TALLER NO. 3

HERMAN COLLAZOS CASTAÑEDA

TALLER NO. 3

TRABAJO DE INTELIGENCIA ARTIFICIAL

CARLOS ALBERTO LONDOÑO LOAIZA

INGENIERO DE SISTEMAS

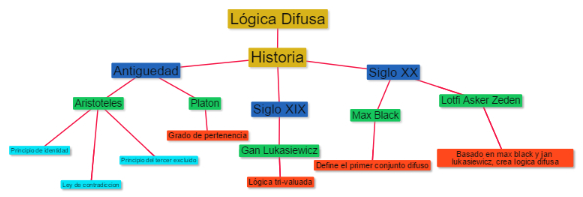
CORPORACIÒN DE ESTUDIOS TECNOLOGICOS DEL NORTE DEL VALLE

CIENCIAS INFORMÁTICAS, TECNOLÓGICAS E INGENIERÍA

[TECNOLOGIA EN SISTEMAS DE INFORMACION](http://www.cnotas.cotecnova.edu.co/servicios/servicios_est2.php)

CARTAGO VALLE

1 Realizar un mapa conceptual que permita conocer los sucesos más importantes hasta la fecha de la historia de la lógica difusa.



https://www.text2mindmap.com/sRtrW2T

2 Nombre 5 aplicaciones de la lógica difusa, que te parezcan importantes, da una breve descripción.

**Controlador Difuso Adaptativo**

Quizás la aplicación en que la Lógica Difusa ha conseguido un éxito mayor, y por ende un mayor número de seguidores, se encuentra en el Control Industrial. Aun cuando existen numerosas versiones de control

emplean lógica difusa, suele asignarse el término *Controlador Difuso* a un sistema de control cuya estructura interna corresponde a la de la figura 12.

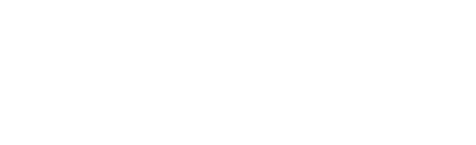


Figura 1

Estructura interna de un Controlador Difuso

Una de las ventajas que suelen mencionarse de los Controladores Difusos, frente a otro tipo de controladores, es que pueden diseñarse, aunque no se tenga un modelo matemático exacto de la Planta a controlar, gracias a que están basados en reglas. Esta situación, sin embargo, no es del todo sencilla: el no tener un modelo matemático de la Planta implica no poder realizar simulaciones sobre la misma, de tal manera que los ajustes del controlador deben realizarse en línea.

Este hecho ha promovido la aparición de controladores autoajustables, que cuentan con algún algoritmo que les permite evaluar su desempeño, y de acuerdo con cierto criterio ajustar su diseño. Estos algoritmos de autoajuste (también conocidos como algoritmos de auto sintonía) son muy útiles también en situaciones en las que la Planta a controlar varía en el tiempo, lo que implica que el controlador deba ajustarse a dichos cambios.

Cuando un Controlador Difuso cuenta con un algoritmo de autoajuste, se dice que es un Controlador Difuso Adaptativo. Sin embargo, existen Controladores con otros algoritmos que no son exactamente de autoajuste, y que podríamos llamar de entrenamiento, que algunos autores denominan también Controladores Difusos Adaptativos.

Los algoritmos de entrenamiento permiten ajustar el diseño del Controlador para que tenga un comportamiento deseado, pero fuera de línea. En general estos algoritmos no permiten la adaptación del controlador a cambios de la planta, pero son muy útiles para diseñar controladores en los que se combina información numérica con información lingüística

A continuación, se presenta un Controlador Difuso Adaptativo desarrollado por Wang & Mendel [20] que cuenta con un algoritmo de entrenamiento. Se ha seleccionado este controlador, porque el algoritmo empleado es uno de los más sencillos conceptualmente, y de mayor simplicidad algorítmica, tal como se presenta en el numeral siguiente.

***Algoritmo de Entrenamiento de Wang y Mendel***

Este algoritmo parte de una tabla que describe cuáles deben ser las salidas concretas, cuando se especifican las entradas concretas, es decir, de tablas como la siguiente:

Tabla 1

Parejas de Entrenamiento Entrada-Salida

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Entrada 1 | Entrada 2 | ... | Entrada p | Salida 1 | Salida 2 | ... | Salida q |
| x11 | x21 | ... | xp1 | y11 | y21 | ... | yq1 |
| x12 | x22 | ... | xp2 | y12 | y22 | ... | yq2 |
| ... | ... |  | ... | ... | ... |  | ... |
| x1w | x2w | ... | xpw | y1w | y2w | ... | yqw |

La tabla anterior puede interpretarse como sigue:

*“Se desea que el Sistema de Lógica Difusa responda con las salidas concretas y11, y21, ... yq1, cuando las entradas concretas son x11, x21, ... xp1; que responda con las salidas concretas y12, y22, ... yq2, cuando las entradas concretas son x12, x22, ... xp2; que responda con las salidas concretas y1w y2w, ... yqw, cuando las entradas concretas son x1w, x2w, ... xpw.”*

Los *w* casos que se quieren obtener con el Sistema de Lógica Difusa pueden ser insuficientes para determinar completamente el diseño, o por el contrario, pueden ser inconsistentes entre sí.

El algoritmo de Wang & Mendel permite determinar la base de reglas a partir de la tabla de patrones de entrenamiento; el diseñador debe seleccionar los demás parámetros de Sistema de Lógica Difusa. El procedimiento es el siguiente para cada uno de los *w* casos:

1. Determinar los grados de pertenencia de *x1k, x2k, ... xpk;y1k, y2k, ... yqk;* a cada uno de los Valores Lingüísticos de las respectivas Variables Lingüísticas.

2. Seleccionar los Valores Lingüísticos *Lx1k,Lx2k,... ,Lxpk, Ly1k,Ly2k,... ,Lyqk*, para los cuales los grados de pertenencia respectivos son máximos

3. Crear una regla de la forma IF Entrada 1 es *Lx1k* AND Entrada 2 es *Lx2k* AND ... AND Entrada p es *Lx1k* THEN Salida 1 es *Ly1k* AND Salida 2 es *Ly2k* AND ... AND Salida q es *Lyqk* .

4. Asignar a la regla anterior un factor de certeza, calculado como el producto de los grados de pertenencia a cada Valor Lingüístico

5. Verificar si en la Base de Reglas existe ya una regla con el mismo antecedente (y quizás distinto consecuente); de ser así, dejar en la Base aquella que tenga un mayor factor de certeza. Si aún no hay en la Base de Reglas una regla con el mismo antecedente, adicionar la nueva regla a la Base.

6. Complementar la Base de Reglas con la información lingüística disponible

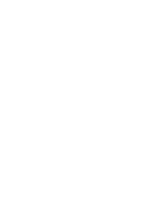
(si existe alguna).

***Ejemplo 1***

Presentamos aquí el ejemplo de un controlador aplicado al problema de llevar un vehículo con marcha hacia atrás a velocidad constante para que busque una cierta línea recta y la siga. El problema ha sido desarrollado por Wang en [16].

La figura 2 muestra gráficamente el problema. El vehículo está inicialmente ubicado a una distancia *x* de la línea, y formando un ángulo *A* con la recta normal a la línea. Se necesita desear un Controlador Difuso para que decida cuál es el ángulo de giro *B* que deben tener las ruedas del vehículo.

Figura 2



Planteamiento del problema

Wang encuentra una Tabla de Parejas Entrada-Salida (como la Tabla 1) a partir de su propia experiencia como conductor, y con ella entrena un Controlador Difuso que tiene 5 valores lingüísticos para la distancia *x*, 7 para el ángulo *A*, y 7 para el ángulo de salida *B*. Los resultados los compara con los obtenidos por Nguyen & Widrow con redes neuronales a partir de la misma Tabla Entrada-Salida, y las trayectorias seguidas por el vehículo son semejantes.

Wang también efectúa el entrenamiento con información numérica incompleta, es decir, considerando sólo una parte de la Tabla de entrenamiento; en estas condiciones el Controlador no es capaz de cumplir su objetivo. Sin embargo, al emplear algunas reglas If-Then extraídas de su experiencia como conductor, Wang obtiene resultados que son iguales a los del primer caso.

Lo interesante de este último diseño es que allí se han combinado dos tipos de información de naturaleza diferente: por una parte, está la información numérica proveniente de la Tabla, y por otra parte está la información lingüística contenida en las reglas If-Then.

**Identificador de Imágenes Aéreas**

El problema aqui planteado es el siguiente: Se tienen tres imágenes aéreas de la misma área; las imágenes se han captado empleando cámaras a blanco y negro, de tal forma que muestran en cada pixel un cierto nivel de gris (nivel de luminancia); las tres fotografías no son iguales, porque cada una de ellas se ha tomado anteponiendo a la cámara un filtro que sólo permite captar una franja de colores (unas ciertas longitudes de onda), y los filtros para cada fotografía han sido diferentes.

