UNIVERSITAT JAUME•I

GRADO EN MATEMÁTICA COMPUTACIONAL

ESTANCIA EN PRÁCTICAS Y PROYECTO FINAL DE GRADO

Sistemas de Control con Lógica Difusa: Métodos de Mamdani y de Takagi-Sugeno-Kang (TSK)

Supervisor:
David SÁEZ BAIXAULI
Tutores académicos:
Juan José FONT FERRANDIS
Marina MURILLO ARCILA
Autor:
Samuel DICIEMBRE SANAHUJA

Fecha de lectura: 2 de Octubre de 2017
Curso académico 2016/2017

Agradecimientos

Quiero agradecer a mis padres la oportunidad que me han dado de poder estudiar en la universidad. A la orden de las Escuelas Pias y de los Cooperadores de la Verdad de la Madre de Dios que me han animado a estudiar y a esforzarme a pesar de mis debilidades y me han ayudado a ver el amor en la educación. A los profesores que he tenido en toda mi etapa de formación, desde primaria hasta la Universidad, todos y cada uno de ellos me han aportado algo para llegar hasta donde he llegado.

Resumen

Este documento trata sobre la Estancia en Prácticas en el Departamento de Informática del Hospital General de Castellón de la Plana y el Trabajo Fin de Grado de la asignatura *MT1030-Prácticas Externas y Proyecto Fin de Grado* del Grado en Matemática Computacional de la Universidad Jaume I.

El documento se divide en dos partes. En la primera, se detalla mi estancia en el Departamento de Informática del Hospital General de Castellón. Se describe la migración de una base de datos y el desarrollo de una aplicación web para trabajar con ella. En la segunda parte, se describe el Trabajo de Fin de Grado realizado. El trabajo consiste en una introducción a la Lógica Difusa, con una presentación de los conceptos básicos y el desarrollo de los dos métodos básicos utilizados en el Control Difuso, una de las aplicaciones más exitosas de la Lógica Difusa. Estos métodos son el de Mamdani y el de Takagi-Sugeno-Kang (TSK).

Palabras clave

Lógica Difusa, Conjuntos Difusos, Control Difuso, Mamdani, TSK.

Keywords

Fuzzy logic, Fuzzy Sets, Fuzzy Control, Mamdani, TSK.

Índice general

Capítulo 1

Introducción

En este documento se presentan, en primer lugar, las tareas llevadas a cabo durante la Estancia en Prácticas y, en segundo lugar, se describe el Trabajo Fin de Grado.

En el segundo capítulo se detalla la Estancia en Prácticas en el Departamento de Informática del Hospital General de Castellón. El objetivo principal de estas prácticas era llevar a cabo una migración de una base de datos en Microsoft Acces a otra en SQL en la plataforma web phpMyAdmin y desarrollar una aplicación web para que los usuarios puedan hacer consultas sobre dicha base de datos. Esta base de datos contiene el inventario con todos los ordenadores, las impresoras, los "switches" y teléfonos instalados en el hospital, así como los diferentes puestos de trabajo y el personal del hospital.

El tercer capítulo desarrolla el Trabajo fin de Grado. En este capítulo se presenta una amplia introducción a la Teoría de Conjuntos Difusos así como la explicación de los métodos más utilizados para la contrucción de sistemas de control difusos: el método de Mamdani y el de Takagi-Sugeno-Kang. Para finalizar este capítulo presentamos un ejemplo en el ámbito de la medicina que resolvemos con los métodos de Mamdani y de Takagi-Sugeno-Kang con el objetivo de poder ver las diferencias a la hora de trabajar con uno u otro.

En el último capítulo se enumeran las conclusiones que se derivan de la Estancia en Prácticas y del Trabajo Fin de Grado.

Capítulo 2

Estancia en Prácticas

2.1. Datos de la estancia en prácticas

2.1.1. Datos del Alumno

- Nombre y Apellidos: Samuel Diciembre Sanahuja
- DNI: 20905514D
- Dirección: Calle Compromiso de Caspe Nº 1 Piso 2º Puerta 3, Castellón
- Teléfono: 627 895 933
- Correo electrónico: al269413@uji.es

2.1.2. Datos de la Empresa

- Nombre: Hospital General Universitario de Castellón
- Dirección: Av. Benicasim s/n 12004, Castellón
- Teléfono 964 72 65 00 y 964 72 66 61
- Fax: 964 25 23 45
- Dependencia funcional: Conselleria de Sanitat de la Comunitat Valenciana
- Tutor en la Empresa: David Sáez Baixauli

- Puesto del tutor en la Empresa: Director del Departamento de Informática
- Correo electrónico del tutor: saez_dav@dva.es
- Fecha de inicio: 30 de Enero de 2017
- Fecha de finalización: 26 de Abril de 2017
- Horario: De 8:00 a 14:00
- Días de la semana: De Lunes a Jueves

2.1.3. Función de la Empresa

El Hospital General Universitario de Castellón se fundó en 1967 y desde entonces ha estado atendiendo a ciudadanos de toda la provincia. No ha parado de crecer desde su fundación y actualmente cuenta con casi 600 camas. Para ayudar a la gestión, tanto de los pacientes como de los empleados el hospital, ha estado actualizándose tecnológicamente para poder facilitar todas sus tareas. Por esta razón se creó el Servicio de Informática: para desarrollar aplicaciones, gestionar las bases de datos, el servicio de usuarios, atendiendo consultas técnicas y en general cualquier duda que pueda surgir relacionada con las Tecnologías de la Información y la Comunicación (TIC). Este servicio ha crecido en los últimos años y siempre está dispuesto a acoger estudiantes en prácticas para enseñarles y, a la vez, que estos ayuden con las múltiples labores del departamento.

2.2. Introducción

En este capítulo se desarrolla el proyecto que he llevado a cabo durante mi Estancia en Prácticas. La empresa en la que he desarrollado el proyecto es el Departamento de Informática del Hospital General Universitario de Castellón. Durante mi estancia, el tutor que me han asignado ha sido David Sáez Baixauli, director del Servicio de Informática del Hospital General junto con Victoria Martí.

2.3. Objetivos del trabajo de prácticas

El objetivo es reformar la base de datos de los puestos de trabajo del Hospital General de Castellón. El constante crecimiento de éste y por consiguiente el aumento del número de trabajadores y de equipamiento ha provocado que su base de datos empiece a quedarse obsoleta.

Se disponía de una base de datos en Microsoft Acces formada por 49 tablas y el objetivo era hacer una *migración* a una base de datos en MySQL.

Además, se quiere desarrollar una aplicación web utilizando los lenguajes HTML y PHP para que los usuarios del hospital (médicos, enfermeros y administrativos) puedan acceder a esta nueva base de datos y realizar las consultas que necesiten.

2.4. Conceptos previos

2.4.1. Migración de Datos

La *migración de datos* consiste en la transferencia de materiales digitales de un origen a un destino. Existen diversos motivos para realizar una migración: la preservación o difusión de los contenidos, mejoras en el funcionamiento, cumplir con nuevos requerimientos de usuario o de software, la interoperabilidad, la actualización de versiones, la estandarización de la tecnología, la reducción de costos al optar por un software libre, el aumento en el volumen de datos, nuevos procesos de negocio o mejoras en la seguridad o el control de la información, entre otros escenarios posibles.

Del mismo modo, puede ser necesario migrar de un proveedor de base de datos a otro o actualizar la versión del software de gestión que se utiliza. Si tenemos una aplicación sobre una base de datos como, por ejemplo, Access y posteriormente "crecemos"de manera que nos hace falta un sistema gestor de bases de datos más potente, lo más seguro es que nos decantemos por Oracle, SQLServer o DB2. En este caso, los datos, que estarán en formato "Access"deberán pasar a formato "SQLserver" o formato "Oracle". La migración conlleva la creación de tablas o modificación de las existentes, cambios en algunos tipos de datos que existen en una base de datos pero no en otras, etc. Especialmente delicados son los campos fecha, los numéricos (enteros, reales, etc) o campos de extensión superior a 256 caracteres, campos para imágenes, etc, ya que cada Sistema de Gestión de Base de Datos (SGBD) los trata de manera diferente.

Los pasos a seguir para llevar a cabo una migración de una base de datos son los siguientes:

- **Planificación**. Lo más importante al migrar una Base de Datos (BD) es llevar a cabo un proceso de planificación y análisis del trabajo, puesto que aunque nos pueda tomar algún tiempo adicional, este será retribuido en el éxito de la operación y menos costos por errores de datos. Es importante que esto sea aplicado cuando la Base de Datos de destino está en producción.
- **Contador de registros**. Si la migración se realiza de forma manual, mediante alguna

consulta de inserción, es recomendable inicializar un contador para cada registro insertado con éxito y otro para los no insertados. De esta manera la suma de ambos debe ser igual a los registros originales.

- **Mapeador de Tipos de datos**. Algunas plataformas no soportan algunos tipos de datos, así que es necesario planificar el mapeo de los campos en la nueva base de datos.
- **Restricciones y Trigers**. Antes de iniciar la migración de la BD, es recomendable deshabilitar los Trigers y/o restricciones que nos puedan generar error en el momento en que el SGBD ejecute el proceso de escritura de los datos.
- **Codificación de Caracteres**. Cuando el copiado se realiza de forma automática, es necesario identificar la codificación de caracteres que la BD destino espera, pues así evitaremos el reemplazo automático de caracteres o en su caso, pérdida de los mismos.

2.4.2. Desarrollo web

Desarrollo web es un término amplio que define la creación de sitios web para Internet o una intranet. Para conseguirlo se hace uso de tecnologías de software del lado del servidor y del cliente que involucran una combinación de procesos de base de datos con el uso de un navegador web a fin de realizar determinadas tareas o mostrar información.

Tradicionalmente, un software departamental o incluso un ambicioso proyecto corporativo de gran envergadura es desarrollado en forma “standalone”; es decir, usando lenguajes ya sea compilados, semicompilados, o interpretados (PHP en nuestro caso) para crear tanto la funcionalidad como toda la interfaz de los usuarios. Pero cabe perfectamente un desarrollo orientado a web para dichos propósitos, siendo más homogéneo y multiplataforma, y dependiendo de las tecnologías utilizadas, más rápido y robusto tanto para diseñar, implementar y probar, como para su uso una vez terminado.

Funcionalmente, el desarrollador web, que es quien realiza esta labor, suele preocuparse únicamente por el funcionamiento del software. Es tarea del diseñador web preocuparse del aspecto final de la página y del “webmaster” (persona responsable de mantenimiento de un sitio web) el integrar ambas partes. En ocasiones el webmaster también se encarga de actualizar los contenidos de la página.

2.5. Explicación detallada del proyecto realizado en la empresa

Comenzaremos por diseñar una nueva base de datos para el inventario del Hospital en MySQL a partir de la base de datos actual. El Hospital General de Castellón disponía de una base de datos en Microsoft Acces formada por 49 tablas. La idea era hacer una migración para que la gestión de datos fuese más cómoda y eliminar los datos redundantes o que no son de interés. La idea de migrar la base de datos a MySQL se basa en la comodidad de poder trabajar desde la web del hospital. Para ello, trabajé en una página web para que los usuarios (auxiliares, médicos y enfermeros) pudieran hacer consultas sencillas sobre la base de datos del servidor.

Para realizar el trabajo comencé repasando el lenguaje SQL, que había visto previamente en la carrera (MT1020-Bases de Datos) y amplié conocimientos con manuales y libros de internet. También fue necesario estudiar el lenguaje de programación PHP desde cero, ya que no tenía conocimientos previos sobre este. Para poder llevar a cabo este objetivo, hablé con varios profesores de la UJI que me facilitaron libros y manuales para llevar a cabo mi aprendizaje. También necesité aprender a utilizar ciertas herramientas para trabajar, como es el caso de MySQLWorkbench, phpMyAdmin y Notepad++.

Software y herramientas utilizadas

2.5.1. MySQLWorkBench

Es una herramienta visual de diseño de bases de datos que integra desarrollo de software, administración de bases de datos, diseño de bases de datos, creación y mantenimiento para el sistema de base de datos MySQL. Permite construir tablas y relaciones tanto con sentencias SQL como manualmente, poniendo por pantalla las sentencias equivalentes a nuestras acciones, lo cual facilita también el aprendizaje del usuario. Dispone de versiones para Windows, Linux y Mac.

2.5.2. phpMyAdmin

Es una herramienta para manejar la administración de MySQL a través de páginas web, utilizando Internet. Actualmente puede crear y eliminar Bases de Datos, crear, eliminar y alterar tablas, borrar, editar y añadir campos, ejecutar cualquier sentencia SQL, administrar claves en campos, administrar privilegios y exportar datos en varios formatos. Se accede mediante usuario y contraseña para restringir el acceso para un proyecto. En nuestro caso solo tendremos acceso alguno de los informáticos del departamento y yo. Mediante programas que crearemos posteriormente los usuarios podrán acceder para hacer consultas o introducir datos pero no podrán realizar grandes cambios (borrar una tabla, cambiar relaciones, etc). Está escrito en PHP.

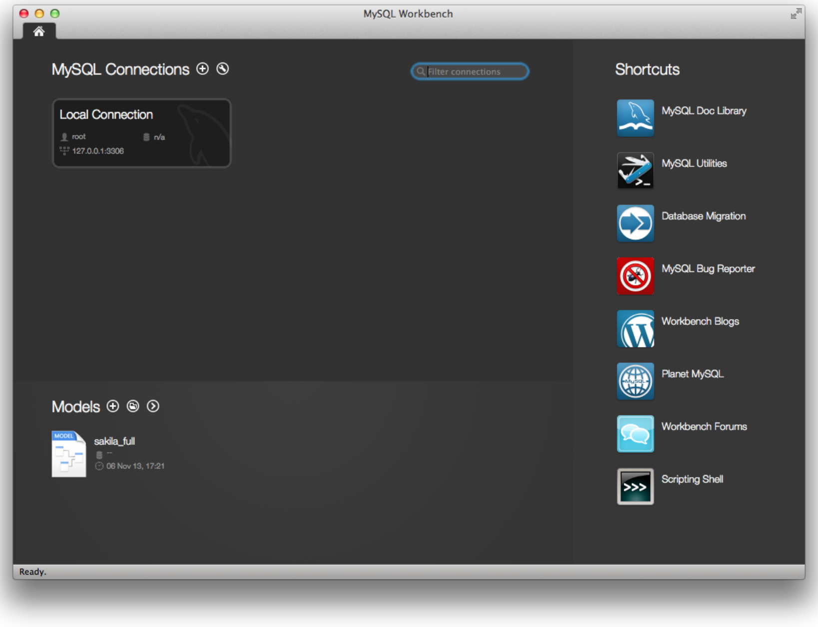

Figura 2.1: Portada de MySQLWorkBench.

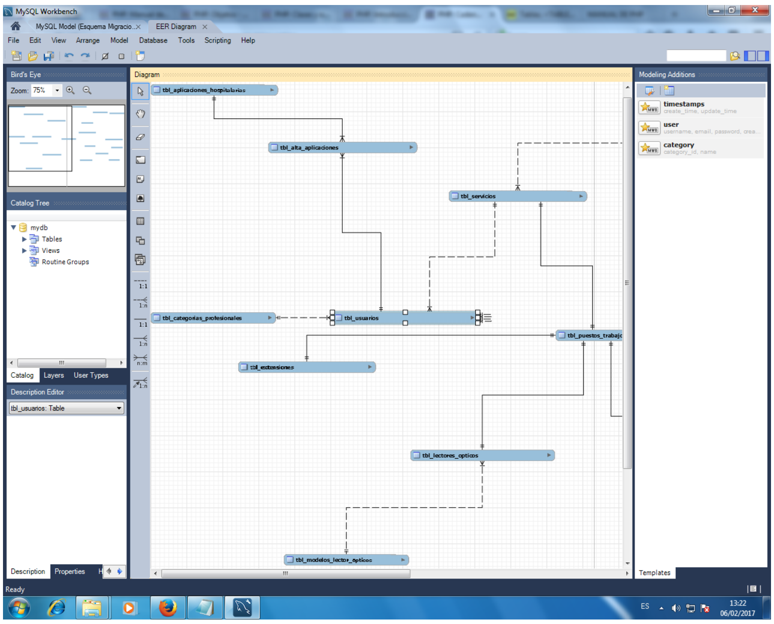

Figura 2.2: Ejemplo de trabajo en MySQLWorkBench.

