E.T.S. de Ingeniería Industrial,
Informática y de Telecomunicación

Nuevo sistema de inferencia utilizando funciones penalty y el método del gradiente para un sistema de reglas

Grado en Ingeniería Informática

Trabajo Fin de Grado

Alberto Maíllo Ruiz de Infante

Javier Fernández Fernández

Pamplona, 28/03/2017

Universidad Pública de Navarra
Nafarroako Unibertsitate Publikoa

Resumen

En este trabajo se va a explicar teóricamente y a través de una aplicación los siguientes apartados:

1. Nueva versión del Modus Ponens Generalizado (GMP) satisfaciendo los axiomas de Fukami

   Este nuevo algoritmo se caracteriza por utilizar índices de igualdad e de overlap.

2. Reducción de un sistema de reglas a una sola aplicando funciones penalty
   A partir de *n* reglas, construimos la relación R para cada una de ellas. Luego, la idea consiste en obtener una única relación aplicando funciones penalty a las *n* anteriores.

3. Reconstrucción de una regla con el algoritmo del gradiente
   Con regla obtenida en el punto anterior, mediante el algoritmo del gradiente calcular la nueva familia de conjuntos A y B.

4. Aplicación del punto 1 a la nueva regla y estudiar los axiomas.

Lista de palabras claves

I. Sistema de reglas
II. Funciones penalty
III. Algoritmo del gradiente
IV. Modus ponens generalizado
V. Índice de overlap
VI. Índice de igualdad

Índice

Introducción

Este trabajo consiste en que a partir de un sistema de reglas, reducirlas todas ellas a una utilizando las funciones penalty. Posteriormente, gracias al algoritmo del gradiente, construimos una única regla. Y por último, usamos esa regla y el nuevo método del Modus Ponens Generalizado para calcular los nuevos consecuentes de las nuevas reglas.

Es decir, dado un sistema de reglas:

R1. Si x es A_1 entonces Y es B_1 $\longrightarrow$ Relación 1 (matriz)
R2. Si x es A_2 entonces Y es B_2 $\longrightarrow$ Relación 2 (matriz)
R3. Si x es A_3 entonces Y es B_3 $\longrightarrow$ Relación 3 (matriz)

$\vdots$ $\vdots$

Rn. Si x es A_n entonces Y es B_n $\longrightarrow$ Relación n (matriz)

donde R1,R2, …, Rn son matrices construidas utilizando un operador de implicación.

Luego utilizamos funciones penalty para:

$$\underbrace{R1, R2, \ldots, Rn}_{\text{tomar una única } R_{final}}$$

Entonces a partir de la R_{final} y el algoritmo del gradiente construimos A* y B* de forma que tenemos:

$$\begin{array}{l} \text{Si } X = A^* \text{ entonces } Y = B^* \\ \quad X = A'^* \\ \hline \qquad\qquad\qquad Y = B'^* \end{array}$$

El cálculo de B'* se hace a partir del nuevo algoritmo del GMP utilizando índices de overlap e de igualdad.

Aplicación

La aplicación en la que se va a basar el proyecto, consiste en un sistema de reglas, cuya regla determina cual es el *rendimiento general*(Y) de un empleado a partir de conocer sus *cualidades individuales*(X_1), *habilidades personales*(X_2), *habilidades técnicas*(X_3) y *habilidades en gestión*(X_4).

Cada regla tiene la siguiente estructura:

- Si X_1=A_1 y X_2=A_2 y X_3=A_3 y X_4=A_4 entonces Y=B.

Tanto el referencial X e Y es el mismo:

X=Y=0, 0.1, 0.2,…, 9.8, 9.9, 10.

Como se ha nombrado anteriormente tenemos cuatro variables difusas como antecedentes, cada una tiene tres etiquetas lingüísticas: pobre, bueno y excelente.

1. Cualidad Individual: Contiene cualidades de una persona tales como adaptabilidad, motivación, innovación, iniciativa, responsabilidad, hábitos de trabajo, habilidades analíticas.