En estas imágenes se ha captado un área extensa con zonas que se pueden clasificar así:

- Zonas de río.

- Zonas de construcciones humanas.

- Zonas dedicadas a la agricultura.

- Zonas boscosas.

El problema consiste en diseñar un algoritmo que, conociendo los niveles de luminancia para un cierto pixel en las tres fotografías, decida a cuál de las zonas anteriores corresponde ese pixel.

No existe una única combinación de luminancias que identifique a cada una de las zonas (no hay un único "verde" en la zona de bosques), y por tanto no se conoce el conjunto de todas las combinaciones de luminancias posibles asociadas a cada zona.

Nótese que el problema puede replantearse en términos de Conjuntos Difusos: Se puede definir como Universo de Discurso el conjunto de todas las combinaciones posibles de luminancias, con lo cual la tarea consiste en encontrar cuatro Conjuntos Difusos definidos sobre dicho universo, cada uno de los cuales debe representar a una de las zonas de las imágenes.

Cómo encontrar esos conjuntos Trivedi [15] propone la utilización de un algoritmo de agrupamiento difuso, cuyos principios generales se exponen a continuación.

Agrupamiento "Fuzzy c-means"

Dado un Universo de Discurso **X**, se define una *c-partición* como una colección de *c* Conjuntos Difusos definidos sobre **X**, y con funciones de pertenencia

*u1(x), u2(x), ..., uc(x)*

siempre y cuando los Conjuntos Difusos cumplan con las siguientes

características:

- Para todos los conjuntos debe existir al menos un elemento de **X** tal que su función de pertenencia al conjunto sea distinta de cero.

- Para todos los elementos de **X**, la suma de sus funciones de pertenencia a los *c* Conjuntos debe ser 1.

Bezdek[3] desarrolló un algoritmo para obtener una c-partición para un Universo de Discurso finito; dicho algoritmo se conoce como el *agrupamiento "fuzzy c-means"*, y es uno de los más populares, pese a que en la literatura se reportan muchos otros algoritmos distintos[2].

Bezdek parte de la existencia de un Universo de Discurso **X** compuesto por

*n* elementos:

**X** = [ **x1 x2 ... xn**]

donde cada uno de los *n* **xi** es una *d-upla* de elementos

**x1** = [ x11 x21 ... xd1]T

**x2** = [ x12 x22 ... xd2]T

...

**xn** = [ x1n x2n ... xdn]

Para encontrar los *c* Conjuntos de la partición, Bezdek define para cada uno de ellos un *centro* **vi**, que identifiqué al conjunto, y define a **V** como la colección de centros de dichos Conjuntos:

**V** = [ **v1 v2 ... vc**]

**v1** = [ v11 v21 ... vd1]T

**v2** = [ v12 v22 ... vd2]T

...

**vc** = [ v1c v2c ... vdc]T

También define la matriz **U** como la matriz de orden *c\*n* que contiene las funciones de pertenencia de cada uno de los *n* **xi** casos a los *c* Conjuntos difusos identificados por los **vi centros**.

*u*11

*u*21

*u*12

*u*22

. . .

. . .

*u*1*n*

*u*2*n*

*U* . . . .

*uc*1

*uc* 2

. . .

*uc n*

Como los *c* conjuntos son una c-partición de **X**, debe cumplirse que:

*n*

*ui k k* 1

*c*

*ui k*

*i* 1

1; *i*, *i*

0; *k* , *k*

1,2. . . *c*

1,2. . . *n*

Con las definiciones anteriores, el problema del agrupamiento de **X** en *c* grupos puede formularse como la búsqueda de la pareja (**U**,**V**) que clasifique "mejor" los datos de **X**. Para medir qué tan bien está hecha una clasificación, Bezdek emplea la función *Jm(****U****,****V****)*

*n c*

*m* 2

*Jm* (*U* , *V* ) (*ui k* ) *xk vi*

*k* 1 *i* 1

en donde *m* es un real mayor que uno, y ||.|| es una norma diferenciable. Bezdek demuestra que la pareja (**U\***,**V\***) que optimiza *Jm* debe satisfacer:

*c xk*

*v*

2 1

\* *m* 1

*i*

\*

*u*

*i k* \*

; *i*, *k*

{1}

*j* 1 *xk v j*

*n*

*m*

\*

*i k xk*

*u*

\*  *k* 1

*v*

;

*i*

*i n*

{2}

\* *m*

*i k k* 1

*u*

Con lo anterior puede plantearse un algoritmo iterativo para la búsqueda de la pareja (**U\***,**V\***) así:

1) - Fijar *c*, un entero mayor que 1 y menor o igual que *n*.

- Fijar *m* un real mayor que 1.

- Seleccionar ||.|| una métrica inducida por producto interno para medir las distancias entre los miembro de **X** y los miembros de **V**.

- Inicializar **U**0.

- p=1

2) Calcular **V**p empleando {1}

3) Calcular **U**p empleando {2}

4) Comparar **U**p con **U**p-1 mediante alguna norma matricial. Si ||**V**p-**V**p-1|| es menor que algún valor de tolerancia entonces terminar, de lo contrario p=p+1 e ir al paso 2.

El algoritmo anterior es el conocido como *agrupamiento fuzzy c-means*. Para ilustrar qué es lo que hace este algoritmo, se han desarrollado los ejemplos

2, 3, 4 y 5:

***Ejemplo 2***

Se desea agrupar los números enteros del 1 al 20 en 3 conjuntos difusos empleando agrupamiento fuzzy c-means.

En este ejemplo **X**=[1 2 3 ....19 20], con lo que

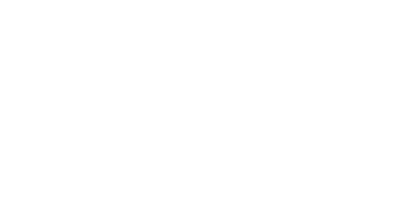
*d*=1

**x1**=[1]; **x2**=[2]; **x3**=[3]; ...; **x19**=[19]; **x20**=[20];

*c*=3

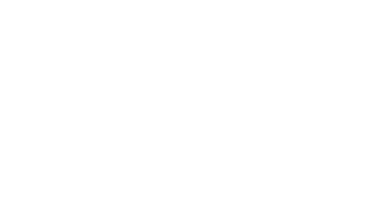
Las figuras 33 a 8 muestran cómo evolucionan los tres conjuntos difusos en cada una de las iteraciones. La partición inicial es aleatoria, y se ha escogido *m*=1.5.

Figura 3



3-partición para p=1

Figura 4



3-partición para p=2

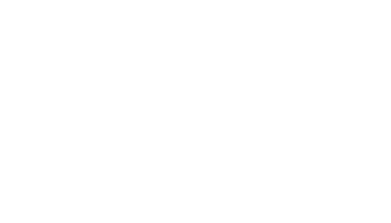
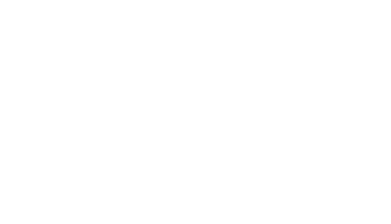


Figura 5

3-partición para p=3

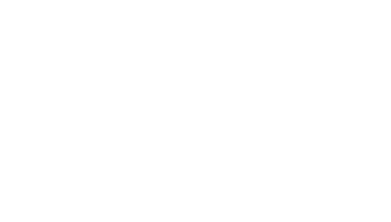


Figura 6

3-partición para p=4

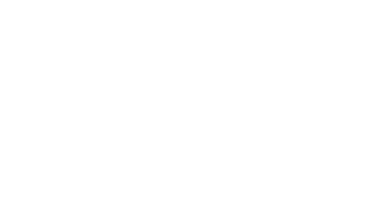


Figura 7

3-partición para p=5

Figura 8

3-partición para p=27

***Ejemplo 3***

Se desea conocer el efecto de cambiar el parámetro *m* en la partición del ejemplo 2.

3 En las figuras 3 a 18 el eje horizontal corresponde al intervalo real [0,20], y el eje vertical a las funciones de pertenecía de los conjuntos definidos por el algoritmo *fuzzy c-mean*

Las figuras 9 a 14 muestran los resultados finales de las particiones, cuando se selecciona *m1*=1.2, *m2*=1.5, *m3*=2, *m4*=3, *m5*=5, *m6*=10. Nótese que aun cuando la forma de los conjuntos difusos varía fuertemente, el centro de éstos no.

