

# **"Un Motor de Inferencia basado en Lógica Difusa para la detección de Patrones de Comportamiento en Datos"**



**UADY**

*"Luz, Ciencia y Verdad"*

**LCC Miguel Angel Uc Dzib**

Directores

Dr. Víctor Hugo Menéndez Domínguez

M. en C. Salvador Medina Peralta

Facultad de Matemáticas

Universidad Autónoma de Yucatán

*Maestría en Ciencias de la Computación*

Noviembre 2016



## **Dedicatoria**

A Dios, y a mi familia. ...



## **Agradecimientos**

A Dios, por bendecirme y permitirme terminar esta etapa de mi vida.

A mi familia, por su incondicional apoyo en todo momento.

A mis asesores, Dr. Víctor Hugo Menéndez Domínguez y M. en C. Salvador Medina Peralta. Por sus valiosos consejos y sugerencias para la elaboración de este trabajo, así como de su compañerismo y paciencia para conmigo.

A los maestros de la Facultad de Matemáticas, por haberme transmitido parte de sus conocimientos y disciplina durante los años de mi formación profesional.

Al CONACyT por el apoyo económico que me brindó para poder estudiar la maestría.



## Resumen

La reciente necesidad de utilizar modelos matemáticos predictivos y de clasificación en disciplinas como la Biología, Medicina, Educación, entre otras, ha estimulado el interés en el empleo de nuevas técnicas para su generación. El problema estriba en que muchos de esos modelos son extremadamente difíciles de modelar por una fórmula o ecuación matemática exacta y precisa debido a la complejidad de la estructura del sistema como por ejemplo la no linealidad, la incertidumbre, la aleatorización, etc. En este sentido, los sistemas de inferencia difusa permiten modelar procesos complejos o sistemas incompletos o inciertos. En este trabajo se presenta una técnica denominada Tablas de Repertorio Difuso (TRD) que se representa como una matriz rectangular con elementos (columnas) y constructos (filas). Se desarrolló un motor de inferencia basado en lógica difusa para la detección de patrones de comportamiento en colecciones de datos. Para su evaluación se usaron datos provenientes de experimentos en las áreas de Biología, Medicina, Educación. Las predicciones se generaron con funciones de pertenencia triangular, trapezoidal y gaussiana a través de dos propuestas, una con variable de salida de tipo nominal (discreto) y otra en escala de razón (continua). Los resultados obtenidos para la detección de patrones de comportamiento muestran una metodología computacional y estadística adecuada para asegurar la evaluación correcta del motor de inferencia y los datos.





# Índice general

<b>1. Introducción</b>	<b>1</b>
1.1. ENTORNO DE LA INVESTIGACIÓN . . . . .	1
1.2. HIPÓTESIS Y OBJETIVOS . . . . .	2
1.3. METODOLOGÍA DE INVESTIGACIÓN . . . . .	3
1.4. TRABAJOS RELACIONADOS . . . . .	4
1.5. MARCO DE TRABAJO . . . . .	9
1.6. APORTACIONES . . . . .	9
1.6.1. Aportaciones teóricas . . . . .	10
1.6.2. Aportaciones prácticas . . . . .	10
1.7. ORGANIZACIÓN DEL DOCUMENTO . . . . .	10
<b>2. Estado del Arte</b>	<b>13</b>
2.1. LÓGICA DIFUSA . . . . .	13
2.1.1. Función de pertenencia . . . . .	14
2.1.2. La representación borrosa del conocimiento . . . . .	14
2.1.3. Razonamiento aproximado . . . . .	16
2.2. SISTEMAS DE INFERENCIA DIFUSO . . . . .	18
2.2.1. Definición y estructura . . . . .	19
2.3. SISTEMA DIFUSO BASADOS EN REGLAS . . . . .	21
2.3.1. Notación . . . . .	21
2.3.2. Procesos en los sistemas difusos . . . . .	22
2.4. REPRESENTACIÓN DEL CONOCIMIENTO . . . . .	23
2.4.1. Representaciones básicas . . . . .	23
2.4.2. Representaciones avanzadas . . . . .	23
2.4.3. Representación de incertidumbre . . . . .	24
2.4.3.1. Factores de certeza . . . . .	24
2.4.3.2. Lógica difusa . . . . .	25
2.5. APRENDIZAJE . . . . .	26

2.5.1. Métodos de aprendizaje automático . . . . .	27
2.6. TABLAS DE REPERTORIO DIFUSO . . . . .	28
2.6.1. ¿Por qué usar las tablas de repertorio difuso? . . . . .	28
2.6.2. Salidas o respuestas y atributos . . . . .	28
2.7. APLICACIONES EN LOS SISTEMAS DE INFERENCIA DIFUSA . . . . .	29
2.8. CONCLUSIONES . . . . .	32
<b>3. MODELOS MATEMÁTICOS EN LA LÓGICA DIFUSA</b>	<b>33</b>
3.1. Modelación . . . . .	33
3.2. Modelo basado en sistemas difusos estáticos . . . . .	37
3.3. Modelo basado con un factor de Peso . . . . .	37
3.4. Modelo basado en criterios de clasificación . . . . .	38
3.5. Evaluación predictiva de modelos de inferencia difusa . . . . .	39
3.5.1. Prueba de desempeño con proporciones caso discreto . . . . .	40
3.5.2. Prueba de desempeño caso continuo . . . . .	40
3.5.3. Algoritmo de desempeño caso discreto y continuo . . . . .	41
3.5.4. Algoritmo de evaluación . . . . .	42
3.6. CONCLUSIONES . . . . .	43
<b>4. IMPLEMENTACIÓN DE MOTOR DE INFERENCIA DIFUSO</b>	<b>45</b>
4.1. ESPECIFICACIONES TÉCNICAS . . . . .	45
4.1.1. Tecnologías y técnicas de desarrollo . . . . .	45
4.1.2. Bibliotecas de desarrollo en R . . . . .	47
4.1.3. Flujo de datos . . . . .	47
4.1.4. Arquitectura . . . . .	48
4.1.5. Arquitectura UML . . . . .	49
4.2. APLICACIÓN SWID . . . . .	51
4.2.1. Funcionalidades . . . . .	51
4.2.2. Estructura de navegación . . . . .	51
4.2.3. Presentación del sitio web . . . . .	52
4.3. CONCLUSIONES . . . . .	56
<b>5. EXPERIMENTACIÓN</b>	<b>57</b>
5.1. VALIDACIÓN MEDIANTE TÉCNICAS NO INVASIVAS . . . . .	57
5.1.1. Configuración de la evaluación . . . . .	58
5.1.2. Proceso de experimentación . . . . .	59
5.1.3. Descripción del conjunto de datos . . . . .	59

---

5.1.4. Rendimiento . . . . .	61
5.2. VALIDACIÓN DE MODELOS PREDICTIVOS EN BIOLOGÍA . . . . .	66
5.2.1. Configuración de la evaluación . . . . .	66
5.2.2. Descripción del conjunto de datos . . . . .	67
5.2.3. Resultados . . . . .	68
5.3. VALIDACIÓN DE MODELOS PREDICTIVOS EN MEDICINA . . . . .	71
5.3.1. Configuración de la evaluación . . . . .	71
5.3.2. Descripción del conjunto de datos . . . . .	72
5.3.3. Resultados . . . . .	72
5.4. VALIDACIÓN DE MODELOS PREDICTIVOS EN EDUCACIÓN . . . . .	74
5.4.1. Configuración de la evaluación . . . . .	74
5.4.2. Descripción del conjunto de datos . . . . .	74
5.4.3. Resultados . . . . .	75
5.5. CONCLUSIONES . . . . .	77
<b>6. CONCLUSIONES</b>	<b>79</b>
6.1. ANÁLISIS DE LA CONSECUCIÓN DE LOS OBJETIVOS . . . . .	79
6.2. ANÁLISIS DE LAS APORTACIONES . . . . .	80
6.3. CONTRASTE DE RESULTADOS . . . . .	81
6.3.1. Artículos en revistas . . . . .	81
6.3.2. Artículos en revisión . . . . .	81
6.3.3. Memorias de congresos . . . . .	82
6.3.4. Seminarios . . . . .	82
6.4. TRABAJO A FUTURO . . . . .	82
6.4.1. Mejoras en las técnicas utilizadas . . . . .	82
6.4.2. Mejoras en la implementación . . . . .	83
<b>Índice de figuras</b>	<b>85</b>
<b>Índice de tablas</b>	<b>87</b>
<b>Bibliografía</b>	<b>89</b>



# Capítulo 1

## Introducción

Este capítulo tiene como propósito ofrecer una visión del trabajo desarrollado para esta tesis. Se inicia describiendo el entorno de la investigación, resaltando los problemas relacionados con la lógica difusa (lógica borrosa) y sus aplicaciones, que han dado origen a la hipótesis y los objetivos de esta tesis. Se describen, en forma breve, el método y el marco de trabajo empleado dentro la investigación. Posteriormente, se listan las aportaciones derivadas del trabajo, tanto en su aspecto teórico como práctico. Finalmente, se presenta la estructura organizativa del documento.

### 1.1. ENTORNO DE LA INVESTIGACIÓN

Internet ha tenido una marcada evolución en los últimos años , permitiendo el acceso y el intercambio de información de manera ágil, con características de flexibilidad en la tecnología de acceso y con capacidades de integración a nivel de servicios (Lago, 2010).

Existen disciplinas como la Biología, Medicina, Educación, entre otras, donde se requiere ejecutar diferentes modelos matemáticos en la computadora, sin embargo, debido a la complejidad en su ejecución, se requiere generar ambientes fáciles de utilizar para los usuarios, el rendimiento o el tiempo de ejecución en la generación de ellos utilizando herramientas web permite decidir si es viable desarrollar la aplicación a través de una arquitectura propuesta. La modelación utilizando lógica difusa es una alternativa relevante dada la posibilidad de representar la incertidumbre y la aleatorización presente en los problemas existentes en las áreas antes mencionadas. Esta modelación difusa, tiene impacto práctico en el control, la predicción y la deducción (Takagi, 1985).

Un sistema de inferencia difuso emplea reglas difusas “si-entonces” que pueden modelar los aspectos cualitativos del conocimiento humano y procesos de razonamiento sin emplear análisis cuantitativos precisos. En 1974, el profesor Mamdani experimentó con

éxito un controlador difuso en una máquina de vapor, pero la primera implantación real de un controlador de este tipo fue realizada en 1980 por F. L. Smidth & Co. en una planta cementera en Dinamarca (Jose, 2007). En 1983, Fuiji aplicó lógica difusa para el control de inyección química para plantas depuradoras de agua, por primera vez en Japón. En 1987 la empresa OMRON desarrolla los primeros controladores difusos comerciales con el profesor Yamakawa. A partir de ese momento, el control difuso ha sido aplicado con éxito en muy diversas ramas tecnológicas, por ejemplo la metalurgia, los robots de fabricación, controles de maniobra de aviones, ascensores o trenes (tren-metro de Sendai, Japón, 1987), sensores, imagen y sonido (sistema de estabilización de imagen en cámaras fotográficas y de video Sony, Sanyo, Canon), electrodomésticos (lavadoras de Panasonic o Bosch, aire acondicionado Mitsubishi, rice cooker), automoción (sistemas de ABS de Mazda o Nissan, cambio automático de Renault, control automático de velocidad, climatizadores) y una larga lista de aplicaciones comerciales (Jose, 2007).

## 1.2. HIPÓTESIS Y OBJETIVOS

A continuación se presentan la hipótesis, el objetivo general y los objetivos específicos del proyecto, a partir de la pregunta de investigación.

***Pregunta de investigación:*** ¿Es posible utilizar un motor de inferencia difusa en áreas como la Biología, Medicina, Educación, entre otras, para generar modelos de predicción y clasificación en el área de la lógica difusa?

Se pretende probar que desde una misma interfaz web y el lenguaje de programación R como motor de inferencia difuso se pueda facilitar la generación de modelos por usuarios de diferentes áreas de conocimiento. Después del estudio de la situación actual, la generación de modelos difusos, así como el uso de aplicaciones web se plantea la siguiente hipótesis:

***Hipótesis:*** La utilización de un sistema web y un motor de inferencia difusa puede facilitar la generación de modelos de predicción y clasificación en el área de la lógica difusa.

A partir de la hipótesis establecida y una vez estudiados los aspectos relacionados con la problemática presentada, se indica el objetivo general que se pretende con este trabajo:

***Objetivo General:*** Desarrollar un motor de inferencia basado en lógica difusa que simplifique la generación de modelos de inferencia difusa con respecto a la detección de patrones de comportamiento en datos.

En función de este objetivo principal se plantean los siguientes objetivos específicos:

1. Investigar los modelos de inferencia disponibles en el área de lógica difusa.
2. Seleccionar el modelo difuso que permitirá obtener resultados óptimos en términos de cálculos computacionales.
3. Diseñar una arquitectura Web que faciliten la integración de los elementos de la aplicación.
4. Desarrollar una aplicación Web que permita generar modelos de inferencia difusa y realizar predicciones con el modelo generado.
5. Definir un conjunto de instrumentos e indicadores para medir la eficiencia de los modelos de inferencia difusa generados.
6. Plantear nuevos algoritmos de inferencia difusa que permita evaluar la capacidad predictiva de las funciones de pertenencia triangular, trapezoidal y gaussiana.
7. Analizar experimentalmente el funcionamiento de cada uno de los elementos de la aplicación web.

### 1.3. METODOLOGÍA DE INVESTIGACIÓN

Esta investigación se centra en aspectos relativos a la generación de modelos de inferencia difusa dentro de una aplicación Web. En forma general, se describen a continuación las principales fases consideradas para la realización del proyecto:

1. **Definir la problemática y analizar el estado del arte.** Se refiere a buscar la forma de generar modelos de inferencia difusa y mostrar los resultados relevantes en un tiempo considerado óptimo. Esto origina la necesidad de definir los parámetros que se tomarán en cuenta para medir los resultados en la interfaz web. Se realizó una revisión sistemática de las temáticas afines a la gestión de modelos de inferencia difusa así como sus principales especificaciones y estándares, las técnicas y métodos relacionados a la generación de modelos de inferencia difusa a través de un servicio web. La revisión sistemática sigue una secuencia estricta y bien definida de pasos metodológicos, de acuerdo con un protocolo definido a priori (Kitchenham y Charters, 2007).
2. **Proposición de un marco arquitectónico para la generación de modelos de inferencia difusa.** Se divide en las siguientes actividades:

- Definir modelos de inferencia difusa.
  - Definir las características de relevancia de los modelos de inferencia difusa.
  - Definir un modelo que permita generar los parámetros estimados de los modelos de inferencia difusa a través de la aplicación web.
3. **Implementación de un prototipo.** Esta fase se subdivide en las siguientes actividades:
- Generar una aplicación Web que genere los resultados de un modelo de inferencia difusa.
  - Ejecutar modelos y predicciones difusas.
  - Recolectar las respuestas obtenidas del servicio web.
  - Ordenar los resultados de acuerdo a ciertos criterios que permitan describir resultados relevantes.
4. **Experimentación.** Se realizaron estudios empíricos de los métodos y los productos desarrollados para demostrar su validez. Para ello, se utilizaron diferentes colecciones de datos utilizados en diferentes experimentos en el área de la Biología, Medicina, Educación, entre otras. Se evalúa la efectividad y la eficiencia del uso de la metodología y el modelo en escenarios reales.
5. **Difusión.** Se comparten los resultados obtenidos en las fases anteriores a la comunidad investigadora mediante la presentación de trabajos en distintos foros nacionales e internacionales, donde se difunden los productos generados. La retroalimentación se da en la forma de las revisiones previas a la presentación de las ponencias. Este proceso es afín al propuesto por el método de investigación denominado “Investigación en acción” (Avison, 1999, ver figura 1.1). Esto se realiza identificando las cuestiones relevantes que guían la investigación para realizar una recolección de información útil.

## 1.4. TRABAJOS RELACIONADOS

Los sistemas difusos se han utilizado gracias a su capacidad para resolver con éxito una amplia gama de problemas en diferentes campos de aplicación. Sin embargo su replicación y la aplicación requiere un alto nivel de conocimiento y experiencia. Además, algunos investigadores publican el código de software y / o de origen asociado con sus propuestas, lo cual es un obstáculo para el progreso científico en otras disciplinas y en la industria. En los últimos años, el software de sistemas difusos (FSS en inglés) se ha desarrollado con el fin de facilitar el uso. Algunos programas de software se distribuyen comercialmente, pero la



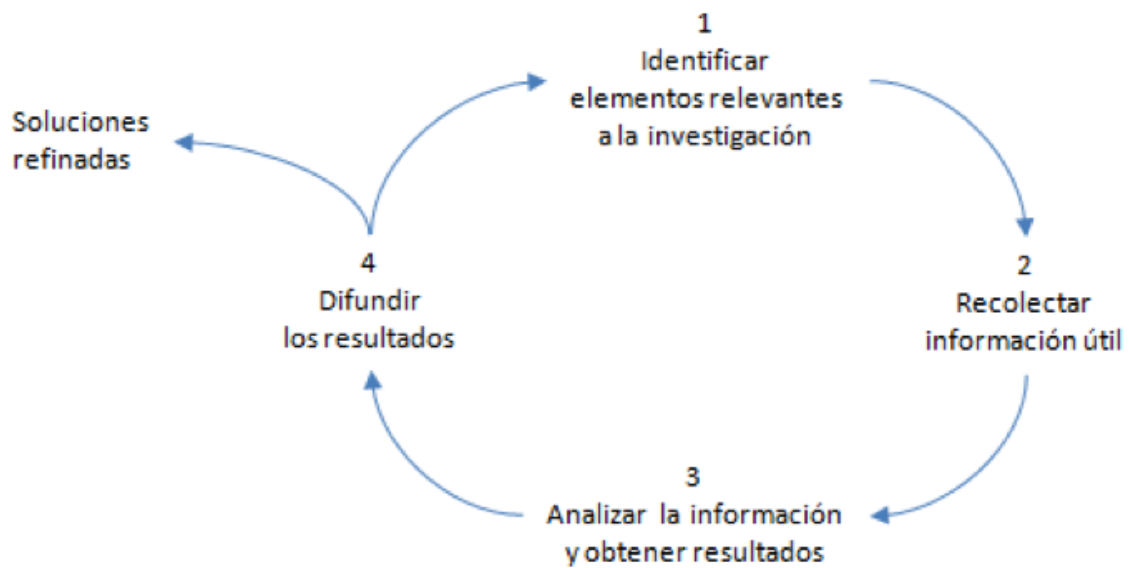


Figura 1.1 Proceso cíclico de “Investigación en acción”.

mayoría del software está disponible como software libre y de código abierto. La reducción de esos obstáculos ha proporcionado muchas ventajas, por ejemplo, la detección rápida de errores, aplicaciones innovadoras, adopción más rápida de los sistemas difusos, etc. En la figura 1.2, se presenta una visión general de software libre y código abierto de sistemas difusos de origen con el fin de proporcionar un marco bien establecido que ayuda a los investigadores a encontrar fácilmente las propuestas existentes y desarrollar futuros trabajos (Alcala, 2016); para lograr esto, se propone una taxonomía de dos niveles y se describen las principales contribuciones relacionadas con cada campo.

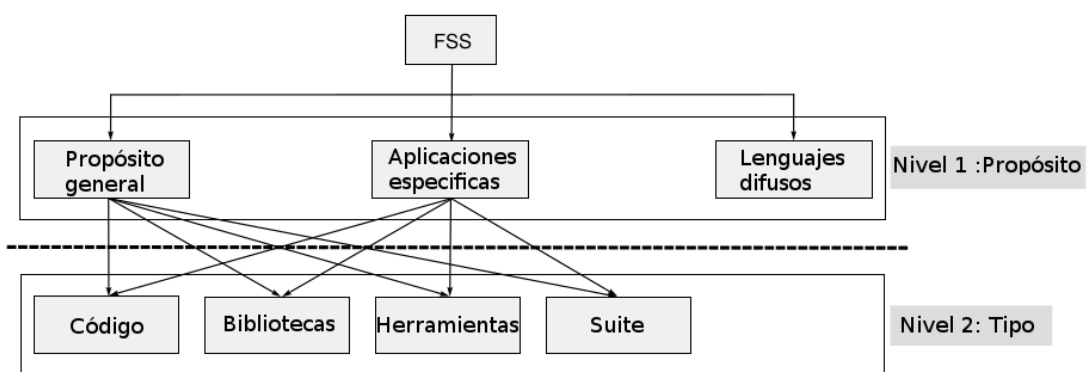


Figura 1.2 Taxonomía basada en el propósito (primer nivel ) y el tipo (segundo nivel) de sistemas difuso.

En el primer nivel, la primera categoría (de propósito general FSS) reúne las contribuciones en las que se desarrolla el FSS con un propósito general. Este tipo de FSS nos permite diseñar y analizar sistemas difusos (sistemas de reglas difusas, los sistemas difusos multicriterio de toma de decisiones, etc.) para diferentes problemas en relación con todas las investigaciones en áreas abordadas por la comunidad difusa (agrupamiento, clasificación, regresión, y así sucesivamente) entre los cuales se encuentra el horno de cemento de Dinamarca (1982), el metro de Sendai (1986) y Fuzzy Logic for Matlab (MathWorks, 2013). La segunda categoría (aplicaciones) se relaciona con las contribuciones que describen FSS para los propósitos de aplicación específicos como por ejemplo en otros campos de investigación (Procesamiento de Imágenes, Redes de Telecomunicaciones, Medicina, etc.), estos fueron sometidos a un rápido crecimiento de la comunidad difusa y se propusieron muchos sistemas difusos para resolver problemas específicos en estas áreas (Kerre, 2000) (Pedrycz, 2000) (Lisboa, 2000) (Tessem, 1994) (Vailikos, 1998). La tercera categoría principal incluye contribuciones en las que se proponen nuevos lenguajes para sistemas difusos, tales lenguajes proporcionan a los investigadores una forma de trabajar que hace más fácil el intercambio entre sistemas difusos y mejora la reutilización de los sistemas difusos desarrollados, Acampora (vicepresidente del Comité de Normas de la Sociedad de Inteligencia Computacional IEEE) ha formado un grupo de trabajo para desarrollar un lenguaje estándar basado en el Lenguaje de Marcado de Sistemas Difusos (Acampora, 2013).

En el segundo nivel de la taxonomía, el descubrimiento de las categorías correctas se convierte en una tarea difícil porque la terminología comúnmente utilizada en la literatura es algo engañosa. Considerando que los investigadores tienden a utilizar diferentes términos o definiciones para referirse al mismo tipo de software. Para aclarar esta confusión y para hacer más fácil la investigación futura, se ha revisado la terminología utilizada en la literatura y se han propuesto cuatro términos que tratan de captar la esencia de los diferentes tipos de software que aparecen en la literatura. Estos tipos son: código, biblioteca, caja de herramientas y suite.

El código recoge las contribuciones en las que solo algoritmos de sistemas difusos escritos en un lenguaje de programación se comparten, por ejemplo, una aplicación inteligente de una cierta clase de algoritmos, el grupo de investigación NNandFS de la Universidad de Magdeburg ofrece documentación y código de Redes Neuronales y Sistemas Difusos (Magdeburg, 2013), Lin y Chen publicaron bibliotecas fáciles de entender y reutilizar para aquellos investigadores que deseen desarrollar y personalizar sus propias aplicaciones Java para sistemas difusos expertos (Lin, 2012).

La biblioteca hace referencia a los grupos en el que un conjunto de funciones se presentan con una funcionalidad relacionada (código encapsulado), escrita en términos de un lenguaje y

de la interfaz de programación de aplicaciones (API) que facilita su integración con software diferente. Se pueden distinguir dos grupos de bibliotecas, el primer grupo incluye bibliotecas difusas que ya están integradas con otras de ingeniería y matemáticas, por ejemplo: DANA-FLSA (Foscarini, 2010) es una biblioteca para el análisis de sensibilidad de los sistemas difusos que resulta de la integración de la biblioteca SimLab en la evaluación y análisis de datos (DANA); FWS (Servicio Web difuso) (Javor, 2012) es una biblioteca que utiliza un motor de inferencia difuso basado en el contexto de CASANDRA (Sistema de Simulación de adaptación Consciente para aplicaciones en numerosas áreas); MODELICA-ANFIS (Fernandez, 2013), desarrolla el esquema de inferencia neuro difuso adaptativo que está dirigido a modelar sistemas complejos. En contraste con el grupo anterior, el segundo grupo considera bibliotecas para el desarrollo de sistemas difusos de distribución libre, los lenguajes de programación para su desarrollo están C ++, Java, Python y R.