2.5.3. Notepad++

Notepad++ es un editor de texto y de código fuente libre con soporte para varios lenguajes de programación, en nuestro caso PHP, HTML, CSS y Javascript. Se parece al Bloc de notas en cuanto al hecho de que puede editar texto sin formato y de forma simple. No obstante, incluye opciones más avanzadas que pueden ser útiles para usuarios avanzados como desarrolladores y programadores. Está escrito en C++ y desarrollado tanto para Windows como para Linux, aunque en este segundo no se suele usar.

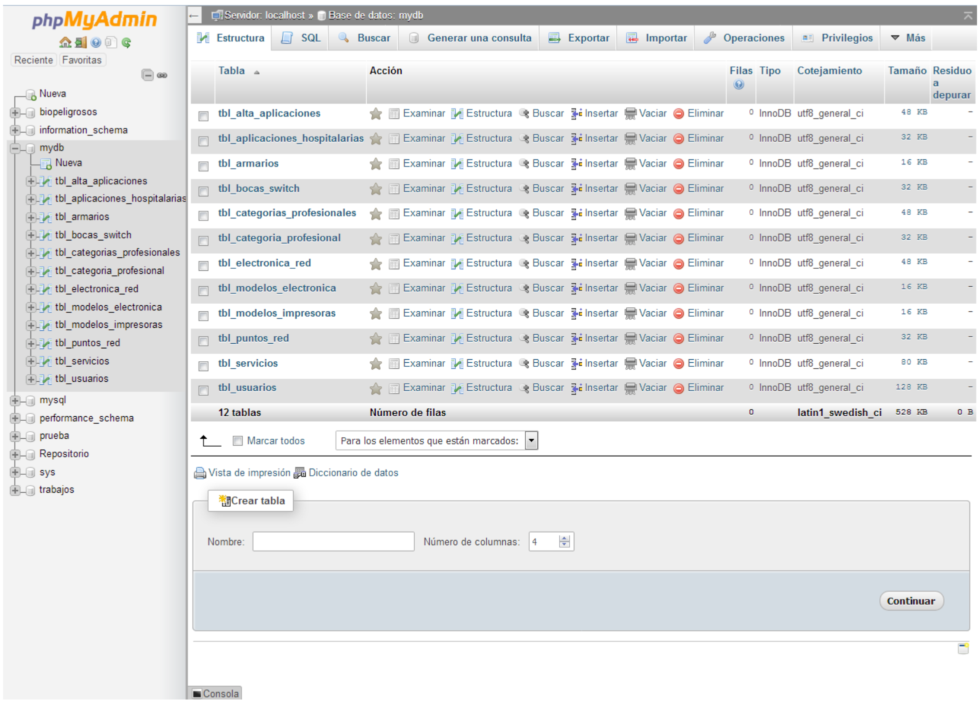

Figura 2.3: Base de Datos en phpMyAdmin.

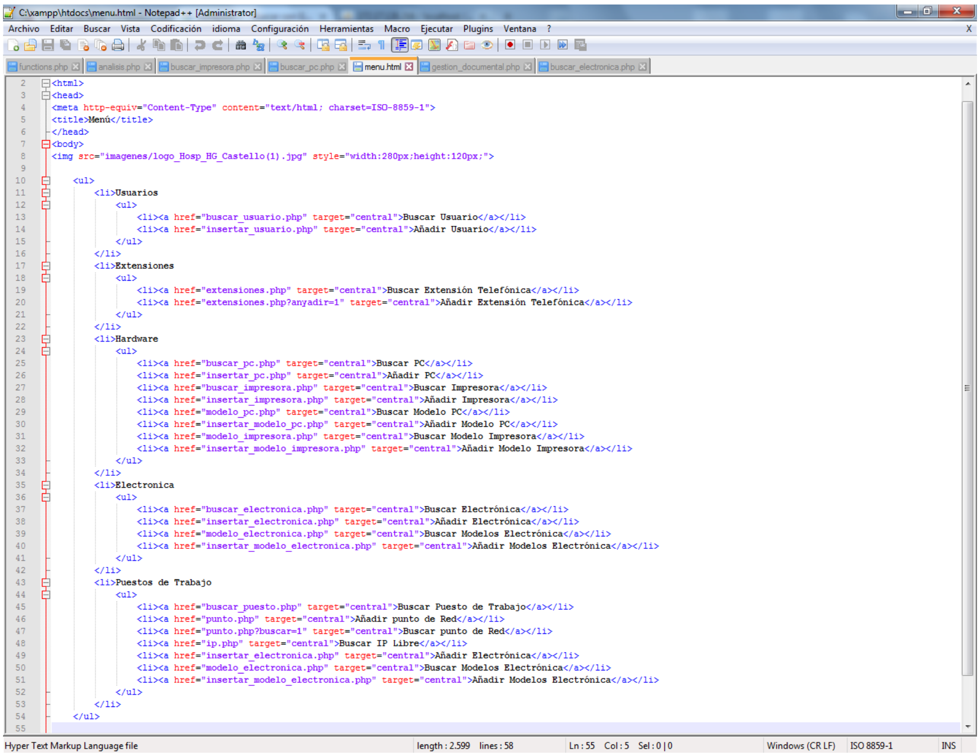

Figura 2.4: Programa escrito en Notepad++.

2.6. Tablas empleadas en la migración

En la nueva base de datos en MySQL disponemos de las siguientes tablas:

- tbl_aplicaciones_hospitalarias
- tbl_alta_aplicaciones
- tbl_armarios
- tbl_bocas_switch
- tbl_categoria_profesional
- tbl_extensiones
- tbl_electronica_red

- tbl_impresoras
- tbl_ips
- tbl_lectores_opticos
- tbl_modelos_impresora
- tbl_modelos_electronica
- tbl_modelos_lectores_optico
- tbl_modelos_pc
- tbl_pcs
- tbl_puesto_trabajo
- tbl_puntos_red
- tbl_rangos_ip
- tbl_servicios
- tbl_usuarios

2.7. Planificación temporal de las tareas

2.7.1. Primera Quincena

En primer lugar, me reuní con mi tutor y me propuso la tarea a realizar. La idea era hacer una migración de base de datos en Microsoft Acces a una desarrollada en MySQL.

La migración permite trabajar de forma más sencilla con la base de datos por parte de los usuarios, en este caso los trabajadores del hospital. También hicimos una reestructuración de las tablas, eliminando las que no eran necesarias y modificando algunos atributos de otras.

Para hacer posible todo esto, dediqué mis primeros días a repasar conceptos de bases de datos que había visto en la carrera y algunos más que no había visto. A partir de manuales de internet me fui familiarizando con el lenguaje SQL que utilizaremos para la creación de la nueva base de datos.

Cuando ya estuve preparado, me dispuse a realizar el diseño de la nueva base de datos. En primer lugar realicé el diseño conceptual, basándome en la base de datos en Acces que ya tenía el Hospital. Era importante tener claro las relaciones entre las distintas tablas. Para facilitar mi trabajo creé un documento Excel en el que iba escribiendo las distintas tablas, con sus atributos, el tipo de dato, si eran únicos, si aceptan nulos (pueden no tener ningún valor asignado) y su valor por defecto.

Una vez finalizado el diseño empleé el programa MySQLWorkBench. Con este programa construí el esquema con todas las tablas y todas sus relaciones, con el fin de agilizar el trabajo. Puesto que la aplicación para acceder a la base de datos estará basada en PHP y HTML, dedicaría los próximos días a estudiar estos lenguajes y aprender a utilizarlos.

2.7.2. Segunda Quincena

Me puse en contacto con varios profesores de la UJI para que me facilitaran libros y apuntes de PHP y HTML. La respuesta fue positiva y con lo que obtuve comencé a estudiar y a hacer algunos programas sencillos para ir practicando.

A mitad de este periodo tuvimos una reunión todo el departamento para discutir algunas cuestiones de la base de datos que estaba realizando. Les mostré el esquema que había hecho y en general estaban satisfechos. Como puntos a mejorar, me aconsejaron cambiar la clave primaria de la tabla "tbl_puesto_trabajo". No nos poníamos de acuerdo sobre cómo debíamos identificar un puesto de trabajo (la ip del equipo, el nombre del equipo o el número de inventario). Finalmente decidimos identificarlo con el nombre del puesto seguido de los últimos 6 números de la ip. Otro tema que discutimos fueron las relaciones que debían tener algunas tablas, es decir, si debían ser 1:1, 1:n ó n:n. Una vez acabada la reunión, me puse a corregir las tablas de mi base de datos.

A continuación, me crearon un usuario para poder acceder y utilizar la herramienta phpMyAdmin, con la cual íbamos a crear la base de datos. A partir de mi esquema fui creando las tablas y las relaciones que previamente había diseñado. En primer lugar creé otra base de datos idéntica a la que llame "prueba" para realizar pruebas y ver como funcionaban las sentencias.

Con todo esto llegamos al momento de programar con el lenguaje PHP. Después de hacer programas sencillos de prueba, me puse a diseñar programas para realizar sentencias SQL y mostrar el resultado en el navegador, utilizando instrucciones de PHP pensadas ya para el manejo de bases de datos SQL. Para esto utilicé el editor de textos Notepad++. Fui aprendiendo poco a poco y fui mejorando la salida por pantalla de mis programas para que quedaran lo más estéticos posibles, ya que al fin y al cabo, la aplicación tiene que ser lo más legible posible.

2.7.3. Tercera Quincena

En esta quincena me dediqué a desarrollar varias páginas y funciones de la aplicación web. Para hacerlo lo mejor posible necesitaba emplear scripts en Javascript y clases en CSS. Una vez más me encontraba con dos lenguajes prácticamente desconocidos para mi. A través de internet, viendo manuales y sobre todo programas de ejemplo, aprendí muchas cosas que me vendrían bien para llevar a cabo mis tareas.

Realicé programas de ejemplo para ver como iban funcionando los lenguajes, elaboré tablas, menús dinámicos y portadas. Tras estar dos días estudiando, me reuní con Pablo Gil, el encargado de este proyecto, y me explicó la idea que él tenía de cómo debía visualizarse la aplicación en la página: una portada, un menú lateral, una ventana que diga si el usuario está registrado o no, eran alguna de las ideas.

Comencé con lo más fácil: la portada y el menú, que poco a poco iría ampliando. En el menú el usuario puede elegir diferentes acciones que le pueden interesar. La primera opción que puse fue la de buscar un usuario de la base de datos, es decir, un trabajador del hospital. La salida muestra los datos de dicho usuario como si de una sentencia SQL se tratara pero a modo de tabla para que sea más estético.

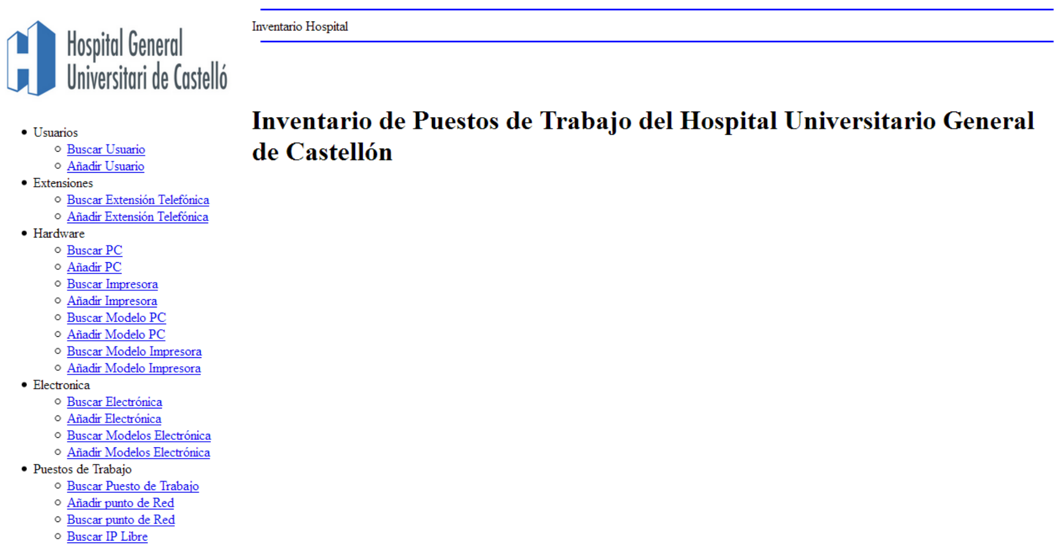

Figura 2.5: Portada de la aplicación web

2.7.4. Cuarta Quincena

Esta quincena la dediqué mayoritariamente a implementar un sistema de administración de sesiones. La idea era que todos los usuarios tuvieran que autenticarse para poder realizar cualquier tarea en la aplicación. Los usuarios se identificarían con su DNI y usarían como contraseña una previamente elegida por ellos. Una vez iniciada la sesión, esta tendría una duración de una hora de forma que en cualquier momento los usuarios podrían cerrar su sesión

Me encontré con algunos problemas a la hora de elaborar este sistema. Las sesiones se cerraban inmediatamente y ninguno de mis compañeros del departamento ni yo sabíamos por qué pasaba. Finalmente opté por utilizar *cookies* para mantener la sesión abierta de forma que el programa guardaba los DNIs de los usuarios y sus nombres como variables de sesión.

Como había ido adelantando las tareas y debido a que varios compañeros del departamento cogieron vacaciones en este periodo, acompañé a Vicente Ramos, técnico del departamento, a instalar y cambiar ordenadores durante unos días.

2.7.5. Quinta Quincena

En esta quincena acabé la aplicación web para la base de datos de los trabajos del hospital. Realicé los últimos programas que me mostraban los detalles de los distintos modelos de impresoras, ordenadores, lectores ópticos y electrónicas de red. También acabé los programas para insertar nuevos datos en la base de datos, comprobé que todo funcionaba correctamente y se lo mostré a mis compañeros.

Contentos con mi trabajo, para las tres semanas que me quedaban en la empresa me pusieron a trabajar en un proyecto similar para aprovechar todo lo que había aprendido de PHP y HTML. Esta vez se trataba de una base de datos para almacenar las visitas que recibe un paciente por parte de un médico y las pruebas que se le realizan. También debía realizar una aplicación web para consultar estas visitas e introducir datos en ella.

Esta aplicación resulta útil ya que los enfermeros guardan muchísimos libros de visitas en papel que ocupan gran cantidad de espacio y el objetivo a largo plazo es digitalizarlo para facilitar el trabajo.

2.7.6. Sexta Quincena

Me dejaron ver uno de estos libros de visitas para saber los campos que debería tener la base de datos, ya que esta estaría formada sólo por una tabla. De cada visita nos interesaba saber la fecha, la hora, el nombre del paciente, el médico que le atendió, las pruebas que le realizaron, diagnóstico y el servicio al que se le mandó.

Hice una base de datos de prueba y la rellené con datos distintos. Resultaba bastante sencillo porque sólo había que implementar dos acciones: insertar una visita y consultar visitas de un paciente.

Realicé una página web sencilla con su correspondiente menú de opciones, su formulario de inserción y su sistema de sesión de usuario. No me dio tiempo presentarlo a los supervisores ni a los enfermeros ya que mi estancia en prácticas finalizó antes de que esto pasara.

2.8. Grado de consecución de los objetivos propuestos

El objetivo principal de la Estancia en Prácticas era realizar una migración de una base de datos en Acces a MySQL. Lo logré creando una nueva base de datos con SQLWorkBench y usando phpMyAdmin.

Además de esto, también desarrollé una página web en la que se podían realizar acciones sobre la base de datos, tales como insertar puestos de trabajo, hardware y usuarios del hospital.

En general se podría decir que he cumplido con los objetivos que me marcaron. He ido trabajando poco a poco y aprendiendo nuevos lenguajes de programación y conceptos que no había visto antes. Por otra parte, en mis dos últimas semanas tambien empecé a trabajar en un proyecto para digitalizar el control de visitas que el hospital tiene sobre los pacientes.

Capítulo 3

Memoria Trabajo Final de Grado

3.1. Introducción

El concepto intuitivo de conjunto clásico corresponde al de una colección de objetos, reunidos usualmente mediante la referencia a alguna propiedad que los caracteriza. En la teoría de conjuntos clásica, un conjunto está bien definido si se puede determinar de manera absoluta si un elemento pertenece o no al conjunto sin ninguna ambigüedad.