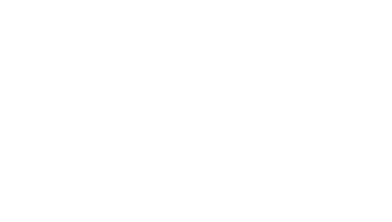
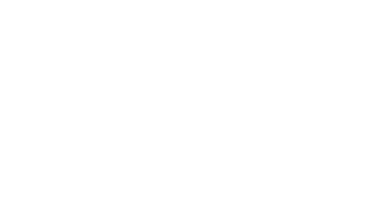


Figura 9

3-partición para m=1.2

Figura 10



3-partición para m=1.5

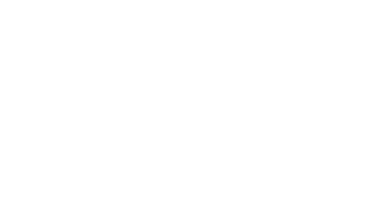
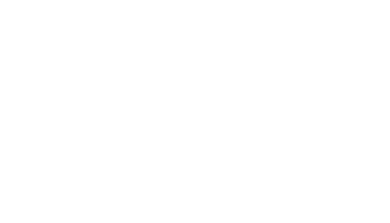


Figura 11

3-partición para m=2

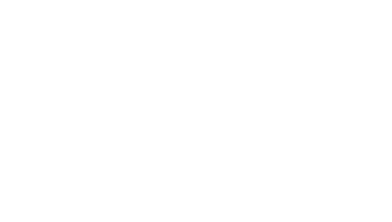


Figura 12

3-partición para m=3

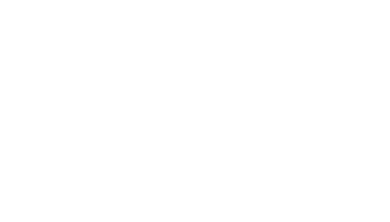


Figura 13

3-partición para m=5

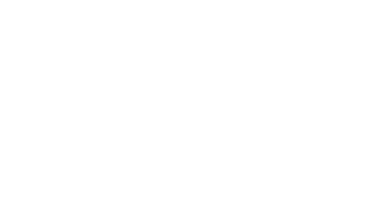
Figura 14

3-partición para m=10

***Ejemplo 4***

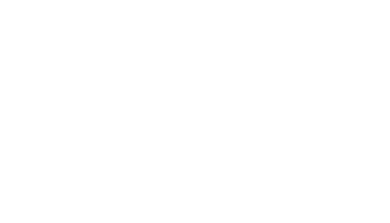
Se desea conocer el efecto de cambiar el número de conjuntos difusos de agrupamiento, en el ejemplo 2.

Las figuras 15 a 18 muestran los resultados finales de las particiones, para



*c1*=2, *c2*=3, *c3*=5, *c4*=10. En todos los casos se ha seleccionado *m*=2

Figura 15



2-partición

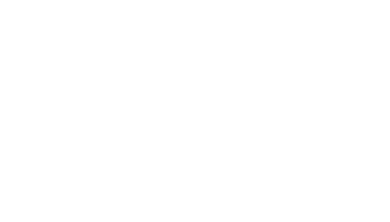


Figura 16

3-partición

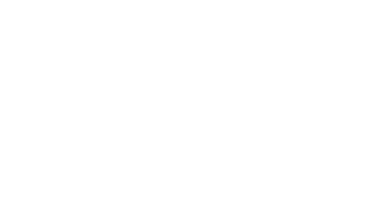


Figura 17

5-partición

Figura 18

10-partición

***Ejemplo 5***

Dado el problema del Identificador de Imágenes Aéreas anteriormente presentado, se desea saber cómo debe plantearse el problema para emplear el agrupamiento fuzzy c-means, si cada una de las tres imágenes tiene

32\*32=1024 pixels.

Los tres grupos de 1024 pixels pueden organizarse en 1024 3-uplas así:

**X** = [**x1 x2 x3 ... x1023 x1024** ]

**x1** =[x1,1 x2,1 x3,1 ]

**x2** =[x1,2 x2,2 x3,3 ]

...

**x1024** =[x1,1024 x2,1024 x3,1024 ]

xi,k= luminancia del pixel *k* en la imagen *i*

Una vez determinado **X** se selecciona *c=*4 como el número de conjuntos que se desean obtener, debido a que se desean obtener cuatro zonas distintas en las imágenes, y se emplea el algoritmo fuzzy c-means.

Trivedi [15] emplea los centros de los conjuntos obtenidos con este procedimiento para caracterizar cada una de las cuatro zonas. Posteriormente, para decidir a qué zona pertenece cada pixel, observa a cuál de los cuatro conjuntos difusos hallados pertenece en mayor medida, con lo que se completa el proceso de identificación de la imagen.

**Base de Datos Difusa**

La Lógica Difusa busca desarrollar un conjunto de procedimientos para manejar la información precisa y/o vaga. Ahora bien, los Sistemas de Bases de Datos tienen por propósito, hablando en términos muy generales, la organización de la información; por lo tanto, no es de sorprender que se haya intentado incorporar las técnicas de Lógica Difusa en el diseño de Bases de Datos.

Miyamoto & Umano [12] distinguen dos tipos de técnicas difusas en las

Bases de Datos:

- Bases de Datos Difusas.

- Técnicas Difusas para la recuperación de la información.

En la primera de estas técnicas el concepto de Conjunto Difuso se incorpora en la estructura misma de la Base de Datos, mientras que en la segunda se emplea en las estrategias de recuperación de la información.

A continuación, se presenta una de las técnicas del primer tipo, reseñadas en [12]; como en los anteriores ejemplos de este artículo, se ha seleccionado buscando un ejemplo conceptualmente sencillo. La técnica en cuestión se denomina *Búsqueda Difusa*, y fue propuesta inicialmente pot Tahani [14].

En esta técnica la Base de Datos tiene dos componentes4: por una parte, se tiene una Base de Datos tradicional, como la representada en la Tabla 2, y por otra parte se tiene una definición Difusa de las variables cuantificables de dicha Tabla, como la que se muestra en la figura 19.

Tabla 2

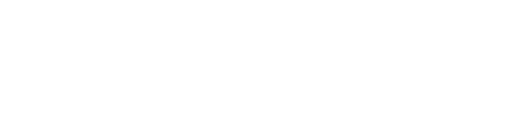
Base de Datos tradicional del Ejemplo

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre** | **Edad** | **Salario** | **Año de Ingreso** |
| Anderson | 30 | 20.000 | 1995 |

El ejemplo presentado es extraído de [12], con algunas adiciones

|  |  |  |  |
| --- | --- | --- | --- |
| Brown | 30 | 15.000 | 1995 |
| Long | 25 | 40.000 | 1993 |
| Nelson | 55 | 20.000 | 1980 |
| Smith | 25 | 23.000 | 1996 |

Figura 19



Variables lingüísticas para el ejemplo

Ante una consulta a la Base de Datos de la forma:

*"Cuáles son los nombres de las personas jóvenes o recientemente empleadas, pero con sueldo alto"*

Tahani propone evaluar la función de pertenecía de cada registro a cada uno de los valores lingüísticos involucrados en la consulta, y entregar como resultado de la búsqueda un conjunto difuso con funciones de pertenencia obtenidas mediante la utilización de operadores AND, OR y NOT difusos [6]; los operadores empleados por Tahani son el mínimo para el AND, el máximo para el OR y el complemento para el NOT.

Con esta metodología la consulta anterior podría representarse por la operación

*ubúsqueda* ( *x*, *y*, *z*)

*u joven* (*x*) *OR ureciente* ( *y*)

*AND ualto* (*z*)

En donde *x* es la edad, *y* el año de ingreso, *z* el salario, *ujoven(x)* es la función de pertenecía al conjunto *Joven* definido sobre la variable edad, *ureciente(y)* es la función de pertenecía al conjunto *Reciente* definido sobre la variable Año de Ingreso, *ualto(z)* es la función de pertenecía al conjunto *Alto* definido sobre la variable salario, y *ubúsqueda(x,y,z)* es la función de

pertenecía al conjunto resultante de la búsqueda.

Empleando la información de la Tabla 2 y de la figura 19 se puede construir la Tabla 3 que muestra los resultados de la búsqueda para cada registro.

Tabla 3

Resultados de la búsqueda del ejemplo en cada registro

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nombre** | ***ujoven(x)*** | ***ureciente(y)*** | ***ualto(z)*** | ***ubúsqueda(x,y,z)*** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Anderson | 0.5 | 0.6 | 0.5 | 0.5 |
| Brown | 0.5 | 0.6 | 0.0 | 0.0 |
| Long | 1.0 | 0.2 | 1.0 | 1.0 |
| Nelson | 0.0 | 0.0 | 0.5 | 0.0 |
| Smith | 1.0 | 0.8 | 0.8 | 0.8 |

El resultado final del ejemplo sería el conjunto:

*Búsqueda={0.5/Anderson, 1.0/Long, 0.8/Smith}*

**Psicología Cognoscitiva: Reconocimiento de Palabras**

El ejemplo que se presenta a continuación difiere de los anteriores sensiblemente. Este caso no emplea los algoritmos asociados a la lógica difusa, sino el concepto mismo de los Conjuntos Difusos, y ha sido seleccionado para resaltar que la importancia de la Lógica Difusa radica en la noción de Conjuntos con fronteras no exactas, lo que implica gradualidad en los cambios.