Las herramientas o caja de herramientas incluyen las contribuciones que proponen FSS para realizar tareas útiles con sistemas difusos. Este FSS puede utilizar una o más bibliotecas y puede ser ejecutado como un programa autónomo a través de línea de comandos, interfaz gráfica de usuario (GUI) o incluso interfaz web. Existen varias herramientas en las áreas de minería de datos y reconocimiento de patrones, entre ellos: FUAT (Bozkir, 2013) tiene por objeto el análisis de agrupamiento difuso; NIP (Cadenas, 2013) maneja la información imperfecta en los conjuntos de datos; CI-LQD (Palacios, 2013) se refiere a la extracción de conocimiento a partir de conjuntos de datos de baja calidad, que automatiza los cálculos involucrados en las comparaciones estadísticas de diferentes algoritmos, con ambas técnicas numéricas y gráficas; SAMT (Wieland, 2011) combina el análisis de datos estadísticos con la formación semi automática de reglas difusas; FID (Abu, 2009) genera árboles de decisión difusos de un conjunto de datos; VisualFCM (Glykas, 2013) apoya la generación y visualización de mapas cognitivos difusos; FingramsGenerator (Pancho, 2013) se ocupa de la generación, la representación visual y análisis de reglas difusas representadas como redes sociales. Con relación al modelado difuso, algunas de las herramientas que existen son: GUAJE (Alonso, 2011) se refiere al diseño de sistemas difusos interpretables, que combina el conocimiento experto con conocimiento extraído automáticamente de los datos. Fispro (Guillaume, 2012) también se destaca por el cuidado que garantiza la interpretación de los sistemas difusos aprendidos de forma automática a partir de datos y es la única herramienta que implementa reglas implicativas graduales (Jones, 2009). Xfuzzy (Baturone, 2007) ofrece varios módulos complementarios para hacer frente al diseño completo de sistemas difusos en C, C ++ o Java (para formar parte de proyectos de software) o incluso en VHDL (para proyectos de hardware). E-Fuzz (Chantrapornchai, 2010) se centra en el diseño de sistemas difusos embebidos. Otra área de investigación en la que se han propuesto varias herramientas

es el razonamiento lógico. Algunos ejemplos son: KIRQ (Rubinson, 2013) que se extiende para dar cabida a las condiciones contradictorias; VisualFLOPER (Morcillo, 2010) que proporciona el entorno de programación lógica difusa para la Investigación (FLOPER) con una interfaz gráfica; RFuzzy (Munoz, 2011), que ofrece la representación del conocimiento, el modelado y el razonamiento con información difusa basada en Prolog. En el área de toma de decisiones esta : FuzzME (Holecek, 2010) considera criterios cuantitativos y cualitativos para la agregación de las evaluaciones parciales en el contexto de la toma de decisiones difusas; MaxAgr (Vrana, 2012) se centra en la toma de decisiones en grupo, ayudando en la agregación de opiniones heterogéneas proporcionados por un conjunto de expertos. Finalmente analicemos las herramientas de R y Matlab. Tenga en cuenta que, a pesar de que Matlab se distribuye comercialmente, las herramientas que se enumeran a continuación son liberadas por sus autores. Algunos ejemplos de Matlab son: IT2FLS (Castillo, 2013) que apoya el diseño de tipo-2 para sistemas difusos por intervalo; XTRIG (Liu, 2009) que realiza la trigonometría cualitativa difusa en términos de números difusos; Fuzzy Calculus CORE (Zahariev, 2009) que es compatible con el álgebra difusa y se ocupa de los sistemas lineales difusos de ecuaciones y desigualdades. Un ejemplo de R como herramienta es fuzzy toolbox (Wagner, 2009) que se espera que sea equivalente a la de Matlab (MathWorks, 2013) con la característica de ser de distribución libre y código abierto.

El último grupo, suite está relacionado con las aportaciones en las que se presentan los marcos más grandes que permiten utilizar (entre otras características) una colección de herramientas o bibliotecas de funcionalidad relacionada, considerando una interfaz de usuario común y cierta capacidad para intercambiar datos entre sí. KEEL (Alcala, 2009) es probablemente la suite más conocida para la extracción de conocimiento a partir de aprendizaje evolutivo. Incluye una amplia variedad de algoritmos difusos para la regresión, clasificación, agrupamiento, patrones en minería de datos. Además permite un análisis completo de los modelos de aprendizaje evolutivo en comparación con herramientas de software existentes para llevarse a cabo. Ofrece varios módulos de investigación y enseñanza. Otras dos suites bien conocidas son Weka (Witten, 2011) y KNIME (Berthold, 2013). Weka incluye algunos algoritmos de aprendizaje de reglas difusas que pueden ser analizados junto con una amplia colección de algoritmos de aprendizaje automático para tareas de minería de datos. KNIME ofrece algunos plugins para sistemas difusos que son principalmente dedicados a la lógica difusa, aunque también se pueden utilizar para clasificadores basados en reglas difusas de conjuntos de datos.

Los sistemas difusos son una de las áreas más importantes para la aplicación de la teoría de conjuntos difusos (Dubois, 1980). Usan la lógica difusa para proporcionar un marco conceptual para la representación del conocimiento en virtud de la imprecisión y la consiguiente

incertidumbre (Zade, 1980). Los sistemas difusos se han aplicado con éxito en muchos campos de aplicación (Por ejemplo, el control (Zhao, 2012), el modelado (Gacto, 2014) (Pedrycz, 1999), la clasificación (Ishibuchi, 2013), la minería de datos (Alcala, 2009), etc.) debido a su capacidad para incorporar el conocimiento experto humano y la computación granular (Yao, 2013), para manejar la imprecisión y la incertidumbre y para describir el comportamiento de los sistemas complejos sin necesidad de un modelo matemático preciso.

Hoy en día el lenguaje de programación R (Ihaka, 2009) está ganando importancia como una solución no comercial que es competitivo con Matlab. Algunas de las bibliotecas de investigación más destacados (normalmente llamados paquetes por parte de la comunidad R) incluyen los siguientes: FRBS (Riza, 2015) que implementa la funcionalidad y algoritmos para construir y utilizar sistemas basados en reglas difusas; SETS (Meyer, 2009) que maneja conjuntos generalizados y personalizables (Incluyendo conjuntos difusos); Por último SAFD (Trutschnig, 2013) que proporciona herramientas básicas para las estadísticas elementales con datos difusos.

## 1.5. MARCO DE TRABAJO

El presente trabajo de tesis tiene como soporte diversos proyectos que se gestan en el Cuerpo Académico de Ingeniería de Software para la Informática Educativa, el cual tiene como objetivo la investigación e innovación de los procesos de desarrollo, operación y mantenimiento de software, así como en las aplicaciones educativas para asistir los procesos de enseñanza y de aprendizaje. Sus líneas de investigación se enfocan en:

- Ingeniería de Software: Esta línea tiene como propósito el estudio y aplicación de enfoques sistemáticos, disciplinados y cuantificables para el desarrollo, operación y mantenimiento de software.
- Informática Educativa: Esta línea tiene como propósito el estudio del uso, efectos y consecuencias de las tecnologías de la información en el proceso educativo.

## 1.6. APORTACIONES

Las contribuciones del trabajo de investigación se enumeran a continuación en dos grupos: aportaciones teóricas, en la forma de modelos y metodologías, y aportaciones prácticas, dadas mediante componentes software.

### 1.6.1. Aportaciones teóricas

- Metodología estadística para evaluar un sistema web mediante técnicas no invasivas.
- Metodología estadística para evaluar el rendimiento de un sistema web difuso.
- Metodología estadística para evaluar modelos difusos cuando la variable de salida sea de tipo nominal (discreto) y otra en escala de razón (continua).
- Algoritmo para evaluar modelos difusos cuando la variable de salida sea de tipo nominal (discreto) y otra en escala de razón (continua).

### 1.6.2. Aportaciones prácticas

- Un motor estadístico que genera reglas, gráficas y predicciones de modelos de inferencia difusa.
- Una herramienta computacional para evaluar y generar modelos de inferencia difusa utilizando funciones de pertenencia triangular, trapezoidal y gaussiana.

## 1.7. ORGANIZACIÓN DEL DOCUMENTO

En los capítulos de esta tesis se describen distintos aspectos de la investigación realizada: revisión documental, trabajos relacionados, metodologías, implementaciones y validaciones. A continuación se detallan los contenidos de los distintos capítulos:

1. **Introducción.** Inicia indicando el marco de la investigación, la hipótesis y los objetivos perseguidos en la realización de esta investigación. Se establece el método de trabajo y el espacio donde se enmarca en términos del grupo de investigación y los proyectos relacionados. Por último, se exponen las aportaciones de esta investigación.
2. **Estado del arte.** Presenta una revisión documental relativa a las temáticas más relacionadas y que resultan primordiales para la investigación desarrollada en este trabajo: conceptos de lógica difusa, sistemas de inferencias difuso, sistema difuso basado en reglas, representación del conocimiento, aprendizaje, tablas de repertorio difuso y finalmente las aplicaciones de los sistemas de inferencia difuso.
3. **Modelos matemáticos en la lógica difusa.** Detalla el modelo basado en sistemas difusos estáticos, con un factor de peso, con criterios de clasificación, aprendizaje difuso, prueba de desempeño para variable de salida de tipo nominal (discreto) y otra en escala de razón (continua).

4. **Implementación.** Presenta la implementación de la metodología descrita en el capítulo previo. Se describe la aplicación del motor de inferencia difuso, con las especificaciones técnicas para evaluar y generar modelos de inferencia difuso.
5. **Experimentación.** Describe el diseño de los distintos experimentos realizados para la validación de las propuestas presentadas. Se establecen los criterios para evaluar el rendimiento y calidad del motor de inferencia difuso. Posteriormente se analizan y discuten los resultados computacionales y estadísticos obtenidos al ejecutar los algoritmos.
6. **Conclusiones.** Describe las principales aportaciones de esta tesis, el cumplimiento de los objetivos planteados y las principales líneas de trabajo futuro que se abren a partir de esta investigación.





# Capítulo 2

## Estado del Arte

### 2.1. LÓGICA DIFUSA

La lógica difusa se estudió por primera vez en la Universidad de Berkely a mediados de los años sesenta. Fue Lofty A. Zadeh (brillante ingeniero iraní nacionalizado en Estados Unidos) quien introdujo el concepto de conjunto difuso (fuzzy set), que se basa en la idea de que los elementos sobre los que se construye el pensamiento humano no son números ni valores de verdad excluyentes, sino etiquetas lingüísticas. La lógica difusa permite representar el conocimiento común, que en gran medida es de tipo lingüístico cualitativo, a partir de los conjuntos difusos y las funciones características asociadas a ellos (Rosanna, 2005).

El aspecto central de los sistemas basados en lógica difusa es que tienen la capacidad de reproducir los modos usuales de razonamiento considerando que la certeza de una proposición es una cuestión de grado. De este modo podemos considerar la lógica difusa como una extensión de la lógica clásica que engloba a ésta, siendo la lógica clásica el caso extremo de la lógica difusa (en que los grados de certeza alcanzan sus límites, 0 y 1). La Lógica difusa se puede inscribir en el contexto de la Lógica Multivaluada. En 1922 Lukasiewicz cuestionaba la Lógica Clásica bivaluada (valores cierto y falso). Además, adelantaba una lógica de valores ciertos en el intervalo unidad como generalización de su lógica trivaluada. En los años 30 fueron propuestas lógicas multivaluadas para un número cualquiera de valores ciertos (igual o mayor que 2), identificados mediante números racionales en el intervalo  $[0,1]$  (Jose, 2007).

Uno de los objetivos de la Lógica difusa es proporcionar las bases del razonamiento aproximado que utiliza premisas imprecisas como instrumento para formular el conocimiento.

En un conjunto clásico (crisp) se asigna el valor 0 ó 1 a cada elemento para indicar la pertenencia o no a dicho conjunto. Esta función puede generalizarse de forma que los valores asignados a los elementos del conjunto caigan en un rango particular, y con ello indiquen el grado de pertenencia de los elementos al conjunto en cuestión (Jose, 2007).

### 2.1.1. Función de pertenencia

La función de pertenencia  $\mu_A$  por la que un conjunto difuso A se define, siendo  $[0, 1]$  el intervalo de números reales que incluye los extremos, tiene la forma (Jose, 2007):

$$\mu_A = X \longrightarrow [0, 1] \quad (2.1)$$

Es decir, mientras que en un conjunto clásico los elementos pertenecen o no pertenecen a él totalmente (por ejemplo un número puede pertenecer o no al conjunto de los pares, pero no pertenecerá con un determinado grado), en los conjuntos difusos hay grados de pertenencia en referencia a un universo local. Por ejemplo, en el contexto de nuestra sociedad actual, una persona de 45 años pertenecerá al conjunto difuso “viejo” con un grado, supongamos de 0.5. Si en vez de usar de referencia nuestra sociedad actual aludimos a una sociedad donde la esperanza de vida fueran 40 años este grado cambiaría.

Originalmente la teoría de conjuntos difusos se formuló en base a un conjunto de operadores también válidos para conjuntos clásicos (Jose, 2007):

- Negación:  $\mu_{\neg A}(x) = 1 - \mu_A(x)$
- Unión:  $\mu_{A \cup B}(x) = \max[\mu_A(x), \mu_B(x)]$
- Intersección:  $\mu_{A \cap B}(x) = \min[\mu_A(x), \mu_B(x)]$
- A será un Subconjunto Borroso de B cuando:  $\mu_A(x) \leq \mu_B(x), \forall x \in X$

Por último, existe un principio que permite la generalización de conceptos matemáticos concisos a la teoría de Conjuntos Difusos. Cualquier función que asocie puntos  $x_1, x_2, \dots, x_n$  del conjunto crisp X al Y puede generalizarse de forma que asocie subconjuntos difusos de X en Y, es el denominado “Principio de extensión” (Jose, 2007).

### 2.1.2. La representación borrosa del conocimiento

En lenguaje natural se describen objetos o situaciones en términos imprecisos: grande, joven, tímido, etc. El razonamiento basado en estos términos no puede ser exacto, ya que normalmente representan impresiones subjetivas, quizá probables pero no exactas. Por ello, la Teoría de Conjuntos Borrosos se presenta más adecuada que la lógica clásica para representar el conocimiento humano, ya que permite que los fenómenos y observaciones tengan más de dos estados lógicos. Para la construcción de Conjuntos Borrosos para ser usados en Sistemas Inteligentes son necesarias técnicas específicas de Adquisición de Conocimiento. Las más usadas son las entrevistas y formularios, pero parece adecuado adaptar otras técnicas al campo

Borroso (Jose, 2007). En los Sistemas Basados en el Conocimiento la función de pertenencia debe ser obtenida del experto en ese dominio de conocimiento. Esta función no ha de ser confundida con una función de distribución de probabilidad basada en la repetición de las observaciones, sino en la opinión del experto. La representación habitual del conocimiento en términos difusos se realiza por medio de reglas, del tipo:

$$\text{Si } x_1 \text{ es } A_{1,1} \text{ y/o } x_2 \text{ es } A_{2,1} \text{ y /o } x_n \text{ es } A_{1,n} \text{ Entonces } y \text{ es } B_1 \quad (2.2)$$

Cada variable que interviene como hipótesis en una regla tiene asociado un dominio. Cada dominio puede estar dividido en tantos Conjuntos Borrosos como el experto considere oportuno. Cada una de estas particiones tiene asociada una Etiqueta Lingüística.

Un conjunto de terminos (term-set, Fig. 2.1) es un conjunto finito, prioritariamente con  $7 \pm 2$  elementos, que son restricciones de una variable lingüística borrosa. Este conjunto de elementos debe ser suficiente para describir cualquier situación relativa al contexto en el que se sitúa el problema (Jose, 2007).

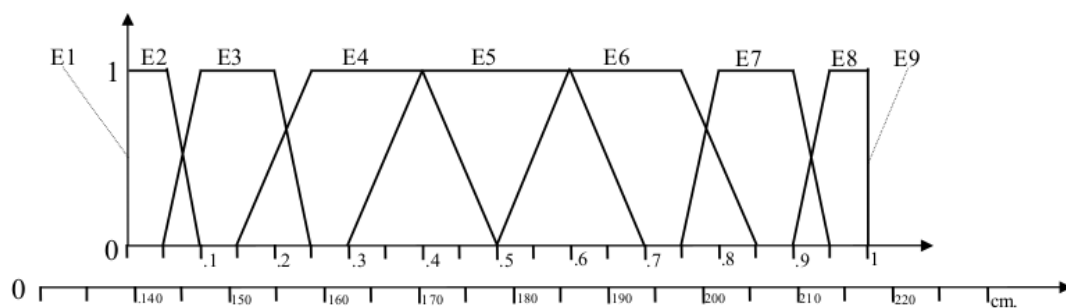


Figura 2.1 Representación de un term set.

Por ejemplo, el siguiente conjunto de términos pretende reflejar una descripción estándar de lo que se entiende por “altura” (referida a personas) (Ver tabla 2.1). El Universo de discurso (alturas) está normalizado entre 0 y 1 aunque refleja, por ejemplo, entre 130 y 230 cm.

Tabla 2.1 Tabla difusa de Longitud.

Valores Lingüísticos	(a, b, c, d)
E1: Bajísimo	(0, 0, 0, 0)
E2: Muy bajo	(0, 0, 0.05, 0.1)
E3: Bastante bajo	(0.05, 0.1, 0.2, 0.25)
E4: Ligeramente bajo	(0.15, 0.25, 0.4, 0.5)
E5: Normal	(0.3, 0.4, 0.6, 0.7)
E6: Ligeramente alto	(0.5, 0.6, 0.75, 0.85)
E7: Bastante alto	(0.75, 0.8, 0.9, 0.95)
E8: Muy alto	(0.9, 0.95, 1, 1)
E9: Altísimo	(1, 1, 1, 1)

### 2.1.3. Razonamiento aproximado

Uno de los objetivos de la Lógica difusa es proporcionar las bases del razonamiento aproximado que utiliza premisas imprecisas como instrumento para formular el conocimiento. Zadeh (Jose, 2007) introdujo la teoría del razonamiento aproximado y otros muchos autores han hecho contribuciones importantes a este campo. Aunque superficialmente pueda parecer que la teoría del razonamiento aproximado y la lógica clásica se diferencian enormemente, la lógica clásica puede ser vista como un caso especial de la primera. En ambos sistemas, se pueden ver a las premisas como inductoras de subconjuntos de mundos posibles que las satisfacen, aunque en el caso de la teoría del razonamiento aproximado esos conjuntos serán subconjuntos difusos. La inferencia en ambos sistemas está basada en una regla de inclusión: una hipótesis se infiere de una colección de premisas si el subconjunto de mundos posibles que satisfacen la conjunción de las premisas está contenido en el subconjunto de mundos posibles que satisfacen la hipótesis. La contribución fundamental del razonamiento aproximado es el uso que hace de las variables y la representación de las proposiciones en términos de valores de verdad lingüísticos - subconjuntos difusos- como valores de esas variables. La lógica clásica sólo usa de modo implícito de idea de variable, en el sentido de valor de verdad asociado a una proposición. Sin embargo, su naturaleza binaria le permite ocultar este hecho, ya que nos podemos referir a una proposición que es verdadera por su denotación,  $p$ , y a una que es falsa simplemente por su negación,  $\neg p$ , evitando así la introducción de una variable  $V_p$  cuyo valor sea la valoración de la proposición  $p$ . El uso del concepto de variable en la teoría del razonamiento aproximado conduce a tratar dominios que no están dentro del ámbito de la lógica clásica, como es el caso de los problemas que tratan los Sistemas Expertos Difusos o los Controladores Difusos. La teoría del razonamiento aproximado permite representar también cuantificadores lingüísticos situados entre el "para todo" y el "existe" clásicos. Esto

facilita representar enunciados como "la mayoría de los coches lujosos son caros" o "bastantes electores votarán en blanco". Zadeh indicó que un cuantificador como "la mayoría" puede ser representado como un subconjunto difuso sobre un universo de discurso (Jose, 2007). Los cuantificadores aproximados se usan para representar conocimiento de sentido común. Una extensión interesante de la teoría del razonamiento aproximado es la posibilidad de tratar con ella conocimiento prototípico. Reiter sugirió una aproximación a la representación de conocimiento de sentido común usando reglas por defecto y Yager lo estudió en el marco de la teoría del razonamiento aproximado (Jose, 2007). De acuerdo con Reiter (Jose, 2007), una regla por defecto tal como "típicamente los pájaros vuelan", puede ser interpretada así: si un objeto es un pájaro y nuestro conocimiento disponible no es incompatible con que el objeto vuele, entonces asumimos que el pájaro vuela.

La lógica binaria puede ser vista como un caso especial de la teoría del razonamiento aproximado en el cual los conjuntos base tienen dos elementos  $\{T, F\}$  y los grados de pertenencia se restringen a 1 o 0 (Jose, 2007). La lógica posibilística puede ser vista como una extensión de ésta, en tanto que, aunque se restringen los conjuntos base de valores a dos,  $T$  y  $F$ , se permiten que los grados de pertenencia sean números en el intervalo unidad. La Lógica difusa extiende la lógica binaria permitiendo su formalización en términos de la teoría del razonamiento aproximado. Así,  $p$  es verdadero alcanzaría la representación  $V_p$  es  $\{1/T, 0/F\}$ ,  $p$  es falso,  $V_p$  es  $0/T, 1/F$  y  $V_p$  es  $1/T, 1/F$  indica que el valor de verdad de la proposición es desconocido. En cualquiera de los casos, el conjunto base asociado a la variable valor de verdad de la proposición  $p$  es  $\{T, F\}$ . La regla principal de inferencia en lógica clásica, modo de razonamiento ya introducido por los Megáricos y Estoicos en tiempos de Aristóteles, es el Modus Ponens (nombre asignado en la Edad Media), que consiste en que si se tiene la regla  $A \rightarrow B$ , y se da el hecho  $A$ , se puede concluir  $B$ ; por ejemplo, si la regla es "Si llueve entonces me mojo" si se da el hecho cierto de que "llueve", entonces podré concluir que "me mojo". En lógica borrosa se puede generalizar esta regla, quedando su esquema de la siguiente forma (ver 2.3):

$$\begin{aligned}
 &\text{REGLA : Si } x \text{ es } A \text{ Entonces } y \text{ es } B \\
 &\text{HECHO: } x \text{ es } A' \\
 &\text{CONCLUSION: } y \text{ es } B'
 \end{aligned}
 \tag{2.3}$$

Por ejemplo, la regla podría ser "Si la ciudad es grande ( $x$  es  $A$ ), el tráfico es muy denso ( $y$  es  $B$ )", el hecho podría ser "la ciudad no es muy grande ( $x$  es  $A'$ )", ¿Qué se podría decir del tráfico ( $B'(x)$ ). Supongamos que las variables están relacionadas no necesariamente por una función, sino por cualquier relación. Supongamos que es una relación binaria borrosa  $R$

en el universo  $X \times Y$ .  $A'$  y  $B'$  son conjuntos difusos en  $X$  e  $Y$  respectivamente. Si conocemos  $R$  y  $A'$  podríamos conocer  $B'$  mediante la denominada **Regla composicional de inferencia**:  $B' = A'(x) R(x, y) B'(y) = \sup_{x \in X} \min[A'(x), R(x, y)]$  Donde  $R(x, y) = I(A(x), B(y))$  (Función de Implicación) (Jose, 2007).

## 2.2. SISTEMAS DE INFERENCIA DIFUSO

Los sistemas difusos basados en reglas para tareas de la clasificación y regresión (FRBS) se basan en el concepto de conjuntos difusos, propuesta por Zadeh en 1965, que tiene por objeto representar el razonamiento humano en un conjunto de reglas SI-ENTONCES para manejar los problemas de la vida real. Por ejemplo el control, la predicción e inferencia, minería de datos, procesamiento de datos en bioinformática y robótica. También se conocen como sistemas de inferencia difusos y modelos difusos. Durante el modelado de un FRBS, hay dos pasos importantes que deben llevarse a cabo: identificación de la estructura y estimación de parámetros. Generalmente FRBS se puede modelar de cuatro partes funcionales :

- a) Un difusor que transforma las entradas en datos en su correspondiente función de pertenencia de la expresión lingüística de cada variable.
- b) Una base de conocimiento que consiste de un conjunto de datos y reglas. Mientras que los conjuntos de datos incluyen las definiciones de conjuntos difusos, las reglas difusas del tipo SI - ENTONCES que representan el conocimiento tienen la siguiente estructura :  
*SI la premisa (antecedente) ENTONCES conclusión (consecuente)*
- c) Un motor de inferencia que realiza las operaciones de inferencia sobre las reglas difusas SI - ENTONCES. Hay dos tipos de inferencia para sistemas difusos basados en reglas lingüísticas: El Mamdani y el modelo Takagi Sugeno Kang.
- d) Un proceso desdifusor para obtener los valores nítidos de los valores lingüísticos. Hay varios métodos para el proceso desdifusor tales como la media ponderada, centroide, etc.

Ya que puede ser difícil obtener información de los humanos en la forma requerida, una forma alternativa y eficaz para adquirir el conocimiento es generar reglas difusas del tipo SI - ENTONCES a partir de los datos de entrenamiento numéricos.

### 2.2.1. Definición y estructura

La base de conocimientos de un controlador difuso consiste en una colección de reglas que describen las acciones de control. El rendimiento del control difuso depende en gran medida de si las normas de control son razonables o no (Herrera, 1995).

Las posibilidades de aplicación de los algoritmos evolutivos en el campo de la lógica difusa está documentada por una serie de publicaciones recientes de investigación, que a grandes rasgos permite optimizar las funciones de pertenencia de los conjuntos difusos, además de optimizar el aprendizaje automático de las reglas difusas (Thomas, 1995).

Las funciones de pertenencia difusas proporcionan la caracterización de conjuntos difusos mediante el establecimiento de una conexión entre términos lingüísticos (como "lenta", "media", "rápida" para una variable de velocidad) y los valores numéricos precisos de variables en un sistema físico. Una función de pertenencia difusa se aproxima a la confianza con la que un valor numérico es descrito por un término lingüístico, la correcta elección de las funciones de pertenencia; sin embargo, no es en absoluto trivial pero juega un papel crucial en el éxito de una aplicación (Thomas, 1995).

La determinación de los parámetros correctos para un controlador de lógica difusa es un problema complejo y es también una tarea que consume tiempo considerable (Dorigo, 2004).