Las personas son capaces de tomar decisiones basadas en información imprecisa. Los $conjuntos$ $difusos$ proveen una herramienta para representar esta información, y poder así también construir un sistema de ayuda para la toma de decisiones. Los conjuntos difusos son un concepto que puede acercar el razonamiento computacional al razonamiento humano. La Lógica Difusa es una solución al problema de intentar simplificar problemas complejos con la lógica binaria. Esta lógica se basa en la idea de que dado un conjunto y un elemento, este puede, o bien estar, o no estar en el conjunto. Es decir, dada una aplicación que indica si un elemento pertenece a un conjunto, sólo se pueden obtener dos valores: 0 (si no pertenece) ó 1 (si pertenece). Aunque muchas veces esto sirve para muchos problemas de clasificación, lo cierto es que el pensamiento humano y muchos problemas de la vida real no resultan tan sencillos de clasificar. Por ejemplo, consideramos que una persona puede ser alta o baja. Si decimos que consideramos que una persona es alta a partir de 1.80m, estamos diciendo que una persona que mide 1.799m es baja, pero ¿es correcto afirmar esto? La Lógica Difusa nos da una posible solución a este tipo de problemas.

La Lógica Difusa fue introducida en 1965 por el ingeniero iraní Lofti A. Zadeh, guiado por el principio de que las matemáticas pueden ser usadas para encadenar el lenguaje con la inteligencia humana. La teoría de conjuntos difusos es un intento de desarrollar una serie de conceptos para tratar de un modo sistemático el tipo de imprecisión que aparece cuando los límites de las clases de objetos no están claramente definidos. Tras la publicación de la obra de Lotfi A. Zadeh, la Lógica Difusa comenzó a experimentar un auge y se desarrollaron rápidamente aplicaciones en base a ella.

La matemática de los conjuntos difusos trabaja con conjuntos que no tienen límites perfectamente definidos, es decir, la transición entre la pertenencia y no pertenencia de una variable a un conjunto es gradual. Estos conjuntos se caracterizan por las funciones de pertenencia, que dan flexibilidad a la modelización utilizando expresiones lingüísticas, tales como mucho, poco, leve, severo, escaso, suficiente, caliente, frío, joven, viejo, etc. Surgió de la necesidad de solucionar problemas complejos con información imprecisa, para los cuales la matemática y lógica tradicionales no son suficientes. La Lógica Difusa es un lenguaje que permite trasladar sentencias sofisticadas del lenguaje natural a un formalismo matemático. Este concepto recibe su nombre en inglés (*fuzzy logic*) de la palabra *fuzzy* que viene de *fuzz* (vello o pelusa) y se traduce como difusa o borrosa.

3.2. Aplicaciones de la Lógica Difusa

Desde mediados de los años 70, la Lógica Difusa se ha utilizado ampliamente debido a varios factores. Uno de ellos es que el uso del conocimiento experto permite la automatización de tareas. En muchas áreas de aplicación se reduce considerablemente la necesidad de operadores que basan su conocimiento en la experiencia (y que difícilmente podría ser expresado con ecuaciones diferenciales). De este modo, si existe un conocimiento del proceso, es posible modelizarlo mediante Lógica Difusa.

Los sistemas basados en Lógica Difusa son fáciles de diseñar, modificar y mantener. Pese a la pérdida de precisión, la reducción de tiempo de desarrollo y mantenimiento es muy relevante para su uso industrial.

Otro factor a tener en cuenta es que el control difuso permite diseñar soluciones de alta calidad que eviten las patentes existentes en otros sistemas de control. En Japón este tipo de controladores se asocia a modernidad, alta calidad y tecnológicamente potente. En Europa sin embargo se trata de ocultar el término “difuso” por su significado negativo. En la actualidad multitud de productos de electrónica de consumo emplean Lógica Difusa.

Por citar algunos ejemplos de uso, la empresa Japonesa Matsuhita utiliza en sus lavadoras un sistema de control que determina automáticamente el ciclo de lavado según el tipo, cantidad de suciedad y tamaño de la colada. Los estabilizadores de imágenes en sus cámaras digitales incorporan reglas que eliminan las vibraciones involuntarias de la mano del operario, comparando la imagen actual con las imágenes anteriores de la memoria. La empresa coreana LG cuenta en sus lavadoras con un sensor y un microprocesador que detectan la carga de lavado y fijan las condiciones óptimas de lavado: el nivel de agua o el tiempo de lavado entre otras. En el ámbito de la automoción, Mitsubishi y General Motors emplean sistemas de transmisión automática y control de temperatura basados en Lógica Difusa.

Otro caso de éxito es el metro de Sendai (Japón), que cuenta con 16 estaciones. El sistema de control difuso está dividido en dos módulos: uno para el control de la velocidad y otro para la parada automática. Este controlador difuso ofrece importantes ventajas sobre los controladores convencionales, como el mayor confort en el viaje para los pasajeros y menor consumo de energía.

3.3. Conceptos básicos

Para poder ver las aplicaciones de esta rama de estudio al control de procesos, vamos a introducir primero algunos conceptos. En la lógica tradicional o lógica binaria únicamente se presentan dos casos: o un elemento pertenece a un conjunto o no pertenece. Sin embargo, la pertenencia a un conjunto difuso es *gradual*, según la cual el valor de pertenencia 0 indica que el elemento no está en el conjunto y el valor 1 indica que se encuentra totalmente dentro del conjunto. Por tanto decimos que un elemento forma parte de un conjunto difuso con un determinado *grado de pertenencia*.

Definición 1 *Se define* ***universo de discurso*** *como el conjunto X de todos los posibles valores que puede tomar una determinada variable x.*

Definición 2 *Un* ***conjunto clásico*** *o* ***crisp set*** *es un conjunto empleado en la lógica binaria. Son los conjuntos que surgen por la necesidad del ser humano de clasificar objetos y conceptos. Únicamente contemplan la pertenencia o no pertenencia al conjunto.*

Definición 3 *Un* ***conjunto difuso*** *o* ***fuzzy set*** *es un conjunto que puede contener elementos de forma parcial, es decir, que la propiedad de que un elemento x pertenezca al conjunto A ($x \in A$) puede ser cierta con un grado parcial de verdad.*

Definición 4 *Sea A un conjunto difuso y sea $x \in X$ un valor del conjunto universal. Su* ***función de pertenencia*** *o* ***función de membresia*** *μ_A es una aplicación que indica el grado de pertenencia de un valor a dicho conjunto difuso.*

$$\mu_A(x) : \mathbb{X} \longrightarrow [0, 1].$$

Así pues, un conjunto difuso A puede definirse de la siguiente manera

$$A = \{(x, \mu_A(x)) : x \in X, \mu_A(x) : \mathbb{X} \longrightarrow [0,1]\},$$

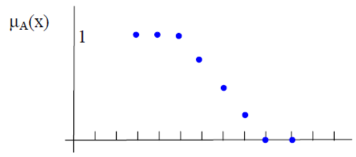

Figura 3.1: Función de membresía en un Universo discreto.

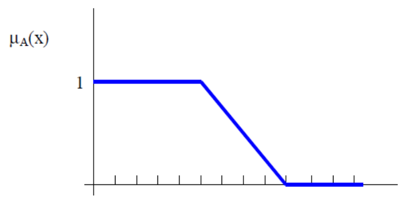

Figura 3.2: Función de membresía en un Universo continuo.

donde $\mu_A(x) = 1$ si x está totalmente en A, $\mu_A(x) = 0$ si x no está en A y $0 < \mu_A(x) < 1$ si x está parcialmente en A. Este valor entre 0 y 1 representa el **grado de pertenencia** (también llamado valor de pertenencia) de un elemento x al conjunto A. Cuanto más cerca esté $\mu_A(x)$ del valor 1, mayor será la pertenencia del objeto x al conjunto A.

Otra forma de representar conjuntos difusos es la siguiente:

$$A = \frac{\mu_A(x_1)}{x_1} + \frac{\mu_A(x_2)}{x_2} + ... + \frac{\mu_A(x_n)}{x_n} = \sum_{i=1}^{n} \frac{\mu_A(x_i)}{x_i}$$

donde el símbolo "+" representa una enumeración y no una operación de suma. En esta representación no se tienen en cuenta los valores de X tal que $\mu_A(x) = 0$. En el caso de que la variable X tome valores continuos y no discretos tendríamos que

$$A = \int_x \frac{\mu_A(x_i)}{x_i},$$

donde la integral tampoco debe considerarse como una operación algebraica sino como una forma de representar los pares de valores.

Ejemplo 1. Si hablamos de temperatura, tendremos que para una persona de Alaska el concepto de caliente puede estar por encima de 10°C, mientras que para un mexicano caliente estaría por encima de 30°C o en un proceso de fundición el concepto de caliente sería para aquellas temperaturas superiores a 300°C. Por esta razón los conjuntos "Caliente", "Tibio" y "Frío" son llamados conjuntos difusos ya que tienen límites borrosos o "no muy bien" definidos.

Ejemplo 2. Supongamos que queremos clasificar a las personas de un país o de una zona en "Ricos" y "No Ricos". Lo más sencillo parece asignar un valor numérico y decir que una persona pertenece al conjunto ricos o no dependiendo de si su fortuna sobrepasa esa cantidad. ¿Y si a una persona le faltaran 100€ para llegar a este valor? ¿Y si le faltaran 10€? Aquí vemos un ejemplo de la representación de los conjuntos aplicando la lógica clásica y la lógica difusa.

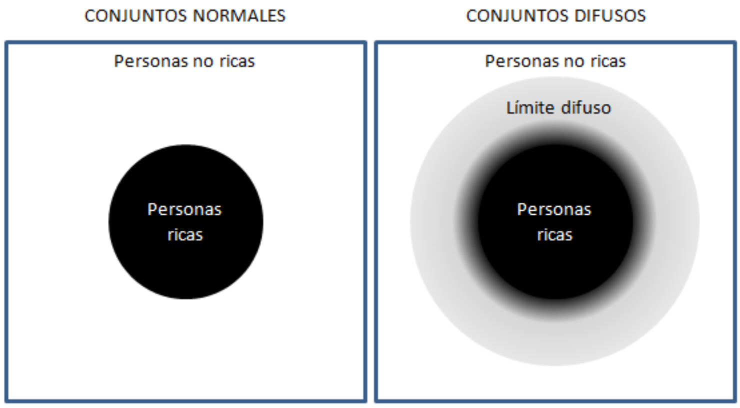

Figura 3.3: Diferencia de límites entre los conjuntos crisp y fuzzy.

Ejemplo 3. Imaginemos que queremos recoger datos de las cualidades físicas de los futbolistas de la liga española de fútbol. Entre estas cualidades nos interesa la altura. Parece obvio que un jugador que mide 1.95m es alto y que uno que mide 1.60m no es alto. Un ejemplo del conjunto difuso "Jugadores Altos" (expresando la altura en cm) podría ser el que sigue

$$Jugadores\ Altos = \{(160, 0), (165, 0'1), (172, 0'25), (178, 0'60), (185, 0'9), (200, 1)...\}$$

Esto nos permite, por ejemplo, afirmar que un jugador de 1.78m tendría un grado 0.60 de pertenencia al conjunto difuso "Jugadores Altos".

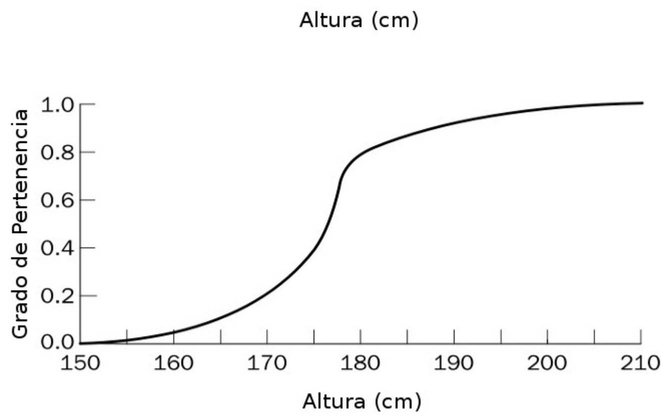

Figura 3.4: Representación grafica de la función de pertenencia del conjunto difuso "Jugadores Altos".

3.3.1. Características de los conjuntos difusos

Un conjunto difuso presenta algunas características que nos pueden resultar útiles a la hora de estudiarlos:

Definición 5 *Sea A un conjunto difuso. La* ***altura*** *de un conjunto difuso es el valor más grande de su función de pertenencia:*

$$h(A) = sup_{x \in X} \mu_{A(x)}.$$

Definición 6 *El **soporte** de un conjunto difuso A es el conjunto de elementos de X que pertenecen a A con grado de pertenencia mayor que 0:*

$$S(A) = \{x \in X | \mu_A(x) > 0\}.$$

*Un conjunto difuso se dice que es **singleton** si su soporte es un único punto.*

Definición 7 *El **núcleo** de un conjunto difuso A es el conjunto de elementos de X que pertenecen a A con grado de pertenencia 1:*

$$Núcleo(A) = \{x \in X | \mu_A(x) = 1\}.$$

*Un conjunto difuso se dice que es **normal** si su núcleo es no vacío.*

Definición 8 *Una **normalización** es una aplicación que convierte un conjunto difuso no normal en un conjunto difuso normal. Sea A un conjunto difuso no normal ($\forall x \in X \longrightarrow \mu_A(x) \neq 1$), un conjunto difuso B normal vendría dado por:*

$$\mu_B(x) = \frac{\mu_A(x)}{h(A)}.$$

Definición 9 *Los **puntos de cruce** o **puntos de equilibrio** de un conjunto difuso son todos los valores que tienen grado de pertenencia 0.5 ($x \in X | \mu_A(x) = 0{,}5$).*

Definición 10 *Un α**-corte** es un conjunto de los valores de X con grado mayor o igual que α:*

$$A_\alpha = \{x \in X | \mu_A(x) \geq \alpha\}.$$

Definición 11 *La **Frontera** de un conjunto difuso A es el conjunto de valores que tienen un grado de pertenencia $0 < \mu_A(X) < 1$ y se define como:*

$$Fr(A) = \{x \in X | 0 < \mu_A(X) < 1\}.$$

3.3.2. Tipos de funciones de pertenencia

Aunque en principio cualquier función sería válida para definir conjuntos difusos, en la práctica hay ciertas funciones que son más usadas que el resto. Esto se debe tanto a la facilidad de computación que su uso conlleva como a su estructura lógica para definir su valor lingüístico asociado. Así, las funciones de pertenencia más comunes son las siguientes:

- Función Gamma

$$\mu(x) = \begin{cases} 0 & \text{si } x \leq a \\ \frac{x-a}{m-a} & \text{si } a < x < m \\ 1 & \text{si } x \geq m \end{cases}$$

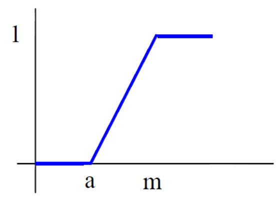

Figura 3.5: Gráfica de la función Gamma.

- Función L
  Puede definirse como la función opuesta a la función Gamma, es decir, 1 menos la función Gamma.

$$\mu(x) = \begin{cases} 1 & \text{si } x \leq a \\ \frac{m-x}{m-a} & \text{si } x \in (a, m) \\ 0 & \text{si } x \geq m \end{cases}$$

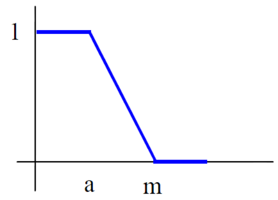

Figura 3.6: Gráfica de la función L.

- Función Lambda (o Función triangular)

$$\mu(x) = \begin{cases} 0 & \text{si } x \leq a \\ \frac{x-a}{m-a} & \text{si } a < x \leq m \\ \frac{b-x}{b-m} & \text{si } m < x \leq b \\ 0 & \text{si } x \geq b \end{cases}$$

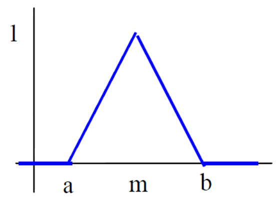

Figura 3.7: Gráfica de la función Lambda.