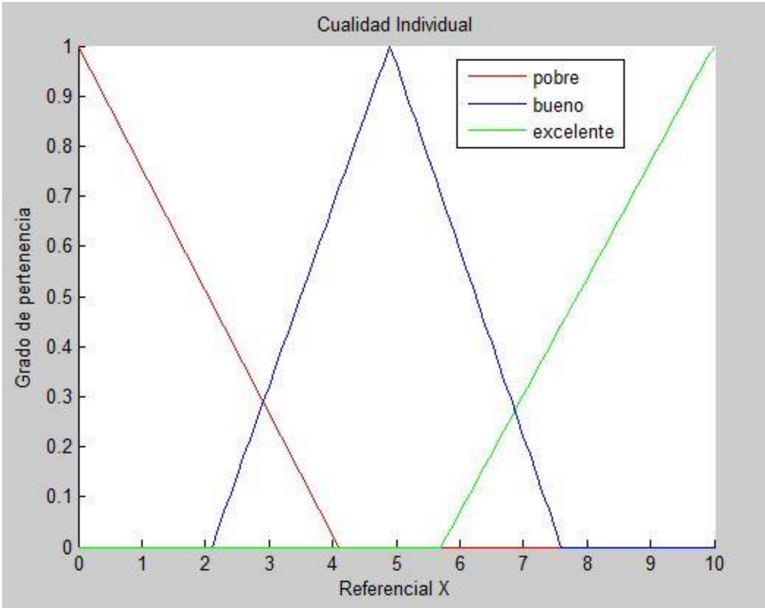


Fig.1.1. Primer antecedente *Cualidad Individual*

2. Habilidades personales: Incluye habilidades comunicativas, tolerancia al estrés.

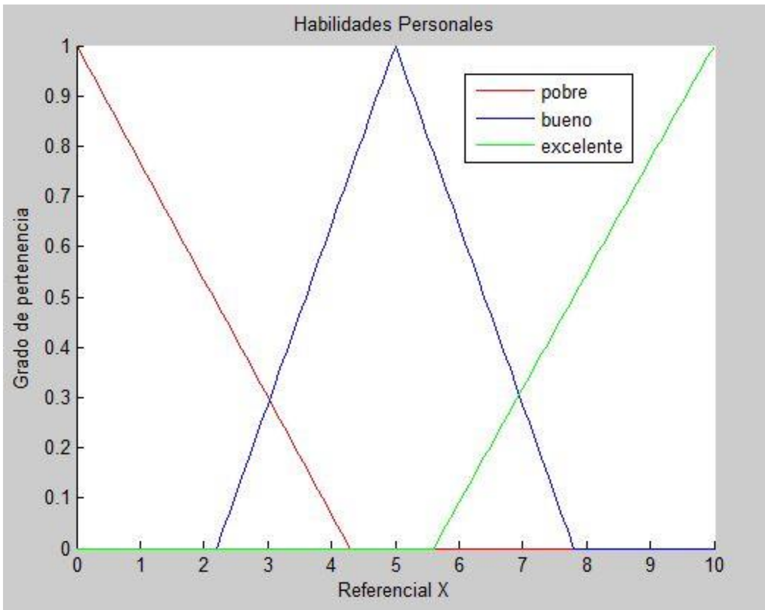


Fig.1.2. Segundo antecedente *Habilidades Personales*

3. Habilidades técnicas: Incluye el conocimiento del trabajo y las herramientas utilizadas para ello.

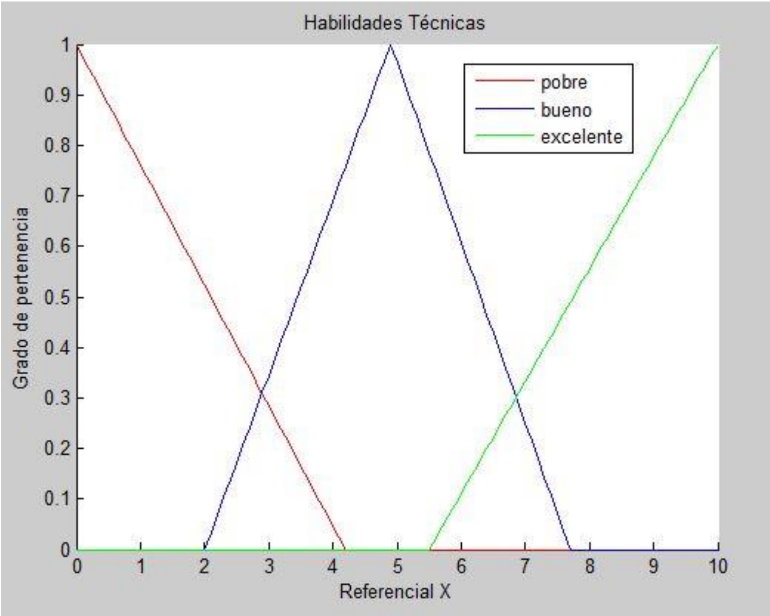


Fig.1.3. Tercer antecedente *Habilidades Técnicas*

4. Habilidades en gestión: Medida en que el empleado demuestra habilidades de gestión efectivas. Incluye planificación y organización, resolución de problemas y toma de decisiones, productividad, trabajo de calidad, ingenio y gestión del tiempo.

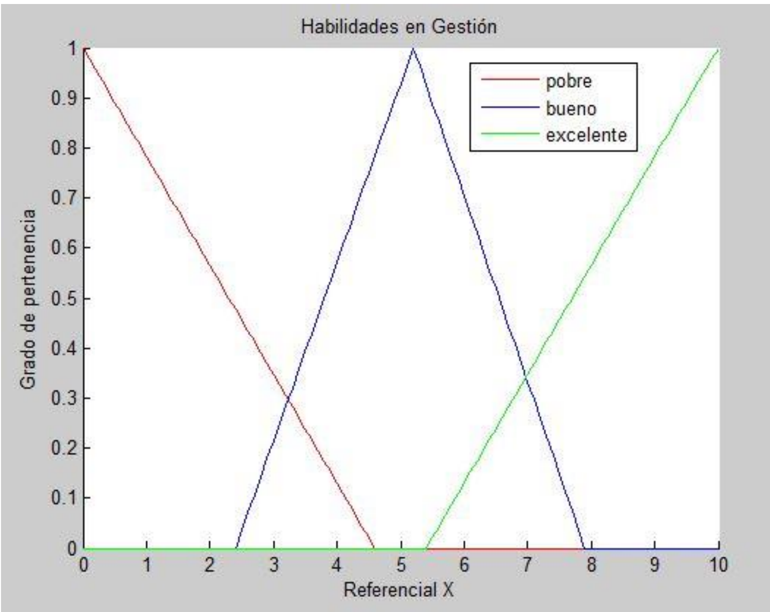


Fig.1.4. Cuarto antecedente *Habilidades en Gestión*

Luego el consecuente *rendimiento general* , que surge entre los medios empleados para obtener algo y el resultado obtenido, está formado por cuatro etiquetas lingüísticas: pobre, medio, bueno y excelente.

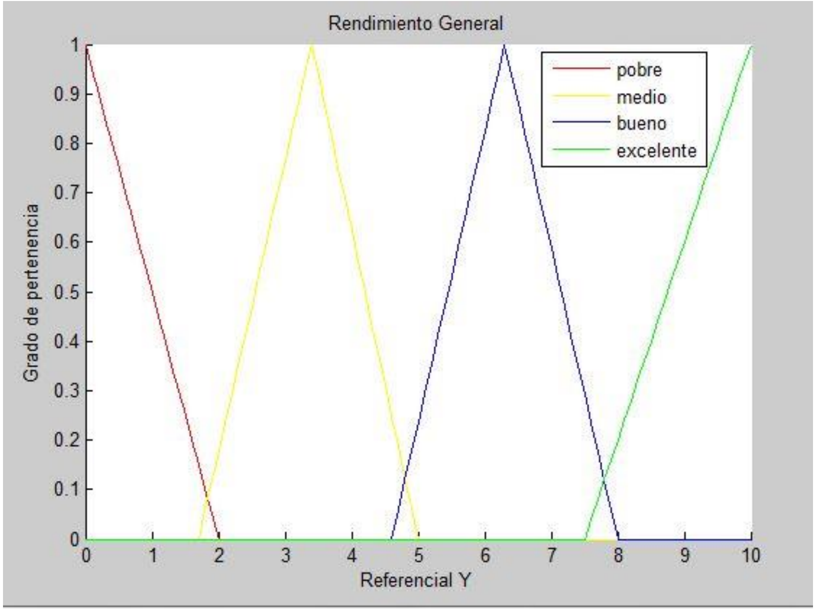


Fig.1.5. Consecuente *Rendimiento General*

Sistema de reglas

Vamos a utilizar el siguiente sistema de reglas:

R1. Si cualidad individual=pobre y habilidades personales=pobre y habilidades técnicas=pobre y habilidades de gestión = pobre entonces rendimiento general = pobre.