La modelación difusa, tiene impacto práctico en el control, la predicción y la deducción (Takagi, 1985). Las tablas de repertorio difuso representan una matriz rectangular con elementos (columnas) y constructos (filas). Cada intersección fila-columna contiene una razón, la cual consiste de un conjunto de funciones triangulares, trapezoidales y gaussianas (ver Ec. 2.4, 2.5 y 2.6) que muestran cómo un usuario aplica un constructo dado a un elemento en particular (Castro, 2004). Las funciones triangulares, trapezoidales y gaussianas extienden las tablas de repertorio difuso a formar restricciones, las cuales pueden ser representadas en una rejilla en orden para darle mayor poder a la representación, mientras se le permite al usuario expresar su conocimiento sin restricciones.

$$f(u; a, b, c) = \begin{cases} 0 & u < a \\ (u - a)/(b - a) & a \leq u < b \\ (c - u)/(c - b) & b < u \leq c \\ 0 & u > c \end{cases} \quad (2.4)$$

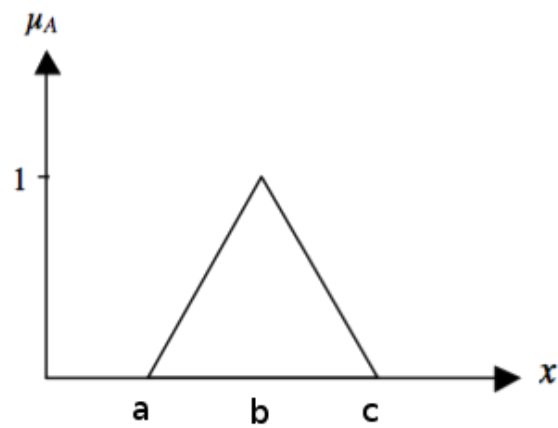


Figura 2.2 Gráfica de función de pertenencia triangular.

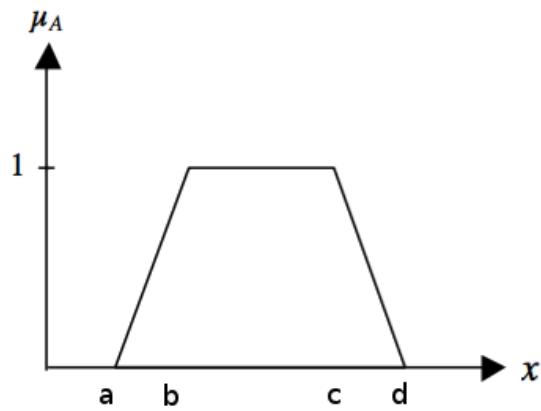


Figura 2.3 Gráfica de función de pertenencia trapezoidal.

$$f(u; a, b, c, d) = \begin{cases} 0 & u < a \\ (u-a)/(b-a) & a \leq u < b \\ 1 & b \leq u \leq c \\ (d-u)/(d-c) & c < u \leq d \\ 0 & u > d \end{cases} \quad (2.5)$$

$$f(u; a, b) = e^{-\frac{1}{2}\left(\frac{u-a}{b}\right)^2} \quad a = \mu, b = \sigma \quad (2.6)$$



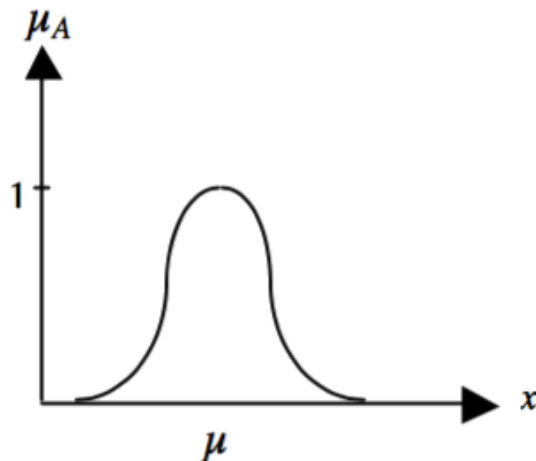


Figura 2.4 Gráfica de función de pertenencia gaussiana(a = media, b = desviación estándar).

## 2.3. SISTEMA DIFUSO BASADOS EN REGLAS

### 2.3.1. Notación

Las reglas difusas SI - ENTONCES para  $c$  patrones de clasificación con  $n$  atributos tiene la notación siguiente (Takagi, 1985) (ver 2.7):

$$\begin{aligned} \text{Una Regla } R_j : & \text{ Si } x_1 \text{ es } A_{j1} \text{ y } \dots \text{ y } x_n \text{ es } A_{jn} \\ & \text{ entonces } C_j, j = 1, 2, \dots, N \end{aligned} \quad (2.7)$$

donde :

$x = (x_1, \dots, x_n)$	vector patrón de dimensión $n$
$A_{ji}$	valor de la lingüística antecedente ( $i = 1, 2, \dots, n$ )
$C_j$	clase consecuente
$N$	número de reglas difusas del tipo SI - ENTONCES

Para problemas con función de aproximación (ver 2.8), las reglas difusas SI - ENTONCES con números reales como valores consecuentes frecuentemente se usa la siguiente notación:

$$\begin{aligned} \text{Una Regla } R_j : & \text{ Si } x_1 \text{ es } A_{j1} \text{ y } \dots \text{ y } x_n \text{ es } A_{jn} \\ & \text{ entonces } y \text{ es } r_j, j = 1, 2, \dots, N \end{aligned} \quad (2.8)$$

donde  $y$  es la variable de salida y  $r_j$  el consecuente número real.

Para estimar el valor de salida  $\hat{y}$  de un vector de entrada  $x = (x_1, \dots, x_n)$  se calcula como el promedio de los pesos utilizando la siguiente fórmula (ver 2.9):

$$\hat{y} = \frac{\sum_{j=1}^N \mu_j(x) \cdot r_j}{\sum_{j=1}^N \mu_j(x)} \quad (2.9)$$

donde  $r_j$  es el consecuente número real y  $\mu_j(x)$  es el grado de compatibilidad de la regla  $R_j$ . El grado de compatibilidad  $\mu_j(x)$  se calcula como el operador producto  $\mu_j(x) = \mu_{j1}(x_1) \times \dots \times \mu_{jn}(x_n)$  donde  $\mu_{ji}(\cdot)$  es la función de pertenencia de la lingüística antecedente  $A_{ji}$  ( $i = 1, \dots, n$ ).

### 2.3.2. Procesos en los sistemas difusos

En general cuando se modela un sistema de inferencia difuso hay dos procesos importantes que deben llevarse a cabo, la identificación de la estructura y estimación de parámetros. La identificación de la estructura es un proceso para encontrar reglas difusas apropiadas y para determinar el número total de reglas.

La estimación de parámetros se aplica para ajustar los parámetros de la antecedente y consiguiente parte de las reglas difusas. Se han propuesto muchos enfoques con el fin de realizar este modelado por ejemplo, como un esquema de tabla de consulta, procedimientos heurísticos, técnicas neuro-difusa, métodos de agrupamiento, algoritmos genéticos, métodos de mínimos cuadrados, descenso de gradiente, etc. Los enfoques mas comunes para generar reglas difusas SI-ENTONCES son los siguientes:

1. Basado en la partición del espacio (Riza, 2015).

- Wang y Mendel técnica (WM): se utiliza para resolver tareas de regresión.
- La técnica de Chi (RBCS.CHI): se utiliza para resolver las tareas de clasificación.
- Técnica Ishibuchi que emplea un factor de peso (FRBCS.W): Se utiliza para resolver la clasificación.

2. Basado en redes neuronales (Riza, 2015).

- El sistema de inferencia difuso basado en la adaptación de la red (ANFIS en ingles): se utiliza para resolver tareas de regresión.
- El sistema inferencia difuso neuronal híbrida (FIS): se utiliza para resolver tareas de regresión.

## 2.4. REPRESENTACIÓN DEL CONOCIMIENTO

La representación del conocimiento se utiliza para clasificar y procesar conceptos en un sistema de información. La resolución de problemas puede ser simplificada con la elección apropiada de representación del conocimiento. Si bien representar el conocimiento de una manera hace la solución más simple, elegir una representación inadecuada puede hacer que la solución sea difícil (Chico, 2008).

### 2.4.1. Representaciones básicas

- a) Reglas de producción. En este modelo parte del conocimiento humano se representa en forma de reglas producciones o reglas de producción. Los expertos tienden a expresar sus técnicas de solución de problemas en forma de reglas situación – acción. El conocimiento representado por la regla de producción es relevante para la línea de razonamiento seguida sólo si la condición de la regla es cierta; consecuentemente, se realiza la acción definida por la regla.
- b) Redes semánticas. Una red semántica está compuesta por nodos, los cuales se relacionan entre sí por medio de enlaces. Los nodos representan objetos o conceptos del dominio del problema y los enlaces sus relaciones o asociaciones. Se llama red semántica porque originalmente se usaron para representar el sentido en expresiones del lenguaje natural. Cada nodo se considera la cabeza de un plano que tiene su definición (si una palabra tiene tres significados, entonces existirán 3 planos para ella). Los enlaces en el plano representan su definición. Pueden existir apuntadores a superclases, modificaciones, disyunciones y conjunciones.
- c) Marcos. Son estructuras de datos que representan situaciones prototípicas. Una de las ideas intuitivas detrás de los marcos es que la memoria se basa mucho en estereotipos (propiedades típicas de los objetos). La idea es tener una sola estructura de datos para poner el conocimiento relevante acerca de una clase de objetos, en lugar de tener el conocimiento distribuido en forma de reglas o fórmulas lógicas.

### 2.4.2. Representaciones avanzadas

- a) Modelos cualitativos, temporales, causales. Describen objetos y procesos en términos de sus relaciones temporales, espaciales y causales.
- b) Sistemas híbridos. Emplean, en paralelo, una combinación de modelos de inteligencia artificial, métodos y técnicas de éstos subcampos.

- c) Razonamiento basado en casos. Se ocupa del estudio de los mecanismos mentales necesarios para repetir lo que se ha hecho o vivido con anterioridad, ya sea por experiencia propia o por casos concretos recopilados en la bibliografía o en la sabiduría popular.
- d) Redes neuronales. Inspirado en la forma en que funciona el sistema nervioso de los animales. Se trata de un sistema de interconexión de neuronas en una red que colabora para producir un estímulo de salida.

### **2.4.3. Representación de incertidumbre**

El modelo ideal del razonamiento (humano o mecánico) es el razonamiento exacto pero, por lo general, el conocimiento que se maneja en la mayoría de los dominios susceptibles de ser tratados por un SBC(Sistema Basado en el Conocimiento) no es de naturaleza determinista (Chico, 2008). En el mundo real se suele razonar con información que es incierta y/o imprecisa, ya sea por deficiencias en la información o por las propias características del mundo real. Entre los enfoques que se han propuesto para trabajar con incertidumbre destacan el uso de factores de certeza, lógica difusa y redes probabilísticas.

#### **2.4.3.1. Factores de certeza**

La limitación que se tiene en la mayoría de los casos prácticos de no disponer de una gran cantidad de datos históricos obligó al desarrollo de un método de inferencia aproximado, que en general proporcione resultados exactos aún en cuando estén basados en datos limitados. A este se lo denominó el formalismo del factor de certeza (Chico, 2008). Los resultados son más o menos ciertos en función de la certeza o falsedad de los hechos y conocimientos utilizados. Los factores de certeza que van asociados a los predicados, por lo general pueden ir de 0 a 100. El procedimiento a seguir en el modelo aproximado es el siguiente:

- a) El factor de un conjunto de condiciones unidas por el operador lógico Y es igual al mínimo de los factores que intervienen.
- b) El factor de un conjunto de condiciones unidas por el operador lógico O es igual al máximo de los factores que intervienen.
- c) El factor de certeza de una conclusión es igual al producto del factor de certeza de las condiciones por el de la regla.
- d) Para el cálculo del coeficiente resultante de la totalidad de reglas que se han encadenado en la deducción de un hecho, se aplican la siguiente relación (ver 2.10):

$$F_{final} = F_1 + F_2 - (F_1 * F_2) \quad (2.10)$$

#### 2.4.3.2. Lógica difusa

La lógica difusa es una extensión de la lógica matemática basada en la teoría de conjuntos difusos y sistemas lógicos de varios valores infinitos. La lógica difusa se ha convertido en un campo de la investigación y el desarrollo importante en muchas disciplinas como las matemáticas, la lógica, la inteligencia artificial y la filosofía desde 1965 (Zhaohao, 2007).

La arquitectura básica de un sistema basado en lógica difusa, es la que se muestra en la figura 2.5 (McNeill, 1994; Olivas, 2007; Pérez-Pueyo, 2005; Kosko, 1997; Slany, 1996).

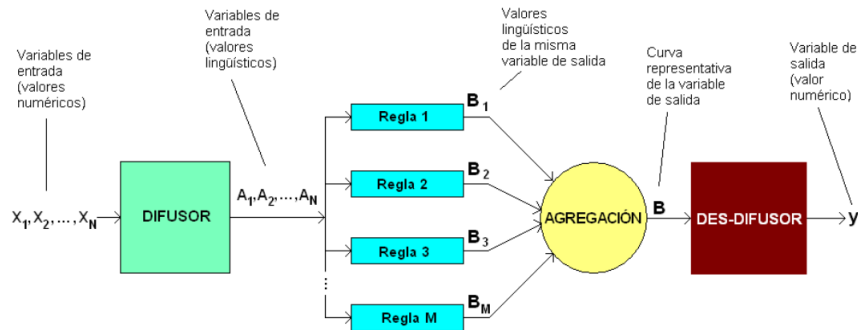


Figura 2.5 Sistema basado en lógica difusa.

Los componentes del sistema son los siguientes ( Jang y Gulley, 2007 ) :

1. Un bloque difusor al que cada variable de entrada se le asignará un valor entre 0 y 1, en función de su grado de pertenencia a los conjuntos difusos que se están considerando.
2. Un bloque de inferencia con reglas del tipo:  
(**Condición 1**) y (**Condición 2**) y ... y (**Condición N**) Entonces (**Salida**) que relacionan los conjuntos difusos de entrada con los de salida.
3. Un bloque de agregación lógica, que a partir de los conjuntos difusos obtenidos a la salida del bloque de inferencia, se obtiene un único conjunto de salida. La agregación lógica puede realizarse hallando el máximo de las funciones de pertenencia de los conjuntos de salida ( OR ) o mediante la suma algebraica de las funciones de pertenencia de los conjuntos de salida.
4. Un bloque desdifusor que realiza una operación inversa a la que realiza el bloque difusor; es decir, que a partir del conjunto difuso de salida del bloque de agregación lógica, obtiene un valor numérico. La desdifusión se puede realizar mediante diversos métodos, algunos de los cuales esta la técnica del Máximo. Hallando el valor al que corresponde el máximo de la función de pertenencia del conjunto de salida del bloque de agregación.

## 2.5. APRENDIZAJE

Ya se indicó anteriormente que la adquisición del conocimiento constituye uno de los principales cuellos de botella en el proceso de construcción de SBC. Por lo tanto, sería

ideal que la base de conocimiento pudiera ser aprendida de forma automática a partir de un conjunto de ejemplos. Este es el objetivo del Aprendizaje Automático.

### **2.5.1. Métodos de aprendizaje automático**

Los principales métodos de aprendizaje automático son (Chico, 2008):

1. Aprendizaje de memoria: el conocimiento es directamente suministrado y memorizado. No se necesitan reglas de inferencia.
2. Aprendizaje por narración: también llamado aprendizaje a partir de instrucciones, está basado en la adaptación del conocimiento proveniente de una fuente, a una forma que puede ser acoplada y utilizada.
3. Aprendizaje por analogía: es realizado adquiriendo nuevo conocimiento que fue de ayuda en la realización de una tarea en similares circunstancias.
4. Aprendizaje inductivo: Se clasifica en aprendizaje a través de ejemplos y aprendizaje a partir de la observación.
  - Aprendizaje a través de ejemplos: también llamado adquisición de conceptos. Un conjunto de ejemplos positivos y negativos son dados, y de ellos se induce una descripción de conceptos de alto nivel.
  - Aprendizaje a partir de la observación: también llamado generalización descriptiva o aprendizaje no supervisado. Se trata de crear nuevas características de teorías a partir de unos hechos.

## 2.6. TABLAS DE REPERTORIO DIFUSO

Las Tablas de Repertorio Difuso (TRD) se desarrollan a partir de la Rejilla de Repertorio (RR) clásica de Kelly (Kelly, 1955); la RR no permite la introducción de información incierta o imprecisa y la TRD solventa estas limitaciones y proporciona importantes ventajas.

### 2.6.1. ¿Por qué usar las tablas de repertorio difuso?

La razón de usar la técnica de la TRD en este Proyecto es por sus siguientes aportaciones (Lozano, 2004):

1. Permite el uso y tratamiento de información concreta e imprecisa, dada esta última por medio de etiquetas o valores lingüísticos. Para ello, se usa un formalismo de representación basado en el concepto de conjunto difuso que será empleado para representar cualquier valor.
2. Su generalidad, es decir, la posibilidad de poder ser empleada en un amplio espectro de dominios. Esto es debido a que tiene en cuenta las características generales del problema de clasificación y la libertad que ofrece a la hora de expresarse.
3. Ofrece la posibilidad de introducir varios valores cuando se valora un atributo.
4. Facilidad de uso, ya que está basada en una idea intuitiva, y ampliamente usada en la realidad, como es la obtención de información basada en comparaciones entre elementos similares.
5. Realiza un análisis dimensional que tiene en cuenta la dualidad en la cualificación de la información (incertidumbre - precisión). Este análisis hace uso de una medida, denominada separabilidad, que es definida para medir la proximidad conceptual entre los elementos que se quieren clasificar. De esta forma, esta técnica ayuda a decidir sobre la alternativa óptima distinguiendo una a una entre las distintas ofertas existentes.
6. La técnica se ha mostrado efectiva para la obtención de reglas generales que muestren qué atributos y valores son los que discriminan cada uno de los elementos del conjunto de salidas del sistema de clasificación.

### 2.6.2. Salidas o respuestas y atributos

En un problema de clasificación bien delimitado el conjunto de salidas ( $E$ ) estará lo suficientemente bien definido como para prestar atención al proceso de extracción de este



conjunto. Toda salida o respuesta que se pueda producir en un sistema de clasificación estará definida por un conjunto de atributos o características significativas  $V_i$  que se nota como  $V$ . Estos atributos o características pueden ser de distinta naturaleza (ver Figura 2.6): lógicos (booleanos y graduados), numéricos (fijos o intervalos) y valores lingüísticos o nominales (expresados en lenguaje natural).

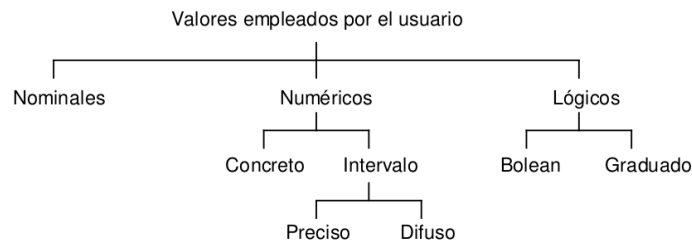


Figura 2.6 Taxonomía de las clasificaciones empleadas por el usuario.

## 2.7. APLICACIONES EN LOS SISTEMAS DE INFERENCIA DIFUSA

En esta sección, se presentan algunas propuestas sobre FSS (en inglés Fuzzy System Software) para resolver problemas específicos en diferentes áreas de investigación. Aquí el énfasis está en la especialización en lugar de generalización. Por lo tanto, la mayoría de FSS disponibles corresponde a herramientas y suites. Debido a la enorme cantidad de problemas específicos para los cuales se ha propuesto FSS, se presentan las obras existentes para cada categoría de segundo nivel en la taxonomía (véase la figura 1.2) clasificada por áreas de investigación, con el fin de proporcionar una visión más completa como sea posible con respecto a los principales dominios de aplicación, se analiza los siguiente puntos :

1. El control difuso (Driankov, 1993) es probable que sea el dominio de aplicación con la más antigua tradición y el mayor número de contribuciones en la literatura. Hay varias bibliotecas para aplicaciones de control, tales como jFuzzy-Logic (Cingolani, 2013) para hacer el diseño de controladores de lógica difusa más fácil. Muchas herramientas se pueden encontrar para diferentes problemas de control, entre los cuales están los siguientes: la herramienta de simulación dinámica del sistema de propulsión WARPS-TAR2 para el modelado de vehículos eléctricos (Cheng, 2013); una herramienta gráfica destinada a controlar y supervisar la temperatura y la humedad relativa en el contexto de la agricultura (Yang, 2013); ASAFES2 utiliza una función de aproximación neu-

ronal difusa que combina el método de razonamiento difuso de Takagi-Sugeno con aprendizaje por refuerzo estocástico (Zikidis, 2009).

2. Para resolver problemas específicos en la toma de decisiones, los programas mas conocidos son FMCGDSS- DECIDER (Lu, 2009) que es una herramienta para la lingüística con grupos de varios criterios de toma de decisiones aplicado a la evaluación material textil; VisualFCM (Glykas, 2013) que se aplica a mapas cognitivos difusos para apoyar la toma en varios escenarios de planificación financiera; Xfuzzy que ha sido utilizado para generar sistemas difusos para ser integrado en una herramienta de mercado de bienes raíces en la toma de decisiones (Lopez, 2010). Algunas herramientas de soporte de decisiones también se han integrado los sistemas de información geográfica (GIS en inglés). Entre ellos se encuentran los siguientes: ArcGIS es una herramientas para el manejo de análisis espacial y se extendió por Eldrandaly y Abdelaziz con el fin de manejar múltiples criterios de análisis de decisión (Eldrandaly, 2012); Geofis DSS (Guillaume, 2013) es un paquete de código abierto destinado a proporcionar GIS con nuevas funcionalidades, como modelado difuso y el razonamiento con datos georreferenciados en Agronomía (observe que el uso de los GIS en la agricultura se está volviendo muy popular). Por otra parte, algunos investigadores han propuesto Matlab para tareas de toma de decisiones con el inconveniente que es software comercial. ANGULO (Angulo, 2012) desarrolló una herramienta relacionadas con la medición de los episodios de la calidad del agua a partir del comportamiento de las variables medidas en las redes de control de agua; FCM-uUTI-DSS (Papageorgiou, 2012) se ocupa de la gestión de tratamiento de la infección urinaria basada en mapas cognitivos difusos.
3. Biomedicina es la rama de la ciencias médicas que se ocupa de la aplicación de principios de las ciencias naturales a la medicina clínica (Sargent, 2005). En el contexto de la biomedicina hay muchas aplicaciones del FSS. Varias herramientas se pueden encontrar para resolver problemas específicos, tales como: MSClust (Tikunov, 2012) que es una herramientas destinada a la realización de agrupamiento difuso con la información del metabolismo; Fuzzy SPike Sorting (FSPS) (Oliynyk, 2012) que proporciona a los laboratorios de neurociencia una nueva herramienta para la clasificación rápida y robusta de la actividad neuronal; Pinti propuesta como una herramienta para el análisis de correspondencias múltiples de datos morfométricos de cráneo (Pinti, 2010).
4. Hay varios cursos que están destinados a la enseñanza de materias que requieren el aprendizaje de técnicas difusas avanzadas. Con el fin de comprender adecuadamente las ventajas y desventajas de estas técnicas, se requiere de estudiantes para

llevar a cabo el trabajo práctico. Por desgracia, este tipo de trabajo se hace por lo general para tareas de programación y los estudiantes olvidan el objetivo principal de analizar los componentes que caracterizan a cada técnica. Como resultado, son incapaces de discernir cuál es la técnica más adecuada para un caso particular. Varias bibliotecas para las tareas de enseñanza se pueden encontrar en la literatura, como COALA (Jurado, 2009) que es un conjunto de plugins de Eclipse que constituye un entorno asistido por ordenador diseñado para que sea más fácil para los estudiantes de informática y de ingeniería aprender algoritmos. Dentro de las herramientas educativas se encuentran las siguientes: CAVUS (Cavus, 2010) que es un sistema basado en web para ayudar en la evaluación de los sistemas de gestión de aprendizaje; eRiskGame (Oliveira, 2013) que se utiliza como una herramienta de apoyo a la enseñanza en los cursos de ingeniería de software; VirtualLab (Bingol, 2010) y FEUP (Moreiral, 2010) que tienen por objeto la enseñanza de control difuso; EDUrobot (Abiyev, 2010) que facilita la comprensión de la robótica móvil; FIRS-trainer (Herrera, 2009) que se centra en los cursos relacionados con los sistemas de recuperación de información difusa. Finalmente en el área educativa podemos encontrar FUZZY RAMSET (Martinez, 2013) que se aplica a los cursos de ingeniería de software con el fin de dar a los estudiantes una experiencia práctica; en el ambiente de laboratorio en tiempo real (Dandil, 2012) esta dSPACE DS1103 DSP y Matlab / Simulink / RTW con el objetivo de ayudar a los estudiantes a modelar y controlar las unidades de motor de inducción; MNTOOL (Melin, 2013) proporciona un entorno visual y amigable para ayudar a los estudiantes en la experimentación con sistemas múltiples en redes neuronales.

5. La herramienta DeFusi utiliza métodos e ideas de distintas áreas de conocimiento, como la psicología, la inteligencia artificial y la ingeniería del software. Del campo de la psicología, se han tomado ideas de la teoría de los constructores personales, del campo de la inteligencia artificial se han tomado ideas del aprendizaje automático y de la ingeniería del software se ha seguido la metodología del proceso unificado de desarrollo (Chico, 2008).
6. J.J. Castro Sánchez diseñó e implementó una herramienta para la adquisición automática de conocimiento (Castro, 2001). Para alcanzar este objetivo desarrollo un método para la obtención automática de la estructura conceptual de un experto, enfrentado a un problema de clasificación, que extiende ideas provenientes del campo de la psicología. El método desarrollado presenta como características más interesantes: la libertad que otorga al experto a la hora de definir los conceptos empleados por él, los atributos que poseen esos conceptos y los valores que estos pueden tomar, así como la posibilidad

de emplear el conocimiento generado durante este proceso para inferir un conjunto de reglas generales que conforman las directrices principales de su razonamiento (Castro, 2001).