- Función Pi (o Función trapezoidal)

$$\mu(x) = \begin{cases} 0 & \text{si } x \leq a \\ \frac{x-a}{b-a} & \text{si } a < x \leq b \\ 1 & \text{si } b < x \leq c \\ \frac{d-x}{b-c} & \text{si } c < x \leq d \\ 0 & \text{si } x > d \end{cases}$$

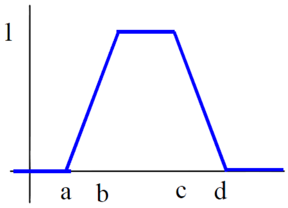

Figura 3.8: Gráfica de la función Pi.

- Función S

$$\mu_S(x) = \begin{cases} 0 & \text{si } x \leq a \\ 2(\frac{x-a}{c-a})^2 & \text{si } a < x \leq c \\ 1 - 2(\frac{x-a}{c-a})^2 & \text{si } \frac{a+c}{2} < x < c \\ 0 & \text{si } x \geq c \end{cases}$$

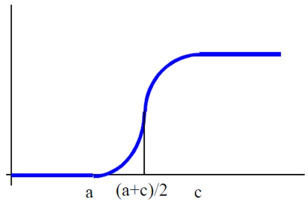

Figura 3.9: Gráfica de la función S.

- Función Z
  Puede definirse como la función opuesta a la función S, es decir, 1-$\mu_S(x)$.

$$\mu_Z(x) = \begin{cases} 1 & \text{si } x \leq a \\ 1 - 2(\frac{x-a}{c-a})^2 & \text{si } a < x \leq c \\ 2(\frac{x-a}{c-a})^2 & \text{si } \frac{a+c}{2} < x < c \\ 1 & \text{si } x \geq c \end{cases}$$

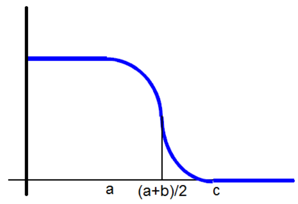

Figura 3.10: Gráfica de la función Z.

- Función Gaussiana

$$\mu_\Pi(x) = \begin{cases} \mu_S(x) & \text{si } x \leq b \\ \mu_Z(x) & \text{si } x > b \end{cases}$$

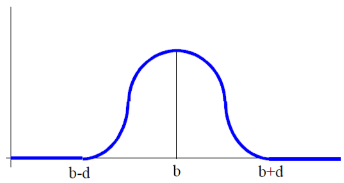

Figura 3.11: Gráfica de la función Gaussiana.

3.3.3. Operaciones con conjuntos difusos

Las tres operaciones básicas que se definen sobre conjuntos clásicos (complemento, unión e intersección) pueden generalizarse de varias formas en conjuntos difusos. Estas operaciones "estándar" sobre conjuntos difusos se comportan de modo similar a las operaciones sobre conjuntos clásicos.

Igualdad

Dos conjuntos difusos A y B, definidos en el mismo Universo X, son iguales si tienen la misma función de pertenencia:

$$\mu_A(x) = \mu_B(x) \;\; \forall x \in X.$$

Inclusión

Sean A y B conjuntos difusos. Diremos que A es **subconjunto difuso** de B ($A \subseteq B$) si su función de pertenencia toma valores mas pequeños

$$\mu_A(x) \leq \mu_B(x) \;\; \forall x \in X.$$

Si existe un valor $x \in X$ tal que $\mu_A(x) \neq \mu_B(x)$ entonces escribiremos que $A \subset B$.

Unión

Sean A y B conjuntos difusos. La forma generalizada de la unión es una T-conorma. Podemos definirla de la siguiente forma:

$$\begin{gathered} \perp : [0,1] \times [0,1] \longrightarrow [0,1] \\ \mu_{A \cup B}(x) = \perp(\mu_A(x), \mu_B(x)). \end{gathered}$$

Para que una función se pueda considerar como una T-conorma, debe satisfacer los siguientes axiomas $\forall a, b, c \in [0,1]$.

- Elemento neutro: $\perp(a, 0) = a$.
- Conmutatividad: $\perp(a, b) = \perp(b, a)$.
- Monotonicidad: Si $a \leq c$ y $b \leq d$ entonces $\perp(a, b) = \perp(c, d)$.
- Asociatividad: $\perp(\perp(a, b), c) = \perp(a, \perp(b, c)$.

Algunas de las T-conormas más utilizadas son las siguientes:

- Máximo: $\perp(a, b) = max(a, b)$.
- Producto: $\perp(a, b) = (a + b) - (ab)$.
- Suma limitada(o de Lukasiewick): $\perp(a, b) = min(a + b, 1)$.

La T-conorma que se corresponde con la Unión de conjuntos difusos es el máximo, por tanto tenemos que:

$$\mu_{A\cup B}(x) = max\,\{\mu_A(x), \mu_B(x)\}.$$

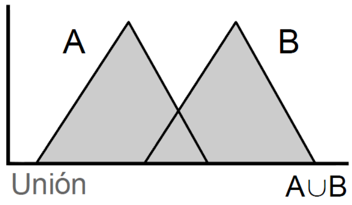

Figura 3.12: Ejemplo de la Unión.

Intersección

Sean A y B conjuntos difusos. La forma generalizada de la intersección es una T-norma. Se define de la siguiente forma:

$$T : [0,1] \times [0,1] \longrightarrow [0,1]$$
$$\mu_{A\cap B}(x) = T(\mu_A(x), \mu_B(x)).$$

Una T-norma satisface los siguientes axiomas $\forall a, b, c \in [0,1]$:

- Elemento unidad: $T(a,1) = a$.
- Conmutatividad: $T(a,b) = T(b,a)$.
- Monotonicidad: Si $a \leq c$ y $b \leq d$ entonces $T(a,b) = T(c,d)$.
- Asociatividad: $T(T(a,b),c) = T(a,T(b,c))$.

Algunas de las T-normas más utilizadas son las siguientes:

- Mínimo: $T(a,b) = min(a,b)$.
- Producto: $T(a,b) = ab$.
- Diferencia limitada(o de Lukasiewick): Si $a \leq c$ y $b \leq d$ entonces $T(a,b) = min(0, a+b-1)$.

La T-norma que se corresponde con la Intersección de conjuntos difusos es el mínimo, por tanto tenemos que:

$$\mu_{A \cap B}(x) = min\,\{\mu_A(x), \mu_B(x)\}.$$

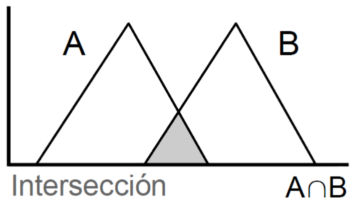

Figura 3.13: Ejemplo de la Intersección.

Complementario

El complementario $\bar{A}$ de un conjunto difuso A se denota por cA y puede definirse de distintas maneras al igual que la unión y la intersección. El complementario esta definido por una funión $c : [0, 1] \longrightarrow [0, 1]$ y tiene que satisfacer los siguientes axiomas $\forall a, b \in [0, 1]$:

- Condiciones frontera: $c(0) = 1; \quad c(1) = 0$.
- c es una función contínua.
- Monotonicidad: Si $a \leq$ b entonces $c(a) \geq c(b)$.
- Involución: $c(c(a)) = a$.

Al igual que sucedía con los operadores de unión y de intersección, también para el complemento existen gran variedad de clases. Nosotros utilizaremos el complemento clásico:

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x).$$

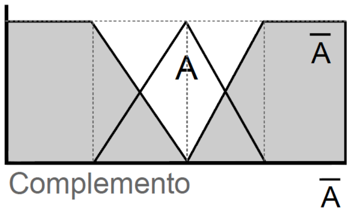

Figura 3.14: Ejemplo del Complementario.

Sin embargo existe una función que es de las más utilizadas conocida como el λ-complemento de Sugeno y que viene definido por la siguiente expresión:

$$\mu_{\bar{A}^{\lambda}}(x) = \frac{1-\mu_A(x)}{1+\lambda\mu_A(x)} \text{ con } \lambda \in (-1, \infty).$$

Distancia

También podemos calcular la distancia entre dos conjuntos difusos A y B en el mismo universo X. Esta función mide la cercanía entre ambos conjuntos difusos. En general se puede usar la **Distancia de Minkowski**:

$$d(A,B) = \sqrt[p]{\int_x |\mu_A(x) - \mu_B(x)|dx}, \quad p \geq 1.$$

Casos particulares:

- **Distancia de Hamming** (p=1): $d(A,B) = \int_x |\mu_A(x) - \mu_B(x)|dx.$
- **Distancia Euclídea** (p=2): $d(A,B) = \sqrt{\int_x |\mu_A(x) - \mu_B(x)|dx}.$
- **Distancia de Tchebyschev** ($p = \infty$): $d(A,B) = \sup_{x \in X} |\mu_A(x) - \mu_B(x)|.$

En universos de discurso discretos, la integral se sustituye por un sumatorio.

3.3.4. Propiedades de los conjuntos difusos

Los conjuntos difusos presentan algunas propiedades que coinciden con las de los conjuntos clásicos. Esto se debe a que, en el fondo los conjuntos clásicos son un caso particular de los conjuntos difusos. Sean A, B y C conjuntos difusos, entonces se cumplen las siguientes propiedades:

- **Conmutativa**: $A \cup B = B \cup A$; $A \cap B = B \cap A$
- **Asociativa**: $A \cup (B \cup C) = (A \cup B) \cup C = A \cup B \cup C$; $A \cap (B \cap C) = (A \cap B) \cap C = A \cap B \cap C$
- **Idempotencia**: $A \cup A = A$; $A \cap A = A$
- **Distributiva**: $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$; $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$
- **Involución**: $c(c(A)) = A$
- **Condiciones Frontera**: $A \cup \emptyset = A$; $A \cup X = X$; $A \cap \emptyset = \emptyset$; $A \cap X = A$
- **Transitividad**: Si $A \subset B$ y $B \subset C$ entonces $A \subset C$
- **Leyes de Morgan**: $c(A \cup B) = c(A) \cap c(B)$; $c(A \cap B) = c(A) \cup c(B)$

A partir de estas propiedades se pueden deducir estas otras:

- $(A \cap B) \subset A \subset (A \cup B)$
- Si $A \subset B$ entonces $A = A \cap B$ y $B = A \cup B$

3.3.5. Variables lingüísticas

Tradicionalmente se han utilizado modificadores de los conjuntos difusos a los que se les llaman variables lingüísticas, equivalentes a lo que en lenguaje natural serían los adverbios (alto, bajo, caliente, frío, etc). Una *variable lingüística* es aquella cuyos valores son palabras o sentencias en un lenguaje natural o artificial. De esta forma, una variable lingüística es un medio para trasladar cualquier elemento que sea demasiado complejo, o del cual no tengamos una definición concreta; a descripciones numéricas que pueden ser tratadas automáticamente: relaciona o traduce el proceso simbólico a proceso numérico. Usando el principio de extensión, muchas herramientas ya existentes pueden ser extendidas para manejar variables lingüísticas, obteniendo las ventajas de la lógica difusa en gran cantidad de aplicaciones.

Tenemos que una variable lingüística está caracterizada por una quíntupla

$$(X, T(X), U, G, M)$$

donde:

- X:nombre de la variable.
- T: conjunto de términos de X; es decir, la colección de sus valores lingüísticos (o etiquetas lingüísticas).
- U: es el universo del discurso (o dominio subyacente). Por ejemplo, si hablamos de temperatura "Cálida" o "Aproximadamente 25°C", el dominio subyacente es un dominio numérico (los grados centígrados).
- G: es una gramática libre de contexto mediante la que se generan los términos en T(X), como podrían ser "muy alto", "no muy bajo", ...
- M: es una regla semántica que asocia a cada valor lingüístico de X su significado M(X) (M(X) denota un subconjunto difuso en U).

Los símbolos terminales de las gramáticas incluyen:

- **Términos primarios** (adjetivos): "bajo", "alto", ...
- **Modificadores** (adverbios): "muy", "más", "menos", "cerca de", ...
- **Conectores lógicos**: Normalmente NOT, AND y OR.

Ejemplo . En el ejemplo de la temperatura tendríamos lo siguiente:

- X: Temperatura.
- T: Caliente, Tibia y Fría.
- U: grados Celcius (°C).
- G: Términos como "muy frio", "muy caliente", etc.
- M: Las funciones de pertenencia de los tres conjuntos difusos ("Caliente", "Tibia" y "Fría").

Definición 12 *Los **cuantificadores** se usan para medir (o cuantificar) la cantidad o la proporción de elementos que cumplen o satisfacen cierta condición.*

De la Lógica Clásica ya conocemos dos muy importantes:

- $\forall$ (Para todo): Se refiere a todos los elementos u objetos.
- $\exists$ (Existe): Se refiere al menos a uno de los elementos u objetos

En la Lógica Difusa se emplean más cuantificadores que se clasifican en dos categorías:

- **Cuantificadores Absolutos**: Se refieren a una única cantidad determinada para medir si esa cantidad son "muchos", "pocos", "muchísimos", "aproximadamente entre 6 y 9", "aprox. más de 43"...
- **Cuantificadores Relativos**: Se refieren a una proporción de elementos respecto del total de los que existen. Por ejemplo: "la mayoría", "la minoría", "casi todos", "casi ninguno"...

Definición 13 *Los **modificadores** son un modo de representar estrategias o técnicas apropiadas cuando el conocimiento proviene de la experiencia o de la intuición (careciendo de demostración matemática o física).*

Ejemplo. Supongamos que tenemos dos conjuntos difusos A y B. El conjunto A será el conjunto de las personas "Viejas" mientras que el conjunto B será el conjunto formado por las personas "Muy Viejas". Una persona que pertenece parcialmente al conjunto "Viejo", tendrá un menor grado de pertenencia al conjunto "Muy Viejo". Así pues, vemos que podríamos obtener el conjunto B aplicando una modificación sobre el conjunto A:

$$\mu_B(x) = (\mu_A(x))^2$$

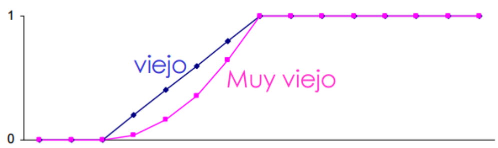

Figura 3.15: Funciones de pertenencia de "Viejo" y "Muy Viejo".

Existe todo un catálogo de posibles adverbios y modificadores asociados, pero las modificaciones que más usualmente se aplican a un conjunto difuso son las siguientes:

- **Concentración**: El efecto es que la función de pertenencia toma valores más pequeños, centrándose en los valores mayores. Se obtiene al aplicar una función tipo $\mu_B(x) = (\mu_A(x))^p$ con $p > 1$. El efecto de aplicar la concentración puede verse en la siguiente figura (la función de pertenencia base es la azul, y la modificada la rosa):

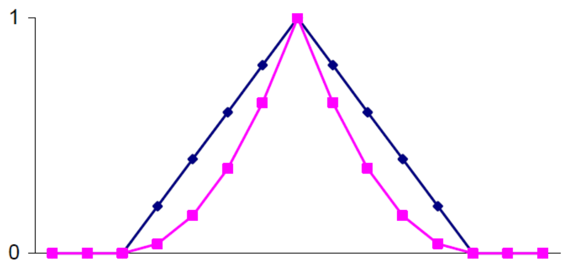

Figura 3.16: Ejemplo de Concentración.

- **Dilatación**: Se obtiene al aplicar una función tipo $\mu_B(x) = (\mu_A(x))^p$ con $0 < p < 1$. Provoca el efecto contrario al de la concentración. La función toma valores mayores. El efecto de aplicar la dilatación puede verse en la siguiente figura (la función de pertenencia base es la azul, y la modificada la rosa):

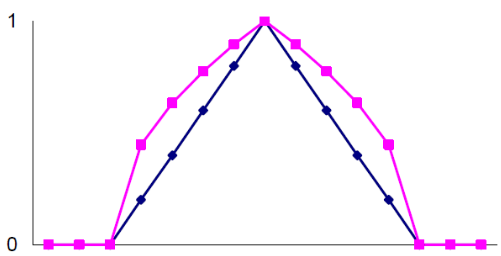

Figura 3.17: Ejemplo de Dilatación.