R2. Si cualidad individual=bueno y habilidades personales=pobre y habilidades del trabajo=bueno y habilidades en gestión=pobre entonces rendimiento general= medio.

R3. Si cualidad individual=bueno y habilidades personales=excelente y habilidades del trabajo=excelente y habilidades en gestión = bueno entonces rendimiento general = excelente.

R4. Si cualidad individual=excelente y habilidades personales=excelente y habilidades técnicas=pobre y habilidades en gestión = pobre entonces rendimiento general = pobre.

R5. Si cualidad individual=bueno y habilidades personales=bueno y habilidades técnicas=pobre y habilidades en gestión = bueno entonces rendimiento general = bueno.

R6. Si cualidad individual=excelente y habilidades personales=excelente y habilidades técnicas=excelente y habilidades en gestión = excelente entonces rendimiento general = excelente.

R7. Si cualidad individual=excelente y habilidades personales=excelente y habilidades técnicas=pobre y habilidades en gestión = bueno entonces rendimiento general = bueno.

R8. Si cualidad individual=pobre y habilidades personales=excelente y habilidades técnicas=excelente y habilidades en gestión = bueno entonces rendimiento general= medio.

Preliminares

Definición 1: un conjunto difuso *A* está definido en un universo U finito y no vacío:

$A=\{(u_{i,}, \mu_A (ui)) | ui \in U\}$

donde μ_A: U ->[0, 1] se le llama grado de pertenencia.

A todos los conjuntos difusos definidos a partir de un referencial finito (U) se denota FSs(U). Un conjunto difuso *A* se dice que es normal cuando existe al menos un $u_i \in U$ tal que $\mu_A(u_i)=1$. Dos conjuntos difusos *A* y *B* del mismos referencial U son completamente disjuntos si $\mu_A(u_i)\mu_B(u_i)=0$ para todo $u_i \in U$.

Dado un conjunto difuso *A*, la expresión $A_c=\{(u, c(\mu_A(u_i))) | u_i \in U\}$ se denomina el complementario de *A*, donde *c* es una negación.

Una función T: $[0, 1]^2$ -> [0, 1] se llama t-norma si es conmutativa, asociativa, creciente y con elemento neutral 1. Luego, una función S: $[0, 1]^2$ ->[0, 1] es una t-conorma si es conmutativa, asociativa, creciente y con elemento neutral 0.

Operador de implicación

Un operador de implicación es una función I: $[0,1]^2$ ->[0, 1] que satisface las siguientes propiedades:

I1- Si x<=z entonces I(x, y)>=I(z,y) para todo $y \in [0,1]$

I2- Si y<=t entonces I(x, y)<=I(x, t) para todo $x \in [0, 1]$

I3- I(0, x)=1 para todo $x \in [0,1]$

I4- I(x, 1)=1 para todo $x \in [0, 1]$

I5- I(1, 0)=0

Ejemplo 1: Implicación de Lukasiewitch I_L=min(1, 1-x+y).

Funciones de agregación

Definición 2 [1, 2]**:** M: $[a, b]^n$ ->[a, b] es una función de agregación si es monótona no decreciente en cada componente y satisface M(**a**)=M(a, a,…, a) y M(**b**)=(b, b, …, b)=b

Ahora vamos a demostrar que el mínimo, el máximo y la media geométrica son medidas de agregación elegidas. Para ello vamos a tomar el intervalo [0, 1].

Demostración 1: El mínimo es una función de agregación

- min(0,…,0)=0
- min(1, ..., 1) =1
- min(x, y)≤min(x, z) si y≤z

Demostración 2: El máximo es una función de agración;

- max(0,..,0)=0
- max(1,..,1)=1
- max(x, y)≤max(x, z) si y≤z

Demostración 3: La media geométrica es una función de agregación

- $MG(0, \dots, 0_n) = \sqrt[n]{0 \cdot \dots \cdot 0} = \sqrt[n]{0} = 0$
- $MG(1, \dots, 1_n) = \sqrt[n]{1 \cdot \dots \cdot 1} = \sqrt[n]{1} = 1$
- MG(x, y)≤MG(x, z) si y≤

Funciones e índices de overlap

Una función de overlap("solapamiento") es la representación del desconocimiento en determinar si pertenece a un conjunto o a otro.

Definición 3 [3]**:** Una función de overlap G_O: $[0,1]^2$ -> [0, 1] cumple que:

1. $G_O(x, y)=G_O(y, x)$ para todo x, y ϵ [0, 1]
2. $G_O(x, y)=0$ si y solo si xy=0
3. $G_O(x, y)=1$ si y solo si xy=1
4. G_O es creciente
5. G_O es continuo

Definición 4 [4]**:** El índice de consistencia entre *A* y *B*, siendo *A* y *B* ϵ FSs(U), se define como

$$O_Z(A,B) = {}_{\mathrm{i=1}}^{\mathrm{n}}\sup(\min(\mu_A(u_i), \mu_B(u_i)))$$

Una característica es que si los conjuntos son completamente difuso entonces $O_z(A,B) = 0$. Otra es que si para todo u_i ϵ U tenemos que $\mu_A(u_i)=\mu_B(u_i)=1$ entonces Oz(A, B)=1.

Ejemplo 2: el *sup* de la ecuación de arriba es el máximo

$$O_Z(A,B) = \vee_{i=1}^{n}(\mu_A(u_i) \wedge \mu_B(u_i))$$

Definición 5 [3]**:** Un índice de overlap O: FSs(U) x FSs(U) -> [0, 1] tal que

1. O(A,B) = 0 si y solo si A y B son disjuntos, esto es $\mu_A(u_i)\mu_B(u_i)=0$ para todo u_i ϵ U
2. O(A,B)=O(B, A)
3. Si B<=C entonces O(A,B)<=O(A,C)

Definición 6: A un índice de overlap, que satisfaga las propiedades dadas en la Definición 5, se le llama normal si cumple también:

Si existe un u_i ϵ U tal que $\mu_A(u_i)=\mu_B(u_i)=1$, entonces O(A,B)=1.

Teorema 1: Siendo M: $[0, 1]^n$ -> [0, 1] una función de agregación tal que $M(x_1, \ldots, x_n)=0$ si y solo si $x_1=\ldots=x_n=0$. Y siendo G_O una función de overlap bajo las características de Definición 3. Entonces, el índice de overlap resultante O: FSs(U)x FSs(U) -> [0, 1] es:

$$O(A,B) = M_{i=1}^{n}(G_O(\mu_A(u_i), \mu_B(u_i)))$$

con las mismas propiedades escritas en Definición 5.