7. Un agente es un programa software que puede desarrollar una serie de tareas por él mismo con el fin de conseguir unos fines particulares (Jennings, 2002). Estos agentes se proponen para modelizar el comportamiento de un agente comprador y vendedor en el comercio electrónico. Acomdi (Un Agente Comprador Difuso Automático) es un programa que modela la fase de negociación de un comprador y pretende ser útil en la automatización de la negociación competitiva entre muchos vendedores y un comprador, sobre varios aspectos del servicio o producto, en contextos B2C, pudiéndose emplear también en contextos B2B (Lozano, 2004).

## 2.8. CONCLUSIONES

En este capítulo se ha analizado el estado del arte con el fin de proporcionar un marco bien establecido que ayuda a los investigadores a encontrar fácilmente las propuestas existentes que están relacionadas con una rama particular y centrarse en los acontecimientos más importantes. Sin embargo, en los últimos años el número de contribuciones en la que el propósito general de la lógica difusa se ha desarrollado ha tendido a aumentar, ya que se puede aplicar para resolver diferentes tipos de problemas en diversos ámbitos de aplicación. Las publicaciones relacionadas con sistemas difusos representan un grupo que es probable que crezca más rápido en el futuro. Por otra parte, muchos investigadores están trabajando actualmente para desarrollar un lenguaje estándar que nos permitirá intercambiar sistemas difusos a través de diferentes plataformas con el fin de mejorar la interoperabilidad. Además, los programas de lógica difusa están listos para ayudar a los ingenieros y desarrolladores de software en la mayoría de las tareas comunes relacionadas. La visibilidad para los propósitos educativos es lenta pero en constante crecimiento. Por otra parte, el estudio bibliográfico muestra cómo esta zona presenta una investigación madura, la presentación de más de 100 artículos y un mayor número de citas cada año. Por otra parte, el número de contribuciones en revistas ha tendido a aumentar en los últimos años, la mejora de la difusión de las propuestas y la cooperación con otras áreas de investigación. Por último, las tendencias actuales más destacadas y problemas abiertos se han puesto de manifiesto en la lógica difusa, con el fin de atraer la atención de la comunidad científica.

## Capítulo 3

# MODELOS MATEMÁTICOS EN LA LÓGICA DIFUSA

En este capítulo se revisa la literatura actual acerca de los temas de modelación y su relación con la lógica difusa. La modelación en un sentido general, se refiere a la creación de una descripción de un sistema (una planta, un proceso, etc.) en términos matemáticos, que caracteriza el comportamiento de la entrada-salida del sistema subyacente. Por ejemplo para describir un sistema físico, como un circuito o un microprocesador, tenemos que utilizar una fórmula matemática o una ecuación que puede representar el sistema tanto cualitativa como cuantitativamente. Tal formulación es una representación matemática, llamado un modelo matemático, del sistema físico (Chen, 2000).

### 3.1. Modelación

La modelación a menudo es necesaria y práctica en situaciones reales, de manera intuitiva el modelado aproximado es siempre posible. Sin embargo, las preguntas claves son qué tipo de aproximación es buena, donde el sentido de "bondad" tiene que ser definido en primer lugar, por supuesto, y cómo formular una buena aproximación como en el modelado de un sistema tal que es matemáticamente rigurosa y puede producir resultados satisfactorios tanto en la teoría y aplicaciones. Siguiendo la clasificación tradicional en el campo de sistemas de control, un sistema que describe el comportamiento de entrada-salida de una manera similar a una función o relación matemática sin la participación de un operador diferencial o ecuación se llama sistema estático; en contraste, un sistema descrito por un operador diferencial o ecuación se llama sistema dinámico. El tema de modelado de sistemas difusos es muy importante y útil por derecho propio, no necesariamente relacionados con el control. Por supuesto, esencial para la investigación de sistemas de control difuso. El concepto

fundamental de modelado de sistemas se puede ilustrar de la siguiente manera. Suponga que tiene un sistema desconocido (ver figura 3.1), para el cual se tiene un conjunto de entradas  $x_1, \dots, x_n$  y salidas  $y_1, \dots, y_m$  que pueden medirse (observarse), en donde las entradas y salidas pueden ser series de tiempo discreto o señales continuas (Chen, 2000).



Figura 3.1 Sistema desconocido de caja negra.

Queremos encontrar una descripción matemática para caracterizar cualitativa o cuantitativamente este sistema desconocido, en el sentido de que mediante la introducción de  $x_1, \dots, x_n$  en la descripción matemática que siempre se puede obtener las correspondientes salidas  $y_1, \dots, y_m$ . El establecimiento de tal descripción matemática se denomina modelado matemático para el sistema desconocido (3.1). Como de costumbre, la descripción matemática puede ser una fórmula matemática, tal como una asignación o una función que relaciona las entradas a las salidas en forma (Chen, 2000) :

$$\begin{cases} y_1 = f_1(x_1, \dots, x_n), \\ \cdot \\ \cdot \\ \cdot \\ y_m = f_m(x_1, \dots, x_n); \end{cases} \quad (3.1)$$

o también puede verse como una declaración lingüística lógica que se puede cuantificar matemáticamente en la forma (3.2):

$$\begin{aligned} &\text{Si entrada } x_1 \text{ y ... y entrada } x_n \\ &\text{Entonces salida } y_1 \text{ y ... y salida } y_m \end{aligned} \quad (3.2)$$

El modelado matemático de sistemas difusos es cuantificar la forma lingüística lógica (3.2) utilizando la lógica difusa y el modelo matemático funcional (3.1), este resultado es llamado un sistema difuso estático.

Los modelos matemáticos son representaciones de mecanismos que gobiernan un fenómeno natural que no es completamente reconocido, controlado o entendido (Tedeschi, 2006).

El proceso de modelado engloba varios pasos que comienzan con una declaración clara de los objetivos del modelo, los supuestos acerca de los límites del modelo, lo apropiado de los datos disponibles, el diseño de la estructura, la evaluación de las simulaciones y el proporcionar retroalimentación para las recomendaciones, además del rediseño de los procesos. Las pruebas del modelo son comúnmente usadas para probar la exactitud de un modelo y son presentadas típicamente como evidencias para promover su aceptación y usabilidad (Sterman, 2002).

Algunos modelos matemáticos pueden ser clasificados según los siguientes criterios (Xiaoyuan, 2013) como lineales y no lineales. El caso más común es el modelo de regresión y el término es comúnmente tomado como sinónimo de modelo de regresión lineal. Sin embargo el término también se utiliza en análisis de series de tiempo con un significado diferente. Los modelos matemáticos se componen generalmente por variables, que son abstracciones de cantidades de interés en los sistemas descritos, y los operadores que actúan sobre estas variables, que pueden ser operadores algebraicos, funciones, operadores diferenciales, etc. Si todos los operadores en un modelo matemático presentan linealidad, entonces el modelo matemático es definido como lineal y en caso contrario como no-lineal. Un acercamiento común con problemas no lineales es la linealización.

- **Determinísticos y probabilísticos.** Un modelo determinísticos es uno en el cual cada conjunto de variables es determinado únicamente por los parámetros en el modelo y por el conjunto de estados previos de esas variables. Por lo tanto los modelos deterministas se desempeñan de la misma forma para un conjunto dado de condiciones iniciales. A la inversa en un modelo estocástico, la aleatoriedad está presente y los estados de las variables no son descritos por valores únicos sino más bien por distribuciones de probabilidad.
- **Estáticos y dinámicos.** Un modelo estático no tiene en cuenta el elemento del tiempo, mientras que un modelo dinámico sí lo hace. Los modelos dinámicos suelen ser comúnmente representados con ecuaciones diferenciales.
- **Discretos y continuos.** Un modelo discreto no toma en cuenta la función del tiempo y comúnmente usa métodos de avance de tiempo, mientras que un modelo continuo sí lo hace. Los modelos continuos son usualmente representados con  $f(t)$  y los cambios son reflejados sobre intervalos de tiempo continuo.

Las técnicas de modelado más comunes (ver tabla 3.1) para las combinaciones de modelos ( continuo / discreto ) y ( determinista / estocásticos ) nos permiten elegir la técnica adecuada (Fey, 2009).

Tabla 3.1 Técnicas de modelado.

	Continuo	Discreto
Determinístico	Ecuaciones diferenciales Parciales	Partículas Interactivas
Estocástico	Ecuaciones diferenciales Estocásticas	Eventos Aleatorios

La siguiente lista debe revisarse cada vez que se desarrollen modelos matemáticos, aun cuando no se apliquen todos los pasos al principio. Esto ayudará a mejorar las destrezas de modelación :

- a) Experimentación. Un buen modelo de un sistema te permite cambiar sus componentes y ver cómo estos cambios afectan el resto del sistema.
- b) Predicción. Un modelo permite la predicción del curso futuro de un sistema dinámico.
- c) Generalización. Un buen modelo permite estimular preguntas adicionales sobre el comportamiento del sistema y la aplicabilidad de los principios que son descubiertos en el proceso de modelación a otros sistemas.

Un modelo matemático es una ecuación o un conjunto de ecuaciones que representan el comportamiento de un sistema (France, 1984), o una construcción matemática diseñada para estudiar un sistema del mundo real o fenómeno. Un modelo es útil cuando captura los elementos adecuados de la realidad con un grado aceptable de exactitud. En consecuencia, es importante hacer cierta evaluación de su validez que permita contar con una medida de protección, tanto al modelador como al usuario del modelo, cuando sea utilizado en predicción del sistema.

Para Mayer y Butler (Mayer, 1993), la validación es un paso necesario para la aceptación de un modelo, y se define como la comparación de las predicciones del modelo con los valores observados del mundo real para determinar si el modelo es adecuado para el propósito establecido. Se considerará la validación de un modelo como en Medina (2006): la comparación por medio de algún método de las predicciones del modelo con observaciones del sistema real para determinar su capacidad predictiva, es decir, la validación se concentra en la determinación de la exactitud y precisión del modelo.

La exactitud se refiere a qué tan cerca están los valores predichos por el modelo de los valores reales y la precisión mide qué tan cerca están entre ellos los valores predichos por el modelo (Tedeschi, 2006), básicamente la exactitud se refiere a qué tan cerca están los valores observados ( $Y$ ) de los predichos ( $Z$ ), o bien, sus diferencias ( $D = Y - Z$ ) con respecto al cero y la precisión a qué tan dispersos están los puntos ( $Z, Y$ ).



## 3.2. Modelo basado en sistemas difusos estáticos

Una declaración finita de implicaciones difusas puede ser descrito por un conjunto general de reglas de la forma SI - ENTONCES que contiene sólo la operación lógica difusa AND, con multiples entradas y una salida, tiene la siguiente forma (Chen, 2000):

(1) "SI  $x_1$  es  $X_{11}$  y ... y  $x_n$  es  $X_{1n}$  Entonces y es  $Y_1$  ".

(2) "SI  $x_1$  es  $X_{21}$  y ... y  $x_n$  es  $X_{2n}$  Entonces y es  $Y_2$  ".

.

.

(N) "SI  $x_1$  es  $X_{N1}$  y ... y  $x_n$  es  $X_{Nn}$  Entonces y es  $Y_N$  ".

Notar que la frase  $x$  es  $X$  es una abreviatura de la declaración completa  $x$  pertenece al subconjunto borroso  $X$  con la correspondiente valor de pertenencia  $\mu_X(x)$  por simplicidad una regla difusa para  $N = 1$  puede expresarse de la siguiente manera (3.3) :

$$R^1 : \text{Si } x_1 \text{ es } X_1 \text{ y ... y } x_n \text{ es } X_n \quad (3.3)$$

$$\text{Entonces } y = a_0 + a_1x_1 + \dots + a_nx_n$$

donde  $\{a_0, a_1, \dots, a_n\}$  son constantes, considere un conjunto particular de entradas,  $x_1 = x_1^0 \in X_1, \dots, x_n = x_n^0 \in X_n$  junto con sus valores de pertenencia  $\mu_{X_1}(x_1^0), \dots, \mu_{X_n}(x_n^0)$  y la salida y con valor (3.4).

$$y^0 = a_0 + a_1x_1^0 + \dots + a_nx_n^0 \quad (3.4)$$

Cuando hay mas de una regla difusa del tipo SI - ENTONCES ( $N > 1$ ) el sistema de reglas difusas queda de la siguiente manera (3.5):

$$R^i : \text{Si } x_1 \text{ es } X_{i1} \text{ y ... y } x_n \text{ es } X_{in} \quad (3.5)$$

$$\text{Entonces } y_i = a_{i0} + a_{i1}x_1 + \dots + a_{in}x_n, i=1, \dots, N$$

considerando un conjunto particular de entradas  $x_1 = x_1^0 \in X_1, \dots, x_n = x_n^0 \in X_n$ .

## 3.3. Modelo basado con un factor de Peso

Considere el peso  $w_j$  de cada regla en (2.8), la estructura queda de la siguiente manera (ver 3.6 y 3.7):

$$\text{Una Regla } R_j : \text{Si } x_1 \text{ es } A_{j1} \text{ y ... y } x_n \text{ es } A_{jn} \quad (3.6)$$

$$\text{entonces y es } r_j \text{ con } w_j, j = 1, 2, \dots, N$$

La ecuación 2.9 queda de la siguiente manera :

$$\hat{y} = \frac{\sum_{j=1}^N \mu_j(x) \cdot r_j \cdot w_j}{\sum_{j=1}^N \mu_j(x)} \quad (3.7)$$

Se asume que el peso  $w_j$  afecta solo la parte consecuente  $r_j$  (Nauck, 1998).

### 3.4. Modelo basado en criterios de clasificación

Este método fue propuesto por Z. Chi , H. Yan y T. Pham (Chi, 1996) que extiende el método de Wang y Mendel para hacer frente a los problemas de clasificación. Básicamente el algoritmo es bastante similar a la técnica Wang y Mendel. Sin embargo el método de Chi sólo toma etiquetas de clase en cada uno de los datos consecuentes para reglas difusas SI - ENTONCES. En otras palabras se generan reglas como en la técnica Wang y Mendel y luego se reemplazan partes consecuentes con sus clases. En cuanto al cálculo de cada regla que están determinadas por las partes antecedente de las reglas, las reglas redundantes se pueden eliminar considerando sus grados. Por último se obtiene reglas difusas SI- ENTONCES basado en el modelo SCBRD. El método consiste de los siguientes cinco pasos :

1. Encuentra los dominios de los intervalos de  $x_1$  a  $x_2$  ,  $[x_1^{(l)}, x_1^{(h)}]$  y  $[x_2^{(l)}, x_2^{(h)}]$  dentro de  $N_i$  regiones ( $i = 1, 2, \dots$ ). Las regiones se etiquetan de la siguiente manera :

$$R1(bajo(l)), R2, \dots, RN, (alto(h)).$$

Una función de pertenencia se adopta por cada región difusa. Las mas comunes son las trapezoidales y triangulares. Para problemas de clasificación con  $L$  clases la salida sera  $C_1, C_2, \dots, C_L$ .

2. Determina los grados de pertenencia  $x_1^{(i)}$  y  $x_2^{(i)}$  para diferentes entradas con sus etiquetas lingüísticas (ver 3.2). Posteriormente dada una entrada asigna la región con el máximo grado de pertenencia. Finalmente produce la regla para cada par de entrada y salida.
3. Asigna el grado a cada regla generada, existen dos estrategias. La primera es determinada por el grado de pertenencia de las entradas y la salida. La segunda se obtiene por la relación entre el número de pares de datos que soporta la regla y el número total de patrones que tiene la misma condición de tipo SI.
4. Crea un conjunto de reglas combinadas (combined ruled bank), la forma de reglas combinadas se ven en la figura 3.2. Las reglas combinadas se llenan con reglas difusas de cualquiera de las generados a partir de datos numéricos o del conocimiento humano.

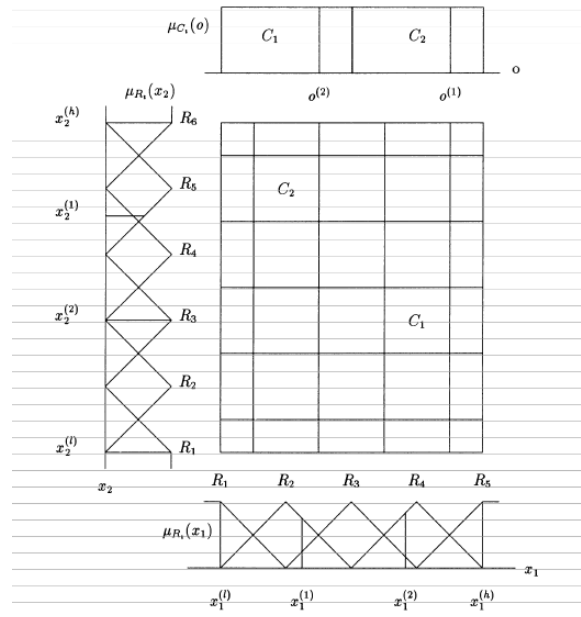


Figura 3.2 Particiones difusas con dos entradas.

- Determina el mapeo entre entrada y salidas a través de una estrategia con el bloque desfusor.

### 3.5. Evaluación predictiva de modelos de inferencia difusa

La evaluación del rendimiento de la predicción de un clasificador es de gran importancia con el fin de ser capaz de juzgar su utilidad, también en comparación con los métodos de la competencia. Las medidas de uso general de rendimiento en la fase de construcción de los modelos son la precisión, tasa de error y las gráficas roc (Hanley, 1982).

Es importante reconocer que la evaluación en la fase de entrenamiento es diferente de la evaluación del modelo final. La primera fase consiste en seleccionar el modelo más eficaz y robusto durante el entrenamiento. Por lo general, se divide un conjunto de datos de entrenamiento en capacitación y validación de subconjuntos, por ejemplo, para la validación cruzada (Ben, 2010). La segunda fase consiste en evaluar el modelo final después del entrenamiento. Idealmente, los datos de prueba de esta fase refleja las distribuciones de la población original a pesar de que tales distribuciones son generalmente desconocidas. En las secciones 3.5.2 y la 3.5.1 se analiza exclusivamente la evaluación del desempeño del modelo final con respecto a las predicciones considerando las funciones de pertenencia triangular, trapezoidal y gaussiana (ver Ec. 2.4, 2.5 y 2.6). Cuando la variable de salida esté en escala nominal sera considerado un caso discreto y cuando esté en escala de razón sera el caso continuo.

### 3.5.1. Prueba de desempeño con proporciones caso discreto

Una predicción se dice que es correcta si el valor de predicción para dato de prueba es igual al de dato real. Se aplicará a un diseño experimental en donde la variable que se mide es dicotómica, es decir, una variable con dos posibles valores de predicciones congruentes : correcta o incorrecta, verdadero o falso. Se asigna “1” por cada éxito y “0” por cada fracaso. Se realizaron observaciones repetidas al mismo tamaño de matriz para funciones de pertenencia trapezoidal, triangular y gaussiana, cada función de pertenencia es considerado un tratamiento, donde  $y_{ij} = 1$  o  $0$ , ( $i = 1, 2, \dots, n$ ;  $j = 1, 2, 3$ ) según sea el caso (ver Tabla 3.2)

Tabla 3.2 Tabla de desempeño discreto.

Predicción	Funciones de pertenencia		
	Triangular 1	Trapezoidal 2	Gaussiana 3
1	$y_{11}$	$y_{12}$	$y_{13}$
2	$y_{21}$	$y_{22}$	$y_{23}$
·	·	·	·
·	·	·	·
n	$y_{n1}$	$y_{n2}$	$y_{n3}$
<b>Total</b>	$g_1$	$g_2$	$g_3$

Para el caso discreto estamos interesados en probar la igualdad de las proporciones de los tratamientos (tipos de función de pertenencia), note que si  $H_0$  es cierta (see Eq. 3.8), todos los tratamientos tienen media común,  $\hat{p}_j = \frac{g_j}{n}$  donde  $g_j =$  número de éxitos en la  $j$ -th columna ( $j = 1, 2, 3$ ). Para ver si existen diferencias significativas entre tratamientos se utilizó la prueba Q de Cochran (Zar, 2007).

$$\begin{aligned}
 H_0 : p_{i1} = p_{i2} = p_{i3} \quad (\text{para } i = 1, \dots, n) \\
 \text{vs} \\
 H_A : p_{ij} \neq p_{ik} \quad (\text{para alg } j \text{ y } k \text{ y alg } i)
 \end{aligned}
 \tag{3.8}$$

### 3.5.2. Prueba de desempeño caso continuo

Para el caso continuo se verificó la exactitud del modelo de predicción con respecto a tres funciones de pertenencia (trapezoidal, triangular y gaussiana). Sea  $D_{ij} = Y_i - Z_{ij}$ , con  $Y_i$  igual a la  $i$ -ésima observación del experimento y  $Z_{ij}$  igual a la  $i$ -ésima predicción bajo el  $j$ -ésima función de pertenencia (ver tabla 3.3).

Tabla 3.3 Tabla experimental caso continuo.

Predicción	Funciones de Pertenencia		
	Triangular 1	Trapezoidal 2	Gaussiana 3
1	$D_{11}$	$D_{12}$	$D_{13}$
2	$D_{21}$	$D_{22}$	$D_{23}$
.	.	.	.
.	.	.	.
n	$D_{n1}$	$D_{n2}$	$D_{n3}$

El modelo estadístico utilizado fue el de diseño por bloques completos (ver Ec. 3.9) :

$$D_{ij} = \mu + \tau_j + \beta_i + \varepsilon_{ij} \quad (3.9)$$

donde  $\mu$  es la media global común a todos los tratamientos si no hubieran diferencias entre ellos,  $\tau_j$  es el efecto del  $j$ -ésimo tratamiento o función de pertenencia ( $j = 1, 2, 3$ ),  $\beta_i$  es el efecto del  $i$ -ésimo bloque o predicción ( $i = 1, \dots, n$ ) y  $\varepsilon_{ij}$  es el error aleatorio.

Los errores  $\varepsilon_{ij}$  se suponen variables aleatorias distribuidas normal e independientemente con media cero y varianza  $\sigma^2$ . Para el caso continuo estamos interesados en probar la igualdad de los tratamientos  $\tau_j$  (ver 3.10) :

$$\begin{aligned} H_0 : \tau_j &= 0 \\ &\text{vs} \\ H_A : \tau_j &\neq 0 \quad (\text{para al menos un } j) \end{aligned} \quad (3.10)$$

Para ver si hay diferencias significativas entre los tratamientos se utilizó la prueba de Friedman, esta es una alternativa no paramétrica cuando los supuestos de normalidad, independencia e igualdad de varianzas para los errores no se cumplen. Para las comparaciones múltiples se consideraron ambos casos, es decir cuando existen o no existen empates entre rangos (?).

### 3.5.3. Algoritmo de desempeño caso discreto y continuo

Se utilizó los siguientes algoritmos para evaluar el rendimiento de la capacidad predictiva por función de pertenencia, considerando el caso discreto (ver Algoritmo 1) y caso continuo (ver Algoritmo 2).

### 3.5.4. Algoritmo de evaluación

---

#### Algorithm 1 Algoritmo de evaluación caso discreto

---

Genera datos de *entrenamiento*  
 Genera datos de *prueba*  
 $Y \leftarrow$  Genera datos *reales*  
 Genera tabla experimental  $T$   
**for** cada función de pertenencia  $f_i$  **do**  
     Genera modelo difuso con datos de *entrenamiento*  
      $Z \leftarrow$  predicciones de modelo difuso con datos de *prueba*  
     **if**  $Y == Z$  **then**  
          $T \leftarrow 1$   
     **else**  
          $T \leftarrow 0$   
     **end if**  
**end for**  
 Realiza prueba Q de Cochran a  $T$   
 Genera gráfica, estadístico de prueba y p-valor

---



---

#### Algorithm 2 Algoritmo de evaluación caso continuo

---

Genera datos de *entrenamiento*  
 Genera datos de *prueba*  
 $Y \leftarrow$  Genera datos *reales*  
 Genera tabla experimental  $T$   
**for** cada función de pertenencia  $f_i$  **do**  
     Genera modelo difuso con datos de *entrenamiento*  
      $Z \leftarrow$  predicciones de modelo difuso con datos de *prueba*  
      $D \leftarrow Y - Z$  diferencia de datos *reales* menos *predicciones*  
      $T \leftarrow D$   
**end for**  
 Realiza prueba de Friedman a  $T$   
 Genera gráfica, estadístico de prueba y p-valor

---

## 3.6. CONCLUSIONES

Se han analizado los principales modelos matemáticos en la lógica difusa, de igual manera se propuso cómo evaluar la capacidad predictiva de las funciones de pertenencia trapezoidal, triangular y gaussiana en relación con la variable de salida, considerando la escala nominal y de razón. Esta metodología define un proceso experimental, para efectuarla se combinaron diseños de experimentos en el área estadística y la validación de modelos matemáticos. Si bien es cierto que solo se aplicaron dos técnicas no paramétricas (prueba Q de Cochran y Friedman), existe la posibilidad de proponer otras técnicas de acuerdo a las necesidades propias del investigador. El incluir los algoritmos facilita el análisis y comprensión, para una correcta aplicación de la lógica difusa en áreas como la biología, medicina, educación entre otras.





## Capítulo 4

# IMPLEMENTACIÓN DE MOTOR DE INFERENCIA DIFUSO

Se seleccionó HTML, PHP y AJAX como elementos base para un sistema web de inferencia difuso (SWID), el cual utiliza el lenguaje de programación R (Ihaka, 2009) como motor de inferencia estadístico, este lenguaje es eficiente en el análisis de datos estadísticos y la simulación de modelos matemáticos. SWID es un sistema web que permite generar modelos de inferencia difusa en áreas como la Medicina, Biología, Educación.