- **Intensificación**: Se disminuyen los valores menores a 1/2 y se aumentan los mayores. La intensificación viene determinada por una función del tipo

$$\mu_B(x) = \begin{cases} 2^{p-1}(\mu_A(x))^p & \text{si } \mu_A(x) \leq 0{,}5 \\ 1 - 2^{p-1}(1-\mu_A(x))^p & \text{si } \mu_A(x) > 0{,}5 \end{cases}.$$

donde $p > 1$. Normalmente se suele poner p=2 (a mayor p, mayor intensificación). El efecto de aplicar la intensificación puede verse en la siguiente figura (la función de pertenencia base es la azul, y la modificada la rosa):

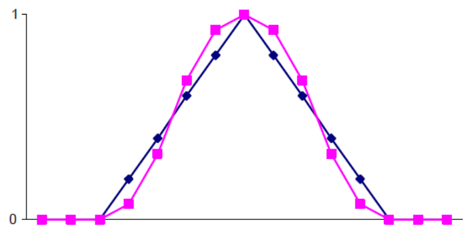

Figura 3.18: Ejemplo de Intensificación.

- **Difuminación**: Provoca el efecto contrario a la intensificación. Viene determinada por una función del tipo

$$\mu_B(x) = \begin{cases} \sqrt{\frac{\mu_A(x)}{2}} & \text{si } \mu_A(x) \leq 0{,}5 \\ 1 - \sqrt{\frac{1-\mu_A(x)}{2}} & \text{si } \mu_A(x) > 0{,}5 \end{cases}$$

El efecto de aplicar la difuminación puede verse en la siguiente figura (la función de pertenencia base es la azul, y la modificada la rosa):

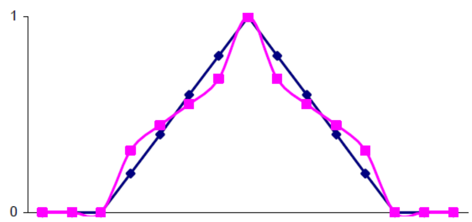

Figura 3.19: Ejemplo de Difuminación.

3.4. Control Difuso

En la actualidad, una de las áreas de aplicación más importantes de la Teoría de Conjuntos Difusos y de la Lógica Difusa la componen los Sistemas Basados en Reglas Difusas (SBRDs). Este tipo de sistemas constituyen una extensión de los Sistemas Basados en Reglas que hacen uso de la Lógica Clásica puesto que emplean reglas del tipo "SI-ENTONCES" (IF-THEN) en las que los antecedentes y consecuentes están compuestos por proposiciones difusas en lugar de proposiciones clásicas.

La **Base de Conocimiento (BC)** representa el conocimiento disponible sobre el problema en el sistema. Se expresa en forma de reglas lingüísticas. La base del conocimiento puede construirse de varias formas, pero una de las más habituales es la de hacerse con los servicios de un experto o experta en la materia, y que él o ella sea el encargado de decidir cuáles son los parámetros de mayor importancia y de establecer las relaciones existentes entre ellos. La Base de Conocimiento se compone de:

- La **Base de Reglas (BR)**, que contiene el conjunto de reglas difusas que serán empleadas en el sistema.
- La **Base de Datos (BD)**, que almacena los conjuntos de términos lingüísticos y las funciones de pertenencia que definen su semántica.

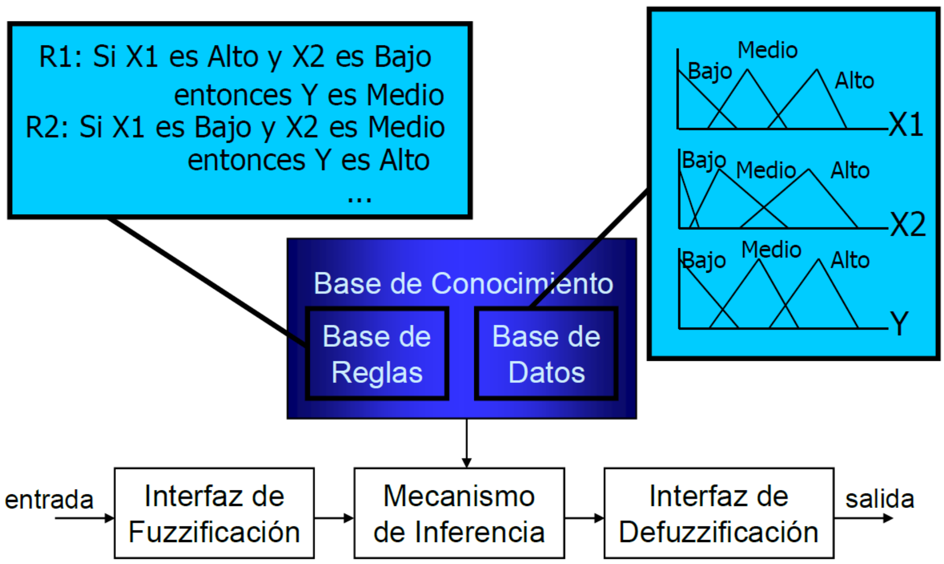

Figura 3.20: Estructura de un Sistema Basado en Reglas Difusas.

3.4.1. Razonamiento aproximado

A principios de los 80, Zadeh introduce el concepto de *Razonamiento Aproximado* y otras componentes que acabarían formando el cuerpo de la Lógica Difusa. De esta forma, propone la utilización de los conjuntos difusos para el manejo cuantitativo de conceptos cualitativos. Muchas veces, la programación clásica no es suficiente para que un sistema realice funciones complejas. Cuando un sistema no ha sido programado explícitamente para realizar una función y se le pide que la realice, el sistema tiene que razonar. Cuando el número de hechos y reglas aumenta, el sistema tiene que poder verificar gran cantidad de hechos que surgen en las etapas de razonamiento. A continuación estudiaremos el concepto de *Regla Difusa* empleada en el Razonamiento Aproximado.

La información para el razonamiento puede venir dada en forma de sentencia o **Proposiciones Atómicas** de la forma "X es A", donde X es el nombre de un objeto y A es el valor que toma dicho objeto. Las Proposiciones Atómicas pueden tomar valores de verdad dentro de un conjunto de posibles valores. Esto implica la existencia de distintas lógicas, clasificadas por el número de valores de verdad posibles: Lógica bivaluada, trivaluada,..., multivaluada. Las proposiciones con atributos imprecisos conllevan el uso de lógica multivaluada o **Lógica Difusa**.

3.4.2. Reglas difusas

Las **reglas difusas** son proposiciones asociadas a adverbios que pueden transformar los conjuntos difusos. Estas reglas emplean sentencias del tipo IF–THEN (SI–ENTONCES):

IF <antecedente o condición> THEN <consecuente o conclusión>

El <antecedente> y el <consecuente> son Proposiciones Difusas que pueden formarse usando conjunciones (AND) o disyunciones (OR). En los sistemas de reglas clásicos, si el antecedente es verdadero, el consecuente es también verdadero. En sistemas donde el antecedente es difuso, todas las reglas se ejecutan parcialmente, y el consecuente es verdadero en cierto grado.

Ejemplo. Trabajamos con un invernadero en el que tenemos una válvula que podemos abrir o cerrar para modificar la temperatura del interior del invernadero. Aquí tendríamos una regla sencilla: SI la Temperatura es Alta ENTONCES Abrir la válvula Poco.

3.4.3. Fuzzificación

Es un proceso que permite asociar a un valor numérico un conjunto difuso, asignándole un grado de pertenencia según un término lingüístico a partir de la función de pertenencia. Este proceso responde a un conjunto de normas preestablecidas, conceptualizadas a partir del conocimiento que brinda el razonamiento humano a través de un sistema implementado vía software. También se conoce como “borrosificación”.

Ejemplo. Volviendo al ejemplo de la temperatura, tenemos tres conjuntos difusos: “Frío”, “Tibio” y “Caliente”. Podemos representar las funciones de pertenencia:

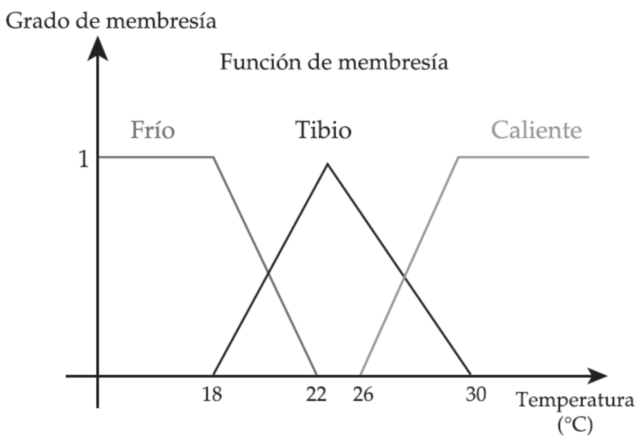

Figura 3.21: Función de pertenencia para “Frío”, “Tibio” y “Caliente”.

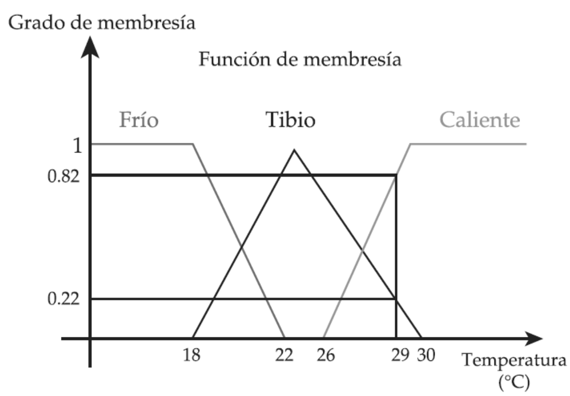

Figura 3.22: Grado de pertenencia para T=29°C.

Así pues, la variable T=29°C presenta los siguientes grados de pertenencia:
$\mu_{Frío}(29) = 0$
$\mu_{Tibio}(29) = 0{,}22$
$\mu_{Caliente}(29) = 0{,}82$.

3.4.4. Defuzzificación

Se trata del proceso inverso a la fuzzificación. El proceso de defuzzificación permite asociar a un conjunto difuso un valor numérico y se lleva a cabo para calcular el valor de salida de los modelos difusos. De hecho, el sistema de inferencia difusa obtiene una conclusión a partir de la información de la entrada, pero se expresa en términos difusos. Esta conclusión o salida difusa es obtenida por la etapa de inferencia borrosa, pero el dato de salida del sistema debe ser un número real y debe ser representativo de todo el conjunto; es por eso que existen diferentes métodos de defuzzificación y arrojan resultados distintos. El más común y ampliamente usado es el método del **centroide**.

Centroide

Asocia el centro del área formada por el número difuso. Matemáticamente se expresa de la siguiente forma:

$$c = \frac{\int_S x\mu(x)dx}{\int_S \mu(x)dx},$$

donde $\mu(x)$ es la función de pertenencia del conjunto de salida, cuya variable es x y S es el dominio o rango de integración.

Bisectriz

Es un método que trata de encontrar el valor numérico del elemento del universo que separa el área de la función de pertenencia del conjunto difuso en dos mitades con la misma área.

Máximo Central (MOM, middle of maximum)

La salida es el valor medio de todos aquellos que generan el valor más alto de la función de pertenencia.

Máximo más pequeño (SOM, smallest of maximum)

La salida es el mínimo valor de todos aquellos que generan el valor más alto de la función de pertenencia.

Máximo más grande (LOM, largest of maximum)

La salida es el máximo valor de todos aquellos que generan el valor más alto de la función de pertenencia.

El siguiente gráfico muestra los anteriores métodos de defuzzificación para un mismo conjunto difuso:

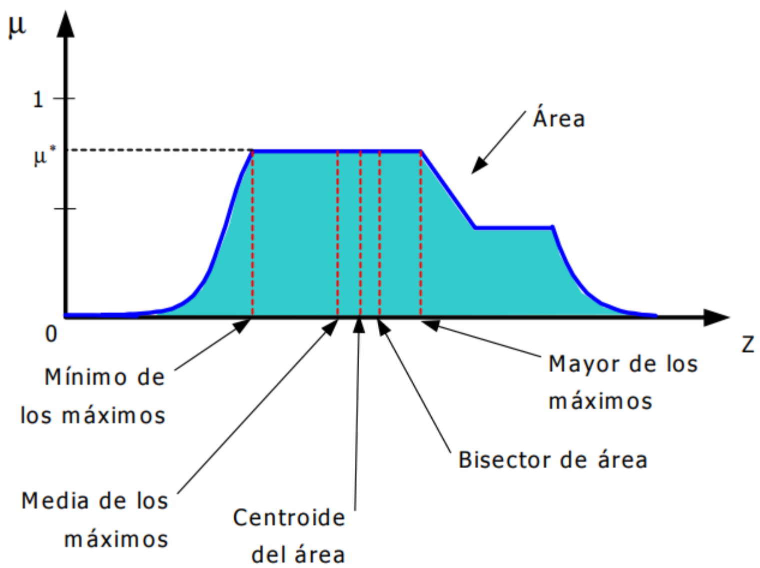

Figura 3.23: Diferentes métodos de defuzzificación.

3.5. Mecanismos de Inferencia

Se entiende por inferencia borrosa la interpretación de las reglas del tipo SI-ENTONCES (IF-THEN) con el fin de obtener las conclusiones de las variables lingüísticas de salida, a partir de los valores de las variables de entrada. La inferencia se basa en el paradigma "Modus Ponens Generalizado" el cual se puede interpretar como la transformación de los grados de cumplimiento del lado izquierdo de las reglas a grados de posibilidad de los lados derechos de las mismas.

Nosotros vamos a analizar únicamente los dos tipos de sistemas de inferencia difusa más usados: los llamados métodos de Mamdani y de Takagi-Sugeno-Kang (TSK).

3.5.1. Inferencia de Mamdani

Este método es posiblemente el más utilizado y fue propuesto por Ebrahim Mamdani en 1975. El proceso se realiza en cuatro pasos:

1. Fuzzificación de las variables de entrada.
2. Evaluación de las reglas.
3. Agregación de las salidas de las reglas.
4. Defuzzificación.

Las etapas de fuzzificación y defuzzificación ya las hemos visto anteriormente. El método de Mamdani utiliza un conjunto de reglas difusas "SI-ENTONCES" (IF-THEN). Toma como entrada los valores de la fuzzificación y se aplican a los antecedentes de las reglas difusas. Si una regla tiene múltiples antecedentes, se utiliza el operador AND u OR para obtener un único número que represente el resultado de la evaluación. Este número (el valor de verdad) se aplica al consecuente.

En el método de Mamdani es habitual listar todas las reglas difusas para saber cómo se obtiene la salida del sistema.

Regla 1: if X es **BAJO** and Y es **BAJO** then Z es **ALTO**
Regla 2: if X es **BAJO** and Y es **ALTO** then Z es **BAJO**
Regla 3: if X es **ALTO** and Y es **BAJO** then Z es **BAJO**
Regla 4: if X es **ALTO** and Y es **ALTO** then Z es **ALTO**

Figura 3.24: Ejemplo de Reglas Difusas en un sistema de Mamdani.

Otra manera muy extendida de presentar las reglas es mediante una tabla:

		X	
		BAJO	**ALTO**
Y	**BAJO**	ALTO	BAJO
	ALTO	BAJO	ALTO

Figura 3.25: Tabla de Mamdani.

Para evaluar la disyunción (operador OR) habitualmente se emplea la T-Conorma estándar (máximo), definida como hemos visto anteriormente. De igual forma, para el operador AND se usa habitualmente la T-Norma estándar del mínimo. Finalmente el resultado de la evaluación del antecedente se relaciona con el consecuente aplicando un **recorte** o **escalado** según el valor de verdad del antecedente para obtener como salida de la regla un conjunto difuso. El método más comúnmente utilizado es el **recorte** (clipping) que corta el consecuente con el valor de verdad del antecedente. El **escalado** proporciona un resultado más preciso, preservando la forma original del conjunto difuso y se obtiene multiplicando todos los valores por el valor de verdad del antecedente.

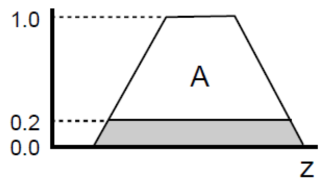

Figura 3.26: Conjunto recortado.

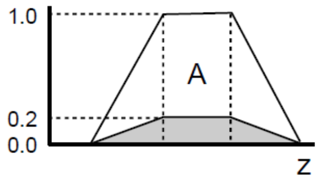

Figura 3.27: Conjunto escalado.