Ejemplo 3: Si tomamos como medida de agregación la media aritmética y como función de overlap $G_O(x,y)=\sqrt{x \cdot y}$, entonces el índice de overlap resultante es:

$$O(A,B) = \frac{1}{n}\sum\sqrt{\mu_A(u_i)\mu_B(u_i)}$$

Índice de igualdad

Medida que determina cuanto se parecen dos conjuntos difusos.

A=B ⟺A⊆ B ⋀ B⊆ A

Se conoce la medida del subconjunto entre dos conjuntos difusos *A* y *B*, que indica cual es el grado de *A* que está contenido en *B*.

Definición 7: si σ_{DI}: FS(U)x FS(U)->[0,1] satisface:

a) σ_{DI}(A,B)=1 si y solo si A≤B, es decir, $\mu_A(u_i) \leq \mu_B(u_i)$ para todo i ϵ {1,.., n}

b) σ_{DI}(A, A_c)=0 si y solo si A={(u_i, $\mu_A(x_i)$=1) | u_i ϵ X}

c) Si A≤B entonces σ_{DI}(A,C)≥ σ(B,C) y σ_{DI}(C,A)≤ σ_{DI}(C,B)

entonces decimos que σ_{DI} es una medida difusa del subconjunto.

Proposición 1: Teniendo M una función de agregación que cumple las propiedades de la Definición 2 e I un operador de implicación, se puede obtener:

$$\sigma_{DI}(\mathrm{A},\mathrm{B}) = M_{i=1}^{n}(I(\mathrm{A}(u_i),\mathrm{B}(u_i)))$$

para todo A,B ϵ FS(U).

Una expresión como índice de igualdad bajo estas condiciones puede ser:

$$EQ_{DI}(A,B) = \wedge\{\sigma_{DI}(A,B), \sigma_{DI}(B,A)\}$$

Proposición 2: la ecuación de arriba cumple las siguientes características:

a) $EQ_{DI}(A,B) = 1$ si y solo si $A = B$
b) $EQ_{DI}(A,B) = EQ_{DI}(B,A)$
c) Si A ≤ B ≤ C, entonces $EQ_{DI}(C,B) \geq EQ_{DI}(C,A)$ y
d) $EQ_{DI}(A,B) \ \geq \ EQ_{DI}(A,C)$
e) $EQ_{DI}(A,A_C) = 1$ si y solo si $A = \{(u, \mu_A(u) = e) | \, u \in U\}$
f) $EQ_{DI}(A,A_C) = 0$ si y solo si $A = 1 \; o \; A = 0$
g) Si σ_{DI} satisface $\sigma_{DI}(\mathrm{A},\mathrm{B}) = \sigma_{DI}(\mathrm{B}_C, A_C)$ entonces $EQ_{DI}(A,B) = EQ_{DI}(A_C,B_C)$

<u>Ejemplo 4</u>: Utilizando la media aritmética y el operador de implicación de Lukasiewitch I_L=min(1, 1-x+y) según la Proposición 1:

$$\sigma_{DI}(\mathrm{A},\mathrm{B}) = \tfrac{1}{n}\textstyle\sum_{i=1}^{n} \wedge(1, \ 1 \ - \ A(x) + B(x))$$

Entonces:

$$EQ_{DI}(A,B) = \wedge\{\frac{1}{n}\sum_{i=1}^{n}\wedge(1,1 - A(x) + B(x))\,,\frac{1}{n}\sum_{i=1}^{n}\wedge(1,1 - B(x) + A(x))\,\}$$

Nueva versión del Modus Ponens Generalizado(GMP) satisfaciendo los axiomas de Fukami

Introducción

Modus ponens o modus tollens son herramientas útiles cuando el conocimiento está expresado en proposiciones que solo toman valores "true" y "false", esto se denomina lógica clásica binaria. En estos casos, la operación lógica de implicación es esencial para construir esquemas de razonamiento en la lógica clásica y para modelar la representación del conocimiento en forma de reglas. El Modus Ponens clásico (MP) tiene por lo general la siguiente estructura:

Regla: If X is A then Y is B

Hecho: X is A

Conclusion: Y is B

donde A y B son valores binarios (true o false)

El MP no contempla las situaciones en las que el conocimiento está determinado por un grado de incertidumbre, es decir, cuando es difuso o borroso. Esto requiere otros mecanismos de representación y de razonamiento. Una herramienta muy eficiente que trabaja en la resolución de estos tipos de problemas es la teoría difusa[12], en donde los esquemas de razonamiento están denotados como razonamiento aproximado[7, 8, 9, 10]. A partir de esto, obtenemos el Modus Ponens Generalizado (GMP) que se trata del modus ponens con razonamiento aproximado:

Regla: If X is A then Y is B

Hecho: X is A'

Conclusion: Y is B'

donde A, A', B y B' son conjuntos difusos

En 1978 Zadeh [4]presentó un método para resolverlo:

$$B'(y)=\wedge_{x\epsilon X}(B(y), O_z(A(x), A'(x))) \text{ para todo } y \in Y$$

Siendo O_z un índice de consistencia que es el concepto clásico de índice overlap extendido a conjuntos difusos.

La principal ventaja de razonamiento difuso es la capacidad de obtener nueva información, incluso cuando el hecho (X is A') es dispar a la condición de la regla (X is A). Y en los casos en los que A=A', volvemos al Modus Ponens (MP).

Otro método para obtener la conclusión B' es el siguiente:

$$B'(y)=\sup_{x\epsilon X} T(A'(x), I(A(x), B(y))) \text{ para todo } y \in Y$$

Donde I es un operador de implicación y T una t-norma[11]. Esta expresión es conocida con la regla composicional de inferencia.

Como los métodos nombrados anteriormente, existente varios que resuelven el Modus Ponens Generalizado intentando encontrar el mejor operador de implicación o función equivalente tal que B' satisfaga la mayoría de los axiomas de Fukami[5] y Baldwin y Pilsworth[6].

En este trabajo se propone un nuevo algoritmo de inferencia que utiliza índices de overlap e igualdad y se comprueba si cumple las propiedades de Fukami[5] y Baldwin y Pilsworth[6].