Este ejemplo consiste en la definición de un experimento cuyo autor es Rueckl [13] para el reconocimiento de palabras, que es uno de los temas abordados por la Psicología Cognocitiva.

La pregunta que se desea contestar es:

¿Qué efectos tiene el contexto de una frase en el reconocimiento de palabras?

A esta pregunta existen dos respuestas opuestas, sustentadas cada una por dos teorías diferentes [13]:

- La teoría del modelo Interactivo sostiene que el contexto si influye en el reconocimiento de palabras.

- La teoría del modelo Autónomo sostiene que el contexto no influye.

Ambos modelos se apoyan en experimentos cuyos resultados son consistentes y robustos, con explicaciones consistentes con las respectivas teorías. Los experimentos son de la siguiente forma:

- Se selecciona una palabra objetivo, por ejemplo *desk*5 .

5 En la explicación de este ejemplo se han mantenido las palabras originales en inglés, sin su traducción, para no desvirtuar la naturaleza del experimento

Se plantean frases con la palabra objetivo, unas en las que el contexto es congruente, y otras en las que el contexto es incongruente, por ejemplo:

Contexto congruente: *Mary's book were pilled up on her*

Contexto incongruente: *Last night Mary read a good*

Se mide que tan fácilmente reconoce un individuo (un grupo de individuos)

la palabra objetivo en los dos tipos de frases, para sacar conclusiones.

Hasta este punto las dos teorías coinciden, pero cuando las frases de contexto incongruente se remplazan por frases de contexto neutral, los experimentos dan resultados distintos, cada uno reforzando una teoría, dependiendo de lo que se entienda por "contexto neutral". Se han utilizado frases como las siguientes:

*They said it was the*

*The the the It was the*

Secuencias aleatorias de palabras.

La propuesta de Rueckl consiste en responder a la pregunta *Qué es un contexto neutral* diciendo que hay *congruencias difusas*, es decir, diciendo que entre los contextos congruentes e incongruentes no hay un único tipo de contexto neutral, sino que la congruencia puede manejarse gradualmente.

El experimento de Rueckl utilizó dos palabras objetivo: *pair pain* . Estas palabras se insertaron en las siguientes frases:

*The cardplayer had a in his hand*

*The shoemaker had a in his hand The piano player had a in his hand The arthritic had a in his hand*

Claramente, el contexto varía para ambas palabras objetivo. Adicionalmente, en el experimento se manipulo la forma de la letra *r - n* que diferencia las dos palabras objetivo, en la forma que muestra la figura 20. Ante este experimento, las dos teorías predicen resultados distintos. Los resultados obtenidos concuerdan más con la teoría Interactiva.



Figura 20

Variaciones de la letra *r - n*

http://www.profesaulosuna.com/data/files/ELECTRONICA/LOGICA%20DIFUSA/TextoAplicaciones.pdf

3 ¿Qué es la lógica booleana, para que sirve y cuales son opciones?

La lógica booleana es una lógica de conjuntos y nos sirve, principalmente, para definir formas de intersección entre conjuntos.  
En este caso, los conjuntos serian lo que quedan definidos por una palabra, es decir, serian conjuntos definidos por intensión. Si uso la palabra "psicoanálisis", esta recubre todo el conjunto de elementos, para el caso, páginas web, en las que dicha palabra se encuentre incluida. Así, a partir de diferentes palabras se definen conjuntos de páginas agrupadas por el hecho de incluir (o no) esa determinada palabra. Estos conjuntos tendrán, entre sí, elementos en común, y elementos que no. Una manera de precisar o afinar nuestra búsqueda consistirá en utilizar estos operadores booleanos para precisar el campo de nuestro interés

Las principales opciones son:

OR - se suman los conjuntos definidos por dos palabras, es decir, la respuesta será todas aquellas referencias donde aparezcan, indistintamente, UNA U OTRA de las palabras indicadas para búsqueda.

AND - se trata de la intersección de los conjuntos definidos por las dos palabras, es decir, solo aquellas referencias que contengan AMBAS palabras a la vez

NOT - en este caso, aquellas referencias que tengan la primera palabra y no la segunda, es decir, un primer conjunto, amputado de su parte común con otro.

NEAR - como el AND pero con la exigencia suplementaria de una cercanía entre las palabras

Es de suponer que las utilidades OR y AND son bastante obvias. Si hay dudas pueden escribirnos para preguntarnos.  
Les daremos, en cambio, algunos ejemplos sobre el uso de las otras opciones, que podrían no ser tan obvias.

La principal utilidad que puede tener la opción NOT es la de eliminar todas las referencias de algún tipo de dominio: por ejemplo, si pensamos que nuestra búsqueda supone páginas puramente académicas y que muy difícilmente pueda encontrarse en algún sitio web comercial, al poner "NOT .com" nos ahorraremos todas las referencias que hayan sido inicialmente seleccionadas por contener palabras con la misma raíz que aquellas que estamos usando para realizar una búsqueda, pero que provengan de dominios comerciales, y que por eso mismo, suponemos que no tienen que ver con el tema buscado.   
Este comando también puede servir para descartar confusiones que pudieran surgir entre el tema de nuestra búsqueda y otros temas conexos. Por ejemplo, si nos interesa el tema drogadicción, pero no en relación al sida, como sabemos que en todos los lugares referidos al sida es probable que haya referencias a la drogadicción, nos ahorraremos muchas referencias que no buscamos si ponemos "NOT aids", o "NOT hiv", o "NOT sida".

La utilidad de NEAR, que por otra parte está implementada en muy pocos lugares, nos permite buscar en forma más precisa definiciones compuestas. Por ejemplo, no nos va a dar lo mismo si buscamos por "neurosis" y "obsesiva" con AND que con NEAR. En el primer caso tendremos todas las referencias donde se hable de neurosis y de obsesión; pero no serán, forzosamente, referencias a la neurosis obsesiva. Es más probable que obtengamos mejores resultados usando el NEAR.

<http://www.psiconet.com/enlaces/internet/boole.htm>

4 Nombrar y dar un ejemplo de cada una de las operaciones entre conjuntos convencionales.

**Operaciones con** **conjuntos.**

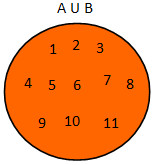
Las operaciones con conjuntos también conocidas como álgebra de conjuntos, nos permiten realizar operaciones sobre los conjuntos para obtener otro conjunto. De las operaciones con conjuntos veremos las siguientes unión, intersección, diferencia, diferencia simétrica y complemento.

**Unión o reunión de conjuntos.**

Es la operación que nos permite unir dos o más conjuntos para formar otro conjunto que contendrá a todos los elementos que queremos unir, pero sin que se repitan. Es decir, dado un conjunto A y un conjunto B, la unión de los conjuntos A y B estará formado por todos los elementos de A y con todos los elementos de B sin repetir ningún elemento. El símbolo que se usa para indicar la operación de unión es el siguiente: ∪.

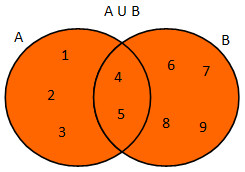
***Ejemplo 1.***

Dados dos conjuntos A={1,2,3,4,5,6,7,} y B={8,9,10,11} la unión de estos conjuntos será A∪B={1,2,3,4,5,6,7,8,9,10,11}. Usando diagramas de Venn se tendría lo siguiente:



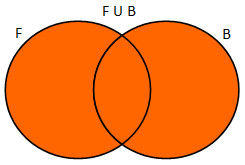
***Ejemplo 2.***

Dados dos conjuntos A={1,2,3,4,5} y B={4,5,6,7,8,9} la unión de estos conjuntos será A∪B={1,2,3,4,5,6,7,8,9}. Usando diagramas de Venn se tendría lo siguiente:



***Ejemplo 3.***

Dados dos conjuntos F={x/x estudiantes que juegan fútbol} y B={x/x estudiantes que juegan básquet}, la unión será A∪B={x/x estudiantes que juegan fútbol o básquet}. Usando diagramas de Venn se tendría lo siguiente:

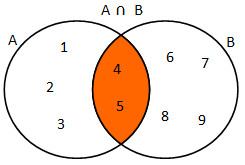


**Intersección de conjuntos.**

Es la operación que nos permite formar un conjunto, sólo con los elementos comunes involucrados en la operación. Es decir, dados dos conjuntos A y B, la de intersección de los conjuntos A y B, estará formado por los elementos de A y los elementos de B que sean comunes, los elementos no comunes A y B, será excluidos. El símbolo que se usa para indicar la operación de intersección es el siguiente: ∩