### 4.1. ESPECIFICACIONES TÉCNICAS

Con la finalidad de comprender mejor los distintos elementos que intervienen en el funcionamiento del sistema, a continuación se describen los aspectos técnicos relacionados con la arquitectura del motor de inferencia difuso.

#### 4.1.1. Tecnologías y técnicas de desarrollo

Las tecnologías involucradas en el desarrollo del sistema constituyen parte de los requerimientos mínimos para su ejecución. A continuación se enlistan con una breve descripción:

1. **Lenguaje PHP** - Para el desarrollo de componentes software, funciones y bibliotecas de SWID. Es el lenguaje fundamental de la plataforma y su intérprete puede integrarse de manera eficiente en el servidor Web Apache.
2. **Lenguaje R** - Es un lenguaje de programación de código abierto para realizar análisis estadístico de datos. Es utilizado por los estadísticos, científico de datos y otros que necesitan hacer un análisis estadístico de datos y recoger la información clave utilizando

mecanismos como la regresión, agrupamiento, clasificación y análisis de textos. R se ha registrado bajo licencia GNU (General Public License). Fue desarrollado por Ross Ihaka y Robert Caballero de la Universidad de Auckland, Nueva Zelanda, que se maneja actualmente por el R Development Core Team. Se puede considerar como una implementación diferente de S, desarrollado por Johan Cámaras en los Laboratorios Bell. Hay algunas diferencias importantes, pero una gran parte del código escrito en S puede ser alterada mediante el intérprete de R (Vignesh, 2013).

3. **Lenguaje HTML** - Para la elaboración de las páginas Web que conforman la interfaz visible de la plataforma SWID. Establece la estructura y contenido de texto o imágenes de la página.
4. **AJAX** - Técnica de desarrollo que pretende aumentar la interactividad de la página web completa, velocidad y facilidad de uso. Se utiliza para la creación de páginas web y aplicaciones web dinámicas programadas en JavaScript para realizar llamadas a los servicios basados en la web (Teague, 2006).
5. **Apache** - Opción automática para un servidor web. Es rápido, potente, ampliamente utilizado, y precompilados disponibles para la mayoría de sistemas operativos (Smith, 2013).
6. **AB de Apache** - Es una herramienta de evaluación comparativa del servidor web que proporcionan estadísticas sobre qué tan bien responde un servidor web durante varias peticiones de usuarios de manera simulada. Permite simular cualquier número de usuarios que solicitan un documento web y lo más importante nos permite simular una visita simultánea por cualquier número de usuarios (solicitudes concurrentes ) a un documento alojado en un servidor web. Proporciona información del tiempo total en que una petición le toma responder (Apache, 2010).
7. **XAMPP** - Facilita la instalación del servidor Apache Tomcat (The Apache Software Foundation 1999) que es una implementación de código abierto de software de las tecnologías Java Servlet y JavaServer Pages. Se utiliza para ejecutar o interpretar el código desarrollado para el sistema SWID.
8. **Lenguaje CSS (Cascade Style Sheets)** - Para definir la presentación de una página Web en términos de un formato y tipografía específica, independiente de la implementación. Los archivos CSS se les denomina hojas de estilo.

9. **Lenguaje JavaScript** - Para manipular dinámicamente los distintos elementos de la interfaz, lo que permite establecer nuevos estilos de interacción. Generalmente los archivos JavaScript constituyen componentes interactivos (widgets).

#### 4.1.2. Bibliotecas de desarrollo en R

1. **FRBS** - Este paquete implementa varios algoritmos para construir y utilizar sistemas difusos basados en reglas. Se basa en el concepto de conjuntos difusos, propuesto por Zadeh en 1965, que tiene por objeto representar el razonamiento humano en un conjunto de reglas SI - ENTONCES (Riza, 2015).
2. **STRINGR** - Es un conjunto de funciones para realizar operaciones en datos de tipo cadena, sus funciones son consistentes, simples y fáciles de utilizar (Hadley, 2012)

#### 4.1.3. Flujo de datos

SWID tiene una interfaz fácil de usar de tal manera que puede ser utilizado en muchas plataformas (sistemas operativos diferentes) y muchos otros dispositivos (por ejemplo computadoras de escritorio, tablets o incluso un teléfono móvil). SWID tiene 4 pasos básicos dentro de su flujo de trabajo (ver figura 4.1): 1) cargando un archivo CSV (archivo separado por comas) 2) la limpieza, comprobación de la lista de variables, e importación de la lista de variables en la base de datos, 3) comprobar el nombre de las variables de entrada y salida, 4) resumen estadístico de las variables de entrada.

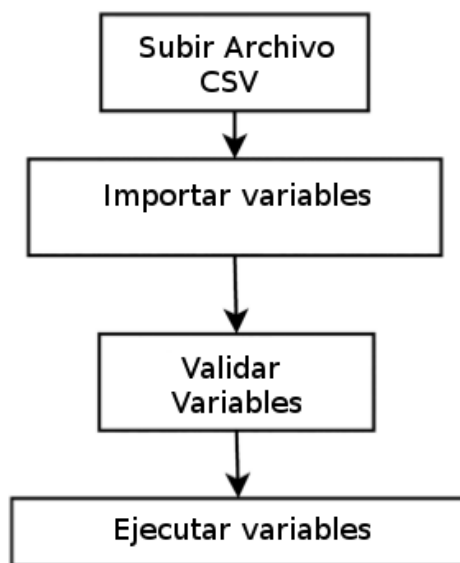


Figura 4.1 Flujo de datos en SWID.

#### 4.1.4. Arquitectura

Para la implementación de SWID se utilizó la Arquitectura SaaS (software como servicio), este patrón puede ser implementado en muchos niveles de complejidad, desde un servicio Web seguro a una interfaz de usuario simple, como una página Web en un navegador. El usuario interactúa con el servicio, que a su vez invoca a la funcionalidad básica. Las respuestas son adecuadamente dirigidas de vuelta al usuario según se requiera. Un proveedor de software de una aplicación SaaS también puede reaccionar rápidamente a los errores en el sistema. El administrador puede supervisar todos los usuarios de forma concurrente para detectar fallas en el software que pueden requerir atención inmediata y pueda ser capaz de solucionarlos antes de que la mayoría de los usuarios se den cuenta. La arquitectura SaaS puede ser especializada mediante el uso de algoritmos avanzados de computación para realizar tareas de razonamiento tales como la inferencia. Estos sistemas avanzados algún día serán capaces de utilizar las habilidades cognitivas para reconocer y actuar sobre ciertos patrones de eventos (Governor, 2009).

La figura 4.2 muestra la arquitectura de software como servicio (Uc, 2016) propuesta para el SWID.

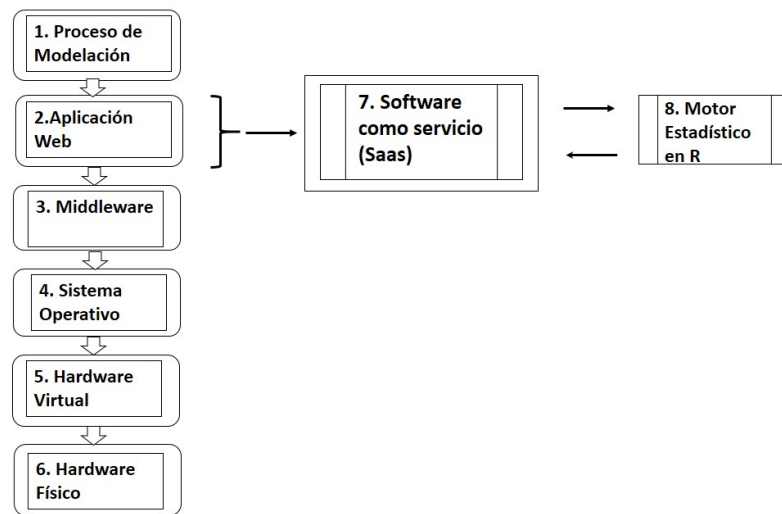


Figura 4.2 Arquitectura SaaS (arquitectura de software como servicio), 1.- Proceso de modelación, 2.- Aplicación Web, 3.- Middleware, 4.-Sistema Operativo, 5.-Hardware Virtual, 6.- Hardware Físico, 7.- Software como Servicio, 8.- Motor Estadístico en R.

#### 4.1.5. Arquitectura UML

El comportamiento dinámico de SaaS puede variar mucho dependiendo de los protocolos, estándares y enfoques arquitectónicos elegidos. La figura 4.3 muestra una representación común del patrón. Un usuario identifica sus necesidades de consumo de los recursos informáticos. Este patrón puede ser implementado en muchos niveles de complejidad, desde un servicio web seguro a una interfaz de usuario simple, como una página web HTML en un navegador. El usuario interactúa con el servicio que a su vez invoca la funcionalidad básica. Las respuestas son adecuadamente dirigidos de vuelta al usuario, según se requiera.

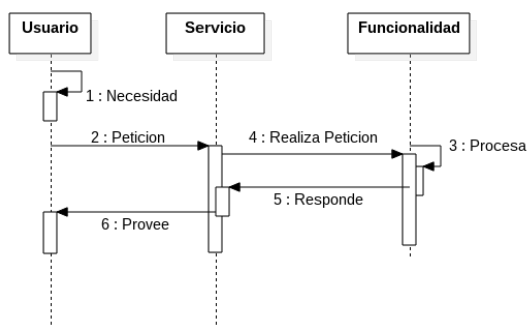


Figura 4.3 Diagrama de Secuencia en SaaS.

Para nuestro caso el diagrama de secuencia de SWID (figura 4.6 ) indica el orden de actividad durante el proceso de participación del usuario, servicio web y motor de inferencia difuso. Hay muchos matices en cada paso de la secuencia. Por ejemplo, cuando un usuario ha iniciado la evaluación de su modelo de inferencia difuso utilizando el servicio web. El servicio web directamente envía al motor estadístico la evaluación de su modelo en el contexto de su variable de salida, en el presente trabajo es discreto o continuo, y una vez realizada la evaluación, el motor de inferencia regresa los parámetros estimados al servicio web.

### 1. Evaluación del modelo caso discreto.

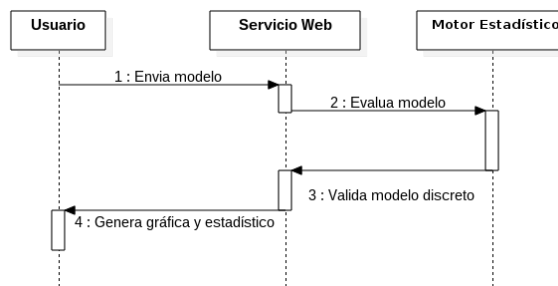


Figura 4.4 Diagrama de secuencia en SWID.

### 2. Evaluación del modelo caso caso continuo.

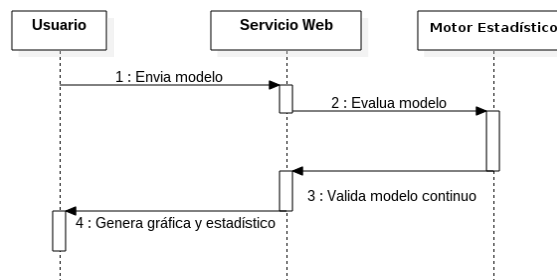


Figura 4.5 Diagrama de secuencia caso continuo.

### 3. Generación del modelo.

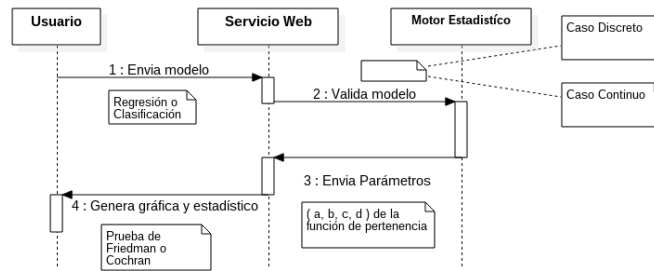


Figura 4.6 Diagrama de secuencia caso discreto.

## 4.2. APLICACIÓN SWID

### 4.2.1. Funcionalidades

SWID ofrece las funciones de gestión de datos, funciones de pertenencia y análisis de datos para dos tipos de algoritmos de clasificación difusa, el primero implementa la técnica Chi's (CHI) (Chi, 1996) y el segundo utiliza un factor de peso (W) (Ishibuchi, 2001). SWID tiene cinco características particulares:

1. Provee una interfaz Web para generar modelos de inferencia difusa.
2. Controla la estructura de los contenidos para una fácil lectura y comprensión de los usuarios.
3. Utiliza sistemas difusos basados en reglas para tareas de clasificación y regresión.
4. Evalúa la capacidad predictiva de las tres funciones de pertenencia (see Eq. 2.4, 2.5 y 2.6) acorde al conjunto de datos. Esto permite la elección de la función de pertenencia con mejor capacidad predictiva.
5. Genera funciones de pertenencia personalizadas y sus gráficas asociadas.

Con todo ello, SWID se convierte en una alternativa interesante para el investigador que desea utilizar los sistemas de inferencia difusa como técnica para la modelación de datos, sin considerar las complejidades propias del uso de técnicas computacionales.

### 4.2.2. Estructura de navegación

La interfaz visible del sistema de inferencia difuso fue implementada con una aplicación web. En su diseño se consideró el método UWE (Koch 2006) ya que especifica de mejor manera una aplicación web en su proceso de creación (Koch et al. 2008). Mantiene una

notación estándar con el uso de UML (Unified Modeling Language) (Object Management Group 2014) para sus modelos y sus métodos están claramente definidos para la construcción de cada uno de ellos. El propósito era crear un sistema funcional y estético en términos de la interfaz de usuario. En una aplicación web es útil saber cómo están enlazadas las páginas. La figura 4.7 representa el modelo de navegación, que se desarrolló en base al análisis de requisitos y el contenido del motor de inferencia difusa.

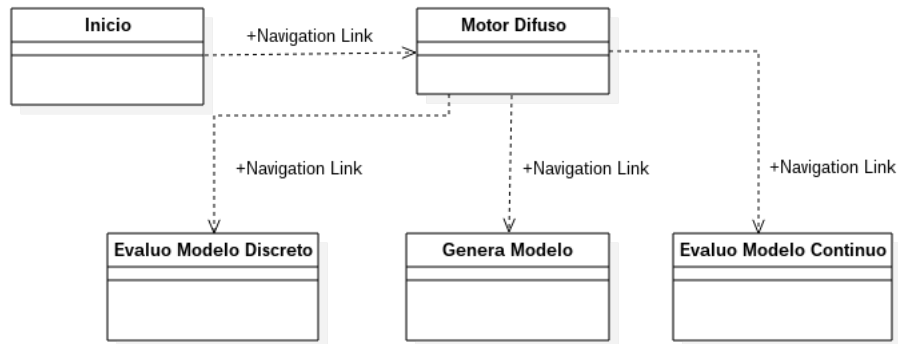


Figura 4.7 Modelo de navegación.

### 4.2.3. Presentación del sitio web

Se desarrolló una aplicación Web para interactuar con los procesos implementados en el prototipo obtenido del modelo propuesto. La aplicación Web está incorporada en un sitio Web con varias secciones, las cuales se describen a continuación:

1. Evalúa modelo difuso cuando la variable de salida es de tipo discreto (ver figura 4.8).

La interfaz web muestra un menú superior con 'Home', 'CONSULTA' (destacado), 'REGISTRATE', 'INICIAR SESIÓN' y 'AYUDA'. El contenido principal está dividido en dos secciones:

- Una sección superior con el título 'Subir Archivo delimitado por comas'. Incluye un botón 'Examinar...' y el texto 'Ningún archivo seleccionado.' con un botón 'Subir Archivo'.
- Una sección inferior con el título 'Evaluar Modelo Discreto' y un botón 'ejecutar'.

Figura 4.8 Interfaz Web para evaluar modelo difuso discreto.

2. Evalúa modelo difuso cuando la variable de salida es de tipo continuo (ver figura 4.9).



The screenshot shows a web interface with a dark navigation bar at the top containing 'Home', 'CONSULTA', 'REGISTRATE', 'INICIAR SESIÓN', and 'AYUDA'. Below the navigation bar, there are three main sections:

- Subir archivo delimitado por comas:** A section with a header and a form area containing an 'Examinar...' button, the text 'Ningún archivo seleccionado.', and a 'Subir archivo' button.
- Evaluar modelo continuo:** A section with a header and a form area containing an 'ejecutar' button.
- Gráfica del Modelo:** A section with a header and an empty form area.

Figura 4.9 Interfaz Web para evaluar modelo difuso continuo.

3. Genera modelo difuso y funciones de pertenencia (ver figura 4.10).

The screenshot shows a web interface with a dark navigation bar at the top containing 'Home', 'CONSULTA', 'REGISTRATE', 'INICIAR SESIÓN', and 'AYUDA'. Below the navigation bar, there are two main sections:

- Subir Archivo delimitado por comas:** A section with a header and a form area containing an 'Examinar...' button, the text 'Ningún archivo seleccionado.', and a 'Subir Archivo' button.
- Generar Modelo:** A section with a header and a form area containing three dropdown menus: 'Etiquetas' (set to '5'), 'Funcion' (set to 'TRAPEZOID'), and 'Modelo' (set to 'FRBCS.W'). An 'ejecutar' button is located to the right of the dropdown menus.

Figura 4.10 Interfaz Web para generación de modelo difuso.

4. Evalúa la capacidad predictiva caso discreto (Fig. 4.11) y caso continuo (Fig. 4.12) de las funciones de pertenencia triangular, trapezoidal y gaussiana.

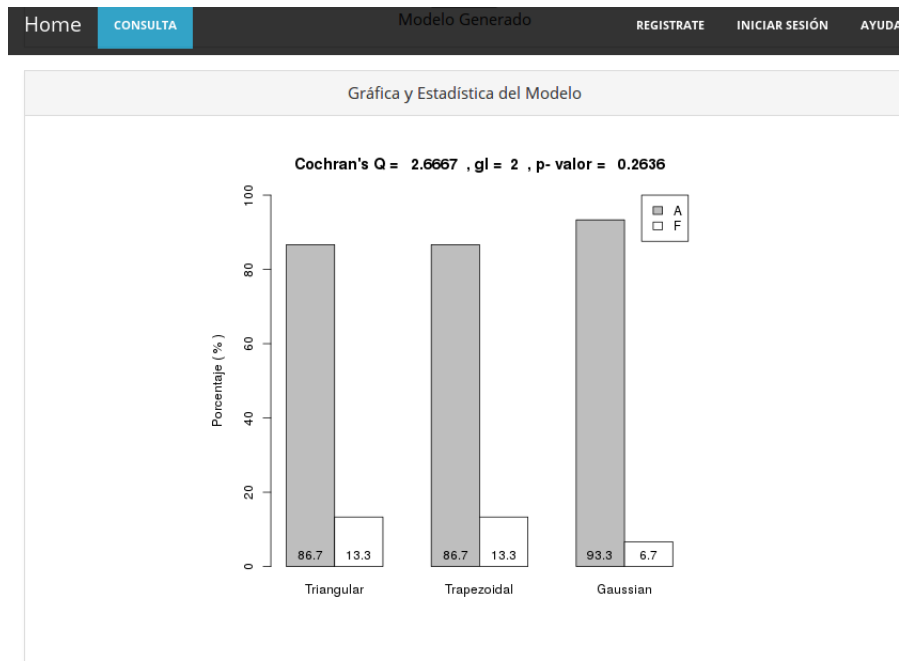


Figura 4.11 Interfaz Web para evaluación de capacidad predictiva.

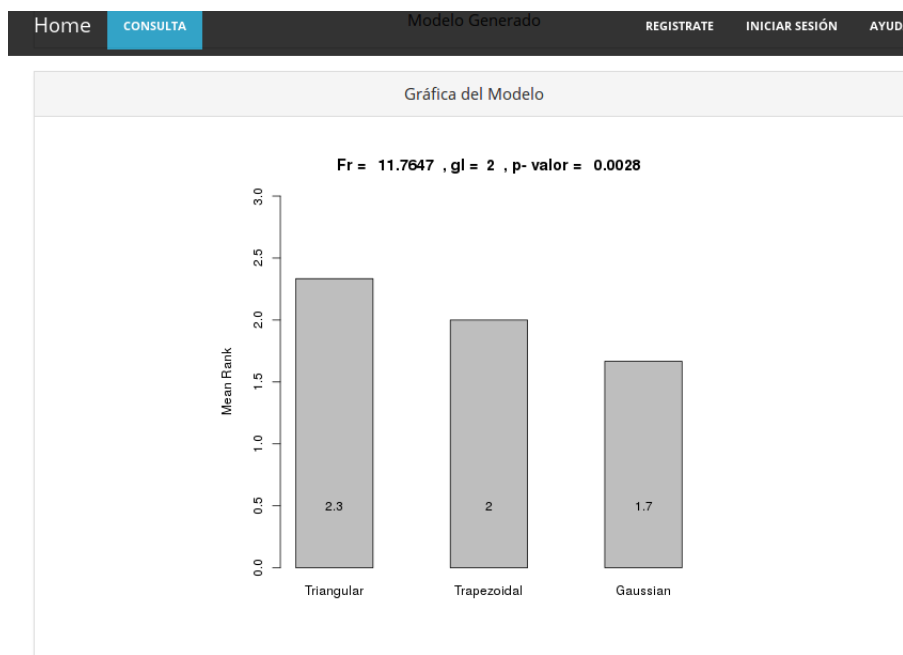


Figura 4.12 Interfaz Web para evaluación de capacidad predictiva.

5. Genera las funciones de pertenencia personalizados y sus gráficas asociadas triangular (Fig. 4.13), trapezoidal (Fig. 4.14) y gaussiana (Fig. 4.15).



Figura 4.13 Interfaz Web de función de pertenencia triangular.



Figura 4.14 Interfaz Web de función de pertenencia trapezoidal.



Figura 4.15 Interfaz Web de función de pertenencia gaussiana.

### 4.3. CONCLUSIONES

La tecnología descrita en el presente capítulo, es una propuesta para generar modelos de inferencia difusa para detección de patrones de comportamiento en datos; el motor de inferencia está integrado por un servicio web en PHP y el lenguaje de programación R. El motor de inferencia está basado en tecnologías de libre distribución que permite la interoperabilidad con sistemas ya existentes. En el siguiente capítulo se validará la tecnología propuesta a través de la metodología descrita en el capítulo 3.

# Capítulo 5

## EXPERIMENTACIÓN

En este capítulo se describe el proceso realizado para validar la propuesta de motor de inferencia difuso que se planteó en el Capítulo 4. El objetivo es comprobar si el motor de inferencia difuso optimiza el rendimiento de la generación de modelos difusos utilizando los algoritmos del Capítulo 3.

En las siguientes secciones, se aplica una evaluación del rendimiento de un motor de inferencia difuso, que permite simular modelos de inferencia difusa (SWID), se presenta una propuesta genérica para optimizar la ejecución de SWID en un servidor Web (Apache).

### 5.1. VALIDACIÓN MEDIANTE TÉCNICAS NO INVASIVAS

La validación mediante técnicas no invasivas, tiene como objetivo determinar si los modelos y algoritmos propuestos, aportan una mejora en rendimiento al motor de inferencia difuso. Por ello, se realiza una evaluación de las técnicas y pruebas estadísticas siguientes:

1. **Técnicas no Invasivas** - En el desarrollo de aplicaciones Web se han establecido colecciones de directrices (Souders, 2007) que generalmente se enfocan en controlar la estructura de los contenidos para una fácil lectura y comprensión por los usuarios. Dos técnicas muy socorridas para mejorar el desempeño de una aplicación Web son la compresión de información y el uso de caché en el cliente y el servidor Web (Souders, 2007).
2. **Prueba t – Student y Wilcoxon para muestras pareadas** - En ocasiones se desea experimentar sobre la efectividad de algún tratamiento en el que se emplean los mismos elementos pero en distintos momentos o bajo distintas condiciones, estas observaciones

son por su naturaleza formadas como muestras relacionadas o dependientes. La prueba *t* para muestras dependientes tiene el supuesto de que la distribución de las diferencias siga una distribución normal. Sin embargo, no siempre se cumple dicho requisito, para ésta situación puede emplearse la prueba de Wilcoxon para muestras relacionadas, es decir, es la prueba análoga no paramétrica a la prueba *t* para dos muestras relacionadas (Zar, 2007).

3. **Análisis de Covarianza** - Es un procedimiento para comparar medias de tratamientos que incorpora información sobre una variable cuantitativa  $x$ . Es una técnica que resulta útil para mejorar la precisión de un experimento. Supongamos que en un experimento, la variable de respuesta  $y$ , está relacionada linealmente con la variable independiente  $x$ . Además, el experimentador no puede controlar la variable  $x$ , pero puede medirla al mismo tiempo que a  $y$ . A  $x$  se le conoce como covariable o variable concomitante y no es aleatoria, su medición se realiza prácticamente sin error. Con el análisis de covarianza se busca adaptar el valor observado de la respuesta para tomar en cuenta el efecto de la variable concomitante. Si se no lleva a cabo dicho ajuste, la variable concomitante puede aumentar el cuadrado medio del error, con lo que hay mayor dificultad para detectar diferencias reales en las respuestas debidas a los tratamientos. Por lo tanto, el análisis de covarianza es un método para tomar en cuenta el efecto de alguna variable que no pueda ser controlada (Zar, 2007).