La "agregación" es el proceso de unificación de las salidas de todas las reglas; es decir, se combinan (normalmente mediante la unión) las funciones de pertenencia de todos los consecuentes, previamente recortados o escalados, para obtener un único conjunto difuso por cada variable de salida.

Vamos a ver a continuación un ejemplo de un controlador difuso aplicando el método de Mamdani. A partir de las variables de entrada y de las reglas difusas facilitadas por un experto obtendremos un valor de salida del sistema.

Ejemplo. Supongamos que queremos deducir la propina que un comensal ha de entregar a partir de la calidad del servicio y de la comida. Para ello, haremos uso de un sistema de reglas difusas. El sistema cuenta con dos variables de entrada *Servicio* (Calidad del Servicio, que se evalúa de 0 a 10), y *Comida* (Calidad de la Comida, que se evalúa igualmente de 0 a 10). El porcentaje de propina se modela con la variable *Propina* (definida entre 5 % y 25 % del precio total).

A la variable de entrada Servicio le asociaremos tres conjuntos difusos con las etiquetas lingüísticas "Pobre", "Bueno" y "Excelente". La calidad de la Comida tendrá asociada dos conjuntos difusos, con las etiquetas "Rancia" y "Deliciosa". Por último, la Propina que dejaremos tendrá asociada tres conjuntos difusos con las etiquetas "Tacaña", "Promedio" y "Generosa".

Contamos con un sistema de reglas difusas que ha sido elaborado por un experto en el ámbito de la hostelería, una persona que pueda valorar correctamente tanto el servicio como la comida en un restaurante. Dicho experto nos proporciona las siguientes reglas, aunque podrían ser más:

$$R_1 : \text{Si Servicio es Pobre} \vee \text{Comida es Rancia} \longrightarrow \text{Propina es Tacaña}$$
$$R_2 : \text{Si Servicio es Bueno} \longrightarrow \text{Propina es Promedio}$$
$$R_3 : \text{Si Servicio es Excelente} \vee \text{Comida es Deliciosa} \longrightarrow \text{Propina es Generosa}$$

Supongamos que un comensal puntúa el Servicio con un 3 y la Comida con un 8. Vamos a ver qué Propina tiene que dejar dicho comensal.

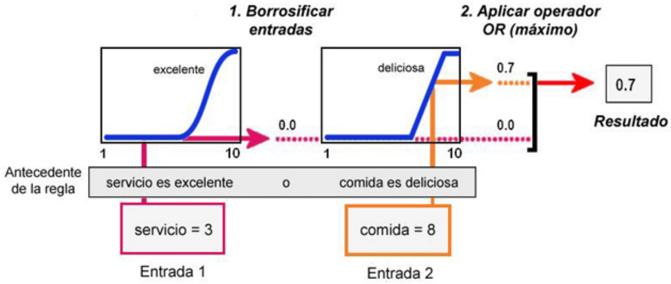

Figura 3.28: Fuzzificación de las variables de entrada en la tercera regla (R_3).

Si nos fijamos en la tercera regla (R_3) y puesto que el comensal ha puntuado con un 3 el "Servicio", en la fuzzificación le corresponde un grado de pertenencia 0 al conjunto difuso "Excelente". Por otra parte, puesto que ha puntuado con un 8 la calidad de la "Comida", al fuzzificar le corresponde un valor de pertenencia de 0.7 al conjunto difuso "Deliciosa" (ver Figura 3.28).

Ya que los dos términos del antecedente están unidos por una disyunción (Servicio Excelente o Comida Deliciosa), hemos aplicado un operador OR, en este caso el máximo, a los dos valores de pertenencia anteriores y obtenemos el valor de pertenencia 0.7. Si los términos del antecedente estuvieran unidos por una conjunción ("y"), habría que aplicar un operador AND, por ejemplo el mínimo.

A partir del consecuente de cada regla (un conjunto difuso) y del valor del antecedente obtenido en el paso anterior, aplicamos un operador difuso de implicación obteniendo así un nuevo conjunto difuso. Como hemos visto anteriormente tenemos dos posibles operadores de implicación. Estos son el recorte, que trunca la función de pertenencia del consecuente, y el escalado. En la Figura 3.29 se describe la obtención de la salida de la tercera regla (R_3) aplicando el recorte:

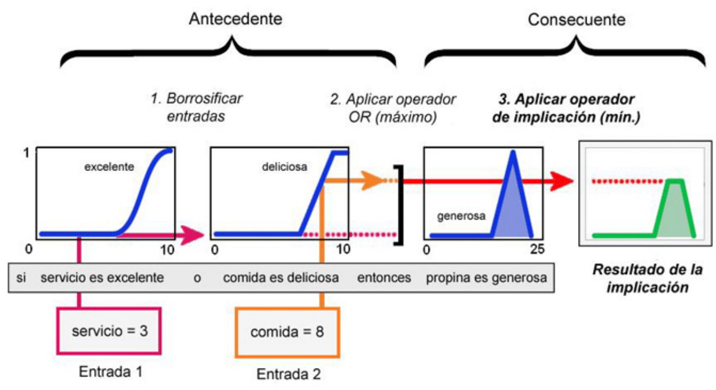

Figura 3.29: Construcción del conjunto difuso de salida de la tercera regla utilizando el recorte.

De modo similar obtendriamos un conjunto difuso para las otras dos reglas (R_1 y R_2). Las salidas obtenidas para cada regla se combinan en un único conjunto difuso utilizando un operador de unión (agregación). Por último basta con aplicar uno de los métodos que hemos visto anteriormente para defuzzificar y obtener un valor representativo del conjunto. El más habitual, como ya hemos dicho, es el método del centroide, así que usaremos este.

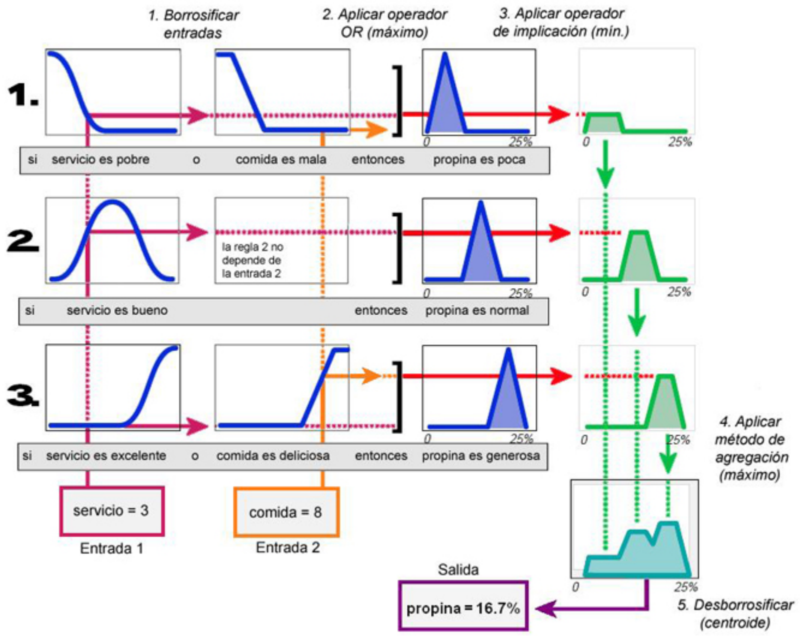

Figura 3.30: Esquema completo del método de Mamdani aplicado a nuestro caso.

En conclusión, si el comensal le asigna al Servicio una puntuación de 3 y a la Comida una puntuación de 8, la propina sugerida es del 16.7 % del monto total de la cena.

3.5.2. Inferencia de Takagi-Sugeno-Kang (TSK)

En 1985, Takagi y Sugeno aportan a la teoría del control difuso un nuevo método llamado de Takagi-Sugeno-Kang (TSK), como alternativa al método de Mamdani. Se trata de un método basado en reglas difusas pero en el que el consecuente no nos da un conjunto difuso sino una serie de funciones lineales. Este modelo es útil para sistemas complejos y de dimensiones mayores que los que podemos resolver por el método de Mamdani.

Sean A_i y B_i, con $i = 1, 2, ..., n$, conjuntos difusos de nuestro sistema. Las reglas tendrían la siguiente forma:

$$
\begin{array}{c}
R_1\text{: Si } x \text{ es } A_1 \wedge y \text{ es } B_1 \text{ entonces } z = f_1(x, y) \\
R_2\text{: Si } x \text{ es } A_2 \wedge y \text{ es } B_2 \text{ entonces } z = f_2(x, y) \\
. \\
. \\
. \\
R_n\text{: Si } x \text{ es } A_n \wedge y \text{ es } B_n \text{ entonces } z = f_n(x, y)
\end{array}
$$

La principal diferencia que presenta el método TSK respecto al de Mamdani es que no es necesario realizar un proceso de defuzzificación. Esto se debe al hecho de que no obtenemos ningún conjunto difuso sino un conjunto de funciones lineales. Así, en el método TSK podemos obtener directamente el valor de salida de sistema con una expresión del tipo:

$$Z_0 = \frac{\sum_{i=1}^{n} \omega_i f_i(x_i, y_i)}{\sum_{i=1}^{n} \omega_i},$$

donde el valor ω_i se obtiene calculando el mínimo de los valores de entrada en cada regla R_i (ver Figura 3.31).

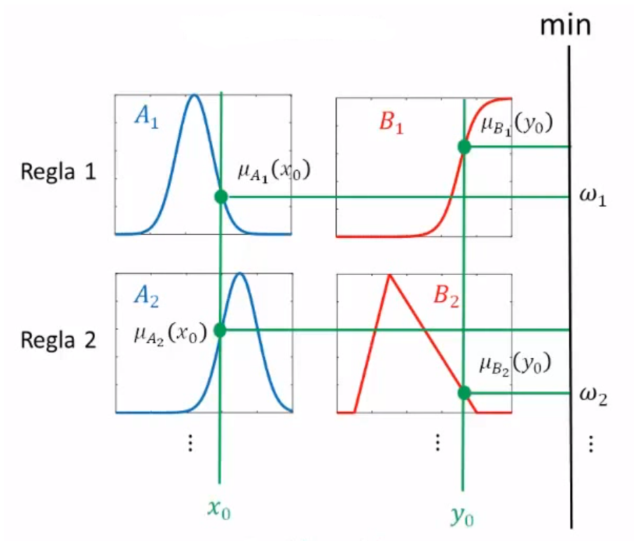

Figura 3.31: Cálculo de las componentes ω_i.

Ejemplo. Consideramos que queremos diseñar un sistema para controlar la posición de un cartucho de una impresora de inyección de tinta. Partimos de una Curva de Control en la cual el eje de abscisas representa el *Error* y la ordenada el *Voltaje* que se le aplica al motor. A partir de ella podemos establecer las Reglas de Control (ver Figura 3.32).

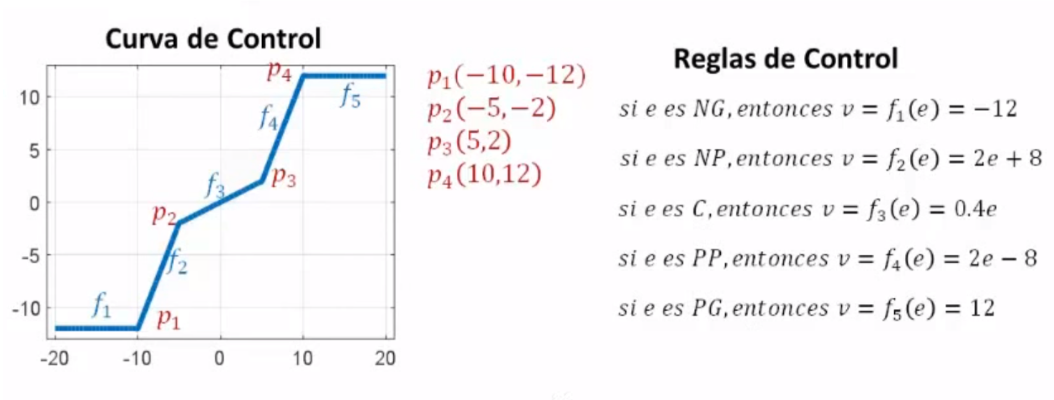

Figura 3.32: Curva de control y reglas de control del problema de la impresora.

A partir de la curva de control obtenemos las cinco funciones lineales y los cuatro puntos característicos que separan dichas funciones. A la variable de entrada *Error* le asociamos cinco términos lingüísticos que representan cinco conjuntos difusos: NG ("Negativo Grande"), NP ("Negativo Grande"), C ("Cero"), PP ("Positivo Pequeño") y PG ("Positivo Grande").

A continuacion se presentan los cinco funciones de pertenencia de los conjuntos difusos del sistema: NG (azul), NP (naranja), C (amarillo), PP (morado) y PG (verde).

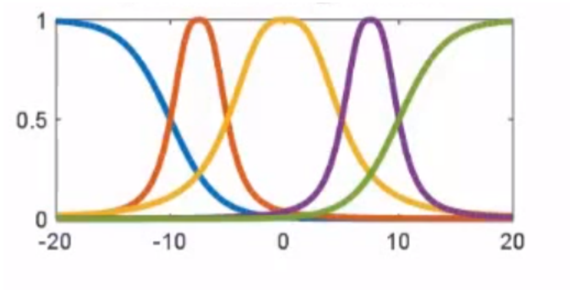

Figura 3.33: Representación de las funciones de pertenencia de los conjuntos difusos asociados a la variable *Error*.

Este ejemplo es muy sencillo, ya que solo utilizamos una variable. Supongamos que tenemos un error nulo ($e = 0$). En este caso tendríamos que:
$\mu_{NP}(0) = 0{,}1$
$\mu_{C}(0) = 1$
$\mu_{PP}(0) = 0{,}1$.
Como para cada regla de control solo tenemos una variable y ω_i se obtiene del mínimo de las variables de entrada, obtenemos de forma inmediata:
$\omega_1 = 0{,}1$
$\omega_2 = 1$
$\omega_3 = 0{,}1$.

Calculamos ahora el valor de salida del sistema para un error nulo:

$$Z_0 = \frac{0{,}1 * f_1(0) + 1 * f_2(0) + 0{,}1 * f_3(0)}{0{,}1 + 1 + 0{,}1}$$

$$= \frac{0{,}1 * 8 + 1 * 0 - 0{,}1 * 8}{1{,}2} = 0$$

Como era de esperar, cuando no hay error alguno, no es necesario aplicar ningún voltaje a la impresora.

3.6. Ejemplo resuelto con el método de Mamdani y TSK

Hasta ahora hemos visto los conceptos básicos de la Lógica Difusa y los dos métodos más habituales para el control de procesos en este contexto. Vamos a ver un ejemplo aplicado a la medicina utilizando los métodos de Mamdani y TSK y analizaremos las diferencias entre ambos métodos.

Queremos calcular el tiempo de vida que puede tener un paciente que sufre un cáncer gástrico. Para calcular el tiempo de vida emplearemos dos variables: X="Edad" e Y="Valor CRP" (C-Reactive Proteins). La variable X nos dirá la edad del paciente mientras que la variable Y es un marcador tumoral, uno de los valores más utilizados a la hora de hacer pronósticos en oncología. La salida de nuestro sistema de control será la variable Z="Tiempo de vida".

Para nuestro ejemplo asociaremos a cada una de nuestras variables cinco conjuntos difusos:

- Para la variable "Edad" consideraremos los conjuntos difusos X_0 = "*Muy joven*", X_1 = "*Joven*", X_2 = "*Mediana Edad*", X_3 = "*Viejo*" y X_4 = "*Muy Viejo*".
- Para "Valor CRP" consideraremos Y_0 = "*Muy Bajo*", Y_1 = "*Bajo*", Y_2 = "*Medio*", Y_3 = "*Alto*" e Y_4 = "*Muy Alto*".
- Para "Tiempo de vida" consideraremos Z_0 = "*Muy Corto*", Z_1 = "*Corto*", Z_2 = "*Mediano*", Z_3 = "*Largo*" y Z_4 = "*Muy Largo*".