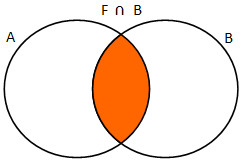
***Ejemplo 1.***

Dados dos conjuntos A={1,2,3,4,5} y B={4,5,6,7,8,9} la intersección de estos conjuntos será A∩B={4,5}. Usando diagramas de Venn se tendría lo siguiente:



***Ejemplo 2.***

Dados dos conjuntos A={x/x estudiantes que juegan fútbol} y B={x/x estudiantes que juegan básquet}, la intersección será A∩B={x/x estudiantes que juegan fútbol y básquet}. Usando diagramas de Venn se tendría lo siguiente:

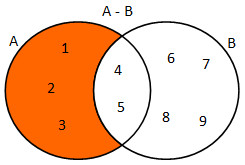


**Diferencia de conjuntos.**

Es la operación que nos permite formar un conjunto, en donde de dos conjuntos el conjunto resultante es el que tendrá todos los elementos que pertenecen al primero, pero no al segundo. Es decir, dados dos conjuntos A y B, la diferencia de los conjuntos entra A y B, estará formado por todos los elementos de A que no pertenezcan a B. El símbolo que se usa para esta operación es el mismo que se usa para la resta o sustracción, que es el siguiente: -

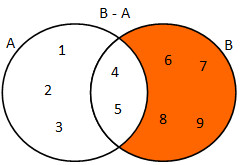
***Ejemplo 1.***

Dados dos conjuntos A={1,2,3,4,5} y B={4,5,6,7,8,9} la diferencia de estos conjuntos será A-B={1,2,3}. Usando diagramas de Venn se tendría lo siguiente:



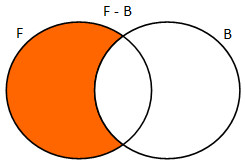
***Ejemplo 2.***

Dados dos conjuntos A={1,2,3,4,5} y B={4,5,6,7,8,9} la diferencia de estos conjuntos será B-A={6,7,8,9}. Usando diagramas de Venn se tendría lo siguiente:



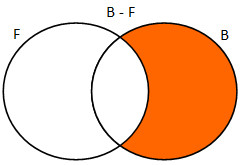
***Ejemplo 3.***

Dados dos conjuntos F={x/x estudiantes que juegan fútbol} y B={x/x estudiantes que juegan básquet}, la diferencia de F con B, será F-B={x/x estudiantes que sólo juegan fútbol}. Usando diagramas de Venn se tendría lo siguiente:



***Ejemplo 4.***

Dados dos conjuntos F={x/x estudiantes que juegan fútbol} y B={x/x estudiantes que juegan básquet}, la diferencia de B con F, será B-F={x/x estudiantes que sólo juegan básquet}. Usando diagramas de Venn se tendría lo siguiente:

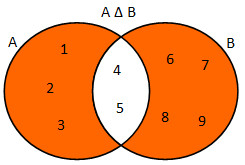


**Diferencia simétrica.**

Es la operación que nos permite formar un conjunto, en donde de dos conjuntos el conjunto resultante es el que tendrá todos los elementos que no sean comunes a ambos conjuntos. Es decir, dados dos conjuntos A y B, la diferencia simétrica estará formado por todos los elementos no comunes a los conjuntos A y B. El símbolo que se usa para indicar la operación de diferencia simétrica es el siguiente: ∆

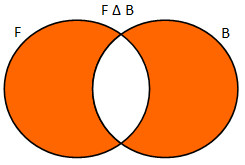
***Ejemplo 1.***

Dados dos conjuntos A={1,2,3,4,5} y B={4,5,6,7,8,9} la diferencia simétrica de estos conjuntos será A∆B={1,2,3,6,7,8,9}. Usando diagramas de Venn se tendría lo siguiente:



***Ejemplo 2.***

Dados dos conjuntos F={x/x estudiantes que juegan fútbol} y B={x/x estudiantes que juegan básquet}, la diferencia simétrica será F∆B={x/x estudiantes que sólo juegan fútbol y básquet}. Usando diagramas de Venn se tendría lo siguiente:

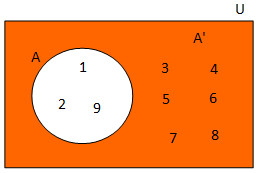


**Complemento de un conjunto.**

Es la operación que nos permite formar un conjunto con todos los elementos del conjunto de referencia o universal, que no están en el conjunto. Es decir, dado un conjunto A que está incluido en el conjunto universal U, entonces el conjunto complemento de A es el conjunto formado por todos los elementos del conjunto universal, pero sin considerar a los elementos que pertenezcan al conjunto A. En esta operación el complemento de un conjunto se denota con un apostrofe sobre el conjunto que se opera, algo como esto A' en donde el el conjunto A es el conjunto del cual se hace la operación de complemento.

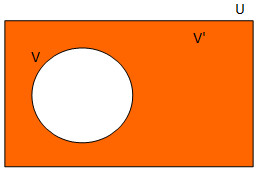
***Ejemplo 1.***

Dado el conjunto Universal U={1,2,3,4,5,6,7,8,9} y el conjunto A={3,4,5,6,7,8}, el conjunto A' estará formado por los siguientes elementos A'={1,2,9}. Usando diagramas de Venn se tendría lo siguiente:



***Ejemplo 2.***

Dado el conjunto Universal U={x/x estudiantes de un colegio} y el conjunto V={x/x estudiantes que juegan voley}, el conjunto V' estará formado por los siguientes elementos V'={x/x estudiantes que no juegan voley}. Usando diagramas de Venn se tendría lo siguiente:



<http://www.conoce3000.com/html/espaniol/Libros/Matematica01/Cap10-03-OperacionesConjuntos.php>

5 ¿Qué son las leyes de Morgan, de un ejemplo de cada una?

**Leyes de Morgan**

Las Proposiciones Una proposición es una afirmación que puede recibir un valor de verdad falso (F), o bien verdadero (V), pero no ambos a la vez.   
  
Su denotación generalmente la encontramos con las letras (p, q, r)   
Conectores Lógicos Podemos formar nuevas proposiciones a partir proposiciones dadas mediante el uso de conectivos lógicos. Algunos de ellos son:   
^ “y” conjunción v“o” disyunción ->“si—,entonces” implicación <-> “si y sólo si” dobleimplicación ¬“no”negación LeyesdeMorgan SonunapartedelaLógicaproposicional,analítica,yfueroncreadaporAugustusdeMorgan. Estas declaran las reglas de equivalencia en las que se muestran que dos proposiciones pueden ser lógicamente equivalentes.   
  
Las Leyes de Morgan permiten: El cambio del operador de conjunción en operador de disyunción y viceversa. Las proposiciones conjuntivas o disyuntivas a las que se aplican las leyes de Morgan pueden estar afirmadas o negadas (en todo o en sus partes).   
  
Casos: ¬(P^Q)≡(¬Pv¬Q)   
Si nos encontramos con una proposición conjuntiva totalmente negada, la ley de Morgan nos permite transformarla en una proposición disyuntiva con cada uno de su miembros negados ¬(PvQ)≡(¬P^¬Q) Si nos encontramos con una proposición disyuntiva totalmente negada, la ley de Morgan nos permite transformarla en una proposición conjuntiva con cada uno de sus miembros negados   
  
(P^Q)≡¬(¬Pv¬Q)   
Si nos encontramos con una proposición conjuntiva afirmada, la ley de Morgan nos permite transformarla en una proposición disyuntiva negada en su totalidad y en sus miembros.   
  
(PvQ)≡¬(¬P^¬Q)   
Si nos encontramos con una proposición disyuntiva afirmada, la ley de Morgan nos permite transformarla en una proposición conjuntiva negada en su totalidad y en sus miembros

<http://logica-icoubb.blogspot.com/2010/07/leyes-de-morgan.html>

6 ¿Cuáles son las formas de representación de un conjunto difuso, ¿cuáles son sus ecuaciones?

Definición: Un conjunto difuso A se define como una Función de Pertenencia que enlaza o empareja los elementos de un dominio o Universo de discurso X con elementos del intervalo [0,1]:

– A: X ® [0,1]

Cuanto más cerca esté A(x) del valor 1, mayor será la pertenencia del objeto x al conjunto A. – Los valores de pertenencia varían entre 0 (no pertenece en absoluto) y 1 (pertenencia total).

