) = 0,5 [C(x) - C(y)] + 0,5 \tag{1.9}$$

2. Operador de negación:

$$n(x) = 1 - x \quad (1.10)$$

3. Operador de conjunción:

$$c(x_1, x_2, \dots, x_n) = (x_1 \cdot x_2 \cdot \dots \cdot x_n)^{\frac{1}{n}} \quad (1.11)$$

4. Operador de disyunción:

$$d(x_1, x_2, \dots, x_n) = 1 - ((1 - x_1) \cdot (1 - x_2) \cdot \dots \cdot (1 - x_n))^{\frac{1}{n}} \quad (1.12)$$

La implicación puede ser definida como

$$i_1(x, y) = d(n(x), y) \quad (1.13)$$

o

$$i_2(x, y) = d(n(x), c(x, y)) \quad (1.14)$$

y la equivalencia se puede definir a partir de la implicación como

$$e(x, y) = c(i(x, y), i(y, x))$$

Los cuantificadores universal y existencial deben ser introducidos de manera natural a partir de los operadores conjunción y disyunción, por lo que introduciendo ya los operadores seleccionados, se tiene:

$$\forall_{x \in U} p(x) = \bigwedge_{x \in U} p(x) = \sqrt[|U|]{\prod_{x \in U} p(x)} = \begin{cases} e^{\frac{1}{|U|} \sum_{x \in U} \ln p(x)} & \forall x, p(x) \neq 0 \\ 0 & \text{en otro caso} \end{cases} \quad (1.15)$$

$$\exists_{x \in U} p(x) = \bigvee_{x \in U} p(x) = 1 - \sqrt[|U|]{\prod_{x \in U} (1 - p(x))} = \begin{cases} 1 - e^{\frac{1}{|U|} \sum_{x \in U} \ln(1 - p(x))} & \forall x, p(x) \neq 0 \\ 0 & \text{en otro caso} \end{cases} \quad (1.16)$$

## 1.4.2. Propiedades

### 1.4.2.1. Relación de la Lógica Difusa Compensatoria con la Lógica Booleana

En [5, 4] se demuestra que las fórmulas válidas del Cálculo Proposicional de la Lógica Compensatoria (CPLC) son exactamente las del Cálculo Proposicional Booleano (CPB) para cualesquiera de las dos selecciones del operador implicación, correspondientes a las fórmulas 1.13 y 1.14 respectivamente. Para ello se utilizaron los axiomas de Kleene [48].

Esta propiedad de la LDC distingue esta Lógica Multivalente; los sistemas multivalentes generalizan unas u otras propiedades de la lógica bivalente, pero ninguno antes ha demostrado una propiedad de carácter general como este teorema de compatibilidad.

### 1.4.2.2. Compatibilidad con la Teoría de la Medición

Desde la concepción básica de la Teoría de la Medición, si atribuimos a un predicado  $C$  la modelación de las preferencias del decisor, entonces:

$$x \geq_c y \Leftrightarrow C(x) \geq_c C(y)$$

En este sentido si  $p$  y  $q$  son predicados que miden las preferencias en relación con dos objetivos diferentes, un operador  $r$  es apropiado para modelar la *confluencia* de objetivos si se cumple la condición:

$$r(x \geq_p y, x \geq_q y) \Leftrightarrow r(p(x), q(x)) \geq (p(y), q(y)) \quad (1.17)$$

Condiciones necesarias y suficientes para la desigualdad de la tesis de la expresión 1.17 han sido estudiadas sólo para algunos operadores. En el marco de la selección de las conectivas de una Lógica Multivalente con intenciones normativas, los operadores de conjunción y disyunción deben satisfacer la condición 1.17. De esta forma dichos operadores pueden ser utilizados para ordenar alternativas según la conjunción o la disyunción de todos los objetivos.

En [4] se demuestra que la definición de los operadores utilizada en la LDC es compatible con la condición 1.17. Otras lógicas multivalentes privilegian diferentes propiedades como la asociativa y distributiva e incumplen propiedades como 1.17.

## 1.5. Conclusiones

El aprendizaje de sistemas de inferencia difusos a través de enfoques evolutivos es un área de activa investigación. Existe una amplia variedad de trabajos. En ellos el aprendizaje abarca los antecedentes de las reglas, los consecuentes, las funciones de pertenencia a usar y el peso de cada regla. Además de esto existen varios proyectos enfocados en la extracción de sistemas de reglas simples. Estos enfoques trabajan con las definiciones clásicas de la lógica de Zadeh y varían la forma de combinar los resultados de las reglas.

Un número elevado de enfoques se han desarrollado para apoyar visualmente la minería de datos. Algunos de los enfoques, además de permitir la identificación del conocimiento, están diseñados para crear modelos donde se use el conocimiento. Entre estos enfoques están varios cuyo propósito es construir clasificadores mediante la interacción visual del usuario con los datos. Los enfoques descritos muestran una precisión en la clasificación competitiva y además brindan todas las ventajas de la visualización a los ingenieros del conocimiento.

El desarrollo de las tecnologías digitales ha creado una demanda creciente de procesar fuentes de datos de gran volumen que se encuentran distribuidas. El uso de agentes inteligentes para la extracción de conocimiento en estos ambientes es un enfoque eficaz. La capacidad intrínseca de los agentes para adaptarse a nuevas condiciones, para comunicarse, cooperar o competir brindan las herramientas conceptuales para rediseñar o crear nuevos enfoques para la minería de datos distribuida. Hacia la seguridad de los datos, la autonomía de los agentes y la interacción mediante el meta-aprendizaje están encaminadas las investigaciones actuales.

## Capítulo 2

# Diseño de una arquitectura basada en agentes para la clasificación en entornos distribuidos conservando la interpretabilidad

En este capítulo se presenta el diseño de una arquitectura basada en agentes para la minería de datos distribuida, aplicada al problema de la clasificación. En este diseño agentes locales realizan minería sobre los conjuntos de datos locales construyendo clasificadores de validez en el sitio. Un agente mediador efectúa meta-clasificación identificando los clasificadores base e integrando sus decisiones para obtener un clasificador global. El funcionamiento de los diferentes clasificadores locales puede ser variado, el enfoque aquí presentado incluye sistemas de reglas difusas a partir de la LDC y construcción visual de clasificadores a partir de la LDC.

Las secciones de este capítulo muestran diferentes partes de la arquitectura propuesta. La sección 2.1 presenta una adaptación a la construcción evolutiva de sistemas de reglas difusas mediante el uso de antecedentes basados en predicados de la LDC. La sección 2.2 contiene la descripción de una técnica para la búsqueda visual de un sistema de inferencia difuso, la técnica está inspirada en la proyección de coordenadas paralelas. Una descripción de la arquitectura multi-agentes es presentada en la sección 2.3. En la nueva arquitectura creada un agente mediador es responsable de integrar los resultados de los diferentes clasificadores locales, en la sección 2.3.1 se realiza una descripción del enfoque de meta-clasificación que

se eligió para la arquitectura.

## 2.1. Adaptación a la construcción evolutiva de sistemas de reglas difusas mediante el uso de antecedentes basados en predicados de la LDC

En la sección 1.1 se presentaron los modelos de clasificación basados en sistemas de reglas difusas. En estos, la inferencia de la clase a partir de una regla de la forma 1.1 se realizaba utilizando una *T-Norma*. El producto y el mínimo son los enfoques elegidos tradicionalmente con este propósito. A su vez, en el proceso de construcción de reglas difusas a partir de un conjunto de entrenamiento se utiliza el grado de compatibilidad entre un antecedente y un patrón de entrenamiento. El cálculo del mismo se efectúa según la ecuación 1.5.

La selección de las reglas que conformarán el sistema de clasificación parte de dos criterios importados de la extracción de reglas de asociación. Estos criterios son la confianza y el soporte. En la sección 1.1.4.2 se presenta una definición de estas medidas para el caso difuso. Las definiciones usan el grado de compatibilidad  $\mu_{A_q}$ .

Puede redefinirse el grado de compatibilidad de los antecedentes con un patrón para usar la definición de conjunción de la lógica difusa compensatoria como en 1.11. Esto es:

$$\mu_{A_q} = \sqrt[n]{\mu_{A_{q1}}(x_1) \times \mu_{A_{q2}}(x_2) \times \dots \times \mu_{A_{qn}}(x_n)} \quad (2.1)$$

Con esta nueva definición se introduce la propiedad de compensación que caracteriza la LDC en los sistemas de inferencia. Aunque los antecedentes de las reglas descritas sólo están compuestas por la conjunción de predicados simples debe tenerse en cuenta que es precisamente en ese operador donde se define la propiedad más importante.

Esta modificación hace uso de la LDC de manera directa en la evaluación del antecedente de cada regla. Un uso adicional de la lógica difusa compensatoria puede estar en el cálculo de las certidumbres o pesos de las reglas.