### 5.1.1. Configuración de la evaluación

Para activar el control de caché del cliente, se han incorporado las siguientes directivas de configuración del servidor Apache :

**control cache:**

```
<IfModule mod_expires.c>  
ExpiresDefault "access plus 1 month"  
FileETag None  
< /IfModule>
```

Para activar la compresión se utilizó el módulo Apache mod\_deflate que permite controlar el grado de compresión y su velocidad para no afectar el desempeño del servidor, las directivas utilizadas son:

**compresión:**

```
<IfModule mod_deflate.c>
AddOutputFilterByType DEFLATE text/ html text /css
application/javascript
BrowserMatch \wedgeMozilla /4gzip - only - text / html
BrowserMatch \wedgeMozilla /4 \.0[678]no - gzip
BrowserMatch \setminusminusbMSIE ! no - gzip ! gzip - only - text /html
</IfModule >
```

### 5.1.2. Proceso de experimentación

La experimentación tiene como objetivo determinar si el motor de inferencia difuso a través del Sistema Web (SWID) optimiza el rendimiento de la generación de modelos difusos. Por ello, se realiza una evaluación de las técnicas, métodos estadísticos y el propuesto.

### 5.1.3. Descripción del conjunto de datos

1. **Técnicas no Invasivas.** Para validar la configuración aplicada al servidor Web se ejecutaron 100, 150, 200, 250, 300, 350, 400, 450, 500, 550 y 600 peticiones de manera aleatoria con un nivel de concurrencia de 10 utilizando la herramienta ab (Apache HTTP server benchmarking tool,) (Apache, 2010).
2. **Prueba t – Student y Wilcoxon para muestras pareadas -** De igual manera para validar la configuración al servidor web con la compresión y sin la compresión para el numero de peticiones contra el tiempo promedio se utilizo la prueba t para muestras dependientes, ya que se cumple el supuesto de que la distribución de las diferencias proviene de una distribución normal. Sin embargo, no siempre se cumple dicho requisito, para ésta situación puede emplearse la prueba de Wilcoxon para muestras relacionadas, que es el caso para el numero de peticiones contra razón de transferencia (Zar, 2007).
3. **Análisis de Covarianza.** Con el propósito de comparar el rendimiento entre los algoritmos Chi y W (Riza, 2015), se utilizaron datos de un muestreo de los laboratorios de investigación marina de la universidad de Tasmania en Australia (Lichman, 2013), se

capturaron un total de 4,117 moluscos del género *Haliotis* (abulones). Las variables morfológicas de entrada fueron longitud, diámetro, altura, peso total, peso sin concha, peso de la víscera, peso de la concha y el número de anillos que fue la variable de salida (Tabla 5.5).

Se ejecutaron 100, 150, 200, 250, 300, 350 y 400 peticiones de manera aleatoria con un nivel de concurrencia de 10 utilizando la herramienta ab (Apache HTTP server benchmarking tool 2.3, Figura 5.1) (Apache, 2010). Se relacionó la variable número de peticiones (covariable) con el tiempo total de ejecución, y se utilizó un análisis de covarianza para verificar diferencias entre dos tipos de algoritmos con respecto al tiempo total de ejecución, el primero implementa las reglas de clasificación usando la técnica Chi's (FRBCS.CHI) (Chi, 1996) y el segundo las reglas de clasificación con un factor de peso (FRBCS.W) (Ishibuchi, 2001).

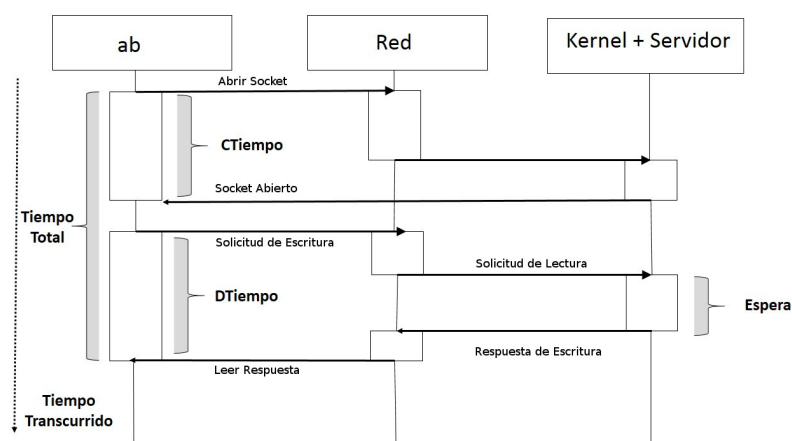


Figura 5.1 Diagrama de tiempos de procesamiento de ab, CTiempo : tiempo que establece la conexión y podemos enviar / escribir nuestra petición de servidor (que es tiempo de conexión a la red + tiempo en aplicación), DTiempo : cantidad de tiempo que se refiere a (tiempo de escribir) + (tiempo para la elaboración del servidor) + (tiempo para leer la respuesta).

Con respecto al tiempo total de ejecución, los parámetros de entrada para los dos algoritmos fueron 7 etiquetas y una función de pertenencia trapezoidal. Los resultados de las pruebas estadísticas fueron considerados significativos si  $P < 0.05$  y el paquete estadístico utilizado fue R 3.1.3 (Ihaka, 2009). El Modelo estadístico (ver Ec. 5.1) empleado en un análisis de covarianza es el siguiente:

$$y_{ij} = \mu + \tau_i + \beta(x_{ij} - \bar{x}_{..}) + \varepsilon_{ij} \quad (5.1)$$



en donde  $y_{ij}$  es el  $j$ -ésimo tiempo de procesamiento bajo el  $i$ -ésimo algoritmo o nivel del único factor,  $x_{ij}$  es el valor del número de peticiones llamada covariable o variable concomitante correspondiente a  $y_{ij}$  ( $ij$ -ésimo ensayo),  $\bar{x}_{..}$  es el promedio de las  $x_{ij}$ ,  $\tau_i$  es el efecto del  $i$ -ésimo algoritmo,  $\beta$  es el coeficiente de regresión lineal que indica la dependencia entre  $y_{ij}$  y  $x_{ij}$ ,  $\varepsilon_{ij}$  es un componente de error aleatorio. Suposiciones: Los errores  $\varepsilon_{ij}$  son NID  $(0, \sigma^2)$ , la pendiente  $\beta$  es diferente de cero, la relación real entre  $x_{ij}$  y  $y_{ij}$  es lineal, la variable concomitante  $x_{ij}$  no depende de los tratamientos.

#### 5.1.4. Rendimiento

Los resultados obtenidos al simular las peticiones por algoritmo se analizan a continuación. La figura 5.2 muestra el número de peticiones (100, 150, 200, 250, 300, 350, 400, 450, 500, 550 y 600) y el tiempo promedio por segundo, resultó que el promedio cuando se aplica la compresión es menor ( $W = 66$ ,  $P = 0.0005$ ).

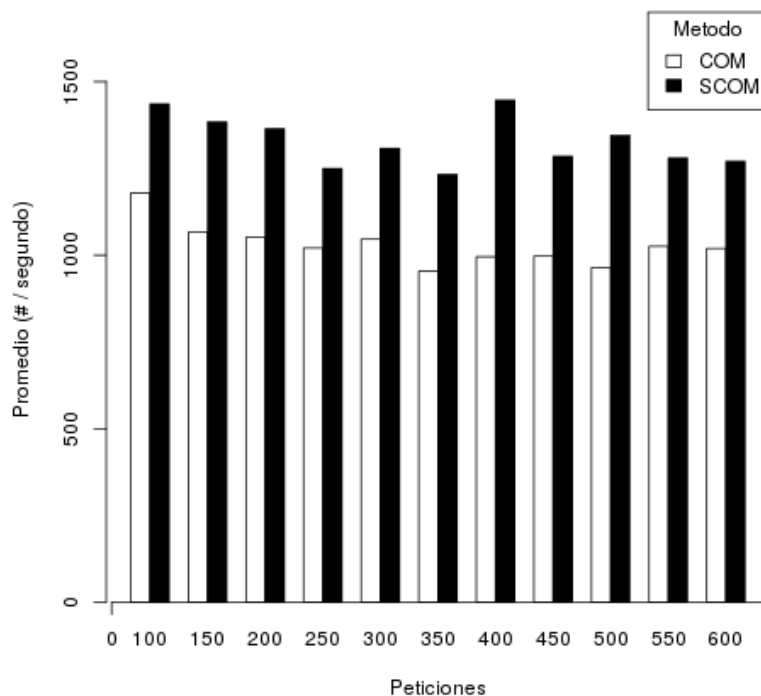


Figura 5.2 Número de peticiones contra el tiempo promedio utilizando compresión (COM) y sin compresión (SCOM).

La figura 5.3 muestra la razón de transferencia por petición (100, 150, 200, 250, 300, 350, 400, 450, 500, 550 y 600) puede verse que la razón de transferencia cuando se aplica la compresión es menor ( $t = 53.1698$ ,  $gl = 10$ ,  $P < 0.0001$ ).

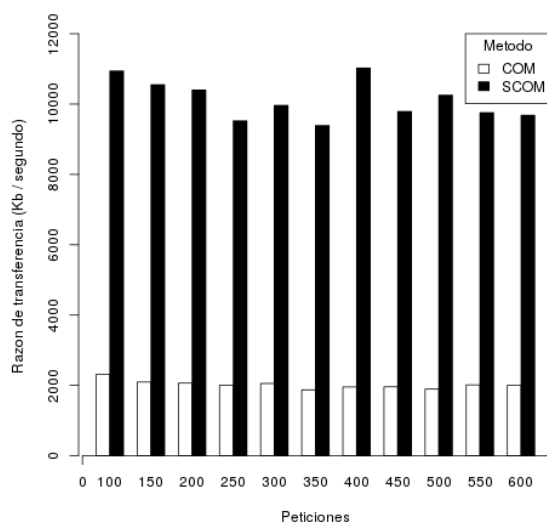


Figura 5.3 Número de peticiones contra razón de transferencia utilizando compresión (COM) y sin compresión (SCOM).

La figura 5.4 evalúa el tamaño de la página considerando todos los archivos (HTML, Java Script,css, imágenes, etc.) utilizando la compresión y sin la compresión la cual resultó de 1657 bytes y de 7471 respectivamente, esto significa que hay una reducción del 88% aproximadamente.

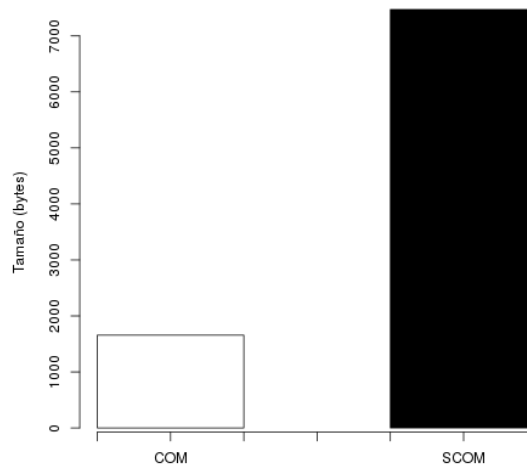


Figura 5.4 Gráfica de tamaño de página cuando se aplica la compresión (COM) y sin compresión (SCOM).

En la tabla 5.1 se observan las etiquetas lingüísticas para cada variable de entrada del sistema difuso a generar para los datos del abulón:

Tabla 5.1 Etiquetas lingüísticas.

<b>EExtra pequeña</b>	<b>Extra pequeña</b>	<b>pequeña</b>	<b>Mediana</b>	<b>Grande</b>	<b>Extra Grande</b>	<b>EExtra Grande</b>
XXS	XS	S	M	L	XL	XXL

La tabla difusa obtenida al ejecutar el entrenamiento del modelo para la variable longitud se encuentra en la tabla 5.2:

Tabla 5.2 Tabla difusa para función de pertenencia trapezoidal sobre la longitud.

	<b>XXS</b>	<b>XS</b>	<b>S</b>	<b>M</b>	<b>L</b>	<b>XL</b>	<b>XXL</b>
a	0	0.1277	0.2555	0.3833	0.5111	0.6388	0.7777
b	0.1111	0.2388	0.3666	0.4944	0.6222	0.7500	0.8888
c	0.2222	0.2944	0.4222	0.55	0.6777	0.8055	1
d	NA	0.4055	0.5333	0.6611	0.7888	0.9166	NA

La tabla 5.3 muestra la predicción (ver Ec. 2.9) del número de anillos con respecto a la Longitud, Diámetro, Altura, Peso total, Peso sin concha, Peso de la víscera y Peso de la cáscara.

Tabla 5.3 Tabla de predicción para un caso particular de abulón.

Regla	Anillos
SI Longitud es S y Diámetro es S y Altura es XXS y Peso total es XXS y Peso sin concha es XXS y Peso de la víscera es XXS y Peso de la cáscara es XXS	26

La figura 5.5a Longitud, 5.5b Diámetro, 5.5c Altura, 5.5d Peso total, 5.5e Peso sin concha, 5.5f Peso de la víscera y 5.5g Peso de la cáscara representa cada variable medida del abulón con sus dominios difusos .

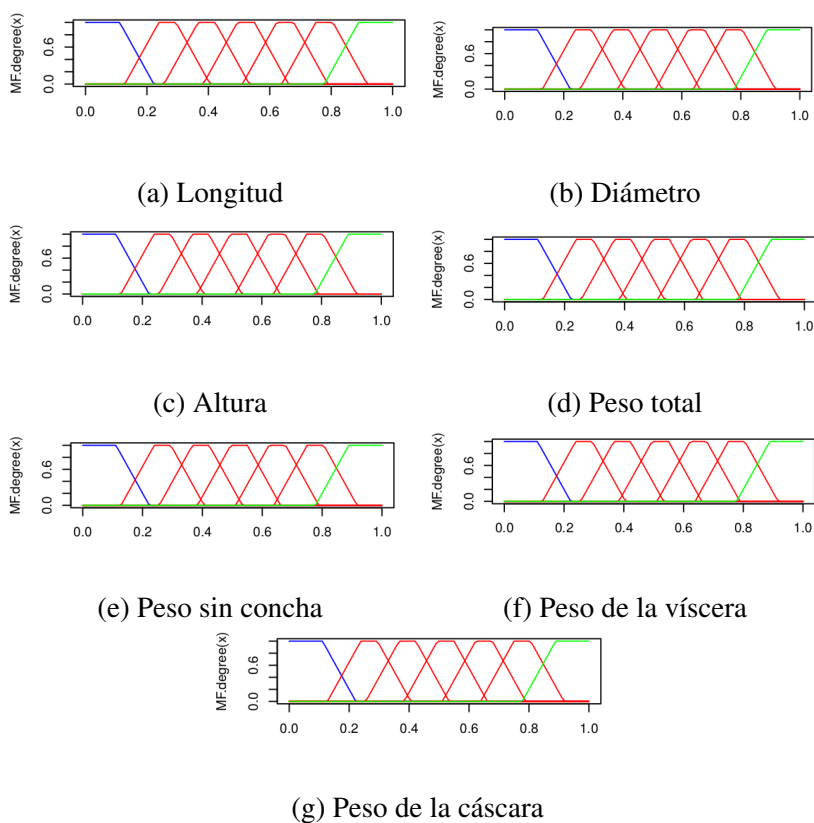


Figura 5.5 Gráfica de las función de pertenencia trapezoidal.

La figura 5.6 muestra el número de peticiones (100,150, 200, 250, 300, 350, 400) y el tiempo de la prueba realizada en milisegundos, puede notarse que a mayor número de peticiones el tiempo de ejecución se incrementa para los dos tipos de algoritmos.

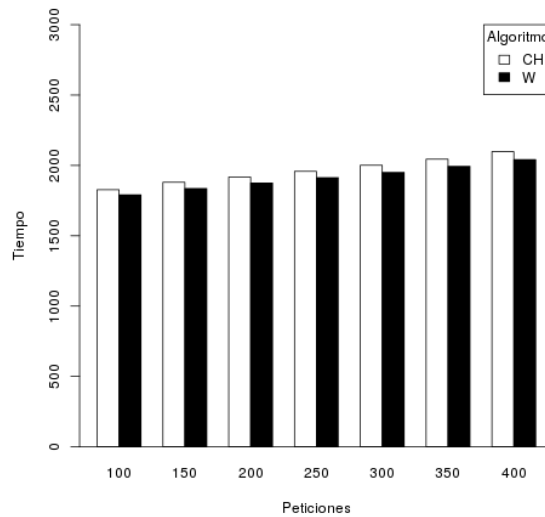


Figura 5.6 Gráfica de Número de peticiones contra el tiempo por algoritmo.

Se cumplió tanto el supuesto de normalidad (prueba de bondad de ajuste de Kolmogorov Smirnov :  $D = 0.0516$ ,  $P = 0.0805$ , figura 5.7) como el de igualdad de varianzas (figura 5.8) para los residuos.

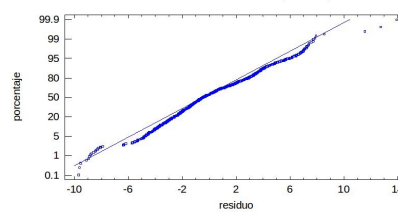


Figura 5.7 Gráfica de probabilidad normal.

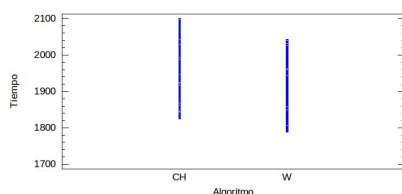


Figura 5.8 Gráfica de residuos vs Algoritmos.

Del análisis de covarianza se determinó una relación lineal del tiempo de procesamiento con el número de peticiones ( $F = 245,577.53$ ,  $P < 0.0001$ ) y diferencias significativas entre algoritmos respecto al tiempo de procesamiento ( $F = 24,246.08$ ,  $P < 0.0001$ , figura 5.9).

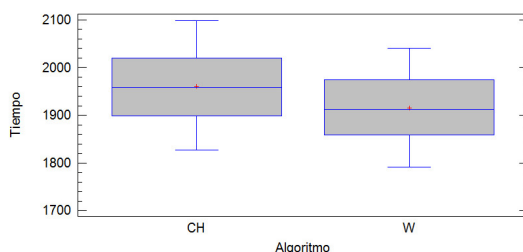


Figura 5.9 Tiempos promedio de procesamiento por algoritmo.

## 5.2. VALIDACIÓN DE MODELOS PREDICTIVOS EN BIOLOGÍA

### 5.2.1. Configuración de la evaluación

En las pruebas, se utilizó el 80% de los casos para el aprendizaje y 20% para evaluar las predicciones de las funciones de pertenencia triangular, trapezoidal y gaussiana. Todos los experimentos se ejecutaron en una computadora Toshiba Satélite con procesador AMD A6, 4 GB de RAM y sistema operativo Debian 8. Con respecto a las predicciones se utilizó el algoritmo de reglas de clasificación con un factor de peso (FRBCS.W) (Ishibuchi, 2001), los parámetros de entrada para el algoritmo fue de 3 y 7 etiquetas en el caso discreto y continuo respectivamente. Las pruebas estadísticas se consideran significativas cuando  $P < 0,05$  y el software estadístico utilizado fue R 3.1.3 (Ihaka, 2009).

### 5.2.2. Descripción del conjunto de datos

#### 1. Evaluación caso discreto en el área de la Biología -

Con el propósito de evaluar el desempeño de modelos de inferencia difusa para el caso discreto, se diseñó un experimento basado en la evaluación de predicciones correctas e incorrectas de funciones de pertenencia triangular, trapezoidal, gaussiana. Se trabajó con la base de datos de plantas Iris, creada por R. A. Fisher (Ver cuadro 5.5). Esta base de datos (iris.dat archivo) se puede obtener por Anonymous FTP en decsai.ugr.es cd pub / arai / tech\_rep / zurita. El conjunto de datos consta de 3 clases de 50 casos cada uno, donde cada clase refiere a un tipo de planta de Iris. El primer atributo indica la longitud del sépalo, la segunda anchura del sépalo, la tercera longitud del pétalo y el cuarto indica la anchura del pétalo, estas representan las variables de entrada. Todos los atributos se miden en centímetros (cm). La clase puede ser iris setosa, iris versicolor o iris virginica que representan la variables de salida. Hay valores para todos los atributos en la base de datos Iris.

Tabla 5.4 Variables morfológicas del iris.

Nombre	Descripción	Unidades	Mínimo	Máximo
Largo de Sépalo	continua	cm	4.3	7.9
Ancho de Sépalo	continua	cm	2.0	4.4
Largo de Pétalo	continua	cm	1.0	6.9
Ancho de Pétalo	continua	cm	0.1	2.5
Clase	nominal	1 = Setosa 2 = Versicolor 3 = Virginica		

2. **Evaluación caso continuo en el área de la Biología** - Para el caso continuo se utilizaron datos de un muestreo de Abulón (*Haliotis* spp., muestra de 120) de los laboratorios de investigación marina de la universidad de Tasmania en Australia (Lichman, 2013), las variables morfológicas de entrada fueron el sexo (M = Macho, F = Hembra e I = infante), longitud, diámetro, altura, peso total, peso sin concha, peso de la víscera, peso de la concha y el número de anillos que fue la variable de salida (Tabla 5.5).

Tabla 5.5 Variables morfológicas del abulón.

Nombre	Descripción	Unidades	Mínimo	Máximo
Sexo	nominal	M, F e I		
Longitud	continua	mm	0.075	0.815
Diámetro	continua	mm	0.055	0.650
Altura	continua	mm	0.000	1.130
Peso total	continua	gr	0.002	2.826
Peso sin concha	continua	gr	0.001	1.488
Peso de la víscera	continua	gr	0.0005	0.760
Peso de la concha	continua	gr	0.0015	1.005
Anillos	razón	número de anillos	1	29

### 5.2.3. Resultados

En el caso discreto los rendimientos entre las funciones de pertenencia no difirió significativamente ( $Q = 4$ ,  $gl = 2$ ,  $P = 0.1353$ , Figura 5.10), con base en la proporción de éxito se obtuvo el orden : trapezoidal, gaussiana y triangular.

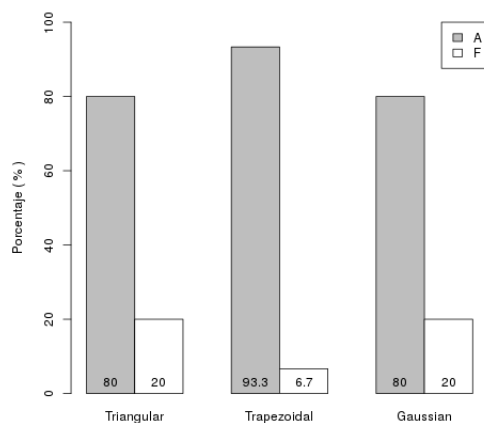


Figura 5.10 Resultados del experimento para funciones de pertenencia triangular, trapezoidal y gaussiana, con A = Aciertos y F = fallos.

La tabla difusa obtenida al ejecutar el entrenamiento del modelo para la función de pertenencia trapezoidal (ver tabla 5.6).



Tabla 5.6 Tabla difusa de largo de sépalo para la función de pertenencia trapezoidal.

<b>Valores Lingüísticos</b>	<b>(a, b, c, d)</b>
S : pequeño	(0, 0.2, 0.4, 0)
M : Mediano	(0.23, 0.43, 0.53, 0.73)
L : Largo	(0.6, 0.8, 1, 0.0)

La tabla 5.7 muestra la predicción del tipo de especie para datos de iris.

Tabla 5.7 Tabla de predicción para un caso particular de iris.

<b>Regla</b>	<b>Especie</b>
<b>SI</b> Longitud de sépalo es S y Anchura sépalo es M y Longitud de pétalo es S y Anchura del pétalo mediana es S	<b>ENTONCES</b> especie es 1

Para el caso continuo los rendimientos entre las funciones de pertenencia difirió significativamente ( $F_r = 9.3000$ ,  $P = 0.0028$ , Figura 5.11), la diferencia mas pequeñas ocurre con la función gaussiana ya que tiene un rango promedio mas bajo.

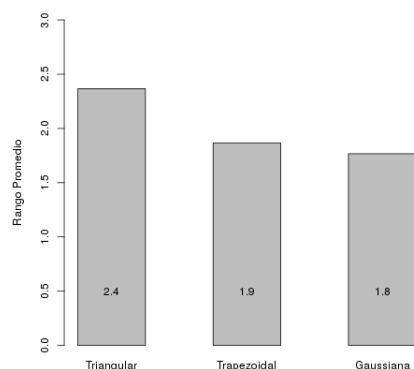


Figura 5.11 Resultados del experimento para funciones de pertenencia triangular, trapezoidal y gaussiana.

De las comparaciones múltiples (ver tabla 5.8, *rango promedio con igual letra no difieren* ( $P > 0,05$ )) se obtuvo que el rango promedio de la función de pertenencia trapezoidal y gaussiana no difieren significativamente, las diferencias se encontraron entre la triangular y las funciones de pertenencia trapezoidal y gaussiana.

Tabla 5.8 Tabla de comparación.

Función	Rango promedio
Triangular	2,4 <sup>a</sup>
Trapezoidal	1,9 <sup>b</sup>
Gaussiana	1,8 <sup>b</sup>

La tabla difusa obtenida al ejecutar el entrenamiento del modelo para función de pertenencia gaussiana (ver tabla 5.9).

Tabla 5.9 Tabla difusa de longitud de abulón.

Valores Lingüísticos	(a, b)
XXS: EExtra pequeña	(0, 0.058)
XS: Extra pequeña	(0.16, 0.058)
S: Pequeña	(0.33, 0.058)
M: Mediana	(0.50, 0.058)
L: Grande	(0.66, 0.058)
XL: Extra Grande	(0.83, 0.058)
XXL: EExtra Grande	(1.0, 0.058)

La tabla 5.10 muestra la predicción del número de anillos para datos de abulón.

Tabla 5.10 Tabla de predicción para un caso particular de abulón.