Presentamos a continuación las funciones de pertenencia de los conjuntos difusos asociados a la variable X="Edad":

$$\mu_{X0}(x) = \begin{cases} 1 - 2(\frac{x}{25})^2 & \text{si } 0 < x \leq 12{,}5 \\ 2(\frac{x-25}{25})^2 & \text{si } 12{,}5 < x \leq 25 \end{cases}$$

$$\mu_{X1}(x) = \begin{cases} 2(\frac{x}{25})^2 & \text{si } 0 < x \leq 12{,}5 \\ 1 - 2(\frac{x-25}{25})^2 & \text{si } 12{,}5 < x \leq 37{,}5 \\ 2(\frac{x-50}{25})^2 & \text{si } 37{,}5 < x \leq 50 \end{cases}$$

$$\mu_{X2}(x) = \begin{cases} 2(\frac{x-25}{25})^2 & \text{si } 25 < x \leq 37{,}5 \\ 1 - 2(\frac{x-50}{25})^2 & \text{si } 37{,}5 < x \leq 62{,}5 \\ 2(\frac{x-75}{25})^2 & \text{si } 62{,}5 < x \leq 75 \end{cases}$$

$$\mu_{X3}(x) = \begin{cases} 2(\frac{x-50}{25})^2 & \text{si } 50 < x \leq 62{,}5 \\ 1 - 2(\frac{x-75}{25})^2 & \text{si } 62{,}5 < x \leq 87{,}5 \\ 2(\frac{x-100}{25})^2 & \text{si } 87{,}5 < x \leq 100 \end{cases}$$

$$\mu_{X4}(x) = \begin{cases} 2(\frac{x-75}{25})^2 & \text{si } 75 < x \leq 78{,}5 \\ 1 - 2(\frac{x-100}{25})^2 & \text{si } 87{,}5 < x \leq 100 \end{cases}$$

Funciones de pertenencia de los conjuntos difusos asociados a la variable Y="Valor CRP":

$$\mu_{Y0}(y) = \begin{cases} 1 - 2(\frac{y}{15})^2 & \text{si } 0 < y \leq 7{,}5 \\ 2(\frac{y-15}{15})^2 & \text{si } 7{,}5 < y \leq 15 \end{cases}$$

$$\mu_{Y1}(y) = \begin{cases} 2(\frac{y}{15})^2 & \text{si } 0 < y \leq 7{,}5 \\ 1 - 2(\frac{y-15}{15})^2 & \text{si } 7{,}5 < y \leq 22{,}5 \\ 2(\frac{y-30}{15})^2 & \text{si } 22{,}5 < y \leq 30 \end{cases}$$

$$\mu_{Y2}(y) = \begin{cases} 2(\frac{y-15}{15})^2 & \text{si } 15 < y \leq 22{,}5 \\ 1 - 2(\frac{y-30}{15})^2 & \text{si } 22{,}5 < y \leq 37{,}5 \\ 2(\frac{y-45}{15})^2 & \text{si } 37{,}5 < y \leq 45 \end{cases}$$

$$\mu_{Y3}(y) = \begin{cases} 2(\frac{y-30}{15})^2 & \text{si } 30 < y \leq 37{,}5 \\ 1 - 2(\frac{y-45}{15})^2 & \text{si } 37{,}5 < y \leq 52{,}5 \\ 2(\frac{y-60}{15})^2 & \text{si } 52{,}5 < y \leq 60 \end{cases}$$

$$\mu_{Y4}(y) = \begin{cases} 2(\frac{y-45}{15})^2 & \text{si } 45 < y \leq 52{,}5 \\ 1 - 2(\frac{y-60}{15})^2 & \text{si } 52{,}5 < y \leq 60 \end{cases}$$

Funciones de pertenencia de los conjuntos difusos asociados a la variable Z="Tiempo de vida":

$$\mu_{Z0}(z) = \begin{cases} 1 & \text{si } 0 < z \leq 0{,}5 \\ 1 - 8(z-0{,}5)^2 & \text{si } 0{,}5 < z \leq 0{,}75 \end{cases}$$

$$\mu_{Z1}(z) = \begin{cases} 8(z-0{,}5)^2 & \text{si } 0{,}5 < z \leq 0{,}75 \\ 1 - 8(z-1)^2 & \text{si } 0{,}75 < z \leq 1 \\ 1 & \text{si } 1 < z \leq 1{,}5 \\ 1 - 8(z-1{,}5)^2 & \text{si } 1{,}5 < z \leq 1{,}75 \\ 8(z-2)^2 & \text{si } 1{,}75 < z \leq 2 \end{cases}$$

$$\mu_{Z2}(z) = \begin{cases} 8(z-1{,}5)^2 & \text{si } 1{,}5 < z \leq 1{,}75 \\ 1 - 8(z-2)^2 & \text{si } 1{,}75 < z \leq 2 \\ 1 & \text{si } 1 < z \leq 2{,}5 \\ 1 - 8(z-2{,}5)^2 & \text{si } 2{,}5 < z \leq 2{,}75 \\ 8(z-3)^2 & \text{si } 2{,}75 < z \leq 3 \end{cases}$$

$$\mu_{Z3}(z) = \begin{cases} 8(z-2{,}5)^2 & \text{si } 2{,}5 < z \leq 2{,}75 \\ 1 - 8(z-3)^2 & \text{si } 2{,}75 < z \leq 3 \\ 1 & \text{si } 1 < z \leq 3{,}5 \\ 1 - 8(z-3{,}5)^2 & \text{si } 3{,}5 < z \leq 3{,}75 \\ 8(z-4)^2 & \text{si } 3{,}75 < z \leq 4 \end{cases}$$

$$\mu_{Z4}(y) = \begin{cases} 8(z-3{,}5)^2 & \text{si } 3{,}5 < z \leq 3{,}75 \\ 1 - 8(z-4)^2 & \text{si } 3{,}75 < z \leq 4 \\ 1 & \text{si } 1 < z \leq 4{,}5 \end{cases}$$

A partir de las funciones de pertenencia, es sencillo representar los distintos conjuntos difusos sobre una misma gráfica:

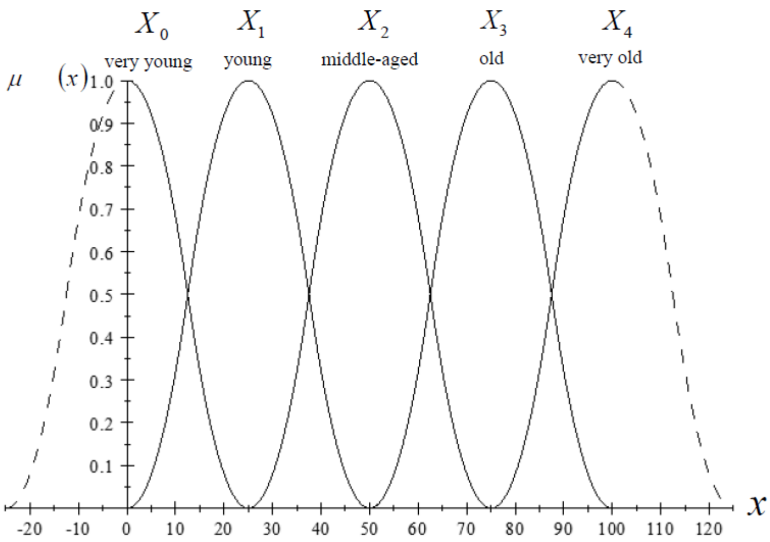

Figura 3.34: Conjuntos difusos de la variable X="Edad".

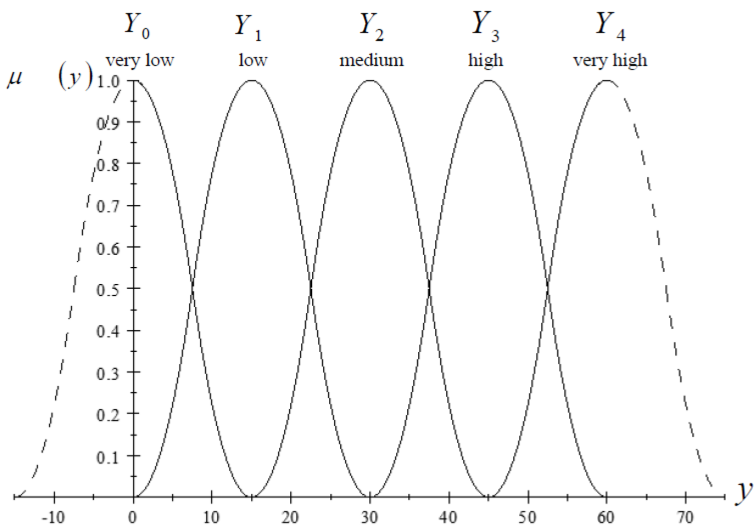

Figura 3.35: Conjuntos difusos de la variable Y="Valor CRP".

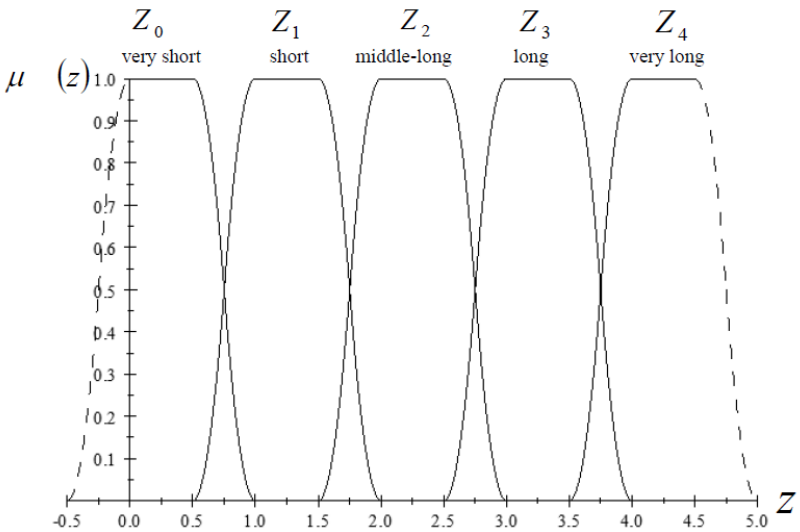

Figura 3.36: Conjuntos difusos de la variable Z="Tiempo de vida".

Analicemos ahora el caso de una persona de 77 años ($x = 77$) cuyo valor de CRP es 16 ($y = 16$). A continuación vamos a pronosticar el tiempo de vida que puede quedarle a esta persona empleando los dos métodos que hemos visto anteriormente.

Análisis con el método de Mamdani

Primero tenemos que llevar a cabo el proceso de fuzzificación. Para esto, evaluamos las funciones de pertenencia en los valores que tenemos. Para $x = 77$ tenemos que concretamente:
$\mu_{X0}(77) = \mu_{MuyJoven}(77) = 0$
$\mu_{X1}(77) = \mu_{Joven}(77) = 0$
$\mu_{X2}(77) = \mu_{MedianaEdad}(77) = 0$
$\mu_{X3}(77) = \mu_{Viejo}(77) = 0'9872$
$\mu_{X4}(77) = \mu_{MuyViejo}(77) = 0'0128.$

Como podemos observar, una persona de 77 años tiene un valor de pertenencia al conjunto difuso "Viejo" de 0'9872 y de 0'0128 para el conjunto difuso "Muy Viejo". Vemos ahora que para $y = 16$ tenemos que:

$\mu_{Y0}(16) = \mu_{MuyBajo}(16) = 0$
$\mu_{Y1}(16) = \mu_{Bajo}(16) = 0'991$
$\mu_{Y2}(16) = \mu_{Medio}(16) = 0'009$
$\mu_{Y3}(16) = \mu_{Alto}(16) = 0$
$\mu_{Y4}(16) = \mu_{MuyAlto}(16) = 0.$

Aquí podemos ver que un CRP de 16 tiene un valor de pertenencia 0’991 para el conjunto “Pequeño” y 0’009 para el conjunto “Medio”.

El siguiente paso en el método de Mamdani consiste en crear un conjunto consecuente a partir de las reglas difusas de nuestro sistema. Como en nuestro caso cada variable solo tiene valor de pertenencia mayor que 0 sobre dos conjuntos difusos, solo tendremos en cuenta las cuatro reglas que presentamos a continuación, obtenidas a partir de la opinión de los especialistas en oncología en casos anteriores.:

$$
\begin{gathered}
R_1 : \text{Si } X \text{ es “Viejo”} \wedge Y \text{ es “Bajo” entonces } Z \text{ es “Corto”} \\
R_2 : \text{Si } X \text{ es “Viejo”} \wedge Y \text{ es “Medio” entonces } Z \text{ es “Corto”} \\
R_3 : \text{Si } X \text{ es “Muy Viejo”} \wedge Y \text{ es “Bajo” entonces } Z \text{ es “Corto”} \\
R_4 : \text{Si } X \text{ es “Muy Viejo”} \wedge Y \text{ es “Medio” entonces } Z \text{ es “Muy Corto”}.
\end{gathered}
$$

Con la evaluación del antecedente de cada regla difusa obtendremos el valor (en nuestro caso ω) que utilizaremos para realizar un recorte sobre el conjunto consecuente, es decir, el conjunto Z_i que corresponda a cada regla.

$$
\begin{gathered}
R_1 \longrightarrow \omega_1 = min(\mu_{X3}(77), \mu_{Y1}(16)) = min(0'9872, 0'991) = 0'9872 \\
R_2 \longrightarrow \omega_2 = min(\mu_{X3}(77), \mu_{Y2}(16)) = min(0'9872, 0'009) = 0'009 \\
R_3 \longrightarrow \omega_3 = min(\mu_{X4}(77), \mu_{Y1}(16)) = min(0'0128, 0'991) = 0'0128 \\
R_4 \longrightarrow \omega_4 = min(\mu_{X4}(77), \mu_{Y2}(16)) = min(0'0128, 0'009) = 0'009.
\end{gathered}
$$

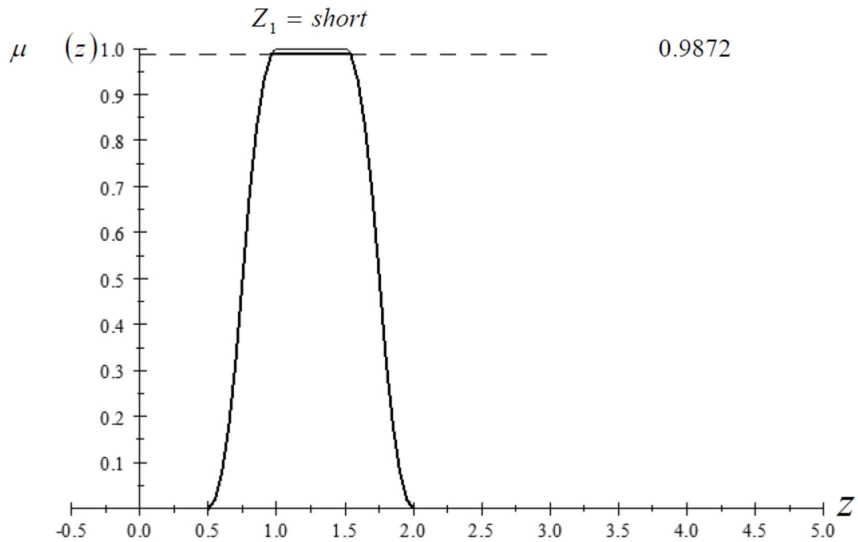

Figura 3.37: Recorte sobre Z_1 aplicando ω_1.

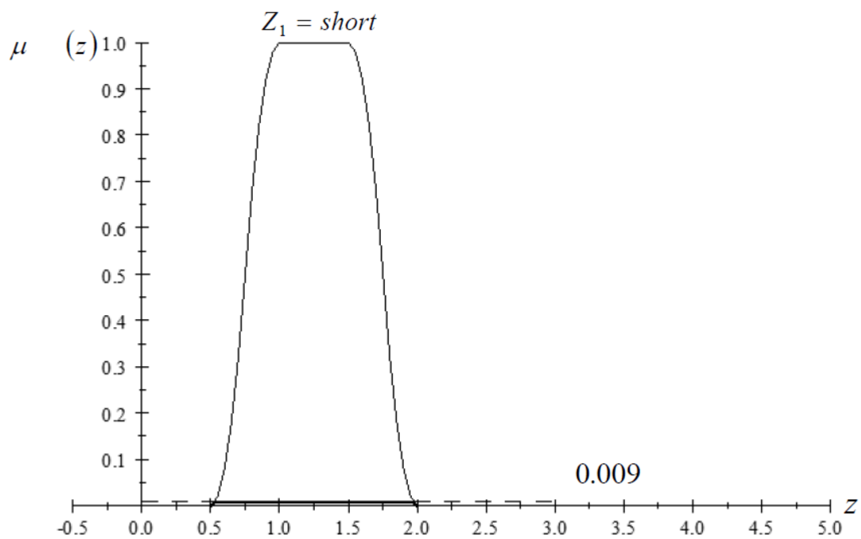

Figura 3.38: Recorte sobre Z_1 aplicando ω_2.