En [40] se presentan y analizan 4 definiciones heurísticas para el peso de las reglas. En esa publicación se estima mediante experimentación que la heurística que más cerca aproxima las fronteras entre clases es 2.4.

$$CF_q^I = c(A_q \Rightarrow C_q) \quad (2.2)$$

$$CF_q^{II} = c(A_q \Rightarrow C_q) - \frac{1}{M-1} \left( \sum_{h=1}^M c(A_q \Rightarrow Class\ h) - c(A_q \Rightarrow C_q) \right) \quad (2.3)$$

$$CF_q^{III} = c(A_q \Rightarrow C_q) - \text{máx} \{c(A_q \Rightarrow Class\ h) \mid h = 1, 2, \dots, M; h \neq C_q\} \quad (2.4)$$

$$CF_q^{IV} = c(A_q \Rightarrow C_q) - \left( \sum_{h=1}^M c(A_q \Rightarrow Class\ h) - c(A_q \Rightarrow C_q) \right) \quad (2.5)$$

Las reglas difusas que se han presentado pueden considerarse predicados de la forma  $R_q : \mu_{A_q}(x_p) \Rightarrow Class_q(y_p)$ . La función  $\mu_{A_q}$  está compuesta por la conjunción de los antecedentes difusos y  $Class_q(y_p)$  evalúa si la clase  $y_p$  del patrón de entrenamiento  $p$  es  $C_q$ . Puesto que se trata de un problema de clasificación se tiene:

$$Class_q(y_p) = \begin{cases} 1 & \text{si } y_p = C_q \\ 0 & \text{si } y_p \neq C_q \end{cases}$$

Puede observarse que en las cuatro definiciones anteriores para el cálculo del factor de certeza se utiliza la confianza de las reglas. La confianza es usada también para obtener el consecuente de las reglas. En las definiciones anteriores, sin embargo, se está considerando que todas las reglas tienen una semántica de co-ocurrencia de un antecedente y un consecuente. Esto no tiene que ser así para todas las reglas. Por otro lado, incluso para esta semántica de las reglas, la confianza como se presentó en 1.3 ha sido simplificada teniendo en cuenta la definición brusca de  $Class_q(y_p)$  y la conjunción de antecedentes según el producto, que constituye una T-norma. La expresión general es:

$$c(A_q \Rightarrow Class_q) = \frac{\sum_{p=1}^m T(\mu_{A_q}(x_p), Class_q(y_p))}{\sum_{p=1}^m \mu_{A_q}(x_p)} \quad (2.6)$$

El operador de conjunción en la LDC no es una T-Norma, por lo que la definición brusca de  $Class_q(y_p)$  no afecta en igual medida la formulación de la confianza. La ecuación 2.7 se ve en cambio afectada por la no asociatividad del operador de conjunción en el caso de la LDC.

Otras interpretaciones semánticas son posibles producto de esto, planteando la confianza como:

$$c(A_q \Rightarrow Class_q) = \frac{\sum_{p=1}^m AndLDC(\mu_{A_q}(x_p), Class_q(y_p))}{\sum_{p=1}^m \mu_{A_q}(x_p)} \quad (2.7)$$

o

$$c(A_q \Rightarrow Class_q) = \frac{\sum_{p=1}^m AndLDC(\mu_{A_q}(x_p), ImplLDC(\mu_{A_q}(x_p), Class_q(y_p)))}{\sum_{p=1}^m \mu_{A_q}(x_p)} \quad (2.8)$$

Las ecuaciones 2.7 y 2.8 que en apariencia pueden redundar en un beneficio, no son útiles producto precisamente de la propiedad de compensación. En la ecuación 2.7 aparece en el numerador la sumatoria del nivel de co-ocurrencia de un antecedente y un consecuente. Para todos los valores de  $p$  en que  $Class_q(y_p) = 1$  se cumplirá que  $numerador_p \geq denominador_p$ , con lo que se obtendrá posiblemente un valor de confianza mayor que uno. Igual problemática surge en la ecuación 2.8. Esto descarta el uso del caso generalizado dado en 2.6 y debe volverse a la definición 1.3 donde no se utiliza la compensación en el numerador. Aparentemente el uso de los operadores de la LDC, dadas sus propiedades, no ofrecen un camino para su aplicación en el cálculo riguroso de la confianza que influya en la obtención del grado de certeza.

## 2.2. Propuesta para la construcción visual de un sistema de reglas difusas

La nueva técnica propuesta está influida por los enfoques analizados en la sección 1.2 y por el trabajo presentado en [10]. Al igual que en las variantes mostradas, esta técnica permite la construcción de un modelo de clasificación mediante la interacción del usuario con los datos y la representación en línea del efecto de sus decisiones. A diferencia de los enfoques anteriores, donde para todos los casos se construía un árbol de decisión, en esta propuesta el modelo creado es un sistema de reglas difusas. La idea de crear un árbol de decisión difuso fue estudiada basándose en varios trabajos alegóricos [43] pero se eligió diseñar y evaluar un enfoque para construir sistemas de reglas como los presentados en la sección 1.1.

La propuesta usa como base la técnica de coordenadas paralelas y la representación gráfica de funciones de membresía. Los usuarios pueden construir iterativamente el sistema de reglas y observar el grupo de instancia de entrenamiento erróneamente clasificadas. La información mostrada tradicionalmente por la técnica de coordenadas paralelas así como datos adicionales (soporte, confianza, etc) les permite a los usuarios retroceder en sus decisiones y construir antecedentes con un grado alto de compatibilidad y reglas con elevados niveles de certidumbre. La incorporación de filtros para la transformación de los datos en tiempo real, aporta valiosas herramientas adicionales para apoyar el análisis.

El uso de la técnica tiene como desventaja que requiere la comprensión del funcionamiento de los sistemas de reglas difusas. Esto avala el uso de sistemas de reglas más simples como el presentado, donde la clasificación es a través de una única regla ganadora y no mediante la agregación de resultados de varias reglas. No obstante este inconveniente, la propuesta se beneficia de la expresividad aportada por las coordenadas paralelas y la representación de funciones de membresía.

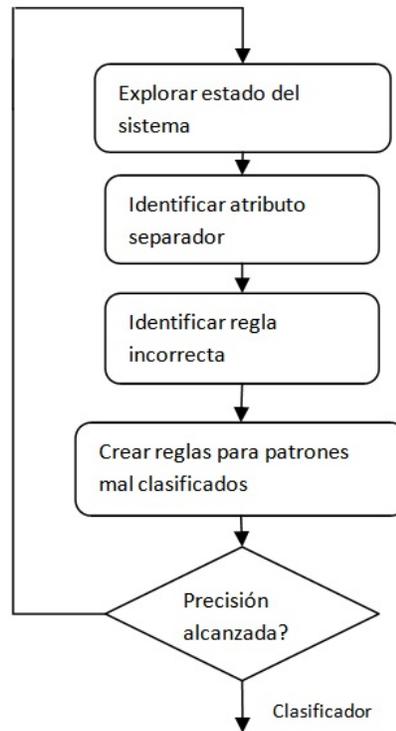


Figura 2.1: Proceso de construcción visual del clasificador

En la figura 2.1 se muestra un diagrama con el flujo de trabajo para construir el clasificador.

Este es un proceso iterativo no algorítmico inspirado en los estudios de Inselberg [36], por tanto constituye solo una guía. Los pasos son:

1. Explorar el estado de clasificación actual del conjunto de datos e identificar qué reglas clasifican determinados patrones apoyándose en la visualización y en el filtrado de los datos mediante consultas.
2. Identificar el atributo, en las instancias mal clasificadas, que separe una clase. Determinar si la granularidad en cada atributo está definida apropiadamente.
3. Identificar reglas que clasifiquen instancias incorrectamente. Refinarlas, agregando/eliminando antecedentes, o inhabilitarlas.
4. Crear reglas para cubrir patrones mal clasificados. Esas reglas pueden comenzar a clasificar erróneamente patrones que con anterioridad eran bien clasificados por lo que se debe ser cuidadoso.
5. Comprobar el estado actual de la clasificación, si se alcanza una precisión aceptable terminar, de lo contrario realizar nuevas iteraciones.

### **2.2.1. Descripción de la técnica.**

La técnica propuesta utiliza una proyección de coordenadas paralelas como base. En cada coordenada se representan las funciones de membresía de la variable lingüística. Un sistema de reglas difusas, inicialmente vacío y en constante edición, es analizado. A cada ejemplo de entrenamiento se le asigna un color diferente, pero sólo son mostrados los patrones erróneamente clasificados por el sistema de inferencia.

La lista de reglas creadas es mostrada en la visualización. Las reglas individuales pueden habilitarse o inhabilitarse. Una de las reglas puede marcarse como activa, lo cual permite editarla. Como en la sección 1.1.4.2 en una regla sólo pueden editarse los antecedentes, agregando o eliminando algunos; el consecuente y el grado de certidumbre son calculados de manera automática. Adicionalmente, se presentan criterios de medida de las reglas como la confianza, el soporte y su producto.

La granularidad de los conjuntos difusos a utilizar en cada variable y el tipo de función de membresía puede editarse en cualquier momento. Los cambios en estos parámetros son reflejados de inmediato aunque algunos pueden inhabilitar reglas. Otras transformaciones

pueden realizarse sobre el conjunto de datos original, como la normalización o estandarización de algunas variables.

En la figura 2.2 se presentan algunas vistas de la técnica propuesta. En la figura 2.2 se muestra la aplicación de la técnica al conjunto de datos *Iris*, con dos reglas creadas, observe el por ciento de clasificación del sistema creado hasta el momento en la esquina inferior derecha. Pueden observarse las reglas editadas.