Regla	Anillos
SI Longitud es S y Diámetro es S y Altura es XXS y Peso total es XXS y Peso sin concha es XXS y Peso de la víscera es XXS y Peso de la cáscara es XXS	26

## 5.3. VALIDACIÓN DE MODELOS PREDICTIVOS EN MEDICINA

### 5.3.1. Configuración de la evaluación

En las pruebas, se utilizó el 80% de los casos para el aprendizaje y 20% para evaluar las predicciones de las funciones de pertenencia triangular, trapezoidal y gaussiana. Todos los experimentos se ejecutaron en una computadora Toshiba satélite con procesador AMD A6, 4 GB de RAM y sistema operativo Debian 8. Con respecto a las predicciones se utilizó el algoritmo de reglas de clasificación con un factor de peso (FRBCS.W) (Ishibuchi, 2001), los parámetros de entrada para el algoritmo fue de 6 etiquetas y función de pertenencia la triangular, trapezoidal y gaussiana. Las pruebas estadísticas se consideran significativas cuando  $P < 0,05$  y el software estadístico utilizado fue R 3.1.3 (Ihaka, 2009).

### 5.3.2. Descripción del conjunto de datos

Con el propósito de evaluar el desempeño de modelos de inferencia difusa en el área de medicina, se diseñó un experimento basado en la evaluación de predicciones correctas e incorrectas de funciones de pertenencia triangular, trapezoidal, gaussiana. Se trabajó con la base de datos diagnóstico del cáncer de pecho en Wisconsin (en inglés WDBC, Ver cuadro 5.11). Las variables medidas en tumores de 569 pacientes son radio (media de las distancias desde el centro a los puntos en el perímetro), textura (desviación estándar de los valores de la escala de grises), perímetro, área, suavidad (variación local de las longitudes de radio), de compacidad ( $\text{perímetro}^2 / \text{zona} - 1.0$ ), de concavidad (severidad de las partes cóncavas del contorno), puntos cóncavos (número de partes cóncavas del contorno), la simetría, la dimensión fractal ("aproximación coastline" - 1) que representan las variables de entrada, para este experimento el Diagnóstico (M = maligno, B = benigno) representa la variable de salida.

Tabla 5.11 Variables de tumores.

Nombre	Descripción	Unidades	Mínimo	Máximo
radio	continua	cm	6.981	28.110
textura	continua	cm	9.71	39.28
perímetro	continua	cm	43.79	188.50
área	continua	cm	143.5	2501.0
suavidad	continua	cm	0.05263	0.16340
compacidad	continua	cm	0.01938	0.34540
concavidad	continua	cm	0.0000	0.4268
puntos cóncavos	continua	cm	0.0000	0.2012
la simetría	continua	cm	0.106	0.304
la dimensión fractal	continua	cm	0.04996	0.09744
Diagnóstico	nominal	1 = Maligno 2 = Benigno		

### 5.3.3. Resultados

Los rendimientos entre las funciones de pertenencia no difirió significativamente ( $Q = 3.8537$ ,  $gl = 2$ ,  $P = 0.1456$ , Figura 5.12), con base en la proporción de éxito, se obtuvo el orden : gaussiana, trapezoidal y triangular.

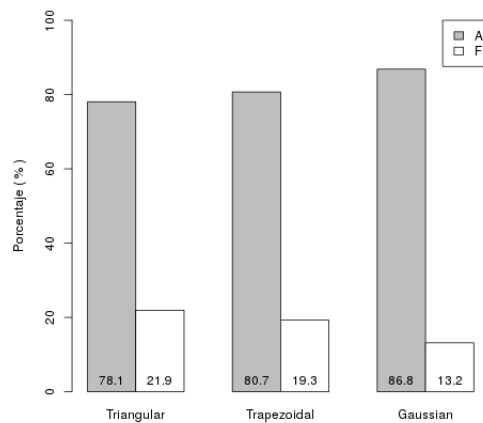


Figura 5.12 Resultados del experimento para funciones de pertenencia triangular, trapezoidal y gaussiana, con A = Aciertos y F = fallos.

La tabla difusa obtenida al ejecutar el entrenamiento del modelo para la función de pertenencia gaussiana (ver tabla 5.12)

Tabla 5.12 Tabla difusa para la función de pertenencia gaussiana

Valores Lingüísticos	(a, b)
S : Pequeño	(0, 0.175)
M : Mediano	(0.5, 0.175)
L : Largo	(1.0, 0.175)

En la tabla 5.13 se observan las predicciones utilizando la función de pertenencia gaussiana.

Tabla 5.13 Tabla de predicción para un caso particular en tumores.

<b>Regla</b>	<b>Diagnóstico</b>
SI radio es M y textura es L y perímetro es M y área es M y suavidad es M y compacidad es M y concavidad es M y puntos cóncavos es M y simetría es M y dimensión fractal es S	<b>ENTONCES</b> tumor es 1
SI radio es S y textura es S y perímetro es S y área es S y suavidad es M y compacidad es S y concavidad es S y puntos cóncavos es S y simetría es M y dimensión fractal es M	<b>ENTONCES</b> tumor es 2

## 5.4. VALIDACIÓN DE MODELOS PREDICTIVOS EN EDUCACIÓN

### 5.4.1. Configuración de la evaluación

En las pruebas, se utilizó el 80 % de los casos para el aprendizaje y 20 % para evaluar las predicciones de las funciones de pertenencia triangular, trapezoidal y gaussiana. Todos los experimentos se ejecutaron en una computadora Toshiba Satélite con procesador AMD A6, 4 GB de RAM y sistema operativo Debian 8. Con respecto a las predicciones se utilizó el algoritmo de reglas de clasificación con un factor de peso (FRBCS.W) (Ishibuchi, 2001), los parámetros de entrada para el algoritmo fue de 3 etiquetas y función de pertenencia la triangular, trapezoidal y gaussiana. Las pruebas estadísticas se consideran significativas cuando  $P < 0,05$  y el software estadístico utilizado fue R 3.1.3 (Ihaka, 2009).

### 5.4.2. Descripción del conjunto de datos

Con el propósito de evaluar el desempeño de modelos de inferencia difusa utilizando el sistema SWID en el área de la educación. Se trabajo con datos de 10, 300 alumnos de 6to año de primaria en el estado de Yucatán. Las variables medidas en los alumnos son habilidad matemática (hm), verbal (hv) y de razonamiento (hr) que representan las variable de entrada

(Ver tabla 5.14). Para este experimento la variable de salida es género (1 = Masculino, 2 = Femenino).

Tabla 5.14 Variables de habilidades.

Habilidad	Descripción	Unidades	Mínimo	Máximo
Matemática	continua	%	8.333	79.1666
Verbal	continua	%	0	100
Razonamiento	continua	%	12	79.1666
Género	nominal	1 = Masculino 2 = Femenino		

### 5.4.3. Resultados

Los rendimientos de la capacidad predictiva para las funciones de pertenencia no difirió significativamente ( $Q = 0.6504$ ,  $g1 = 2$ ,  $P = 0.7224$ , Fig. 5.13).

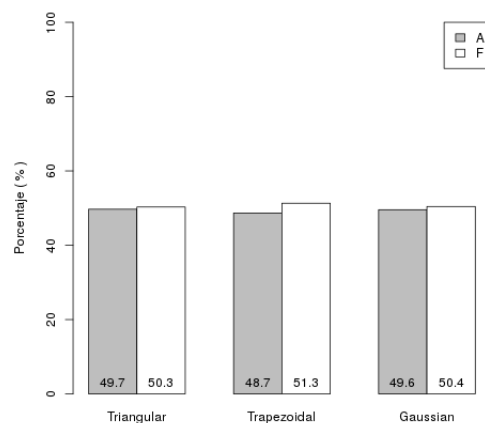


Figura 5.13 Resultados del experimento para variable de salida genero.

La tabla difusa obtenida al ejecutar el entrenamiento del modelo para la variable habilidad matemática para función de pertenencia triangular (ver tabla 5.15), trapezoidal (ver tabla 5.16) y gaussiana (ver tabla 5.17):

Tabla 5.15 Tabla difusa de habilidad matemática para función de pertenencia triangular.

Valores Lingüísticos	(a, b, c)
S : Pequeño	(0, 0, 0.5)
M : Mediano	(0, 0.5, 1.0)
L : Largo	(0.5, 1.0, 1.0)

Tabla 5.16 Tabla difusa de habilidad matemática para función de pertenencia trapezoidal.

Valores Lingüísticos	(a, b, c, d)
S : Pequeño	(0, 0.2, 0.4, 0)
M : Mediano	(0.23, 0.43, 0.53, 0.73)
L : Largo	(0.6, 0.8, 1.0, 0)

Tabla 5.17 Tabla difusa de habilidad matemática para función de pertenencia gaussiana.

Valores Lingüísticos	(a, b)
S : Pequeño	(0, 0.175)
M : Mediano	(0.5, 0.175)
L : Largo	(1.0, 0.175)

En la tabla 5.18 se observan las predicciones utilizando la función de pertenencia triangular.

Tabla 5.18 Tabla de predicción para un caso particular en habilidades.

Regla	Género
SI hm es M y hr es M y hv es M	ENTONCES sexo es 1
SI hm es M y hr es M y hv es M	ENTONCES sexo es 2



## 5.5. CONCLUSIONES

En este capítulo se ha presentado una propuesta no invasiva para mejorar el desempeño del Sistema SWID en un servidor Web, en términos de sus tiempos de respuesta. Los resultados resaltan la importancia de una adecuada configuración del servidor Web para garantizar una mejor y más rápida transferencia de información. El uso del servidor Web Apache y la herramienta ab facilitó en gran medida la aplicación de las técnicas, esto gracias a su arquitectura modular que permite incorporar nuevas características, como la compresión, el control de encabezados y otros elementos que favorecen su optimización. Muchas bases de datos y herramientas de análisis de datos se han creado para la investigación bioinformática, generalmente se combinan para ser utilizados en las tareas de recuperación de datos y de análisis de datos por algunos biólogos y científicos en bioinformática, SWID puede ser usado como herramienta de investigación no solo en el área de la bioinformática sino también en otras disciplinas como Medicina, Psicología, Educación, entre otras (Uc, 2015). Como trabajo futuro se propone evaluar el rendimiento de SWID variando el número de etiquetas y proponer otros tipos de algoritmos en el área de la lógica difusa.

El experimento muestra que el sistema SWID es útil para modelar procesos de inferencia difusa de manera eficiente ya que los tiempos medios de procesamiento fueron de 1960.11 y 1914.98 milisegundos para los algoritmos CH y W respectivamente; el que resultó con mejor rendimiento fue el algoritmo W. Cabe señalar que el lenguaje de programación R es útil como motor de inferencia difusa en tecnologías web con PHP, el uso del servidor web Apache y la herramienta ab facilitó en gran medida la evaluación de SWID.

Con respecto a la evaluación de las predicciones para el caso continuo y discreto, se han presentado los experimentos y resultados obtenidos al evaluar las predicciones de los modelos de inferencia difusa considerando las funciones de pertenencia trapezoidal, triangular y gaussiana utilizando reglas de clasificación. Se han utilizado colecciones de prueba y métricas estándares para garantizar la veracidad de los resultados. En el caso de género la capacidad predictiva se comportó de una manera homogénea, la elección de cualquiera de las tres funciones de pertenencia sería lo adecuado.

Los resultados obtenidos en el caso discreto y continuo de los experimentos muestran un impacto positivo del enfoque presentado en esta investigación. Al caracterizar las predicciones de los modelos de inferencia difusa se transforman en fuente importante de información cualitativa y cuantitativa de las acciones que se llevan a cabo, lo que resulta importante para el desarrollo de nuevas técnicas o incorporar funcionalidades que faciliten la evaluación de los modelos de inferencia difusa. Como trabajo futuro se propone evaluar otros algoritmos en la modelación difusa y variar el número de etiquetas, así como el tipo de función de

pertenencia e.g la Sigmoid y Bell, así como proponer otros tipos de algoritmos en el área de la lógica difusa.

# Capítulo 6

## CONCLUSIONES

Este capítulo tiene como finalidad presentar los resultados de una valoración global de los productos obtenidos en esta investigación. Se inicia analizando la consecución de los objetivos planteados en el trabajo, para proseguir con una revisión de las principales aportaciones generadas como resultado de la investigación. Finalmente se presentan las distintas publicaciones realizadas en estos dos años que corroboran los resultados obtenidos y se establecen las líneas futuras de investigación.

### 6.1. ANÁLISIS DE LA CONSECUCIÓN DE LOS OBJETIVOS

En el capítulo 1 de este documento se expusieron los objetivos parciales que se pretendían cumplir para lograr el objetivo general que enmarca esta tesis que es:

<p><b><i>Objetivo General:</i></b> Desarrollar un motor de inferencia basado en lógica difusa que simplifique la generación de modelos de inferencia difusa con respecto a la detección de patrones de comportamiento en datos.</p>
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

A continuación se hace una valoración de la consecución de los objetivos específicos presentados en el primer capítulo:

1. Investigar los modelos de inferencia disponibles en el área de lógica difusa. Sistemas difusos estáticos, basados con un factor de peso, criterios de clasificación, metodología y algoritmos de aprendizaje difuso, evaluación del desempeño considerando la variable de salida discreta y continua.

2. Seleccionar el modelo difuso que permitirá obtener resultados óptimos en términos de cálculos computacionales.
3. Diseñar una arquitectura Web que faciliten la integración de los elementos de la aplicación.
4. Desarrollar una aplicación web que permita generar modelos de inferencia difusa y realizar predicciones con el modelo generado.
5. Definir un conjunto de instrumentos e indicadores para medir la eficiencia de los modelos de inferencia difusa generados.
6. Plantear nuevos algoritmos de inferencia difusa que permita evaluar la capacidad predictiva de las funciones de pertenencia triangular, trapezoidal y gaussiana.
7. Analizar experimentalmente el funcionamiento de cada uno de los elementos de la aplicación web.

Considerando que se ha logrado la consecución de los objetivos específicos, puede decirse que el objetivo general de esta investigación se ha cumplido, puesto que se ha definido, desarrollado y evaluado.

La hipótesis presentada en el capítulo 1 se basaba en el cumplimiento del objetivo general. De manera que a partir de los resultados obtenidos, puede concluirse que la hipótesis ha quedado demostrada y por tanto puede afirmarse que :

El desarrollo un sistema web y un motor de inferencia difusa puede facilitar la generación de modelos de predicción y clasificación en el área de la lógica difusa.

## **6.2. ANÁLISIS DE LAS APORTACIONES**

En este trabajo se ha especificado un marco metodológico que utiliza un motor de inferencia basado en lógica difusa para evaluar y facilitar su generación en áreas como la Biología, Medicina, Educación. Se han presentado los distintos elementos que conforman la propuesta y cuya realización permiten lograr el objetivo planteado.

El concepto de lógica difusa ha resultado valioso para definir la propuesta. Los procesos y componentes del motor de inferencia difuso han sido diseñados considerando la importancia de contar con una herramienta que facilite la evaluación de las funciones de pertenencia. Esto permite recomendar la función adecuada en una determinado situación experimental.

Este motor de inferencia difuso ha sido implementado considerando una metodología estadística para garantizar una adecuada solución del problema o experimento. Si bien se

ha enfocado a resolver problemas sin considerar las complejidades computacionales, es necesario entender el concepto de lógica difusa y su arquitectura.

La propuesta de diferenciar entre variables de salida discreta y continua ha resultado importante en el momento de crear un modelo en la lógica difusa, más aun el planteamiento de los algoritmos representan una alternativa computacional en diversas áreas de la ciencia. El servicio SWID es el resultado de implementar las metodologías y modelos descritos para mejorar la interacción entre el usuario y el motor de inferencia. Los componentes de SWID al estar basados en servicios, garantizan la reutilización y la adaptabilidad con otros sistemas. Las tecnologías y estándares utilizados garantizan la flexibilidad de la arquitectura para incorporar nuevas funcionalidades y características empleadas en la generación de modelos difusos. De los experimentos realizados se ha comprobado la efectividad de SWID en términos de rendimiento para que sean utilizados en la práctica.

### **6.3. CONTRASTE DE RESULTADOS**

Con el propósito de valorar los distintos resultados obtenidos de la investigación, se han generado distintas publicaciones, las cuales han sido presentadas como artículos de revista (2) y participación en congreso (1). A continuación se lista cada uno de ellos agrupados por categorías y ordenados cronológicamente en forma descendente. Las temáticas de las publicaciones versan sobre la generación de modelos difusos para la detección de patrones de comportamiento.

#### **6.3.1. Artículos en revistas**

- [1] Miguel Uc, Víctor Menéndez, Salvador Medina, “*Optimización del desempeño de un Sistema Web para Modelos de Inferencia Difusa mediante Técnicas no invasivas*”, Consejo Nacional de Acreditación en Informática y Computación, 2395-9061, 27–31, 2016.
- [2] Miguel Uc, Víctor Menéndez, Salvador Medina, “*Evaluación del Rendimiento de un Sistema Web para Modelos de Inferencia Difusa*”, Abstraction and Application, 14, 20–30, 2016.

#### **6.3.2. Artículos en revisión**

- [3] Miguel Uc, Víctor Menéndez, Salvador Medina, “*Performance Evaluation Model of Fuzzy Inference Rules Ranking*”, Computación y Sistemas.

### 6.3.3. Memorias de congresos

- [4] Miguel Uc, Víctor Menéndez, Salvador Medina, “*Optimización del desempeño de un Sistema Web para Modelos de Inferencia Difusa mediante Técnicas no invasivas*”, XVIII Congreso Nacional y el XIV Congreso Internacional de Informática y Computación.
- [5] Miguel Uc, Víctor Menéndez, Salvador Medina, “*Aplicación de Reglas difusas basada en sistemas de clasificación en Bionformática*”, Encuentro Universitario de Sistemas Computacionales EUSICS, 20-27, 2016.

### 6.3.4. Seminarios

- [6] Miguel Uc, Víctor Menéndez, Salvador Medina, “*Aplicación de Reglas difusas basada en sistemas de clasificación en Bionformática*”, Enmarcado dentro del primer ciclo de seminarios del IIMAS-UNAM, Parque Científico y Tecnológico de Yucatán, MAYO – JUNIO 2016.

## 6.4. TRABAJO A FUTURO

Previamente se ha presentado un análisis del cumplimiento de los objetivos inicialmente planteados, así como una revisión de las aportaciones conseguidas. A lo largo del desarrollo de esta investigación se han presentado diferentes líneas de trabajo que permiten el desarrollo, ampliación y mejora de los resultados obtenidos a este momento. Derivado de ese conocimiento se identifican mejoras orientadas a la implementación (prácticas) y a las técnicas aplicadas en los modelos (teóricas).

### 6.4.1. Mejoras en las técnicas utilizadas

- **Estudio de otras técnicas de evaluación de las predicciones en el caso continuo.** Se han empleado metodologías estadísticas que involucran técnicas no paramétricas, esto da la posibilidad de explorar técnicas paramétricas, bootstrap y montecarlo para evaluar las funciones de pertenencia.
- **Mejorar los tiempos de estimación de los parámetros en las funciones de pertenencia.** Con la finalidad de mejorar los tiempos de estimación de los parámetros en la funciones de pertenencia se propone utilizar computo paralelo y distribuido.
- **Incorporar nuevas funciones de pertenencia en modelos de inferencia difuso.** Existen otros tipos de función de pertenencia como por ejemplo la sigmoid y la Bell,

la generación de este tipo de funciones desarrollará nuevos mecanismos de predicción tanto en el caso continuo y discreto.

#### 6.4.2. Mejoras en la implementación

- **Analizar el impacto de las mejoras desde un enfoque de usuario final.** Realizar experimentos fuera de línea que den a conocer la utilidad percibida por el usuario sobre el sistema y la metodología de evaluación en las funciones de pertenencia.
- **Realizar el modelo ontológico de perfiles de usuario.** Con la finalidad de mejorar la representación conceptual de los perfiles de usuarios, es necesario realizar un modelo ontológico para la representación de las relaciones entre diferentes tipos de usuarios y las acciones que realizan en el sistema.





# Índice de figuras

1.1. Proceso cíclico Investigación . . . . .	5
1.2. Taxonomía de software de sistemas difusos . . . . .	5
2.1. Representación de un term set . . . . .	15
2.2. Gráfica de función de pertenencia triangular . . . . .	20
2.3. Gráfica de función de pertenencia trapezoidal . . . . .	20
2.4. Gráfica de función de pertenencia gaussiana . . . . .	21
2.5. Sistema basado en lógica difusa. . . . .	26
2.6. Taxonomía de las clasificaciones empleadas por el usuario. . . . .	29
3.1. Sistema desconocido de caja negra . . . . .	34
3.2. Particiones difusas con dos entradas . . . . .	39
4.1. Flujo de Datos en SWID . . . . .	48
4.2. Arquitectura SaaS . . . . .	49
4.3. Diagrama de Secuencia SaaS . . . . .	49
4.4. Diagrama de secuencia en SWID . . . . .	50
4.5. Diagrama de secuencia caso continuo . . . . .	50
4.6. Diagrama de secuencia caso discreto . . . . .	51
4.7. Modelo de navegación . . . . .	52
4.8. Interfaz Web para evaluar modelo difuso discreto . . . . .	52
4.9. Interfaz Web para evaluar modelo difuso continuo . . . . .	53
4.10. Interfaz Web para generación de modelo difuso . . . . .	53
4.11. Interfaz Web para evaluación de capacidad predictiva discreta . . . . .	54
4.12. Interfaz Web para evaluación de capacidad predictiva continua . . . . .	54
4.13. Interfaz Web de función de pertenencia triangular . . . . .	55
4.14. Interfaz Web de función de pertenencia trapezoidal . . . . .	55
4.15. Interfaz Web de función de pertenencia gaussiana . . . . .	56

---

5.1. Diagrama de tiempos de procesamiento de ab . . . . .	60
5.2. Peticiones por segundo . . . . .	61
5.3. Razón de transferencia por segundo . . . . .	62
5.4. Gráfica de tamaño de página . . . . .	63
5.5. Gráfica de las función de pertenencia trapezoidal. . . . .	64
5.6. Gráfica de número de peticiones contra el tiempo. . . . .	65
5.7. Gráfica de probabilidad normal. . . . .	65
5.8. Gráfica de residuos vs algoritmos. . . . .	66
5.9. Tiempos de procesamiento por algoritmo. . . . .	66
5.10. Resultados del experimento en biología caso discreto. . . . .	68
5.11. Resultados del experimento en biología caso continuo. . . . .	70
5.12. Resultados del experimento en medicina caso discreto. . . . .	73
5.13. Resultados del experimento para variable de salida genero. . . . .	75

# Índice de tablas

2.1. Tabla difusa de Longitud. . . . .	16
3.1. Técnicas de modelado. . . . .	36
3.2. Tabla de desempeño discreto. . . . .	40
3.3. Tabla experimental caso continuo. . . . .	41
5.1. Etiquetas lingüísticas. . . . .	63
5.2. Tabla difusa para función de pertenencia trapezoidal sobre la longitud. . . . .	63
5.3. Tabla de predicción para un caso particular de abulón. . . . .	64
5.4. Variables morfológicas del iris. . . . .	67
5.5. Variables morfológicas del abulón. . . . .	68
5.6. Tabla difusa de largo de sépalo para la función de pertenencia trapezoidal. . . . .	69
5.7. Tabla de predicción para un caso particular de iris. . . . .	69
5.8. Tabla de comparación. . . . .	70
5.9. Tabla difusa de longitud de abulón. . . . .	71
5.10. Tabla de predicción para un caso particular de abulón. . . . .	71
5.11. Variables de tumores. . . . .	72
5.12. Tabla difusa para la función de pertenencia gaussiana . . . . .	73
5.13. Tabla de predicción para un caso particular en tumores. . . . .	74
5.14. Variables de habilidades. . . . .	75
5.15. Tabla difusa de habilidad matemática para función de pertenencia triangular. . . . .	75
5.16. Tabla difusa de habilidad matemática para función de pertenencia trapezoidal. . . . .	76
5.17. Tabla difusa de habilidad matemática para función de pertenencia gaussiana. . . . .	76
5.18. Tabla de predicción para un caso particular en habilidades. . . . .	76



# Bibliografía

- [Abiyev, 2010] R. Abiyev, D. Ibrahim, and B. Erin, “*EDURobot: an educational computer simulation program for navigation of mobile robots in the presence of obstacles*”, International Journal of Engineering Education, vol. 26, no. 1, pp. 18–29, 2010.
- [Abu, 2009] N. M. Abu-halaweh, and R. W. Harrison, “*Practical fuzzy decision trees*”, in IEEE Symposium on Computational Intelligence and Data Mining, Tennessee, USA, 2009, pp. 211–216.
- [Acampora, 2013] G. Acampora, V. Loia, C.-S. Lee, and M.-H. Wang, “*On the power of Fuzzy Markup Language*”, ser. Studies in Fuzziness and Soft Computing. Springer, 2013.
- [Alcala, 2009] J. Alcala-Fdez, L. Sanchez, S. Garcia, M. J. Jesus, S. Ventura, J. M. Garrell, J. Otero, C. Romero, J. Bacardit, V. M. Rivas, J. C. Fernandez, and F. Herrera, “*KEEL: a software tool to assess evolutionary algorithms for data mining problems*”, Soft Computing, vol. 13, no. 3, pp. 307–318, 2009.
- [Alcala, 2009] J. Alcalá-Fdez, R. Alcalá, M. J. Gacto, and F. Herrera, “*Learning the membership function contexts for mining fuzzy association rules by using genetic algorithms*”, Fuzzy Sets and Systems, vol. 160, no. 7, pp. 905–921, 2009.
- [Alcala, 2016] Alcala-Fdez J, and Alonso J, “*A Survey of Fuzzy Systems Software: Taxonomy, Current Research Trends, and Prospects*”, IEEE Transactions On Fuzzy Systems, 2016, 40-56.
- [Alonso, 2011] J. M. Alonso, and L. Magdalena, “*Generating understandable and accurate fuzzy rule-based systems in a java environment*”, in International Workshop on Fuzzy Logic and Applications, ser. Lecture Notes in Artificial Intelligence, A. M. Fanelli, W. Pedrycz, and A. Petrosino, Eds. Springer-Verlag Berlin Heidelberg, 2011, vol. 6857, pp. 212–219.
- [Angulo, 2012] C. Angulo, J. Cabestany, P. Rodriguez, M. Batlle, A. Gonzalez, and S. de Campos, “*Fuzzy expert system for the detection of episodes of poor water quality through continuous measurement*”, Expert Systems with Applications, vol. 39, no. 1, pp. 1011–1020, 2012.
- [Apache, 2010] Apache Software Foundation “*Apache HTTP Server Reference Manual - for Apache version 2.2.17*”, Network Theory Ltd., EUA, 2010.