Representación: Un conjunto difuso A puede representarse como un conjunto de pares de valores: Cada elemento xÎX con su grado de pertenencia a A. También puede ponerse como una “suma” de pares: – A = { A(x)/x, xÎX} – (Los pares en los que A(xi )=0, no se incluyen)

Ejemplo: Conj. de alturas del concepto difuso “Alto” en Personas: – A = 0.25/1.75 + 0.5/1.8 + 0.75/1.85 + 1/1.9 (su universo es discreto)

Si el Universo es Continuo: A A x x x = ò ( ) /

La suma y la integral no deben considerarse como operaciones algebraicas

<http://www.lcc.uma.es/~ppgg/FSS/FSS1.pdf>

7 ¿Qué es la lógica simbólica, que son proposiciones y que son tablas de verdad?, dar un ejemplo.

|  |
| --- |
| La lógica se define como la ciencia del razonamiento, o como el estudio de los métodos y principios usados para distinguir el razonamiento correcto del incorrecto. Por su parte, la lógica simbólica es el estudio de la lógica mediante la matemática, es decir, que incorpora la exactitud y rigor matemáticos.    Un razonamiento es cualquier grupo de oraciones declarativas, tal que una de ellas (conclusión) se afirma que se deriva de otras, llamadas premisas, las cuales se consideran evidencia de la verdad de la primera. Para efectos del curso, estudiaremos dos tipos de razonamiento:  Inductivo: comúnmente, por analogía; afirma probabilidad o cierta evidencia de la verdad de la conclusión.  Deductivo: sus premisas ofrecen una evidencia contundente de la verdad de la conclusión. Su correctitud viene dada por la validez o invalidez del razonamiento. |
| El objetivo de la presente asignatura es introducir al estudiante en los métodos de demostración de validez de razonamientos propios de la lógica simbólica. Para ello, estudiaremos los dos tipos de razonamientos descritos anteriormente:  Razonamiento Inductivo: Inducción Completa y Definiciones Inductivas.  Razonamiento Deductivo:  El Sistema Ss: Lógica de proposiciones o Lógica proposicional.  El Sistema Sp: Lógica de predicados. |

<http://logica-simbolica.globered.com/>

La lógica proposicional o lógica de orden cero es un [sistema formal](https://es.wikipedia.org/wiki/Sistema_formal) cuyos elementos más simples representan proposiciones, y cuyas [constantes lógicas](https://es.wikipedia.org/wiki/Constante_l%C3%B3gica), llamadas conectivas o [conectores lógicos](https://es.wikipedia.org/wiki/Conectores_l%C3%B3gicos), representan [operaciones](https://es.wikipedia.org/wiki/Operaci%C3%B3n_matem%C3%A1tica) sobre proposiciones, capaces de formar otras proposiciones de mayor complejidad.[1](https://es.wikipedia.org/wiki/L%C3%B3gica_proposicional#cite_note-1)

La lógica proposicional trata con [sistemas lógicos](https://es.wikipedia.org/wiki/Sistema_l%C3%B3gico) que carecen de cuantificadores, o variables interpretables como entidades. En lógica proposicional si bien no hay signos para variables de tipo entidad, sí existen signos para [variables proposicionales](https://es.wikipedia.org/wiki/Variable_proposicional) (es decir, que pueden ser interpretadas como proposiciones con un valor de verdad de definido), de ahí el nombre proposicional. La lógica proposicional incluye además de variables interpretables como [proposiciones](https://es.wikipedia.org/wiki/Proposici%C3%B3n) simples signos para [conectivas lógicas](https://es.wikipedia.org/wiki/Conectiva_l%C3%B3gica), por lo que dentro de este tipo de lógica puede analizarse la [inferencia lógica](https://es.wikipedia.org/wiki/Inferencia) de proposiciones a partir de proposiciones, pero sin tener en cuenta la estructura interna de las proposiciones más simples.

<https://es.wikipedia.org/wiki/L%C3%B3gica_proposicional>

Una tabla de verdad, o tabla de valores de verdad, es una tabla que muestra el [valor de verdad](https://es.wikipedia.org/wiki/Valor_de_verdad) de una [proposición](https://es.wikipedia.org/wiki/Proposici%C3%B3n) compuesta, para cada combinación de verdad que se pueda asignar.

Fue desarrollada por [Charles Sanders Peirce](https://es.wikipedia.org/wiki/Charles_Sanders_Peirce) por los años 1880, pero el formato más popular es el que introdujo [Ludwig Wittgenstein](https://es.wikipedia.org/wiki/Ludwig_Wittgenstein) en su *[Tractatus logico-philosophicus](https://es.wikipedia.org/wiki/Tractatus_logico-philosophicus" \o "Tractatus logico-philosophicus)*, publicado en 1921.

<https://es.wikipedia.org/wiki/Tabla_de_verdad>

8 ¿Qué es una tautología, de un ejemplo?

En la lógica proposicional una tautología es aquella fórmula bien formada que resulta siempre verdadera, independientemente del valor de verdad que sus proposiciones atómicas tengan Esto significa que independientemente de los valores de verdad que tengan las premisas, la conclusión siempre va a ser verdadera. La tautología es utilizada en los procesos de deducción de la lógica sentencial. Ejemplos de tautología: Sean las siguientes proposiciones: a: Voy al cine b: Voy a cenar c: Me quedo en casa Entonces la sentencia: (a^b) -> (a v ¬c) Es una tautología, la tabla de verdad de esta construcción es

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| a | b | c | ¬c | a^b | av¬c | (a ^ b) -> (a v ¬c) |
| V | V | V | F | V | V | V |
| V | V | F | V | V | V | V |
| V | F | V | F | F | V | V |
| V | F | F | V | F | V | V |
| F | V | V | F | F | F | V |
| F | V | F | V | F | V | V |
| F | F | V | F | F | F | V |
| F | F | F | V | F | V | V |

La expresión sustituyendo los valores de las sentencias atómicas queda: Si voy al cine y voy a cenar, entonces voy al cine o no me quedo en casa.

La proposición compuesta p v ¬p es una tautología.

Sea p la proposición atómica “El auto es rojo” entonces ¬p queda “El auto no es rojo”

La tabla de verdad para esta proposición compuesta es la siguiente:

|  |  |  |
| --- | --- | --- |
| p | ¬p | p v ¬p |
| V | F | V |
| F | V | V |

La proposición compuesta es entonces:

“El auto es rojo o el auto no es rojo” que es una tautología por ser siempre verdadera.

La proposición p -> (p v q) es una tautología ya que esta proposición únicamente puede ser falsa cuando p es verdadera y (p v q) sean falsas, pero si p es verdadera entonces p v q es verdadera independientemente del valor de verdad que tenga q.

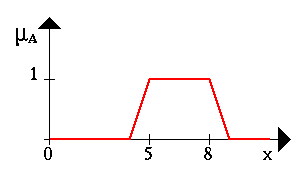
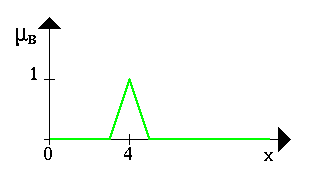
<http://www.ejemplosde.com/29-logica/1573-ejemplo_de_tautologia.html>

9 ¿Cuáles son las operaciones que se pueden realizar en la lógica difusa empleando conjuntos difusos?

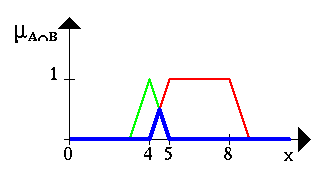
Operaciones con conjuntos difusos

Ahora que tenemos una idea de lo que son conjuntos difusos, podemos introducir las operaciones básicas sobre conjuntos difusos. Parecido a las operaciones sobre conjuntos booleanos nosotros también podemos [interseccionar](http://redeya.bytemaniacos.com/electronica/cursos/fuzzy/fuzzy.htm" \l "AG), [unificar](http://redeya.bytemaniacos.com/electronica/cursos/fuzzy/fuzzy.htm#AG) y [negar](http://redeya.bytemaniacos.com/electronica/cursos/fuzzy/fuzzy.htm#AG) conjuntos difusos. En su primerísimo artículo sobre conjuntos difusos, L. A. Zadeh sugirió el operador mínimo para la intersección y el operador máximo para la unión de dos conjuntos difusos. Es fácil ver que estos operadores coinciden con la unificación booleana, e intersección si nosotros únicamente consideramos los grados miembros 0 y 1.

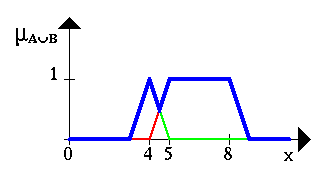
A fin de aclarar esto, mostraremos varios ejemplos. Sea A un intervalo difuso entre 5 y 8, y B un número difuso en torno a 4. Las figuras correspondientes se muestran a continuación:

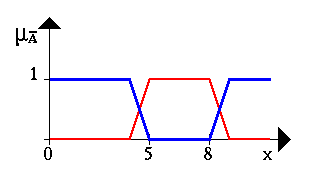
La figura siguiente muestra la operación AND (Y) del conjunto difuso A y el número difuso B (el resultado es la línea azul).



La operación OR (O) del conjunto difuso A con el número difuso B se muestra en la próxima figura (nuevamente, es la línea azul).



Esta figura da un ejemplo para una negación. La línea azul es la NEGACION del conjunto difuso A.