Figura 2.2: Técnica para la construcción visual de sistema de reglas difusas.

Los sistemas de reglas difusas con un número pequeño de reglas y antecedentes han probado ser efectivos para la clasificación al aplicarlos a diversos problemas [40, 39]. Los altos niveles de interpretabilidad del conocimiento que brinda frente a otras alternativas es una de las características que ha fomentado la aplicación de las reglas difusas al aprendizaje automático. En la propuesta para la extracción de reglas que se realiza en esta investigación se tiene en cuenta esa propiedad. En una técnica de extracción interactiva de reglas no debe esperarse la construcción de sistemas de reglas complejos, por el contrario la técnica debe proveer al usuario de herramientas e informaciones que posibiliten la construcción rápida del sistema de inferencia. Los sistemas de aprendizaje automático ofrecen respuesta rápida en las tareas de minería de datos, por tanto un enfoque visual de extracción debe tratar de emular esta

ventaja.

## 2.3. Arquitectura multiagentes para el uso de reglas difusas en ambientes distribuidos

La arquitectura de agentes propuesta, figura 2.3, está enfocada a la solución del problema de clasificación. En la arquitectura existe un agente mediador en la organización, agentes de minería de datos en cada localización, agentes de datos en cada sitio y agentes de usuario en diferentes localizaciones.

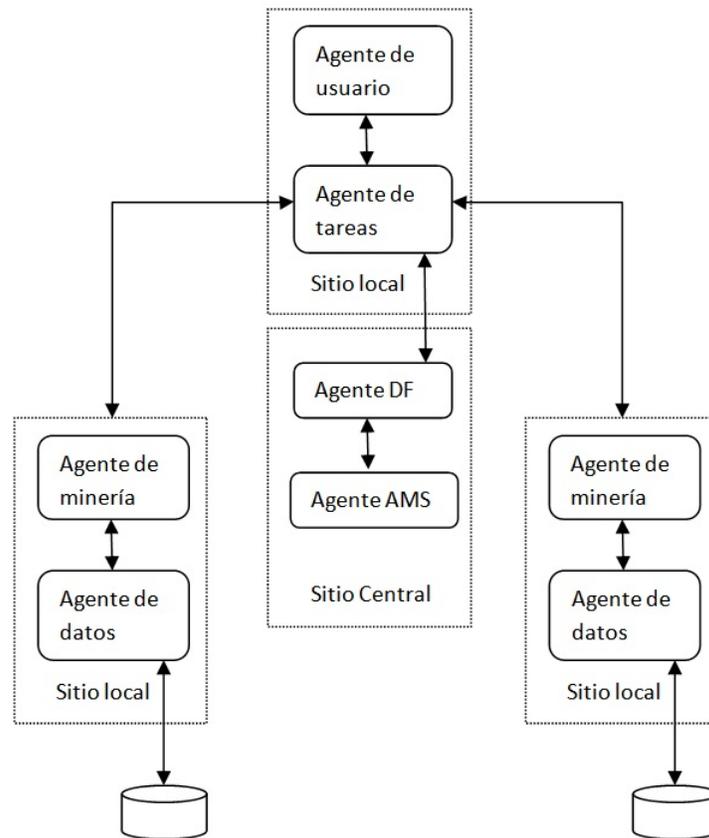


Figura 2.3: Arquitectura multiagentes para la clasificación.

Los agentes para la minería de datos están diseñados para llevar a cabo su comportamiento sobre el conjunto de datos locales. En principio, cada agente puede construir resultados

locales mediante métodos de DM diferentes. En el diseño se proponen dos estrategias: la construcción evolutiva de sistemas de reglas difusas para la clasificación y la construcción visual de un clasificador basado en reglas difusas.

Los agentes de obtención de datos están localizados en cada sitio de la organización. Su responsabilidad es presentar un acceso uniforme a los datos y controlar las opciones de acceso manteniendo las restricciones de seguridad y privacidad. Su uso permite la autonomía de las fuentes de datos.

Los agentes de usuarios son responsables de tomar las entradas o nuevas instancias a clasificar y de presentar los resultados a los usuarios. Un tipo especial de agente de usuario brinda la posibilidad de crear, a través de la interacción visual con los datos, un modelo de clasificación.

El agente mediador o planificador de la tarea tiene la responsabilidad de distribuir los patrones a clasificar a través de los diferentes sitios que han construido modelos locales. Debe además construir una respuesta global a partir de los resultados parciales de los sitios. El agente mediador puede seguir varias estrategias para obtener el resultado final, en este diseño utiliza enfoques de meta-clasificación. El esquema básico para clasificación de una nueva instancia es mostrado en la figura 2.4. Cada agente local aporta su criterio, el agente mediador es responsable de integrar las predicciones.

El comportamiento del sistema de meta-clasificación es dinámico puesto que el número de clasificadores locales disponibles en el tiempo puede variar. Los agentes de minería activos son controlados mediante el sitio central de la organización. Cada fuente de datos al activarse debe registrarse en el sitio central. Los agentes mediadores comprueban al inicio de la clasificación los sitios que están disponibles y los utilizan como base para la meta-clasificación.

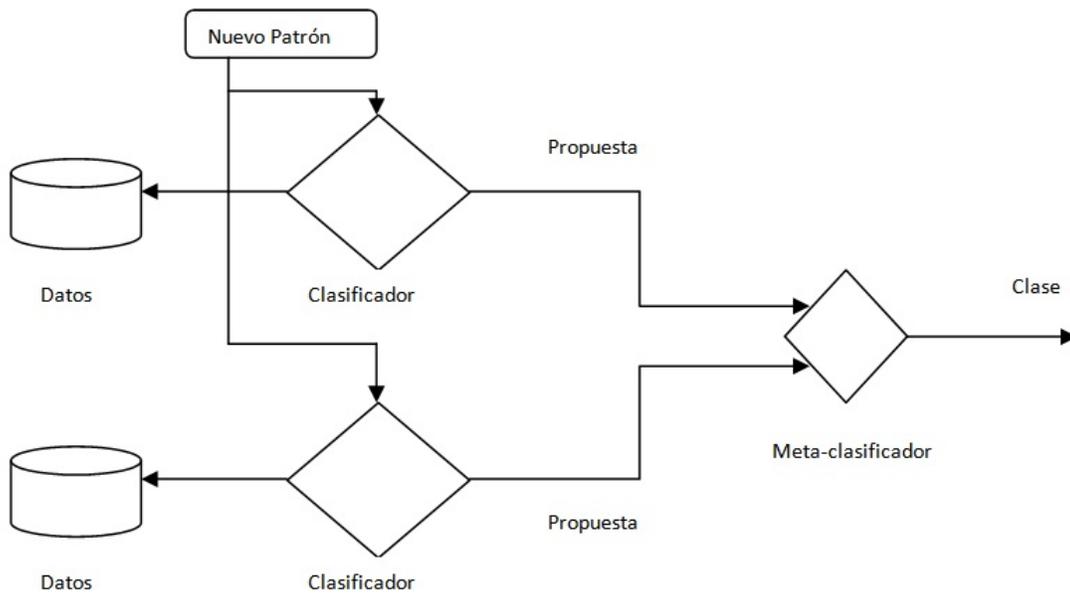


Figura 2.4: Esquema de clasificación de una nueva instancia.

El sistema ha sido implementado utilizando las herramientas JADE para la creación de sistemas multi-agentes. En el contenedor principal de JADE residen dos agentes, AMS(Sistema para el manejo de Agentes) y DF (facilitador de directorio) que se encargan del servicio de nombres y control de agentes y del directorio de páginas amarillas respectivamente [9]. El agente DF provee la interfaz necesaria para el comportamiento dinámico del sistema. Todas las rutinas de comunicación y registro son manejadas por estos agentes de JADE.

### 2.3.1. Agentes para la integración de resultados

Las técnicas de meta-aprendizaje permiten elegir el mejor resultado entre varios modelos aprendidos individualmente. Varios enfoques se han desarrollado para la construcción de meta-clasificadores [68, 8, 26, 30]. En el proceso general de meta-aprendizaje se realiza una caracterización de los conjuntos de datos y luego se crean un grupo de modelos de aprendizaje base. Los modelos construidos son asociados con las características de los datos. En el momento de utilizar el conocimiento adquirido se analizan las características de los datos y se utiliza el modelo base correspondiente [49]. En la sección 1.3 se presentó la herramienta JAM. El enfoque de JAM permite obtener clasificadores globales a partir de clasificadores

locales. Otras herramientas han sido desarrolladas con este propósito .

En [64] se presenta una breve descripción de diferentes tendencias para el meta-aprendizaje aplicado a la clasificación. Entre las técnicas discutidas están las de caracterización de los conjuntos de datos, la asociación de conjuntos de datos con modelos predictivos, la combinación de modelos base y otros.

El enfoque usado por JAM para enfrentar la tarea de clasificación está basado en estrategias de arbitraje y combinación. En [50] se presenta un enfoque basado en el voto, con tres estrategias diferentes, para la integración de resultados. Las estrategias utilizadas son votación mayoritaria, votación con pesos mayoritaria y votación selectiva. Estas tendencias tienen como ventajas y características que permiten utilizar bases de datos distribuidas, el conocimiento colectivo es recolectado de fuentes de conocimiento individuales, permite integrar el comportamiento de varios clasificadores y el conocimiento es compartido sin compartir los datos.