- [Baturone, 2007] I. Baturone, F. J. Moreno-velo, S. Sanchez-solano, A. Barriga, P. Brox, A. Gersnoviez, and M. Brox, “Using Xfuzzy environment for the whole design of fuzzy systems”, in IEEE International Conference on Fuzzy Systems, London, UK, 2007, pp. 1–6.
- [Ben, 2010] Ben-Hur A, and Weston J., *A user’s guide to support vector machines*, Methods Mol Biol. 2010; 609: 223–239. doi: 10.1007/978-1-60327-241-4\_13 PMID: 20221922
- [Berthold, 2013] M. R. Berthold, B. Wiswedel, and T. R. Gabriel, “Fuzzy logic in KNI-ME, modules for approximate reasoning”, International Journal of Computational Intelligence Systems, vol. 6, no. sup1, pp. 34–45, 2013.
- [Bingol, 2010] O. Bingol and S. Pacaci “A virtual laboratory for fuzzy logic controlled DC motors”, International Journal of the Physical Sciences, vol. 5, no. 16, pp. 2493–2502, 2010.
- [Bozkir, 2013] A. S. Bozkir, and E. A. Sezer, “FUAT: a fuzzy clustering analysis tool”, Expert Systems with Applications, vol. 40, no. 3, pp. 842–849, 2013.
- [Cadenas, 2013] J. M. Cadenas, M. C. Garrido, and R. Martinez, “NIP - an imperfection processor to data minin datasets”, International Journal of Computational Intelligence Systems, vol. 6, no. sup1, pp. 3–17, 2013.
- [Castillo, 2013] O. Castillo, P. Melin, and J. R. Castro, “Computational intelligence software for interval type-2 fuzzy logic”, Computer Applications in Engineering Education, vol. 21, no. 4, pp. 737–747, 2013.
- [Castro, 1999] J.L. Castro, J.J. Castro-Schez, and J.M. Zurita, *Learning maximal structure rules in fuzzy logic for knowledge acquisition expert systems*, Fuzzy Sets and Systems, 101, pp. 331-342, 1999.
- [Castro, 2001] José Jesús Castro Sánchez, “Diseño e implementación de una herramienta para la adquisición automática de conocimiento”, Tesis Doctoral, Universidad de Castilla la Mancha, 2001.
- [Castro, 2004] Jose J. Castro, and Juan Luis Zurita, *Fuzzy Repertory Table: A Method for Acquiring Knowledge About Input Variables to Machine Learning Algorithm*, IEEE Transactions on Fuzzy Systems, 123-139, 2004.
- [Cavus, 2010] N. Cavus, “The evaluation of learning management systems using an artificial intelligence fuzzy logic algorithm”, Advances in Engineering Software, vol. 41, no. 2, pp. 248–254, 2010.
- [Cervantes, 2010] H. Cervantes, “Arquitectura de Software”, SG, México, 2010, Vol. 1. 1870 – 0888.
- [Chantrapornchai, 2010] C. Chantrapornchai, K. Sripanomwan, O. Chaowalit, and J. Pipatpaisarn, “Developer toolkit for embedded fuzzy system based on E-Fuzz”, in Future Generation Information Technology, ser. Lecture Notes in Computer Science. Springer, 2010, vol. 6485, pp. 220–233.

- [Chen, 2000] Guanrong Chen, and Trung Tat Pham, *Introduction to Fuzzy Sets, Fuzzy Logic, and Fuzzy Control Systems*, CRC Press, 2000. ISBN: 0849316588.
- [Cheng, 2013] C. Cheng, A. McGordon, R. P. Jones, and P. A. Jennings, “*Development of a comprehensive and flexible forward dynamic powertrain simulation tool for various hybrid electric vehicle architectures*”, *Proceedings of the Institution of Mechanical Engineers Part D-Journal Of Automobile Engineering*, vol. 226, no. 3, pp. 385–398, 2012.
- [Chi, 1996] Z. Chi, H. Yan and T. Pham, *Fuzzy algorithms with applications to image processing and pattern recognition*, World Scientific, Singapore, 1996.
- [Chico, 2008] David Chico García, and José Jesús Castro Sánchez, *Una herramienta para el diseño de sistemas difusos multinivel en problemas de decisión o clasificación*, Universidad de Castilla la Mancha, 2004.
- [Cingolani, 2013] P. Cingolani, and J. Alcalá-Fdez, “*jFuzzyLogic: a java library to design fuzzy logic controllers according to the standard for fuzzy control programming*”, *International Journal of Computational Intelligence Systems*, vol. 6, no. sup1, pp. 61–75, 2013.
- [Conover, 1999] Conover, W., *Practical nonparametric statistics*. Wiley series in probability and statistics: Applied probability and statistics. Wiley. ISBN 9780471160687.
- [Dandil, 2012] B. Dandil, “*An integrated electrical drive control laboratory: Speed control of induction motors*”, *Computer Applications in Engineering Education*, vol. 20, no. 3, pp. 410–418, 2012.
- [Dorigo, 2004] M. Dorigo, and T. Stutzle, “*Ant Colony Optimization*”, MIT Press, Cambridge, MA. 2004.
- [Driankov, 1993] D. Driankov, H. Hellendoorn, and M. Reinfrank, “*An Introduction to Fuzzy Control*”, Springer-Verlag, 1993.
- [Dubois, 1980] D. Dubois and H. Prade, “*Fuzzy sets and systems: Theory and applications*”, New York: Academic Press, 1980.
- [Eldrandaly, 2012] K. A. Eldrandaly, and N. M. AbdelAziz, “*Enhancing ArcGIS decision making capabilities using an intelligent multicriteria decision analysis toolbox*”, *Journal of Environmental Informatics*, vol. 20, no. 1, pp. 44–57, 2012.
- [Fernandez, 2013] J. Fernandez De Canete, A. Garcia-Cerezo, I. Garcia-Moral, P. Del Saz, and E. Ochoa, “*Object-oriented approach applied to ANFIS modeling and control of a distillation column*”, *Expert Systems with Applications*, vol. 40, no. 14, pp. 5648–5660, 2013.
- [Fey, 2009] Feytmans, Ernest, Appel, and Ron D, *Bioinformatics : A Swiss Perspective*, New Jersey : World Scientific Publishing Company, 2009. ISBN: 9789812838773.
- [Foscarini, 2010] F. Foscarini, G. Bellocchi, R. Confalonieri, C. Savini, and G. Van den Eede, “*Sensitivity analysis in fuzzy systems: Integration of SimLab and DANA*”, *Environmental Modelling & Software*, vol. 25, no. 10, pp. 1256–1260, 2010.

- [France, 1984] France, J. and Thornley, J. H. M, *Mathematical models in agriculture*, London: Butterworths, 350, 1984.
- [Gacto, 2014] M. J. Gacto, M. Galende, R. Alcalá, and F. Herrera, “*METSK-HDe: A multiobjective evolutionary algorithm to learn accurate TSK-fuzzy systems in high-dimensional and large-scale regression problems*”, *Information Sciences*, vol. 276, pp. 63–79, 2014.
- [Glykas, 2013] M. Glykas, “*Fuzzy cognitive strategic maps in business process performance measurement*”, *Expert Systems with Applications*, vol. 40, no. 1, pp. 1–14, 2013.
- [Governor, 2009] James Governor, Dion Hinchcliffe, and Duane Nickull, “*Web 2.0 Architectures*”, O’Reilly Media, 2009.
- [Guillaume, 2012] S. Guillaume, and B. Charnomordic, “*Fuzzy inference systems: An integrated modeling environment for collaboration between expert knowledge and data using FisPro*”, *Expert Systems with Applications*, vol. 39, no. 10, pp. 8744–8755, 2012.
- [Guillaume, 2013] S. Guillaume, B. Charnomordic, B. Tisseyre, and J. Taylor, “*Soft computing-based decision support tools for spatial data*”, *International Journal of Computational Intelligence Systems*, vol. 6, no. sup1, pp. 18–33, 2013.
- [Hadley, 2012] Hadley Wickham, “*Make it easier to work with strings*”, *Journal Statistical Software*, 2015, in press.
- [Hanley, 1982] Hanley JA, and McNeil BJ, *The meaning and use of the area under a receiver operating characteristic (ROC) curve*, *Radiology*. 1982; 143: 29–36. PMID: 7063747.
- [Herrera, 1995] Francisco Herrera, Manuel Lozano, and Jose Luis Verdegay, “*Generating Fuzzy Rules From Examples Using Genetic Algorithms*”, World Scientific Publishing Co. Pte. Ltd., 1995.
- [Herrera, 2009] E. Herrera-Viedma, A. G. Lopez-Herrera, S. Alonso, J. M. Moreno, F. J. Cabrerizo, and C. Porcel, “*A computer-supported learning system to help teachers to teach fuzzy information retrieval systems*”, *Information Retrieval*, vol. 12, no. 2, pp. 179–200, 2009.
- [Holecek, 2010] P. Holecek and J. Talasova, “*Designing fuzzy models of multiple-criteria evaluation in FuzzME software*”, in *International Conference on Mathematical Methods in Economics*, 2010, pp. 250–256.
- [Ihaka, 2009] Ihaka, Ross, *R : A Language for Data Analysis and Graphics*, *J. Comput. Graph. Stat.*, 299-314, 1996.
- [Ishibuchi, 2001] Ishibuchi, Hisao and Nakashima, Tomoharu, *Effect of Rule Weights in Fuzzy Rule-Based*, *IEEE Transactions on Fuzzy Systems*, 506 - 515, 2001.
- [Ishibuchi, 2013] H. Ishibuchi, S. Mihara, and Y. Nojima, “*Parallel distributed hybrid fuzzy GBML models with rule set migration and training data rotation*”, *IEEE Transactions on Fuzzy Systems*, vol. 21, no. 2, pp. 355–368, 2013.



- [James, 2009] James G., Dion H. and Duane N, “*Web 2.0 Architectures*”, O’Reilly, Inc. 2009.
- [Javor, 2012] A. Javor, and A. Fur, “*Simulation on the web with distributed models and intelligent agents*”, Simulation-Transactions of the Society for Modeling and Simulation International, vol. 88, no. 9, pp. 1080–1092, 2012.
- [Jennings, 2002] J.J. Castro-Schez, J.L. Castro and N.R. Jennings, “*Assisting negotiations and supporting decisions in small business*”, Technical Report, Universidad de Castilla la Mancha, 2002.
- [Jones, 2009] H. Jones, B. Charnomordic, D. Dubois, and S. Guillaume, “*Practical inference with systems of gradual implicative rules*”, IEEE Transactions on Fuzzy Systems, vol. 17, no. 1, pp. 61–78, 2009.
- [Jose, 2007] José Ángel Olivas Varela, “*La lógica borrosa y sus aplicaciones*”, Universidad de Castilla La Mancha, 2007.
- [Jurado, 2009] F. Jurado, A. I. Molina, M. A. Redondo, M. Ortega, A. Giemza, L. Bollen, and H. U. Hoppe, “*Learning to program with COALA, a distributed computer assisted environment*”, Journal of Universal Computer Science, vol. 15, no. 7, pp. 1472–1485, 2009.
- [Kelly, 1955] G. Kelly, *The Psychology of Constructs*, Ed. Norton, New York, 1955.
- [Kerre, 2000] E. E. Kerre, and M. Nachtgael, “*Fuzzy Techniques in Image Processing*”, ser. Studies in Fuzziness and Soft Computing. Springer, 2000.
- [Lago, 2010] P. Lago, R. Farenhorst, P. Avgeriou, R.C. de Boer, V. Clerc, A. Jansen, and H. Van Vliet,. (2010) “*The GRIFFIN Collaborative Virtual Community for Architectural Knowledge Management*”, in Collaborative Software Engineering, Eds. Springer , 2010, pp. 195-217.
- [Lichman, 2013] M. Lichman. *UCI Machine Learning Repository*. <http://archive.ics.uci.edu/ml>, University of California, School of Information and Computer Science.
- [Lin, 2012] C. Lin, and S. Chen, “*An easy-to-implement fuzzy expert package with applications using existing java classes*”, Expert Systems with Applications, vol. 39, no. 1, pp. 1219–1230, 2012.
- [Lisboa, 2000] S. P. S., P. J. G. Lisboa, and J. Kacprzyk, “*Fuzzy Systems in Medicine*”, ser. Studies in Fuzziness and Soft Computing. Springer, 2000.
- [Liu, 2009] H. Liu, G. M. Coghill, and D. P. Barnes, “*Fuzzy qualitative trigonometry*”, International Journal of Approximate Reasoning, vol. 51, pp. 71–88, 2009.
- [Lopez, 2010] V. Lopez, M. Santos, and J. Montero, “*Fuzzy specification in real estate market decision making*”, International Journal of Computational Intelligence Systems, vol. 3, no. 1, pp. 8–20, 2010.
- [Lozano, 2004] José Felipe Lozano Gijón, *Un Agente Comprador Difuso Automático*, Universidad de Castilla la Mancha, 2004.

- [Lu, 2009] J. Lu, Y. Zhu, X. Zeng, L. Koehl, and J. Ma, G. Zhang, “A linguistic multi-criteria group decision support system for fabric hand evaluation”, *Fuzzy Optimization and Decision Making*, vol. 8, no. 4, pp. 395–413, 2009.
- [Magdeburg, 2013] University of Magdeburg, “Software for neural networks and fuzzy systems”, 2013, <http://fuzzy.cs.uni-magdeburg.de/software.html>.
- [Martinez, 2013] L. G. Martinez, G. Licea, A. Rodriguez, J. R. Castro, and O. Castillo, “Using MatLab’s fuzzy logic toolbox to create an application for RAMSET in software engineering courses”, *Computer Applications in Engineering Education*, vol. 21, no. 4, pp. 596–605, 2013.
- [MathWorks, 2013] MathWorks, “Fuzzy logic toolbox - r2013b”, 2013, <http://www.mathworks.es/products/fuzzy-logic/>.
- [Mayer, 1993] Mayer, D. and Butler, D., Statistical validation. *Ecological Modelling.*, 68, 21–32, 1993.
- [Melin, 2013] P. Melin, O. Mendoza, O. Castillo, and J. R. Castro, “A visual toolbox for modeling and testing multi-net neural systems”, *Computer Applications in Engineering Education*, vol. 21, no. 1, pp. 164–184, 2013.
- [Meyer, 2009] D. Meyer, and K. Hornik, “Generalized and customizable sets in R”, *Journal of Statistical Software*, vol. 31, no. 2, pp. 1–27, 2009.
- [Morcillo, 2010] P. J. Morcillo, G. Moreno, J. Penabad, and C. Vazquez, “A practical management of fuzzy truth-degrees using FLOPER”, in *International Web Rule Symposium*, ser. Lecture Notes in Computer Science. Springer, 2010, vol. 6403, pp. 20–34.
- [Moreiral, 2010] E. J. Moreira Pegoraro, and A. Sousa, “FEUP Fuzzy Tool II. improved tool for education and embedded control”, in *Iberian Conference on Information Systems and Technologies*, Lousada, Portugal, 2010, pp. 331–336.
- [Munoz, 2011] S. Munoz-Hernandez, V. Pablos-Ceruelo, and H. Strass, “RFuzzy: Syntax, semantics and implementation details of a simple and expressive fuzzy tool over Prolog”, *Information Sciences*, vol. 181, no. 10, pp. 1951–1970, 2011.
- [Nauck, 1998] D. Nauck and R. Kruse, *How the learning of rule weights affects the interpretability of fuzzy systems*, in *Proc. 7th IEEE Int. Conf. Fuzzy Systems*, Anchorage, AK, May 4–9, 1998, pp. 1235–1240.
- [Oliveira, 2013] C. De Oliveira, M. Cintra, and F. Mendes Neto, “Learning risk management in software projects with a serious game based on intelligent agents and fuzzy systems”, in *Conference of the European Society for Fuzzy Logic and Technology (EUSFLAT 2013)*, Milano, Italy, 2013, pp. 834–839.
- [Oliynyk, 2012] A. Oliynyk, C. Bonifazzi, F. Montani, and L. Fadiga, “Automatic online spike sorting with singular value decomposition and fuzzy C-mean clustering”, *BMC neuroscience*, vol. 13, no. 96, pp. 1–19, 2012.

- [Palacios, 2013] A. M. Palacios, L. Sanchez, and I. Couso, “*CI-LQD: A software tool for modeling and decision making with low quality data*”, in IEEE International Conference on Fuzzy Systems, Hyderabad, India, 2013, pp. 1–8.
- [Pancho, 2013] D. P. Pancho, J. M. Alonso, and J. Alcalá-Fdez, “*A new fnggram-based software tool for visual representation and analysis of fuzzy association rules*”, in IEEE International Conference on Fuzzy Systems, Hyderabad, India, 2013, pp. 1–7.
- [Papageorgiou, 2012] E. I. Papageorgiou, “*Fuzzy cognitive map software tool for treatment management of uncomplicated urinary tract infection*”, Computer methods and programs in biomedicine, vol. 105, no. 3, pp. 233–245, 2012.
- [Pedrycz, 1999] W. Pedrycz and A. V. Vasilakos, “*Linguistic models and linguistic modeling*”, IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, vol. 29, no. 6, pp. 745–757, 1999.
- [Pedrycz, 2000] W. Pedrycz, and A. V. Vasilakos, “*Computational Intelligence in Telecommunication Networks*”, CRC Press, Taylor & Francis Group, 2000.
- [Pinti, 2010] A. Pinti, F. Rambaud, J. Griffon, and A. T. Ahmed, “*A tool developed in Matlab for multiple correspondence analysis of fuzzy coded data sets: Application to morphometric skull data*”, Computer methods and programs in biomedicine, vol. 98, no. 1, pp. 66–75, 2010.
- [Riza, 2015] L. Riza, C. Bergmeir, F. Herrera, and J. M. Benitez, “*frbs: Fuzzy rule-based systems for classification and regression tasks*”, Journal Statistical Software, 2015, in press.
- [Rosanna, 2005] Rosanna Pérez Pueyo, *Procesado y clasificación de espectros Raman mediante técnicas de lógica difusa: aplicación a la identificación de materiales pictóricos*, Departament Teoria del Senyal i Comunicacions, Universitat Politècnica de Catalunya, 2005.
- [Rubinson, 2013] C. Rubinson, “*Contradictions in fsQCA*”, Quality & Quantity, vol. 47, no. 5, pp. 2847–2867, 2013.
- [Sargent, 2005] M. G. Sargent, “*Biomedicine and the Human Condition: Challenges, Risks, and Rewards Paperback*”, Cambridge University Press, 2005.
- [Smith, 2013] Peter Smith, “*Professional Website Performance : Optimizing the Front End and the Back End*”, John Wiley & Sons, Inc., 2013.
- [Souders, 2007] Souders, S, “*High Performance Web Sites: Essential Knowledge for Front-End Engineers*”, O’Reilly Media, Ed. 1, EUA, 2007.
- [Sterman, 2002] Sterman, J. D., *All models are wrong: Reflections on becoming a systems scientist*, System Dynamics Review, (2002),18(4), 501–531.<http://doi.org/10.1002/sdr.261>.
- [Takagi, 1985] Takagi, Tomohiro and Sugeno, Michio, *Fuzzy identification of systems and its applications to modeling and control*, IEEE Transactions on Systems, Man and Cybernetics, 116-132, 1985.

- [Teague, 2006] Teague, J. C., McPeak, J., and Fawcett, Joe, “CSS, DHTML, & Ajax”, Peachpit Press, 2006, 4th. Ed. .
- [Tedeschi, 2006] Tedeschi, L. O.(2006).*Assessment of the adequacy of mathematical models*. Agricultural Systems, 89(2-3), 225–247. <http://doi.org/10.1016/j.agsy.2005.11.004>
- [Tedeschi, 2006] Tedeschi L., Assessment of the adequacy of mathematical models. *Agricultural Systems.*, (89), 225–247, 2006.
- [Tessem, 1994] B. Tessem, and P. I. Davidsen, “Fuzzy system dynamics: An approach to vague and qualitative variables in simulation”, System Dynamics Review, vol. 10, no. 1, pp. 49–62, 1994.
- [Thomas, 1995] Thomas Back, and Frank Kursawe, “Evolutionary Algorithms for Fuzzy Logic: A Brief Overview”, World Scientific Publishing Co. Pte. Ltd., 1995.
- [Tikunov, 2012] Y. M. Tikunov, S. Laptinok, R. D. Hall, A. Bovy, and R. C. H. de Vos, “MSClust: a tool for unsupervised mass spectra extraction of chromatography-mass spectrometry ion-wise aligned data”, Metabolomics: Official journal of the Metabolomic Society, vol. 8, no. 4, pp. 714–718, 2012.
- [Trutschnig, 2013] W. Trutschnig, M. A. Lubiano, and J. Lastra, “SAFD - an R package for statistical analysis of fuzzy data”, in *Towards Advanced Data Analysis by Combining Soft Computing and Statistics*, ser. Studies in Fuzziness and Soft Computing, C. Borgelt, M. Gil, J. Sousa, and M. Verleysen, Eds. Springer, 2013, vol. 285, pp. 107–118.
- [Uc, 2015] Miguel Uc, Víctor Menéndez, y Salvador Medina, “Optimización del desempeño de un Sistema Web para Modelos de Inferencia Difusa mediante Técnicas no invasivas”, Tecnología educativa CONAIC, 2395-9061, 27–31, 2015.
- [Uc, 2016] Miguel Uc, Víctor Menéndez, y Salvador Medina, “Evaluación del Rendimiento de un Sistema Web para Modelos de Inferencia Difusa”, Abstraction and Application, 14, 20–31, 2016.
- [Vailikos, 1998] A. V. Vasilakos, C. Ricudis, K. Anagnostakis, W. Pedrycz, and A. Pitsillides, “Evolutionary-fuzzy prediction for strategic QoS routing in broadband networks”, in IEEE International Conference on Fuzzy Systems, 1998, pp. 1488–1493.
- [Vignesh, 2013] Vignesh Prajapati, “Big Data Analytics with R and Hadoop”, Packt Publishing, ISBN 978-1-78216-328-2, 2013.
- [Vrana, 2012] I. Vrana, J. Vaníček, P. Kovar, J. Brozek, and S. Aly, “A group agreement-based approach for decision making in environmental issues”, Environmental Modelling & Software, vol. 36, pp. 99–110, 2012.
- [Wagner, 2009] C. Wagner, S. Miller, and J. M. Garibaldi, “A fuzzy toolbox for the R programming language”, in IEEE International Conference on Fuzzy Systems, 2011, pp. 1185–1192.
- [Wieland, 2011] R. Wieland, W. Mirschel, K. Groth, A. Pechenick, and K. Fukuda, “A new method for semi-automatic fuzzy training and its application in environmental modeling”, Environmental Modelling & Software, vol. 26, no. 12, pp. 1568–1573, 2011.

- [Witten, 2011] I. H. Witten, E. Frank, and M. A. Hall, “*Data mining: Practical machine learning tools and techniques*”, Morgan Kaufmann, San Francisco, 2011.
- [Xiaoyuan, 2013] Xiaoyuan, L., and Jun, L., *Review of Mathematical Modeling in Applied Mathematics Education*, 2013 Fourth International Conference on Intelligent Systems Design and Engineering Applications, 549–552.(2013).<http://doi.org/10.1109/ISDEA.2013.530>.
- [Yang, 2013] M.-T. Yang, C.-C. Chen, and Y.-L. Kuo, “*Implementation of intelligent air conditioner for fine agriculture*”, *Energy and Buildings*, vol. 60, pp. 364–371, 2013.
- [Yao, 2013] J. T. Yao, A. V. Vasilakos, and W. Pedrycz, “*Granular computing: Perspectives and challenges*”, *IEEE Transactions on Cybernetics*, vol. 43, no. 6, pp. 1977–1989, 2013.
- [Zade, 1980] L. A. Zadeh, “*Fuzzy sets and systems in System Theory*”, J. Fox, Ed. Polytechnic Press, 1965, pp. 29–39.
- [Zahariev, 2009] Z. Zahariev, “*Software package and API in MATLAB for working with fuzzy algebras*”, in *International Conference on Applications of Mathematics in Engineering and Economics*, vol. 341, Sozopol, Bulgaria, 2009, pp. 341–348.
- [Zar, 2007] Zar, J.H., *Biostatistical Analysis*, 5a Ed. Prentice-Hall. New Jersey, EUA. 944 p,2007.
- [Zhao, 2012] Y. Zhao and H. Gao, “*Fuzzy-model-based control of an overhead crane with input delay and actuator saturation*”, *IEEE Transactions on Fuzzy Systems*, vol. 20, no. 1, pp. 181–186, 2012.
- [Zhaohao, 2007] Zhaohao Sun, and Gavin Finnie, “*A Fuzzy Logic Approach to Experience Based Reasoning*”, *International Journal of Intelligent Systems*, Vol. 22, 867–889 , 2007.
- [Zikidis, 2009] K. C. Zikidis, and A. V. Vasilakos, “*Asafes2: A novel, neuro-fuzzy architecture for fuzzy computing, based on functional reasoning*”, *Fuzzy Sets and Systems*, vol. 83, no. 1, pp. 63–84, 1996.