<http://redeya.bytemaniacos.com/electronica/cursos/fuzzy/fuzzy.htm>

10 Mostrar a través de un ejemplo la representación gráfica de un sistema difuso.

|  |
| --- |
| Herramienta de representación gráfica - Xfplot |

|  |
| --- |
| La herramienta *xfplot* ilustra el comportamiento de un sistema difuso mediante una representación gráfica bidimensional o tridimensional. La herramienta puede ser ejecutada desde la línea de comandos con la expresión "*xfplot file.xfl*", o desde la ventana principal del entorno usando la opción "*Graphical representation*" del menú *Verification*.  La ventana principal de la herramienta consta de un panel principal, en el que se muestra la representación gráfica, y de una barra superior, dedicada a la configuración.  http://www2.imse-cnm.csic.es/Xfuzzy/Xfuzzy_3.3/tools/gif/xfplot_0.gif  El menú *File* de la barra superior permite salvar los datos representados en un fichero externo (opción "*Save Data*"), actualizar la representación (opción "*Actualize*") y salir de la herramienta (opción "*Close*"). El menú *Configuration* permite seleccionar el tipo de representación (opción "*Plot Mode*"), el modelo de colores (opción "*Color Model*") y el valor de las variables de entrada no representadas (opción "*Input Values*"), así como cargar la configuración de un fichero externo (opción "*Load Configuration*") o salvarla (opción "*Save Configuration*"). Tres listas desplegables en la barra superior permiten seleccionar las variables asignadas a cada eje. El último campo contiene el número de puntos usados en la partición de los ejes X e Y. La elección de este parámetro es importante porque determina la resolución de la representación. Un valor bajo del parámetro puede hacer que se excluyan detalles importantes del comportamiento del sistema. Por otra parte, un valor alto hará que la superficie representada sea difícil de entender al usar un grid excesivamente denso. El valor por defecto de este parámetro es 40.  http://www2.imse-cnm.csic.es/Xfuzzy/Xfuzzy_3.3/tools/gif/xfplot_1.gifLa representación gráfica tridimensional incluye la posibilidad de rotar la superficie usando los dos botones deslizantes situados en la parte derecha e inferior de la gráfica. Esta capacidad de rotación facilita la interpretación de la superficie representada.  Al seleccionar el tipo de representación bidimensional, el panel central se modifica mostrando una gráfica plana que representa la variación de la variable de salida seleccionada como eje Z con respecto a la variable de entrada seleccionada como eje X.  http://www2.imse-cnm.csic.es/Xfuzzy/Xfuzzy_3.3/tools/gif/xfplot_2.gif  Cuando el sistema a representar contiene más variables de entrada que las requeridas por el tipo de representación seleccionado, es necesario introducir los valores asignados a las variables de entrada no representadas. Para ello se recurre a la opción "*Input Values*" que abre una ventana de introducción de valores.  http://www2.imse-cnm.csic.es/Xfuzzy/Xfuzzy_3.3/tools/gif/xfplot_3.gif |

|  |
| --- |
| http://www2.imse-cnm.csic.es/Xfuzzy/Xfuzzy_3.3/tools/gif/line.gif |
|  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | [http://www2.imse-cnm.csic.es/Xfuzzy/Xfuzzy_3.3/tools/gif/B_xfuzzy.gif](http://www2.imse-cnm.csic.es/Xfuzzy/Xfuzzy_3.3/index_sp.html) | [http://www2.imse-cnm.csic.es/Xfuzzy/Xfuzzy_3.3/tools/gif/B_index.gif](http://www2.imse-cnm.csic.es/Xfuzzy/Xfuzzy_3.3/tools/verification_sp.html) | [http://www2.imse-cnm.csic.es/Xfuzzy/Xfuzzy_3.3/tools/gif/B_previous.gif](http://www2.imse-cnm.csic.es/Xfuzzy/Xfuzzy_3.3/tools/verification_sp.html) | [http://www2.imse-cnm.csic.es/Xfuzzy/Xfuzzy_3.3/tools/gif/B_up.gif](http://www2.imse-cnm.csic.es/Xfuzzy/Xfuzzy_3.3/tools/xfplot_sp.html) | [http://www2.imse-cnm.csic.es/Xfuzzy/Xfuzzy_3.3/tools/gif/B_next.gif](http://www2.imse-cnm.csic.es/Xfuzzy/Xfuzzy_3.3/tools/xfmt_sp.html) |

<http://www2.imse-cnm.csic.es/Xfuzzy/Xfuzzy_3.3/tools/xfplot_sp.html>

11 ¿Cuáles son las propiedades de los conjuntos difusos?

**Propiedades fundamentales de las operaciones estándar en conjuntos difusos**

|  |  |
| --- | --- |
| *Conmutatividad* | AuB = BuA  AnB = BnA |
| *Asociatividad* | (AuB)uC = Au(BuC)  (AoB)nC = An(BnC) |
| *Distributividad* | Au(BnC) = (AuB)n(AuC)  An(BuC) = (AnB)u(AoC) |

**Propiedades fundamentales de las operaciones estándar en conjuntos difusos**

**(continuación)**

|  |  |
| --- | --- |
| *Idempotencia* | AuA =A  AnA=A |
| *Identidad* | AoA = A  A^X-A |
| *Absorción* | AuX-X  An0=0  Au(AoB) = A  An(AuB) = A |
| *Leyes de De Morgan* | AuB=AnB  AnB=AuB |

De la comparación entre esta tabla y la Tabla 1 se observa que existen dos leyes

fundamentales que no se conservan: la ley de contradicción y la ley del tercero excluido

*(tertium non datur),* que son duales entre sí como ponen de manifiesto las leyes de De

Morgan. Es decir, que para un conjunto difuso A definido en X se tiene que, en general,

AnA^0 y AujA^X. Esto es una consecuencia natural del hecho de abandonar el

concepto de pertenencia o no pertenencia estricta de un elemento a un conjunto, pero

las implicaciones filosóficas de este hecho son numerosas y profundas, pues contradicen

los cimientos en los que se basa la matemática tradicional y, por extensión, todo el

edificio científico construido sobre ellos.

<http://www.tdx.cat/bitstream/handle/10803/6914/03_masCasals_capitol_2.pdf?sequence=3>

12 Definir e implementar las siguientes funciones:

Función de membresía

Función de saturación

Función hombro

Función triangular

Función trapecio o pi

Función s o sigmoidal

13 ¿Qué son números difusos?

Un número difuso es una extensión de un número regular en el sentido que no se refiere a un único valor sino a un conjunto de posibles valores, que varían con un *peso* entre 0 y 1, llamado [función miembro](https://es.wikipedia.org/w/index.php?title=Funci%C3%B3n_miembro&action=edit&redlink=1). Un número difuso es así un caso especial de [conjunto difuso](https://es.wikipedia.org/wiki/Conjunto_difuso) convexo. Así como la [lógica difusa](https://es.wikipedia.org/wiki/L%C3%B3gica_difusa) es una extensión de la [lógica booleana](https://es.wikipedia.org/wiki/L%C3%B3gica_booleana) (que sólo utiliza valores 0 y 1, exclusivamente), los números difusos son una extensión de los [números reales](https://es.wikipedia.org/wiki/N%C3%BAmero_real). Los cálculos con números difusos permiten la incorporación de [incertidumbre](https://es.wikipedia.org/wiki/Azar) en parámetros, propiedades, geometría, condiciones iniciales, etc.

<https://es.wikipedia.org/wiki/N%C3%BAmero_difuso>

14 ¿Qué son relaciones nítidas y difusas?