En este trabajo utilizamos un enfoque basado en el voto. Sea  $p$  un nuevo patrón a clasificar y  $Classifiers = \{cla_1, \dots, cla_n\}$  un conjunto de clasificadores entrenados sobre diferentes bases de datos homogéneas. El patrón  $p$  es pasado a cada elemento de  $Classifiers$  con lo que se obtiene un conjunto

$$Result = \{ \langle Clase_1, Peso_1 \rangle, \dots, \langle Clase_n, Peso_n \rangle \}$$

donde  $Clase_i \in C$  y  $Peso_i \in [0, 1]$ . Por conveniencia definamos los conjuntos  $Result_{v_i}$  como los pesos para los clasificadores que predijeron la clase  $v_i$ . La clase asignada al patrón  $p$  puede definirse de varias formas:

- La media de los pesos por clase

$$clase = arg \max_{v_i \in C} \left\{ \frac{\sum_{peso \in Result_{v_i}} peso}{|Result_{v_i}|} \right\} \quad (2.9)$$

- El peso máximo

$$clase = arg \max_{v_i \in C} \{ \max \{ Result_{v_i} \} \} \quad (2.10)$$

- La suma de los pesos por clase

$$clase = arg \max_{v_i \in C} \left\{ \sum_{peso \in Result_{v_i}} peso \right\} \quad (2.11)$$

La definición 2.10 concede especial importancia a criterios individuales. Este criterio de selección de clase elegirá con mayor probabilidad clases asociadas a ejemplos de entrenamiento muy similares al patrón que se clasifica. Si las bases de datos no se encuentran uniformemente distribuidas diferentes clasificadores se activarán con un gran peso para diferentes regiones del espacio de rasgos predictores. En el caso de clasificadores basados en reglas difusas como los presentados en la sección 1.1 las reglas generadas serán muy similares para dos clasificadores si sus conjuntos de entrenamiento presentan una distribución similar. En caso de que los conjuntos de entrenamientos estén en regiones diferentes del espacio, los conjuntos de reglas diferirán sustancialmente y pueden verse los sistemas de reglas como un todo. En ambos casos el comportamiento de este criterio dará mayor importancia a los casos de entrenamiento más similares.

La definición 2.11 tiene la desventaja de ignorar criterios individuales de gran confianza en favor de criterios más votados. Esta medida de selección concede mayor peso a la confluencia de predicciones. En este sentido, la definición se asemeja al método de combinación de resultados por votos mayoritarios [49], sin embargo su comportamiento varía pues los clasificadores base no tienen un voto de influencia constante.

La definición 2.9 tiene un comportamiento promedio entre las dos anteriores. Las predicciones individuales pueden influir notablemente si su desviación es lo suficientemente grande.

El modo de integración mediante voto aquí propuesto requiere que cada clasificador presente, junto a la clase que asigna al patrón, un valor que indica la certidumbre con que realiza la asignación. En el caso de los clasificadores base que se utilizan en esta arquitectura la elección de este valor es directa, basta tomar el valor de la ecuación 1.2 para cada uno de los FRS locales.

La propiedad de interpretabilidad es fácil de preservar siguiendo este esquema. Los clasificadores base pueden enviar a manera de explicación, además de su voto, la regla que clasifica. El único problema es que las medidas de selección 2.9 y 2.11 utilizan en su formulación más de una regla, la solución es mostrar todas las reglas como explicación de la decisión ordenadas según el valor de certeza. En el caso de la definición 2.10 se elige la regla ganadora.

### 2.3.2. Agentes para la búsqueda visual de reglas difusas

En cada localización puede activarse un agente para la búsqueda visual de sistemas de reglas difusas. Este tipo de agente es tanto un agente de usuario como de minería local. Tiene la responsabilidad de presentar al usuario las vistas para la generación mediante interacción de un FRS, como se describe en 2.2.1. Adicionalmente, el agente debe identificarse en el agente DF como un clasificador y proveer una solución de clasificación para nuevos patrones bajo solicitud.

El funcionamiento de este esquema y de la arquitectura obliga a que en cada localización un ingeniero del conocimiento o experto realice el proceso de construcción visual. Esto, no debe verse como una limitante, sino como una oportunidad de extraer información.

## 2.4. Conclusiones

Cuatro clases de agentes son utilizados para definir una arquitectura para la clasificación en entornos distribuidos. Los agentes de manipulación de datos son responsables de restringir el acceso siguiendo las políticas de seguridad. Los agentes de minería local construyen modelos de clasificación a través de reglas difusas utilizando los datos de un sitio y sin intercambio de información. El agente mediador, bajo solicitud, selecciona los decisores disponibles y distribuye las instancias a clasificar, el rol fundamental del mismo es sin embargo, combinar las propuestas y elegir las reglas que serán presentadas al usuario como explicación. Los agentes de usuario tienen la responsabilidad de recoger las solicitudes y de mostrar las respuestas a los usuarios. Agentes adicionales están presentes en la arquitectura, éstos garantizan la comunicación e identificación de los diferentes agentes en la organización y permiten la incorporación dinámica de nuevos clasificadores base.

Entre las ventajas del enfoque está que minimiza el tráfico de datos y que no necesita compartirlos para producir resultados globales. El cómputo de los modelos se ha distribuido al efectuarse sobre fuentes de datos no centralizadas y por lo mismo más pequeñas. Pueden elegirse alternativas diversas para la creación de los clasificadores base, aunque existen restricciones no planteadas en los modelos de meta-clasificación basados en el voto mayoritario y voto mayoritario pesado.

El clasificador local propuesto se basa en reglas difusas donde la clase inferida es el consecuente de una única regla ganadora. Estos sistemas pueden inducirse a partir de datos

siguiendo diversos procedimientos. Se ha propuesto una modificación a uno de ellos para adaptarlos al uso de la LDC. Esta modificación puede redundar en una reducción del error en la clasificación.

Un método para la construcción visual de FRS se ha propuesto como agente de minería local. El método, basado en la técnica de coordenadas paralelas, apoya una nueva perspectiva de la minería visual. Aunque la técnica no es automática, es una alternativa viable para la extracción, comprensión y uso del conocimiento contenido en los datos.

# Capítulo 3

## Resultados Experimentales

En este capítulo se realiza una validación estadística de las diferentes partes de la arquitectura propuesta. Se validan tres aspectos de la arquitectura: la propuesta para la construcción visual de clasificadores, la utilización de la LDC en lugar de la lógica difusa clásica para calcular el grado de membresía y los métodos de combinación de resultados. Las pruebas son realizadas a partir de los estadísticos obtenidos en experimentos sobre varios conjuntos de datos cuyo número de instancias, de atributos y de clases varía. Los tres grupos de experimentos siguen una metodología general [21] particularizándose en cada caso.

### 3.1. Diseño de los experimentos

Tres grupos de experimentos se efectúan en este capítulo. Cada grupo tiene diferente propósito. El primer grupo de experimentos pretende evaluar el comportamiento de un clasificador construido visualmente (*Grupo I*). El segundo grupo es utilizado para medir el efecto de utilizar la LDC en los antecedentes de un sistema de reglas difusos (*Grupo II*). La validez de los métodos de combinación de resultados es medida a partir del tercer grupo de experimentos (*Grupo III*).

Aunque cada grupo de experimentos tiene diferentes configuraciones, en todos los casos, los individuos de la muestra se corresponden con conjuntos de entrenamiento obtenidos de diversas fuentes y donde las variables para efectuar las comparaciones se obtienen mediante la ejecución de las diferentes propuestas y de métodos conocidos de clasificación de diversos conjuntos de datos. Los métodos seleccionados dependen del propósito de cada grupo de experimentos.

Las variables de los experimentos son cuantitativas. Se utilizan los estadísticos descriptivos producidos por los métodos de clasificación. Se utilizan también otros estadísticos específicos de los FRS, que permiten medir la comprensibilidad, como la cantidad de reglas y la longitud media de las reglas. Las variables para cada individuo de la muestra se obtienen al calcular la media de aplicar 10 repeticiones de validación cruzada con 10 particiones al correspondiente conjunto de datos con los métodos de control y con las propuestas.

Dado que el número de conjuntos de datos es bajo y no puede probarse la normalidad de las muestras, las pruebas realizadas no serán paramétricas. En estos casos, para la comparación de dos algoritmos se utiliza [21] la prueba no paramétrica de signos de Wilcoxon [66]. Si la comparación es entre varios algoritmos puede usarse la prueba de Friedman [27] o de Iman-Davenport. Al detectar diferencias entre los algoritmos deben emplearse las pruebas pareadas de Bonferroni-Dunn y Holm para identificar donde se encuentran las diferencias.

Los algoritmos que se analizan fueron implementados en el ambiente de aprendizaje automático Weka para el caso de los grupos de experimentos *II* y *III*. En el caso del *grupo I* fue elaborada una herramienta específica para la obtención de los estadísticos.

## 3.2. Conjuntos de datos a usar en los experimentos

En los experimentos se utilizan 10 conjuntos de datos de diversas fuentes. En los conjuntos de datos todos los atributos predictores son numéricos. El número de clases, de instancias de entrenamiento y de rasgos predictores varía. Las variables predictoras han sido normalizadas para todos los experimentos. En la tabla 3.1 se muestran las características de todos los datos de prueba.