Axiomas de Fukami

Los axiomas de Fukami [5]:

F1. If A'=A then B'=B

F2. Modificador muy   If $A'=A^2$ then B'=B ó

If $A'=A^2$ then $B'=B^2$

F3. Modificador más o menos   If $A'=A^{½}$ then $B'=B^{½}$

F4. Complementario   If $A'=A_c$ then B'=Y ó

If $A'=A_c$ then $B'=B_c$

Axiomas de Baldwin y Pilsworth

A parte existen otros cinco axiomas escritos por Baldwin y Pilsworth [6], que algunos de ellos están en contracción con los de Fukami.

B1. B'≥B

B2. If $A'_1 \leq A'_2$ then $B'_1 \leq B'_2$ monotono.

B3. If $A'=A_c$ then B'=Y

B4. Simetría entre le modus ponens y el modus tollens: If B'$= A' o\ I(A, B)$ then A'_C=$B'_C\ o\ I(A, B)$

Algoritmo con índices de overlap e igualdad

Se tiene:

Regla:	If X is A then Y is B
Hecho:	X is A$'$
Conclusion:	Y is B$'$

Con A , A’ ϵ FSs(X) y B, B’ ϵ FSs(Y) siendo X, Y referenciales finitos no vacios

1. Se selecciona un índice de overlap O según la Definición 5 y un índice de igualdad EQ_{DI} según la ecuación de Proposición 2.
2. Se construye el conjunto Φ para todo $y \epsilon Y$

$$\Phi(y) = \begin{cases} 0, & si\ B(y) = 0 \\ [B(y)]^{\gamma} & en\ caso\ contrario \end{cases}$$

$$\text{donde } \gamma = \begin{cases} (2.1)\ 0\ , si\ O(A, A') = 0\ ó\ A' = A_C; \\ (2.2)\ \mathbb{K}, si\ A(x_i)A'(x_i) \neq 0\ y\ A(x_i)A'(x_i) \neq 1\ y\ A(x_i) \neq 1\ y \\ \qquad \mathbb{K} = \frac{lnA'(x_i)}{lnA(x_i)}\ \forall x_i \in X; \\ (2.3)\ 1, si\ O_N(A', A) = 1 \\ (2.4)\ K\ \cdot\ EQ_{DI}(A', A) \end{cases}$$

donde

$$K = \frac{1}{n}\sum_{i=1}^{n} w_i\ \ siendo\ \ w_i = \begin{cases} \wedge\left(1, \frac{lnA'(x_l)}{lnA(x_l)}\right), & si\ A(x_l)A'(x_l) \neq 0\ y\ A(x_l) \neq\ 1; \\ 1, & si\ \ (A(x_l) =\ 0\ y\ A'(x_l) \neq 0)\ \ ó \\ & \quad (A(x_l) \neq\ 0\ y\ A'(x_l) = 0)\ \ ó \\ & \quad (A(x_l) =\ 1\ y\ 0 \neq A'(x_l) \neq 1) \end{cases}$$

3. Construimos el conjunto final

$$B'(y) = \bigvee(B(y), \Phi(y))\ \ para\ todo\ y\ \epsilon\ Y$$

Ejemplos

En esta sección vamos a realizar diferentes pruebas aplicando el algoritmo de arriba y posteriormente vamos a estudiar los datos de salida.

Para ello vamos a utilizar el siguiente supuesto caso que consiste en estudiar cual es el *rendimiento general*(Y) de un empleado, a partir de su *cualidad individual*(X). Tienen el siguiente esquema:

- Si cualidad individual =A entonces rendimiento general=B
  cualidad individual = A'

---

rendimiento general=B'?

Como dicta el nuevo método, a priori se debe elegir un índice de overlap e un índice de igualdad, éste último será el del Ejemplo 4:

$$EQ_{DI}(A,B) = \wedge\{\frac{1}{n}\sum_{i=1}^{n} \wedge(1, 1 - A(x) + B(x)), \frac{1}{n}\sum_{i=1}^{n} \wedge(1, 1 - B(x) + A(x))\}$$

En cambio, se va a seleccionar dos índices de overlap los cuales ambos cumplen las propiedades de la Definición 5, y uno de ellos también satisface la característica de índice de overlap normal.

Índice de overlap normal

En este caso, el índice de overlap elegido es:

A=B={x_1, x_2, …, x_n}

O(A,B)=$\vee A(x_i) \wedge B(x_i)$ para i ϵ {1,.., n} (1)

Es normal ya que tiene esta propiedad extra:

- Si existe un u_i ϵ U tal que $\mu_A(u_i)$= $\mu_B(u_i)$=1 entonces O(A,B)=1

<u>Demostración 1</u>: Suponemos que existe un x_i ϵ A, B tal que A(x_i)=B(x_i)=1

A(x_1)=B(x_1)=1

O(A,B)=$\vee A(x_i) \wedge B(x_i) = \vee \min(1,1), \min(A(x_i), B(x_i))$ para i ϵ {2,..,n}

O(A,B)=max(1, $\min(A(x_i), B(x_i))$ y como A y B toman valores entre [0, 1] entonces se obtiene que O(A,B)=1.

Proposición 3: Bajo la condiciones de algoritmo y de los índices elegidos anteriormente, se obtiene el cumplimiento de la siguiente propiedad.

(F1) If A'=A then B'=B

<u>Ejemplo 5</u>: En este ejemplo vamos a calcular el consecuente B', en los casos en los que A' = A

Si cualidad individual = A entonces rendimiento general=B
cualidad individual = A

---

rendimiento general=B'