Relaciones nítidas. Relación entre n conjuntos Con el producto Cartesiano definido en (2.46), puede ahora expresarse una relación entre los conjuntos AI, AZ, ...An, como un subconjunto del producto cartesiano Aix Ajx ... xAn. Esta relación recibe el nombre de relación n-aría y se denota generalmente por R(Ai, A2, ... An) o, si no hay ambigüedades, simplemente por R. Llamamos relación binaria al caso particular de una relación entre dos conjuntos, el primero de los cuales recibe el nombre de conjunto salida, y el segundo conjunto llegada. A continuación, se exponen los métodos más usuales para representar las relaciones nítidas n-arias. Representación de relaciones nítidas Dado que definimos las relaciones como conjuntos nítidos, podemos utilizar cualquiera de los métodos descritos en 2.3.1 (enumeración, regla, función característica) para representarlas, teniendo en cuenta que cada elemento de la relación es ahora una n-pla. En particular, usando la función característica se tiene que para una relación R(Ai, A2, ... · ,xn) — 1 si (x,,\*2,•••,\*„) e R .0 si (x,,x2,"-,x,,) í R (2.47) en donde, evidentemente, x/eAi, ..., xMeAn. En el caso particular de que los conjuntos A¡ sean finitos, existen otras formas convenientes para representar relaciones, como son los diagramas sagitales o las matrices de relación. En la primera de ellas se dibujan agrupados los elementos de los distintos conjuntos sobre los que está definida la relación, y se unen mediante líneas aquellos elementos pertenecientes a un mismo par ordenado. Así, por ejemplo, para los conjuntos P (países) y M (monedas) definidos como P={Alemania, Canadá, EE.UU., Francia} y M= (Dólar, Peseta, Marco}, el diagrama sagital que relaciona los países con monedas es el mostrado en la Figura 2.9: Alemania « SS\_-I< ^' • Canadá®--^^Q «Peseta EE.UU.® ^^ ^ « Marco Francia « Figura 2.9: Representación de relaciones mediante diagramas sagitales El método matricial consiste en crear una matriz de dimensiones |Ai|x|A2|x...x|An| -donde |·| representa la cardinalidad-, y etiquetar cada una de las filas/columnas con los nombres de los elementos de los correspondientes conjuntos. De esta forma cada elemento de la matriz corresponde a una n-pla del producto cartesiano AI x A2x ... x An. Asignando a cada uno de los elementos el valor 1 si la n-pla pertenece a la relación y el valor O en caso contrario, se obtiene la matriz de relación [R] asociada a la relación R. Por razones obvias de representación, el método matricial se aplica principalmente a relaciones binarias, donde la matriz resultante es bidimensional. Para el ejemplo anterior, la matriz de relación se muestra en la Figura 2.10: Alemania Canadá EE.UU. Francia Dólar 0 1 1 0 Peseta 0 0 0 0 Marco 1 0 0 0 Figura 2.10: Representación de relaciones mediante matrices de relación Dado que expresamos las relaciones como conjuntos, podemos definir sobre ellas uniones, intersecciones, complementos, etc. Las particularidades de estas operaciones ya se han tratado en la sección 2.4.1, por lo que es innecesario repetirlas para este caso.

Relaciones difusas. Relación difusa entre n conjuntos En 2.5.2 se definió una relación nítida como un subconjunto de un cierto producto cartesiano. Ello permite una rápida generalización para el caso difuso de la siguiente forma: Definimos una relación difusa R entre los conjuntos nítidos AI, A2, ... An, como un subconjunto difuso del producto cartesiano AI x A2X ... xAn, en donde las n-pla (xl ,x2,···,xn) que constituyen los elementos de la relación llevan asociado un grado de pertenencia |iR(x,,x2,···,xn) entre O y 1. Este grado de pertenencia suele interpretarse como la "intensidad" de la relación que liga entre sí los elementos de la n-pla. Es de destacar el hecho de que en la definición que acaba de darse lo difuso es la relación en sí, pero no los conjuntos involucrados en el producto cartesiano. Recuérdese que la expresión que se utiliza para el producto cartesiano (2.46) se ha definido en base a conjuntos nítidos. Es posible también definir un producto cartesiano difuso9 tomando como base conjunto difusos, pero ello no conlleva que las relaciones definidas sobre un tal producto sean automáticamente difusas. El producto cartesiano difuso se utiliza principalmente como base teórica para otros desarrollos como puede ser el principio de extensión, el cual se detalla en la sección 2.6.2. De la definición de relación difusa también se desprende que toda relación difusa n-aria puede considerarse como un conjunto difuso n-dimensional en AI x A2x ... x An, con función de pertenencia \iA.(xi,X2,...,xn). Este punto de vista permite intercambiar conjunto con relación en conceptos clave como la composición. Un caso interesante lo constituyen las relaciones binarias designadas con R(A, A) o R(A2 ) y caracterizadas por que los conjuntos salida y llegada coinciden. En estas relaciones se definen tres propiedades características: reflexividad, simetría y transitividad. La posesión o no de estas propiedades determina distintos tipos importantes de relación binaria (equivalencia, compatibilidad, orden, etc.) los cuales forman la base para construir interesantes estructuras matemáticas, como los retículos (lattices). Debido a su extensión y escaso impacto en la presente tesis, no se profundizará en el tratamiento de estas cuestiones.

<http://www.tdx.cat/bitstream/handle/10803/6914/03_masCasals_capitol_2.pdf?sequence=3>

15 ¿Que son reglas difusas, ¿cuáles existen?

Reglas difusas. Operadores de implicación. Interpretación. El conocimiento humano se expresa en términos de reglas difusas SI\_ENTONCES SI <proposición difusa> ENTONCES <proposición difusa> Tipos de proposiciones difusas: Atómicas: x es A, donde x es una variable lingüística y A es un valor lingüístico Compuestas: Composición de proposiciones difusas atómicas con las conectivas “y”, “o” o “no”, representando intersección, unión y complemento difuso, respectivamente 3. Reglas difusas. Operadores de implicación. Interpretación. 3. Reglas difusas. Operadores de implicación. Interpretación. Ejemplos: El error es Negativo-Grande La interpretación o significado de una proposición difusa atómica se define mediante la función de pertenencia del conjunto difuso Negativo-Grande El grado de pertenencia de un error concreto al conjunto difuso Negativo-Grande determinará el grado con que se verifica la proposición difusa 3. Reglas difusas. Operadores de implicación. Interpretación.

Reglas difusas. Operadores de implicación. Interpretación. Ejemplos de proposiciones difusas compuestas: X es A o X no es B X es A y X es B X no es A y X no es B (X es A y X no es B) o X es C X es A y Y es D En una proposición difusa compuesta pueden estar implicadas variables distintas Las proposiciones difusas compuestas se pueden considerar relaciones difusas 3. Reglas difusas. Operadores de implicación. Interpretación. 3. Reglas difusas. Operadores de implicación. Interpretación. ¿Cómo determinamos la interpretación de estas relaciones difusas? ¿Cómo determinamos la función de pertenencia? Para las conectivas “y” se deben utilizar intersecciones difusas X es A y Y es B µ A∩ B ( x, y ) = T [ µ A ( x), µ B ( y )] Para las conectivas “o” se deben utilizar uniones difusas X es A o Y es B µ A∪ B ( x, y ) = S[ µ A ( x), µ B ( y )] Para las conectivas “no” se deben utilizar complementos difusos 3. Reglas difusas. Operadores de implicación. Interpretación.

Reglas difusas. Operadores de implicación. Interpretación. Proposiciones difusas condicionales (Reglas Difusas) Ejemplo: SI el error es Negativo-Grande ENTONCES u es Negativo-pequeño El significado se representa mediante una relación difusa entre el error y la variable de salida U La función de pertenencia de esta relación difusa se determina mediante un operador de implicación difuso 3. Reglas difusas. Operadores de implicación. Interpretación. 3. Reglas difusas. Operadores de implicación. Interpretación. Algunos operadores de implicación difusos: Larsen µ R ( x, y ) = µ A ( x ) ⋅ µ B ( y ) Mamdani µ R ( x, y ) = min{µ A ( x), µ B ( y )} Dienes-Rescher µ R ( x, y ) = max{1 − µ A ( x), µ B ( y )} Lukasiewicz µ R ( x, y ) = min{1, 1 − µ A ( x) + µ B ( y )} Zadeh µ R ( x, y ) = max{min{µ A ( x), µ B ( y )}, 1 − µ A ( x)} ⎧ 1, si µ A ( x) ≤ µ B ( y ) Gödel µ R ( x, y ) = ⎨ ⎩µ B ( y ), e.o.c. 3.

<http://es.slideshare.net/mentelibre/lll-variables-lingsticas-variables-difusas-y-reglas-difusas>

BIBLIOGRAFÍA

https://www.text2mindmap.com/sRtrW2T

http://www.profesaulosuna.com/data/files/ELECTRONICA/LOGICA%20DIFUSA/TextoAplicaciones.pdf

<http://www.psiconet.com/enlaces/internet/boole.htm>

<http://www.conoce3000.com/html/espaniol/Libros/Matematica01/Cap10-03-OperacionesConjuntos.php>

<http://logica-icoubb.blogspot.com/2010/07/leyes-de-morgan.html>

<http://www.lcc.uma.es/~ppgg/FSS/FSS1.pdf>

<http://logica-simbolica.globered.com/>

<https://es.wikipedia.org/wiki/L%C3%B3gica_proposicional>

<https://es.wikipedia.org/wiki/Tabla_de_verdad>

<http://www.ejemplosde.com/29-logica/1573-ejemplo_de_tautologia.html>

<http://redeya.bytemaniacos.com/electronica/cursos/fuzzy/fuzzy.htm>

<http://www2.imse-cnm.csic.es/Xfuzzy/Xfuzzy_3.3/tools/xfplot_sp.html>

<http://www.tdx.cat/bitstream/handle/10803/6914/03_masCasals_capitol_2.pdf?sequence=3>

<https://es.wikipedia.org/wiki/N%C3%BAmero_difuso>

<http://www.tdx.cat/bitstream/handle/10803/6914/03_masCasals_capitol_2.pdf?sequence=3>

<http://es.slideshare.net/mentelibre/lll-variables-lingsticas-variables-difusas-y-reglas-difusas>