Dos de los conjuntos de datos son *breast cancer W* y *YinYang*. El primero ha sido modificado para eliminar las instancias con valores perdidos. El segundo es un conjunto de datos artificial, creado a partir de una imagen, del que se han seleccionado un grupo de ejemplos aleatoriamente. Los datos originales así como la simplificación se muestran en el anexo 3.6.

El conjunto de datos *credit* fue procesado antes de su inclusión en los experimentos. Este contiene rasgos continuos y nominales además de valores perdidos. En primer lugar, todos los ejemplos con valores perdidos en algún atributo fueron eliminados. Los valores para los atributos nominales fueron llevados a una escala entre  $[0, 1]$  y tratados como variables reales. Finalmente se normalizaron las variables continuas.

Conjunto de datos	Atributos	Instancias	Clases
diabetes	8	768	2
breast cancer W	9	683*	2
iris	4	150	3
wine	13	178	3
glass*	9	214	6
sonar	60	208	2
ecoli	7	336	8
YinYang	2	6550*	3
credit	15	653*	2
wave	40	5000	3

Cuadro 3.1: Conjuntos de datos usados para la simulación.

### 3.3. Resultados experimentales de la técnica para la extracción visual de reglas difusas.

En esta sección se valida la técnica de extracción visual de reglas difusas. Para ello se comparan los resultados de precisión obtenidos al usar los clasificadores así contruidos con otros modelos de clasificación. La evaluación de la precisión se realiza mediante una comparación con otros clasificadores como modelos de aprendizaje perezoso y árboles de decisión. El rendimiento del nuevo enfoque y de otros clasificadores se muestra en la tabla 3.2.

Los criterios de comprensibilidad de los sistemas contruidos son evaluados comparándolos con resultados de técnicas automáticas [42, 15, 55, 41]. Las tablas 3.4 y 3.5 recogen estas medidas para varios clasificadores al ser aplicados a algunas bases de datos, las utilizadas en dichas publicaciones.

En las pruebas se utilizan como antecedentes de las reglas, para todas las variables lingüísticas, los conjuntos difusos cuya funciones de membresía se muestran en la figura 1.1. El grado de compatibilidad es obtenido usando la LDC. El cálculo del peso de cada regla se realiza, por ser una de las más efectivas, según la heurística dada en 2.4.

El proceso de validación cruzada en un clasificador de construcción visual debe limitar la

cantidad de particiones realizadas al conjunto de entrenamiento. Esto es producto de la naturaleza interactiva de la minería visual. Por este motivo se utilizan solo 5 particiones en los experimentos.

Los criterios de medición que se tienen en cuenta en los experimentos son: la precisión en los datos de prueba, la cantidad de reglas de los sistemas construidos y la media de las longitudes de las reglas que componen cada sistema de inferencia. La comparación se realiza separando estos criterios en dos grupos. En un grupo están los relacionados con la precisión del clasificador. En el otro grupo están los que atañen a las propiedades de comprensibilidad de un sistema de reglas difusas, cantidad de reglas y longitud de las mismas.

La prueba de Iman-Davenport arrojó un valor de  $p = 0,08914$  al aplicarse sobre las muestras en la tabla 3.2. Para un nivel de confianza de  $\alpha = 0,05$  no existen diferencias en el comportamiento de estos clasificadores al ser evaluados como un grupo.

La prueba de signos de Wilcoxon fue usada para comparar el clasificador visual con el resto de los modelos individualmente. Los resultados obtenidos para la prueba en cada caso se muestran en la tabla 3.3.

Conjunto de datos	C4.5	Este trabajo	MLP	KNN
diabetes	71.22	70.31	75.13	73.30
breast cancer W	94.91	96.92	95.29	95.54
iris	96.00	96.00	96.00	94.67
wine	90.44	96.06	97.75	96.07
glass	66.82	48	65.42	70.09
sonar	71.15	78.84	78.84	87.01
ecoli	84.22	81.25	84.82	85.41
wave	75.25	76	83.82	77.48
credit	85.92	85.60	86.92	84.87
YinYang	96.80	85.90	98.03	68.81

Cuadro 3.2: Precisión en los datos de prueba mediante validación cruzada, 5 particiones para la técnica propuesta.

El clasificador visual se comporta de forma similar a los métodos C4.5 y K-NN. En el caso de la comparación con el MLP, usando un nivel de confianza  $\alpha = 0,05$ , se rechaza  $h_0$  por lo

que el comportamiento de los clasificadores no es similar.

	Este Trabajo vs MLP	Este trabajo vs KNN	Este Trabajo vs C4.5
p	0.0059	0.1514	0.2402

Cuadro 3.3: Resultados de la prueba de Wilcoxon

La interpretabilidad de los sistemas de reglas construidos es comparada con la obtenida por otros métodos de generación automática de FRS publicados. La literatura revisada no recoge las medidas de comprensibilidad para todos los conjuntos de datos empleados en este estudio. Las tablas 3.4 y 3.5 tienen un propósito descriptivo.

Obsérvese que la cantidad de reglas ha sido reportada para *iris* y *wine* en todas los trabajos citados, mientras que los resultados sobre *diabetes* y *breast cancer W* están parcialmente disponibles. El resultado obtenido en este trabajo es similar en el caso de *iris* y peor en el caso de *wine*. El resto de la tabla está muy poco poblada, no puede realizarse un análisis.

	diabetes	iris	wine	breast cancer W	sonar	ecoli	wave
Este trabajo	6	4	8	10	23	14	19
Ishibuchi [41]	-	3	3	-	-	-	-
Ishibuchi [38]	-	5	3	-	-	-	-
Wang [65]	-	3	3	2	-	-	-
Abonyi [1]	11.2	3	3.6	-	-	-	-
Chang [15]	3	5	5	3	-	-	-

Cuadro 3.4: Cantidad de reglas.

La cantidad de términos es una forma de medir la complejidad de un sistema de reglas difusas. Aunque no es posible realizar un análisis estadístico sólido, producto de la baja disponibilidad de datos publicados, puede observarse que la técnica visual presentada en este trabajo no se comporta como la peor variante en ningún conjunto de datos.

	diabetes	iris	wine	breast cancer W	sonar	ecoli	wave
Este trabajo	12	5	11	11	37	15	26
Ishibuchi [41]	-	-	-	-	-	-	-
Ishibuchi [38]	-	7	9	-	-	-	-
Wang [65]	-	11	27	18	-	-	-
Abonyi [1]	40	4	8.8		-	-	-
Chang [15]	4	7	13	4	-	-	-

Cuadro 3.5: Cantidad de términos lingüísticos empleados en las reglas.

Unificando ambos resultados puede notarse que las reglas obtenidas visualmente tienen una longitud promedio de uno o dos términos, esto contrasta con el resto de los resultados. Esta conclusión es consistente con el hecho bien conocido de que la técnica de coordenadas paralelas es útil para encontrar correlaciones entre unas pocas variables y que la capacidad para expresarlas dependen del orden de las mismas. Es de esperarse que los FRS construidos estén compuestas por un número grande de reglas cortas.

### 3.4. Resultados experimentales en agentes para el aprendizaje de reglas basados en LDC.

El grado de relación entre el uso de la LDC en los antecedentes y la precisión en la clasificación es evaluada en esta sección. Para ello se comparan los resultados de aplicar el método presentado en 2.1 al ser aplicado sobre los conjuntos de datos usando la LDC o no. El método fue implementado en el ambiente Weka siguiendo los pasos descritos en [41]. Sin embargo, sólo se implementaron los pasos fundamentales del algoritmo y no las mejoras sugeridas para el mismo, como búsqueda local o el aprendizaje de pesos. En el trabajo citado sólo se aplica el algoritmo sobre tres conjuntos de datos, con las mejoras mencionadas, por ello fue necesario desechar los resultados publicados y realizar los experimentos completos, deseándose además una comparación justa.

Los parámetros utilizados en los experimentos se muestran en la tabla 3.6. Además, se utilizaron los valores  $w_1 = 0,9$ ,  $w_2 = 0,05$  y  $w_3 = 0,05$  para la evaluación de la función objetivo 1.8. El cálculo del peso de la reglas se estimó según la ecuación 2.4.

Parámetro	Valor
$N_r$	900
Generaciones	500
Tamaño de la Población	50
Tamaño de la Élite	10
Probabilidad de Cruzamiento	0.9
Probabilidad de mutación parcializada $p(0 \Rightarrow 1)$	$1/N_r$
Probabilidad de mutación parcializada $p(1 \Rightarrow 0)$	0.1

Cuadro 3.6: Parámetros de los métodos usados en los experimentos.

Los valores usados fueron seleccionados de [41]. El único parámetro que difiere notablemente es el número de generaciones. En el trabajo citado fue empleado el valor de 10000 generaciones pero los conjuntos de datos usados en esa publicación son de menor volumen y complejidad. El valor sugerido por los autores resultó demasiado alto para el propósito de este trabajo. Se decidió seleccionar un valor menor, que redunde en menores tiempos de ejecución. Los efectos de esta elección son sobre el tamaño de los FRS generados, pero incide en menor medida en la precisión de los mismos. La comparación de las precisiones son el objetivo de este grupo de experimentos, no la evaluación de la interpretabilidad.