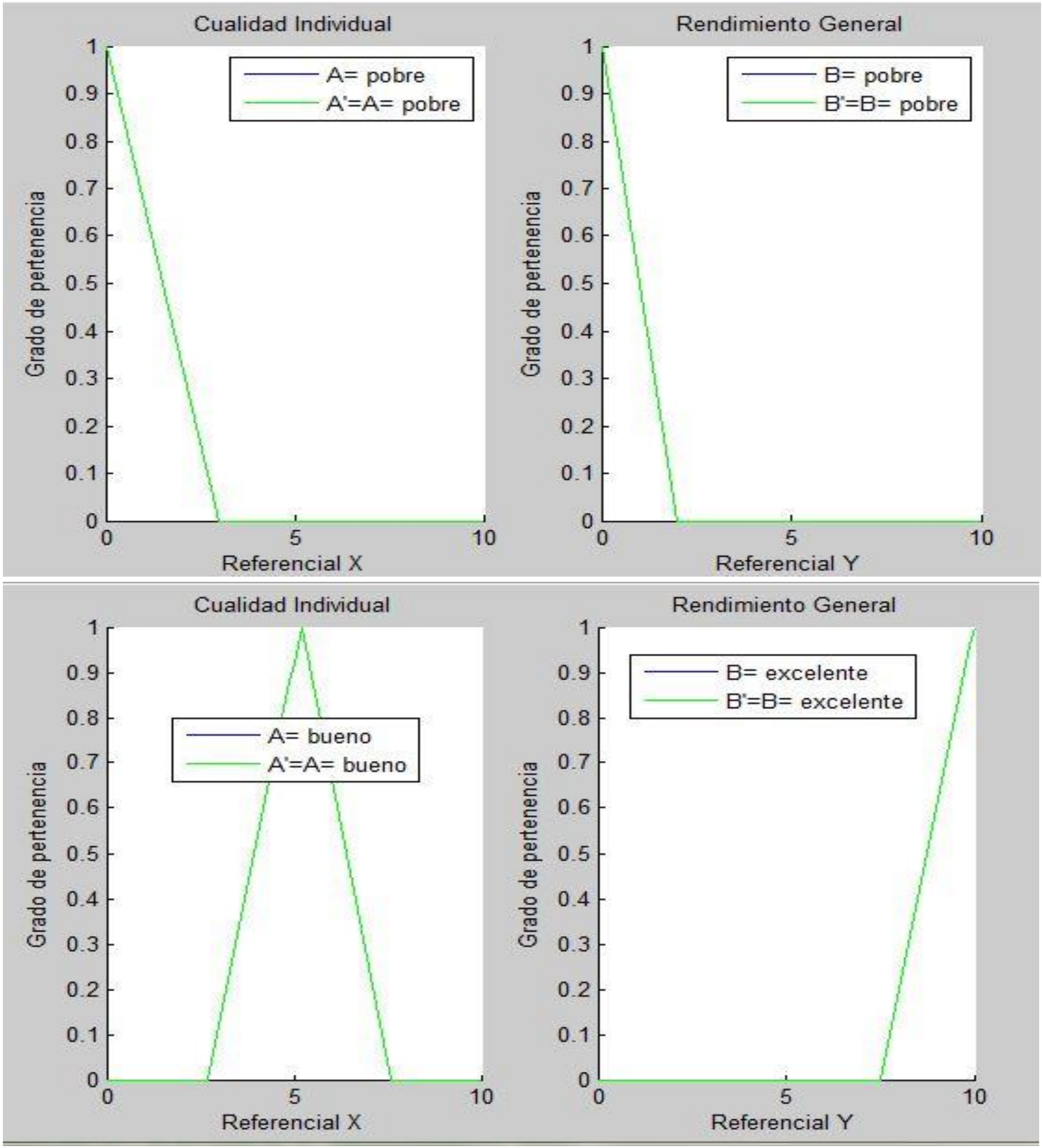


Fig. 2.1. Estudio del primer axioma de Fukami con índice de overlap (1)

Proposición 4: Bajo la condiciones de algoritmo y de los índices elegidos anteriormente, se obtiene el cumplimiento de la siguiente propiedad.

(F2) If A'=A^2 then B'=B

Como se trata de un índice de overlap normal, es decir, A(x_i)A'(x_i)=1, entonces $\gamma = 1$ entre en el paso (2.3)

<u>Ejemplo 6</u>: En este ejemplo vamos a calcular el consecuente B', en los casos en los que A' = A^2, es decir, vamos aplicarle el modificador "muy".

Si cualidad individual = A entonces rendimiento general=B
cualidad individual = A^2

---

rendimiento general=B'