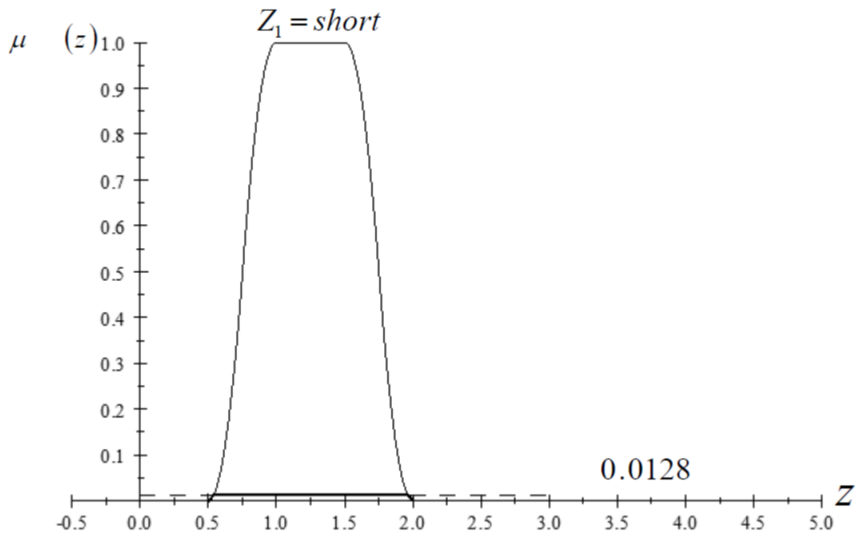

Figura 3.39: Recorte sobre Z_1 aplicando ω_3.

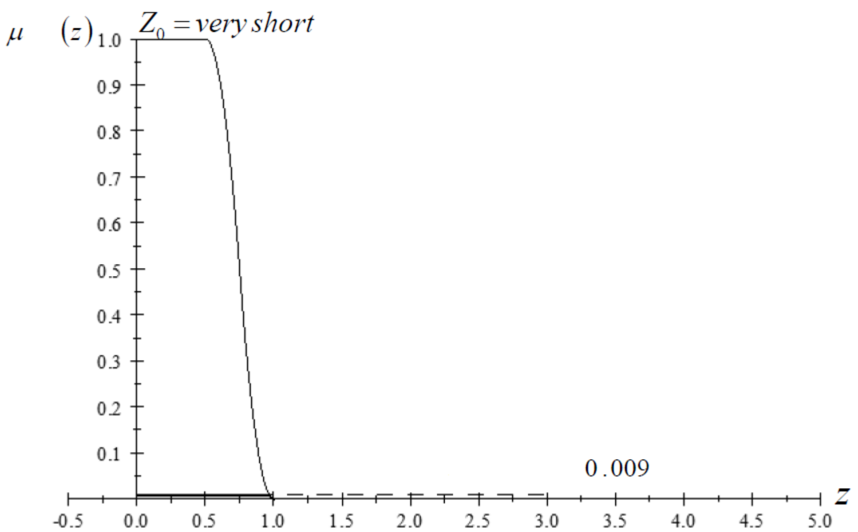

Figura 3.40: Recorte sobre Z_0 aplicando ω_4.

Una vez obtenidos los cuatro conjuntos difusos, realizamos la agregación empleando el operador unión (ver Figura 3.41) y obtenemos un nuevo conjunto difuso de salida sobre el que tenemos que aplicar defuzzificación para obtener un valor representativo del mismo. Para obtener el valor resultante del sistema para la entrada (x, y)=(77, 16) utilizaremos el método del centroide.

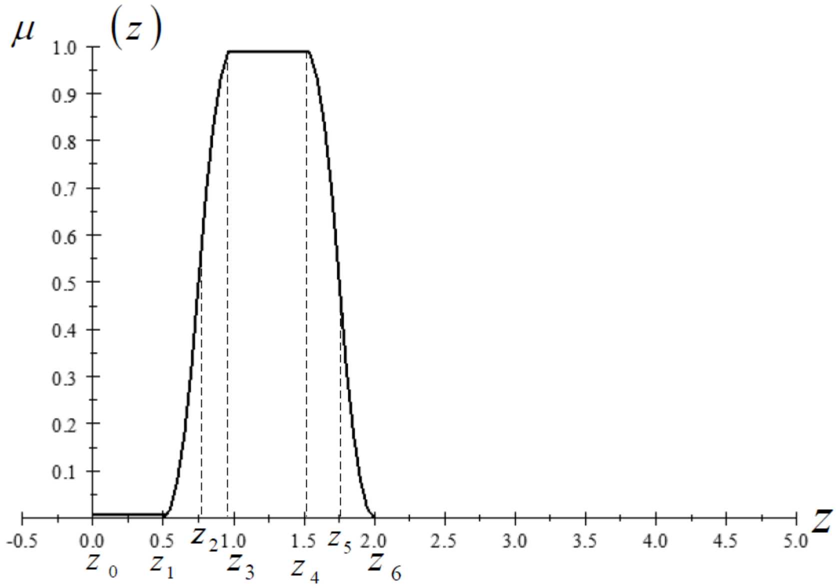

Figura 3.41: Conjunto resultado al realizar la agregación.

La gráfica de la función está definida en el intervalo [0, 2] con los puntos característicos $z_0 = 0, z_1 = 0{,}533, z_2 = 0{,}75, z_3 = 0{,}96, z_4 = 1{,}54, z_5 = 1{,}75$ y $z_6 = 2$.

$$z = f(77, 16) =$$

$$\frac{\int_0^{0,533} 0{,}009 z dz + \int_{0,533}^{0,75} 8(z-0{,}5)^2 z dz + \int_{0,75}^{0,96} 1-8(z-1)^2 z dz + \int_{0,96}^{1,54} 0{,}9872 z dz + \int_{1,54}^{1,75} 1-8(z-1{,}5)^2 z dz + \int_{1,75}^{2} 8(z-2)^2 z dz}{\int_0^{0,533} 0{,}009 dz + \int_{0,533}^{0,75} 8(z-0{,}5)^2 dz + \int_{0,75}^{0,96} 1-8(z-1)^2 dz + \int_{0,96}^{1,54} 0{,}9872 dz + \int_{1,54}^{1,75} 1-8(z-1{,}5)^2 dz + \int_{1,75}^{2} 8(z-2)^2 dz} =$$

$$= \frac{0{,}5064500169645}{0{,}48233334949} = 1{,}05$$

Así pues, según nuestro sistema, para una persona de 77 años y un valor CRP igual a 16 podemos pronosticar que le queda poco más de 1 año de vida.

Análisis con el método TSK

Seguimos queriendo evaluar el tiempo de vida en pacientes con cáncer gástrico a partir de la información sobre su edad y valor de CRP. En este apartado utilizaremos el método TSK en lugar del método de Mamdani para el control difuso.

El proceso de fuzzificación es el mismo que en el método de Mamdani, es decir, tendremos que para $x = 77$:
$\mu_{X0}(77) = \mu_{MuyJoven}(77) = 0$
$\mu_{X1}(77) = \mu_{Joven}(77) = 0$
$\mu_{X2}(77) = \mu_{MedianaEdad}(77) = 0$
$\mu_{X3}(77) = \mu_{Viejo}(77) = 0'9872$
$\mu_{X4}(77) = \mu_{MuyViejo}(77) = 0'0128,$

y para $y = 16$:
$\mu_{Y0}(16) = \mu_{MuyBajo}(16) = 0$
$\mu_{Y1}(16) = \mu_{Bajo}(16) = 0'991$
$\mu_{Y2}(16) = \mu_{Medio}(16) = 0'009$
$\mu_{Y3}(16) = \mu_{Alto}(16) = 0$
$\mu_{Y4}(16) = \mu_{MuyAlto}(16) = 0.$

En este método, a diferencia del anterior, para cada combinación de valores x e y obtenemos una función lineal de la forma general $f(x, y) = ax + by + c$. Para determinar la función f es necesario disponer de ternas de datos (x, y, z) obtenidas experimentalmente.

Así pues, por experiencias anteriores se dispone de una serie de valores para las ternas (x, y, z) (ver Cuadro 3.1). Si nos fijamos en los valores (x, y) que tienen algún grado de pertenencia a X_3 e Y_1 (R_1), obtenemos, empleando la regresión lineal, que para el par de conjuntos difusos X_3 e Y_1 encontramos la función $z_1 = f_1(x, y) = 0{,}13057x + 3{,}4256 \cdot 10^{-3}y - 9{,}4426$. De igual manera procederíamos con el resto de combinaciones.

x	y	z
77	18	0.5
79	18	0.6
81	21	0.9
90	15	0.5
75	17	0.8
73	17	0.5
84	18	0.5
72	19	0.6
73	15	0.6
71	16	0.6
75	19	0.7
70	16	0.8

Cuadro 3.1: Datos pacientes

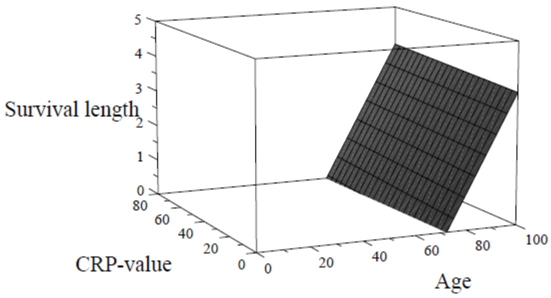

Figura 3.42: Dependencia funcional lineal entre X_3, Y_1 y Z.

Las reglas IF...THEN... en el método TSK no nos proporcionan un conjunto difuso asociado a Z como consecuente sino una función lineal tal y como acabamos de comentar.

$$R_1 : \text{Si } X \text{ es “Viejo”} \wedge Y \text{ es “Bajo” entonces}$$
$$z_1 = f_1(x, y) = -0{,}13057x + 3{,}4256 \cdot 10^{-3}y - 9{,}4426$$
$$R_2 : \text{Si } X \text{ es “Viejo”} \wedge Y \text{ es “Medio” entonces}$$
$$z_2 = f_2(x, y) = -1{,}0077 \cdot 10^{-2}x - 2{,}7872 \cdot 10^{-2}y + 2{,}0658$$
$$R_3 : \text{Si } X \text{ es “Muy Viejo”} \wedge Y \text{ es “Bajo” entonces}$$
$$z_3 = f_3(x, y) = -6{,}2637 \cdot 10^{-2}x + 0{,}26035y + 4{,}1361$$
$$R_4 : \text{Si } X \text{ es “Muy Viejo”} \wedge Y \text{ es “Medio” entonces}$$

$$z_4 = f_4(x, y) = -9{,}7739 \cdot 10^{-3}x + 3{,}3194 \cdot 10^{-4}y + 1{,}1162.$$

Obtenemos los valores ω_i, de la misma manera que en el método de Mamdani.

$$\begin{array}{c}
R_1 \longrightarrow \omega_1 = min(\mu_{X3}(77), \mu_{Y1}(16)) = min(0'9872, 0'991) = 0'9872 \\
R_2 \longrightarrow \omega_2 = min(\mu_{X3}(77), \mu_{Y2}(16)) = min(0'9872, 0'009) = 0'009 \\
R_3 \longrightarrow \omega_3 = min(\mu_{X4}(77), \mu_{Y1}(16)) = min(0'0128, 0'991) = 0'0128 \\
R_4 \longrightarrow \omega_4 = min(\mu_{X4}(77), \mu_{Y2}(16)) = min(0'0128, 0'009) = 0'009.
\end{array}$$

A partir de las funciones lineales y los valores ω_i podemos calcular el valor de salida del sistema según propone el método TSK:

$$z_0 = \frac{\sum_{i=1}^{n} \omega_i f_i(x_i,y_i)}{\sum_{i=1}^{n} \omega_i} =$$

$$= \frac{\omega_1 \cdot f_1(77{,}16) + \omega_2 \cdot f_2(77{,}16) + \omega_3 \cdot f_3(77{,}16) + \omega_4 \cdot f_4(77{,}16)}{\omega_1 + \omega_2 + \omega_3 + \omega_4} =$$

$$= \frac{0{,}9872 \cdot 0{,}8555 + 0{,}009 \cdot 0{,}84392 + 0{,}0128 \cdot 3{,}47865 + 0{,}009 \cdot 0{,}36895}{0{,}9872 + 0{,}009 + 0{,}0128 + 0{,}009} =$$

$$= \frac{0{,}8999215}{1{,}018} \approx 0{,}88$$

Así pues, empleando el método TSK, a una persona de 77 años y con un valor CRP igual a 16 podemos pronosticar que le queda menos de 1 año de vida.

Capítulo 4

Conclusiones

Durante la Estancia en Prácticas diseñé una base de datos en MySQL para llevar a cabo una migración y una aplicación web para poder acceder a dicha base. En este tiempo pude aplicar muchos de los conceptos de programación vistos en la carrera así como algunos nuevos que aprendí. También pude experimentar el trabajo en equipo; todos aportábamos ideas para que nuestra base de datos fuera lo más eficiente posible y siempre podía consultar dudas a mis compañeros. La experiencia ha sido muy positiva.

En el Trabajo Fin de Grado se introduce la Teoria de Conjuntos Difusos y se detallan los dos métodos más utilizados para la construcción de sistemas de control difuso, una de las principales aplicaciones de la Lógica Difusa. Dicha lógica, junto con otras herramientas, han abierto las puertas al tratamiento de una gran variedad de fenómenos y problemas que resultaban, hasta hace algunos años, difíciles de abordar.

El primero de los métodos analizados, el método de Mamdani, exige un gran número de operaciones en la fase de procesamiento, y es muy útil cuando tenemos un número reducido de variables. Cuando el número de variables aumenta, nos encontramos con los siguientes inconvenientes:

- El número de reglas aumenta exponencialmente con el número de variables en la parte premisa.
- Cuantas más reglas haya que construir, más difícil es saber si son adecuadas.
- Si el número de variables en la parte premisa es demasiado grande, será difícil comprender la relación causal entre las premisas y las consecuencias y por tanto, las reglas serán difíciles de construir.

En contraste, El método de Takagi-Sugeno-Kang no requiere de tantas operaciones en el proceso inicial. Sin embargo, su uso es imposible en la práctica cuando no podemos trabajar con conjuntos de datos discretos para lograr el diseño de las funciones lineales. La elección del método depende, en consecuencia, del acceso a los datos.

Bibliografía

Carlos González Morcillo, *Lógica Difusa, Una introducción práctica.*

Manual de usuario LG, lavadora, T1604DPL.

Hang Zettervall, *Fuzzy set theory applied to make medical prognoses for cancer patients.* Department of Mathematics and Natural Sciences. Blekinge Institute of Technology.

J. Galindo Gómez, *Apuntes de Lógica Difusa*, Departamento de Lenguajes y Ciencias de la Computación, Universidad de Málaga.

Oscar Cordón, *Diseño de Sistemas Difusos para Modelado y Clasificación. Aplicaciones*, Dpto. Ciencias de la Computación e Inteligencia Artificial, Universidad de Granada.

Araceli Grande Meza, *Observadores Difusos y Control Adaptable Difuso Basado en Observadores*, Centro de Investigación y de Estudios Avanzados, Instituto Politécnico Nacional.

Pedro Villar, Ana María Sánchez, Rosana Montes, Francisco Herrera, *Diseño de un Sistema de Clasificación Basado en Reglas Difusas para one-class mediante el aprendizaje genético de la base de datos.* Dpto. Lenguajes y Sistemas Informáticos. Universidad de Granada.

D. Guzmán, V. M. Castaño, *La lógica difusa en ingeniería: Principios, aplicaciones y futuro* Centro de Física Aplicada y Tecnología Avanzada. Universidad Nacional Autónoma de México, Campus Juriquilla, 76000, Querétaro, México. 2006. ISSN: 0378-0524.

L. A. ZADEH, *Fuzzy Sets*, Department of Electrical Engineering and Electronics Research Laboratory, University of California, Berkeley, California in 1965.