	Este trabajo	Ishibuchi [41]
Wine	93.48	93.16
breast cancer W	93.99	93.92
Iris	94.53	93.47
Sonar	74.62	71.80
diabetes	74.13	75.23
glass	58.11	56.16
ecoli	78.71	79.62
YinYang	86.71	86.92
Wave-1000	76.81	76.17
credit	86.17	86.54

Cuadro 3.7: Precisión en datos de prueba.

Con el propósito de determinar el efecto de usar la LDC, se aplicó la prueba de signos de Wilcoxon con un nivel de confianza  $\alpha = 0,05$  a los resultados de precisión alcanzados por los dos enfoques y que se muestran en la tabla 14, obteniéndose un valor  $p = 0,4922$ . No existen pues, razones para rechazar la hipótesis fundamental, en este caso que la media de la diferencia entre las dos muestras es igual a cero.

Los valores calculados para los estadísticos *F-Measure* y *Area\_under\_ROC* pueden estudiarse en el anexo 3.6. Al aplicar sobre estos datos la prueba de signos de Wilcoxon se obtuvieron valores  $p = 0,7188$  y  $p = 0,7188$  respectivamente, con lo que se muestra que no existen diferencias significativas.

El resultado de la prueba anterior no apoya la adaptación realizada como una mejora en el rendimiento de los FRS. Las diferencias encontradas son producto del azar. Esto no significa que el uso de la LDC no pueda incrementar el rendimiento de los métodos conocidos, sino que el enfoque aquí planteado no resulta provechoso.

### 3.5. Validación de los métodos de combinación.

Los métodos de combinación fueron evaluados a través de una comparación con los resultados obtenidos en la sección anterior sin utilizar la LDC. Las condiciones de distribución de los datos fueron simuladas desarrollando un meta-clasificador en el ambiente Weka. Este meta-clasificador utiliza como base  $K$  clasificadores como los descritos en la sección anterior con la T-Norma establecida como el producto para los antecedentes. El entrenamiento de este meta-clasificador se efectúa construyendo una partición con  $K$  subconjuntos del conjunto de entrenamiento y entrenando cada clasificador base en un subconjunto.

Este meta-clasificador es entrenado 10 veces con validación cruzada y 10 particiones. Con esto se logra simular el ambiente distribuido.

En los experimentos se ha usado  $k = 5$  y los tres métodos de combinación propuestos.

	Definición 2.11	Definición 2.9	Definición 2.10	Ishibuchi [41]
Wine	97.08	94.08	94.99	93.16
breast cancer W	95.11	93.59	-	93.92
Iris	95.07	95.20	95.07	93.47
Sonar	74.04	69.17	72.48	71.80
diabetes	76.03	74.85	75.04	75.23
glass	60.15	57.16	60.58	56.16
ecoli	83.25	79.95	81.80	79.62
YinYang	86.96	85.91	86.52	86.92
wave-1000	77.41	74.84	-	76.17
credit	86.95	86.60	85.91	86.54

Cuadro 3.8: Precisión en datos de prueba para la validación del meta-clasificador con diferentes enfoques de combinación. Se muestran: Suma máxima, Promedio máximo y Peso máximo.

La prueba de Iman-Davenport arrojó un valor de  $p = 0.0002087$  al aplicarse sobre las muestras de la tabla 14<sup>1</sup>. Esto muestra que para un nivel de confianza de  $\alpha = 0,05$  existen diferencias en el comportamiento de los métodos. En la tabla 3.9 se muestra el ranking de los enfoques.

Algoritmo	Ranking
Suma Máxima	1.099999999999
Media Máxima	2.4
Ishibuchi	2.5

Cuadro 3.9: Ranking promedio de los enfoques.

Con el objetivo de identificar en donde se encuentran las diferencias se realizan las pruebas de Holm y de Shaffer con un valor de significación  $\alpha = 0,05$ . Estas demuestran que existen diferencias significativas entre la combinación de resultados mediante la definición 2.11 y el

<sup>1</sup>Nota del autor: En la prueba no se utilizaron los datos de la definición 2.10, no estaban disponibles en todos los conjuntos de datos.

resto de las muestras del experimento. Por otra parte, no existen diferencias significativas entre la combinación utilizando la definición 2.9 y la aplicación de Ishibuchi [41] sobre todos los datos de manera centralizada. Los resultados de la aplicación de estas pruebas al grupo se muestran en la tabla 3.10.

i	Algoritmos	$z = (R_0 - R_i)/SE$	$p$	Holm	Shaffer
3	Def. 2.11 vs. Ishibuchi	3.130495	0.001745	0.01666666	0.01666666
2	Def. 2.11 vs. Definicion 2.9	2.906888	0.003650	0.025	0.05
1	Def. 2.9 vs. Ishibuchi	0.223606	0.823066	0.05	0.05

Cuadro 3.10: Resultados de las pruebas de Holm/Shaffer para  $\alpha = 0,05$

Los resultados obtenidos a partir de la precisión en la clasificación fueron contrastados mediante el uso de otros estadígrafos. En el anexo 3.6 se presentan los valores de *F-Measure* y *Area\_Under\_ROC*. Al aplicar sobre ellos la prueba de Iman-Davenport se obtuvieron valores  $p = 0,2855442$  y  $p = 0,119852$ . Para un nivel de confianza de  $\alpha = 0,05$  no es posible detectar diferencias en el grupo.

La consecuencia más destacable es el comportamiento del enfoque para la combinación de resultados dado en 2.11, que supera al resto de los métodos del grupo en precisión. Aunque los enfoques 2.10 y 2.9 no logran sobrepasar a los clasificadores locales en las simulaciones efectuadas, el comportamiento competitivo frente al caso centralizado es ya un resultado valioso. Por otro lado, la no diferencia soportada por las pruebas sobre *F-Measure* y *Area\_under\_ROC*, aunque hace disminuir la confianza en los resultados obtenidos en la precisión, apoya el uso de los métodos de combinación frente a la variante centralizada.

### 3.6. Conclusiones

La validación estadística efectuada en este capítulo apoya la hipótesis de que el método propuesto para la creación visual de sistemas de reglas difusas es competitivo frente a otros modelos de clasificación. Los sistemas producidos en los experimentos están compuesto por reglas con pocos antecedentes.

El segundo grupo de pruebas efectuados probó que la adaptación realizada, donde se modifica la forma de calcular el grado de compatibilidad y se establece según los operadores de la

lógica difusa compensatoria, no supera al enfoque tradicional. En los conjuntos de datos estudiados, el uso de la media geométrica no basta para obtener un mejor ajuste de las regiones de decisión en cada regla.

El esquema de combinación de resultados mediante la selección de la máxima suma por clases arrojó diferencias significativas con respecto al resto de las formulaciones. La propuesta en cuestión mostró un comportamiento superior en la precisión para todos los conjuntos de datos usados. El resto de las definiciones en cambio muestra un rendimiento similar entre sí y al compararla con la variante no distribuida.

# Conclusiones

Del resultado de este trabajo se pueden inferir las siguientes conclusiones:

- El diseño de una arquitectura basada en agentes que realizan minería sobre los conjuntos de datos locales, para obtener resultados globales preservando la interpretabilidad y respetando las restricciones de seguridad es viable. Unas pocas clases de agentes, que simplifican la arquitectura, se han definido: agentes de usuario, mediador, agentes de minería local y agentes de manipulación de datos. Múltiples alternativas pueden usarse para cada clase, sólo debe respetarse el esquema general de intercambio de reglas ganadoras.
- El efecto de usar la lógica difusa compensatoria en sistemas de clasificación, con el enfoque empleado en este trabajo, en lugar de la lógica difusa clásica, no superó el enfoque clásico. La simple sustitución del operador producto por la media geométrica, que introduce la propiedad de compensación - que da nombre a este sistema - no es suficiente para provocar cambios significativos en los resultados. No implica esto que la lógica difusa compensatoria no sea útil con el propósito de incrementar la precisión, pero la propuesta efectuada no es suficiente.
- Las técnicas de visualización han sido aplicadas con éxito a la minería de datos, pero el mayor impacto ha estado en el análisis exploratorio, estudio de resultados y prueba de hipótesis; la información extraída ha sido tradicionalmente consumida como conocimiento por expertos, en pocos casos se ha usado para la construcción de modelos de clasificación. Una técnica para la creación visual de sistemas de reglas difusas ha sido diseñada, los resultados arrojados en su validación son competitivos. Los clasificadores construidos mediante esta técnica, pueden ser usados directamente o en cambio empleados para generar conjuntos de reglas iniciales, a manera de heurística, para otros métodos evolutivos que refinen el sistema.

- Las reglas obtenidas visualmente tienen una longitud promedio de uno o dos términos, lo que contrasta con otros resultados publicados. El resultado es, sin embargo, consistente con el hecho bien conocido de que la técnica de coordenadas paralelas es útil para encontrar correlaciones entre unas pocas variables y que la capacidad para expresarlas dependen del orden de las mismas. Es de esperarse que los sistemas de reglas difusas construidos estén compuestas por un número grande de reglas cortas.
- Los métodos de combinación definidos están inspirados en los enfoques basados en el voto, sin embargo, al usar como clasificadores base sistemas de reglas difusas, se han utilizado las propiedades que éstos brindan. Los resultados obtenidos, se corresponden con el comportamiento de los meta-clasificadores al aventajar al clasificador con los datos centralizados en las pruebas realizadas. El método ofrece además, la posibilidad de mostrar una explicación interpretable de sus decisiones a los usuarios.