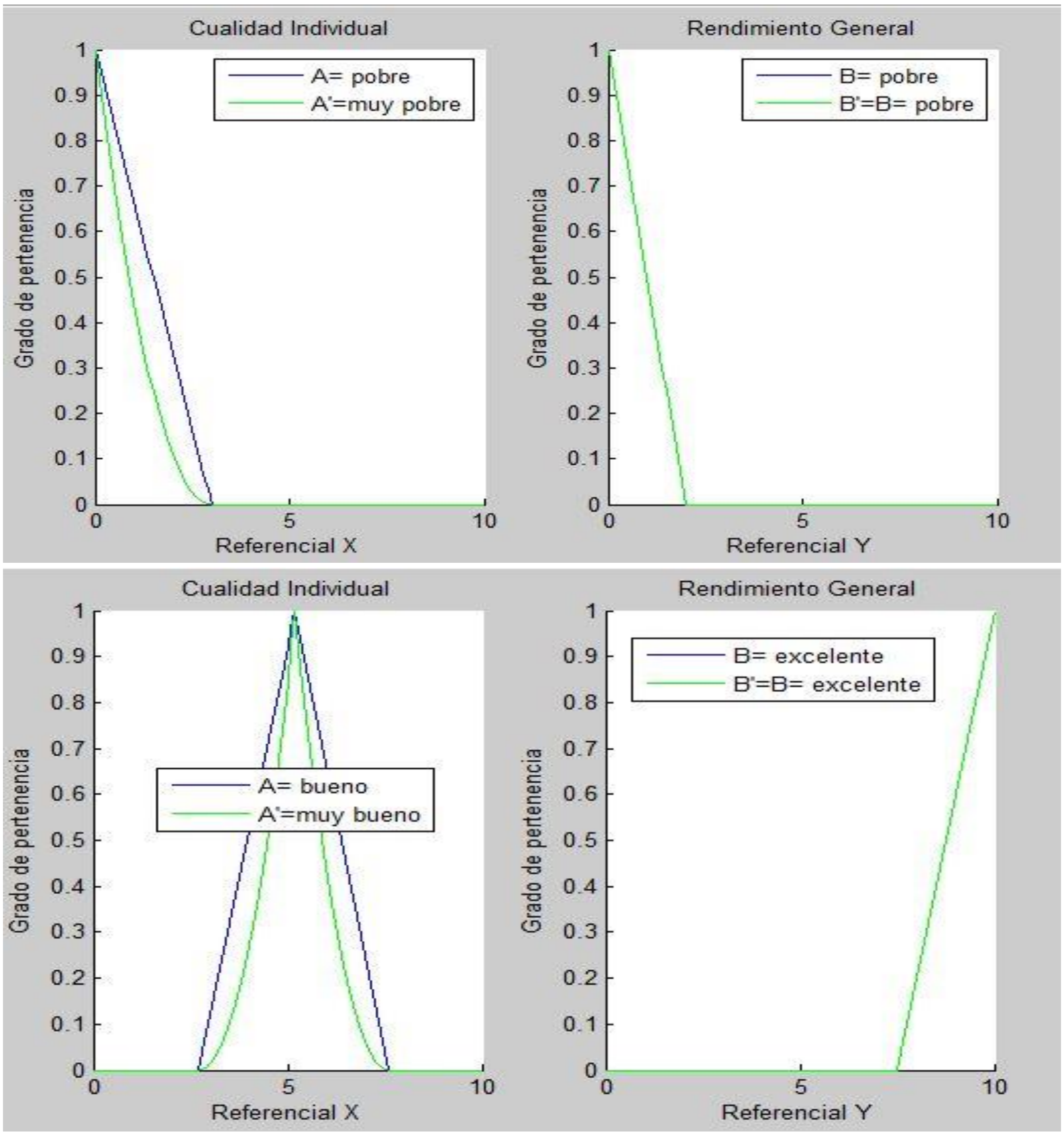


Fig. 2.2. Estudio del segundo axioma de Fukami con índice de overlap (1)

Proposición 5: Bajo la condiciones de algoritmo y de los índices elegidos anteriormente, sí satisface el tercer axioma de Fukami.

(F3) If $A' = A^{½}$ then $B' = B^{½}$

Ejemplo 7: En este ejemplo vamos a calcular el consecuente B' aplicándole el modificador más o menos $A' = A^{½}$,

Si cualidad individual = A entonces rendimiento general=B
cualidad individual = $A^{½}$

---

rendimiento general=B'

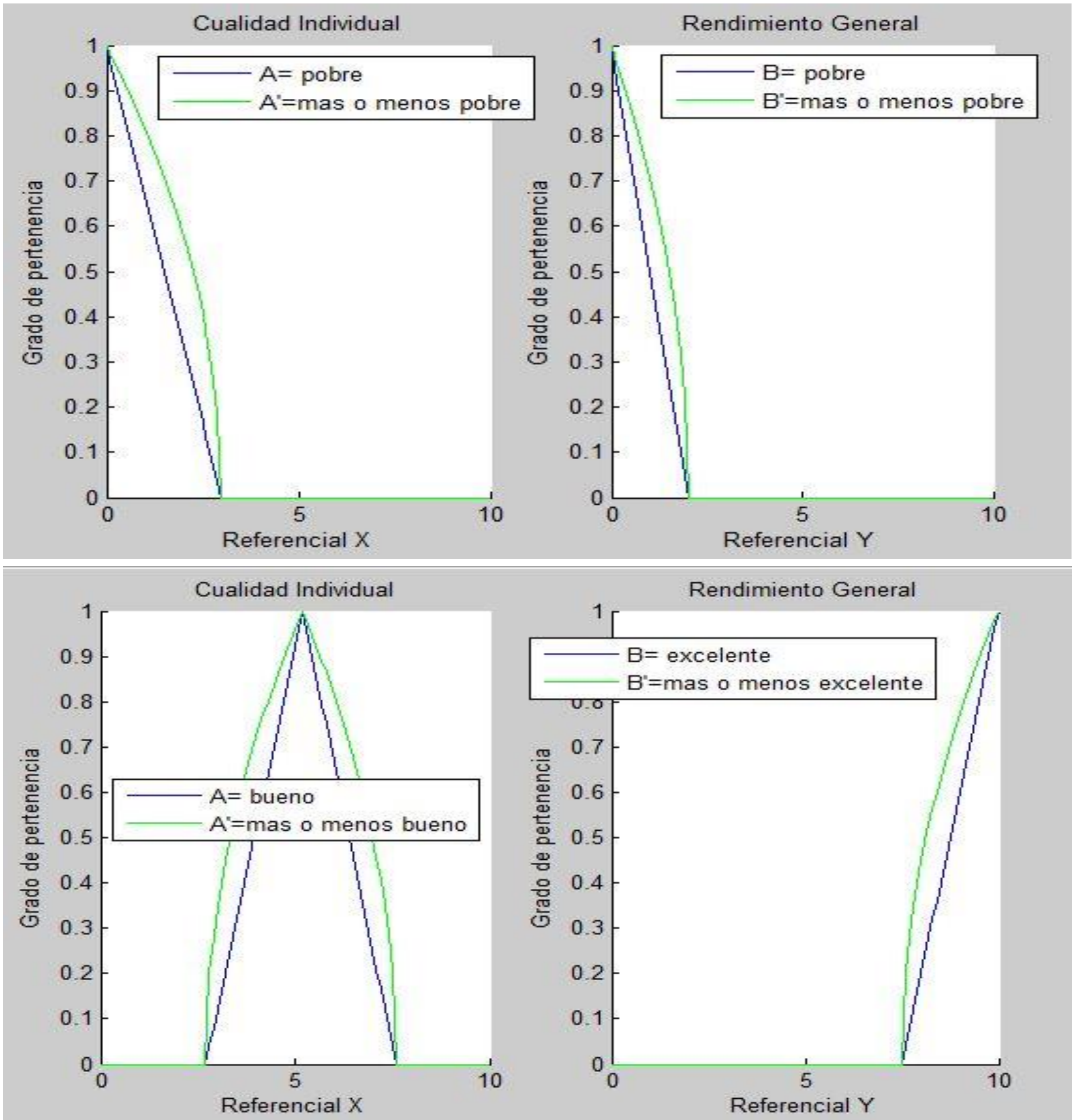


Fig. 2.3. Estudio del tercer axioma de Fukami con índice de overlap (1)

Proposición 7: Bajo la condiciones de algoritmo y de los índices elegidos anteriormente, se obtiene el cumplimiento de la siguiente propiedad.

(F4) If A'=A_C then B'=Y

(B3) If A'=A_C then B'=Y

En este caso el valor de $\gamma = 0$ calculado en el paso (2.1)

<u>Ejemplo 8</u>: En este ejemplo vamos a calcular el consecuente B' siendo A' el complementario de A.