# Recomendaciones

- Realizar otras investigaciones destinadas a identificar y evaluar variantes de adaptación para el uso de la lógica difusa compensatoria en métodos publicados de construcción de sistemas de reglas difusas. Otras modificaciones posibles son la definición de nuevas formas de hallar el consecuente y el grado de certidumbre de la regla. Igualmente pueden tomarse enfoques en que los antecedentes incluyan otros operadores lógicos.
- Estudiar el comportamiento de los métodos de combinación de resultados bajo el supuesto de no igualdad en las distribuciones de los datos para diferentes localizaciones.
- Valorar la posibilidad de construir árboles de decisión difusos a partir de datos usando métodos de minería visual inspirados en los trabajos citados de generación visual de árboles de decisión.

# Bibliografía

- [1] Janos Abonyi, Johannes A. Roubos, and Ferenc Szeifert. Data-driven generation of compact, accurate, and linguistically-sound fuzzy classifiers based on a decision-tree initialization. *International Journal of Approximate Reasoning*, 32:1–21, 2002.
- [2] R. Agrawal and R. Srikant. Fast algorithms for mining association rules. In *Proc. of 20th International Conference on Very Large Data Bases*, pages 487–499, 1994.
- [3] V. S. Ananthanarayana, D. K. Subramanian, and M. N. Murty. Scalable, distributed and dynamic mining of association rules. In *Proceedings of HIPC'00*, pages 559–566, Bangalore, India, 2000.
- [4] Rafael Alejandro Espín Andrade, Gustavo Mazcorro Téllez, and Eduardo Fernández González. Consideraciones sobre el carácter normativo de la lógica difusa compensatoria. Technical report, 2007.
- [5] Rafael Espin Andrade, Gustavo Mazcorro Téllez, Eduardo Fernández González, Jorge Marx-Gómez, and Maria Ines Lecich. Compensatory logic: A fuzzy normative model for decision making. *Operations Research Journal. Havana University*, 2007.
- [6] Mihael Ankerst, Christian Elsen, Martin Ester, and Hans peter Kriegel. Visual classification: An interactive approach to decision tree construction. In *Proc. 5th Int. Conf. on Knowledge Discovery and Data Mining (KDD'99)*, pages 392–396, 1999.
- [7] Mihael Ankerst, Daniel A. Keim, and Hans-Peter Kriegel. Circle segments: A technique for visually exploring large multidimensional data sets. In *Visualization '96, Hot Topic Session*, 1996.
- [8] Roberto Battiti and Anna Maria Colla. Democracy in neural nets: Voting schemes for classification. *Neural Networks*, 7(4):691 – 707, 1994.

- [9] Fabio Luigi Bellifemine, Giovanni Caire, and Dominic Greenwood. *Developing Multi-Agent Systems with JADE*. 2007.
- [10] Michael R. Berthold and Lawrence O. Hall. Visualizing fuzzy points in parallel coordinates. Technical report, Berkeley, CA, USA, 1999.
- [11] R. Bhatnagar. Decision tree induction by cooperating agents. In *Workshop on Multi-Agent Learning*, Providence, RI, July 1997.
- [12] Jorge Casillas, Brian Carse, and Larry Bull. Fuzzy-xcs: A michigan genetic fuzzy system. *IEEE TRANSACTIONS ON FUZZY SYSTEMS*, 15(4), August 2007.
- [13] Jorge Casillas, Oscar Cordón, and Francisco Herrera. Learning fuzzy rules using ant colony optimization algorithms. In *Abstract proceedings of ANTS2000 From Ant Colonies to Arti Ants: A Series of International Workshops on Ant Algorithms*, pages 13–21, 2000.
- [14] Philip K. Chan, Wei Fan, Andreas L. Prodromidis, and Salvatore J. Stolfo. Distributed data mining in credit card fraud detection. *IEEE Intelligent Systems*, 14(6):67–74, 1999.
- [15] Xiaoguang Chang and John H Lilly. Evolutionary design of a fuzzy classifier from data. *IEEE Trans Syst Man Cybern B Cybern*, 34(4):1894–906, 2004.
- [16] F. Chiclana, F. Herrera, E. Herrera-Viedma, and L. Martínez. A note on the reciprocity in the aggregation of fuzzy preference relations using owa operators. *Fuzzy Sets Syst.*, 137(1):71–83, 2003.
- [17] O. Cordón, F. Herrera, F. Hoffmann, and L. Magdalena. *Genetic Fuzzy Systems: Evolutionary Tuning and Learning of Fuzzy Knowledge Bases*. Singapore: World Scientific, 2001.
- [18] O. Cordon, M. J. del Jesus, F. Herrera, and M. Lozano. Mogul: A methodology to obtain genetic fuzzy rule-based systems under the iterative rule learning approach. *INTERNATIONAL JOURNAL OF INTELLIGENT SYSTEMS*, 14:1123–1153, 1994.
- [19] M. Dasgupta and R. Deb. Factoring fuzzy transitivity. *Fuzzy Sets and Systems*, 118(3):489 – 502, 2001.

- [20] S. Datta, C. Giannella, and H. Kargupta. K-means clustering over a large, dynamic network. In *Proceedings of 2006 SIAM Conference on Data Mining*, Bethesda, MD, April 2006.
- [21] Janez Demšar. Statistical comparisons of classifiers over multiple data sets. *Machine Learning Research*, 7:1–30, 2006.
- [22] Marcin Detyniecki. *Mathematical Aggregation Operators and their Application to Video Querying*. Ph.d. dissertation, University of Paris 6. University Pierre and Marie Curie, 2000.
- [23] D. Dubois and H. Prade. A review of fuzzy set aggregation connectives. *Information Sciences*, 36(1-2):85 – 121, 1985.
- [24] R. Espín Andrade and J. Marx Gómez. Lógica compensatoria: Un enfoque axiomático para el desarrollo de aplicaciones administrativas. In *Proceedings of 2da. Conferencia Iberoamericana en Sistemas, Cibernética e Informática CISCI*, 2003.
- [25] R. Espín Andrade and J. Marx Gómez. Knowledge discovery based on compensatory logic and metaheuristic search. In Xuan F. Zha, editor, *Handbook of Research on Artificial Intelligence in Industrial Information Systems: Methodologies, Frameworks and Emerging Technologies*. 2008.
- [26] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 1997.
- [27] M. Friedman. A comparison of alternative test of significance for the problem of m rankings. *Annals of Mathematical Statistics*, 11:86–92, 1940.
- [28] Michael Friendly and Ernest Kwan. Effect ordering for data displays. *Comput. Stat. Data Anal.*, 43(4):509–539, 2003.
- [29] Michelle Galea and Qiang Sheng. *Swarm Intelligence in Data Mining*, volume 34 of *Studies in Computational Intelligence*. 2006.
- [30] Nicolás García-Pedrajas, César García-Osorio, and Colin Fyfe. Non linear boosting projections for ensemble construction. *Machine Learning Research*, 8, 2007.
- [31] Antonio González and Raúl Pérez. Slave: A genetic learning system based on an iterative approach. *IEEE TRANSACTIONS ON FUZZY SYSTEMS*, 7(2), April 1999.

- [32] Vladimir Gorodetsky, Oleg Karsaeyv, and Vladimir Samoilov. Multi-agent technology for distributed data mining and classification. In *IEEE/WIC International Conference on Intelligent Agent Technology (IAT 2003)*, October 2003.
- [33] David Hand, Heikki Mannila, and Padhraic Smyth. *Principles of Data Mining*. The MIT Press, 2001.
- [34] Charles D. Hansen and Chris Johnson. *The Visualization Handbook*. Academic Press, December 2004.
- [35] J. Holland and J. Reitman. Cognitive systems based on adaptive algorithms. In *Pattern-Directed Inference Systems*, pages 313–329. 1978.
- [36] A. Inselberg. Multidimensional detective. In *INFOVIS '97: Proceedings of the 1997 IEEE Symposium on Information Visualization (InfoVis '97)*, page 100, Washington, DC, USA, 1997. IEEE Computer Society.
- [37] A. Inselberg and B. Dimsdale. Parallel coordinates: a tool for visualizing multi-dimensional geometry. pages 361–378, 1990.
- [38] H. Ishibuchi, T. Nakashima, and T. Murata. Three-objective genetic based machine learning for linguistic rule extraction. *Inf. Sci.*, 136:109–133, 2001.
- [39] Hisao Ishibuchi and Yusuke Nojima. Analysis of interpretability-accuracy tradeoff of fuzzy systems by multiobjective fuzzy genetics-based machine learning. *International Journal of Approximate Reasoning*, 44:4–31, 2007.
- [40] Hisao Ishibuchi and Takashi Yamamoto. Comparison of heuristic rule weight specification methods. In *Proc. 11th IEEE Int. Conf. Fuzzy Systems*, pages 908–913, May 2002.
- [41] Hisao Ishibuchi and Takashi Yamamoto. Fuzzy rule selection by multi-objective genetic local search algorithms and rule evaluation measures in data mining. *Fuzzy Sets and Systems*, 141(1):59 – 88, 2004. Genetic Fuzzy Systems: New Developments.
- [42] Hisao Ishibuchi, Takashi Yamamoto, and Tomoharu Nakashima. Hybridization of fuzzy gbml approaches for pattern classification problems. *IEEE Trans. on Systems, Man, and Cybernetics - Part B*, 35:359–365, 2005.
- [43] Cezary Z. Janikow. Fuzzy decision trees: Issues and methods, 1996.

- [44] Eser Kandogan. Visualizing multi-dimensional clusters, trends, and outliers using star coordinates. In *KDD '01: Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 107–116, New York, NY, USA, 2001. ACM.
- [45] Daniel A. Keim. Designing pixel-oriented visualization techniques: Theory and applications. *IEEE Transactions on Visualization and Computer Graphics*, 6:59–78, 2000.
- [46] Daniel A. Keim, Mihael Ankerst, and Hans-Peter Kriegel. Recursive pattern: A technique for visualizing very large amounts of data. In *VIS '95: Proceedings of the 6th conference on Visualization '95*, page 279, Washington, DC, USA, 1995. IEEE Computer Society.
- [47] Matthias Klusch, Stefano Lodi, and Gianluca Moro. Agent-based distributed data mining: the kdec scheme. pages 104–122, 2003.
- [48] Dexter Kozen. A completeness theorem for kleene algebras and the algebra of regular events. *Information and Computation*, 110:366–390, 1994.
- [49] Ludmila I. Kuncheva. *Combining Pattern Classifiers: Methods and Algorithms*. Wiley-Interscience, 2004.
- [50] Andreas D. Lattner, Alexander Grimme, and Ingo J. Timm. An evaluation of meta learning and distribution strategies in distributed machine learning, 2007.
- [51] Eghbal G. Mansoori, Mansoor J. Zolghadri, and Seraj D. Katebi. A weighting function for improving fuzzy classification systems performance. *Fuzzy Sets Syst.*, 158(5):583–591, 2007.
- [52] Yusuke Nojima and Hisao Ishibuchi. Computational efficiency of parallel distributed genetic fuzzy rule selection for large data sets. In *Proc. of Information Processing and Management of Uncertainty in Knowledge-based Systems*, pages 1137–1142, 2008.
- [53] Tatsuya Nomura and Tsutomu Miyoshi. An adaptive fuzzy rule extraction using hybrid model of the fuzzy self-organizing map and the genetic algorithm with numerical chromosomes, 1998.
- [54] Wei Peng, Matthew O. Ward, and Elke A. Rundensteiner. Clutter reduction in multi-dimensional data visualization using dimension reordering. In *INFOVIS '04: Proceedings*

- of the *IEEE Symposium on Information Visualization*, pages 89–96, Washington, DC, USA, 2004. IEEE Computer Society.
- [55] Hans Roubos, Magne Setnes, and Janos Abonyi. Learning fuzzy classification rules from data, 2000.
- [56] Matthew Roughan and Yin Zhang. Secure distributed data-mining and its application to large-scale network measurements. *SIGCOMM Comput. Commun. Rev.*, 36(1):7–14, 2006.
- [57] Jorn Schneidewind, Mike Sips, and Daniel A. Keim. Pixnostics: Towards measuring the value of visualization. In *IEEE Symposium on Visual Analytics Science and Technology*, 2006.
- [58] Harri Siirtola, Tuuli Laivo, Tomi Heimonen, and Kari-Jouko Raiha. Visual perception of parallel coordinate visualizations. *Information Visualisation, International Conference on*, 0:3–9, 2009.
- [59] S. Smith. *A Learning System Based on Genetic Adaptive Algorithms*. PhD thesis, University of Pittsburgh, 1980.
- [60] Tom Soukup and Ian Davidson. *Visual Data Mining: Techniques and Tools for Data Visualization and Mining*. John Wiley & Son, 2002.
- [61] Zbigniew Switalski. General transitivity conditions for fuzzy reciprocal preference matrices. *Fuzzy Sets Syst.*, 137(1):85–100, 2003.
- [62] Soon Tee Teoh. Paintingclass: interactive construction, visualization and exploration of decision trees. In *In Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 667–672. ACM Press, 2003.
- [63] Soon Tee Teoh and Kwan-Liu Ma. Starclass: Interactive visual classification using star coordinates. In *SDM*, 2003.
- [64] R. Vilalta, Christophe Giraud-Carrier, Pavel Brazdil, and Carlos Soares. Using meta-learning to support data mining. *International Journal of Computer Science and Applications*, 1(1):31–45, 2004. DBLP.
- [65] J. S. Wang and G. C. S. Lee. Self-adaptive neuro-fuzzy inference system for classification application. *IEEE Transaction Fuzzy System*, 10:790–802, 2002.

- [66] F. Wilcoxon. Individual comparisons by ranking methods. *Biometrics*, 1:80–83, 1945.
- [67] Ian H. Witten and Eibe Frank. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, second edition, 2005.
- [68] David H. Wolpert. Original contribution: Stacked generalization. *Neural Netw.*, 5(2):241–259, 1992.
- [69] Yonghong Xu, Wenxue Hong, Na Chen, Xin Li, WenYuan Liu, and Tao. Parallel filter: A visual classifier bases on parallel coordinates and multivariate data analysis, 2007.
- [70] Nong Ye, editor. *The handbook of data mining*. Lawrence Erlbaum Associates, 2003.

# Nomenclatura

ACO Optimización mediante colonias de hormigas. Ant Colony Optimization

DDM Minería de datos distribuida (Distributed Data Mining)

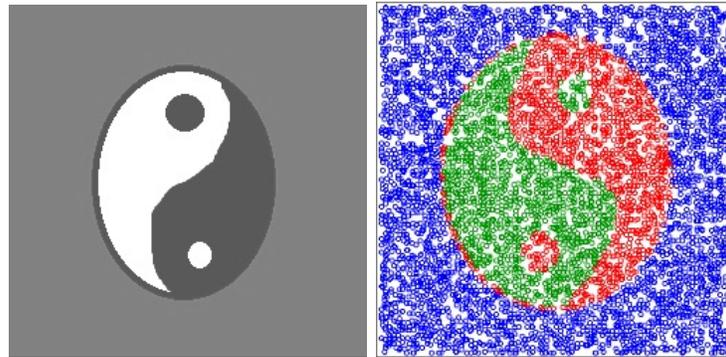
DM Minería de datos (Data Mining)

FRS Sistemas de reglas difusas

LDC Lógica difusa compensatoria

# Anexos

## Anexo 1



(a) El conjunto de datos original tiene 65499 patrones  
(b) Conjunto de datos simplificado, 6550 instancias escogidas aleatoriamente

Figura 1: El símbolo del Yin y el Yang.

## Anexo 2

	Este trabajo	Ishibuchi
Wine	0.95	0.95
breast cancer W	0.96	0.96
Iris	1	1
Sonar	0.72	0.68
diabetes	0.83	0.82
glass	0.66	0.59
ecoli	0.90	0.91
YinYang	0.94	0.95
Wave-1000	0.71	0.73
credit	0.86	0.86

Cuadro 11: Resultados experimentales obtenidos al aplicar la LDC o no a FRS. F-Measure.

	Este trabajo	Ishibuchi
Wine	0.96	0.96
breast cancer W	0.98	0.97
Iris	0.99	0.99
Sonar	0.78	0.77
diabetes	0.67	0.69
glass	0.75	0.70
ecoli	0.91	0.92
YinYang	0.97	0.98
Wave-1000	0.77	0.80
credit	0.89	0.89

Cuadro 12: Resultados experimentales obtenidos al aplicar la LDC o no a FRS. Área bajo la curva.

## Anexo 3

	Definición 2.11	Definición 2.9	Definición 2.10	Ishibuchi [41]
Wine	0.98	0.95	0.95	0.95
breast cancer W	0.95	0.96	-	0.96
Iris	1	1	1	1
Sonar	0.70	0.64	0.67	0.68
diabetes	0.83	0.83	0.83	0.82
glass	0.62	0.59	0.63	0.59
ecoli	0.94	0.91	0.93	0.91
YinYang	0.94	0.94	0.94	0.95
wave-1000	0.74	0.70	-	0.73
credit	0.86	0.86	0.86	0.86

Cuadro 13: F-Measure en datos de prueba para la validación del meta-clasificador con diferentes enfoques de combinación. Se muestran: Suma máxima, Promedio máximo y Peso máximo.

	Definición 2.11	Definición 2.9	Definición 2.10	Ishibuchi [41]
Wine	1	0.98	0.97	0.96
breast cancer W	0.97	0.98	-	0.97
Iris	1	1	1	0.99
Sonar	0.81	0.77	0.74	0.77
diabetes	0.73	0.69	0.72	0.69
glass	0.79	0.71	0.72	0.70
ecoli	0.97	0.91	0.93	0.92
YinYang	0.96	0.98	0.98	0.98
wave-1000	0.87	0.84	-	0.80
credit	0.88	0.90	0.88	0.89

Cuadro 14: Area\_under\_ROC en datos de prueba para la validación del meta-clasificador con diferentes enfoques de combinación. Se muestran: Suma máxima, Promedio máximo y Peso máximo.