Si cualidad individual = A entonces rendimiento general=B
cualidad individual = A_C

---

rendimiento general=B'

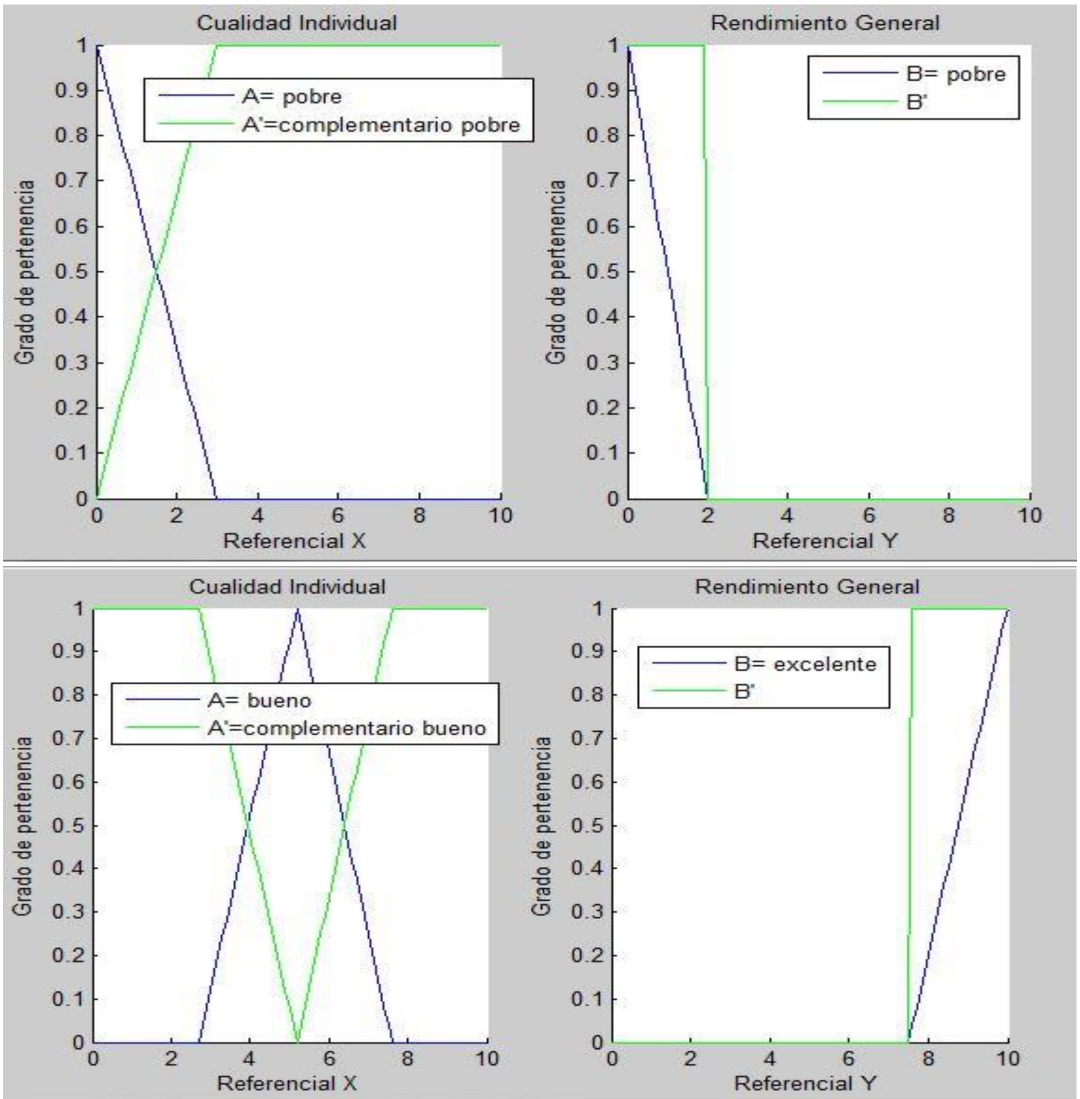


Fig. 2.4. Estudio del cuarto axioma de Fukami con índice de overlap (1)

Índice de overlap no normal

Aquí el índice de overlap es:

A=B={x_1, x_2, …, x_n}

O(A,B)=$\frac{1}{n}\sum_{i=n}\sqrt{A(x_i)\cdot B(x_i)}$ para i ϵ {1,…,n} (2)

No se trata de un índice de overlap normal porque para que O(A,B)=1 todos los valores de A(x_i)=B(x_i)=1 para i ϵ {1,…,n}

Proposición 8: Con el nuevo índice de overlap, y bajo las condiciones del nuevo método y el índice de igualdad elegido al principio, cumple:

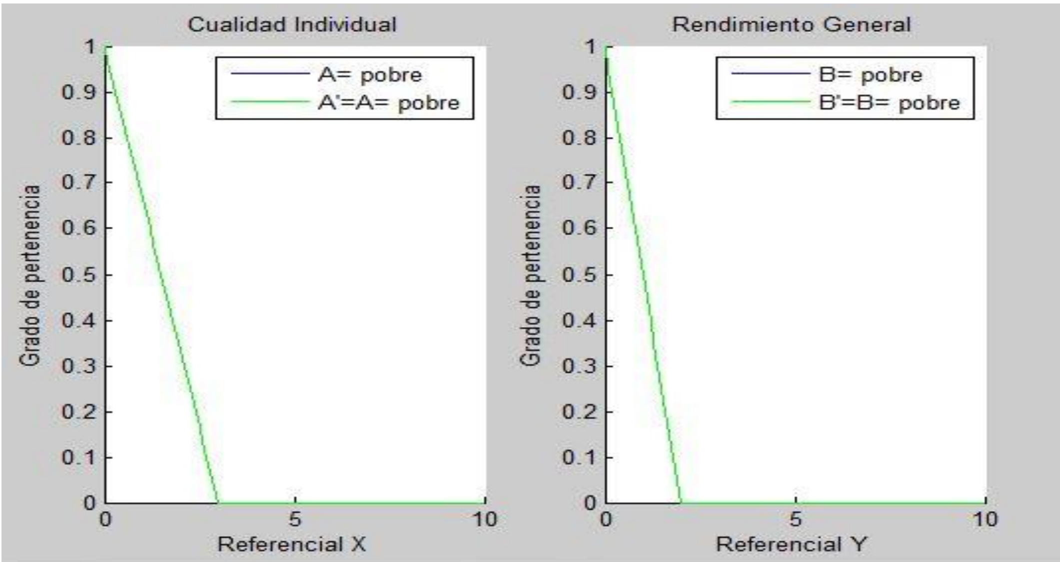
(F1) If A'=A then B'=B

Ejemplo 9: En este ejemplo vamos a calcular el consecuente B', en los casos en los que A' = A

Si cualidad individual = A entonces rendimiento general=B
cualidad individual = A

---

rendimiento general=B'



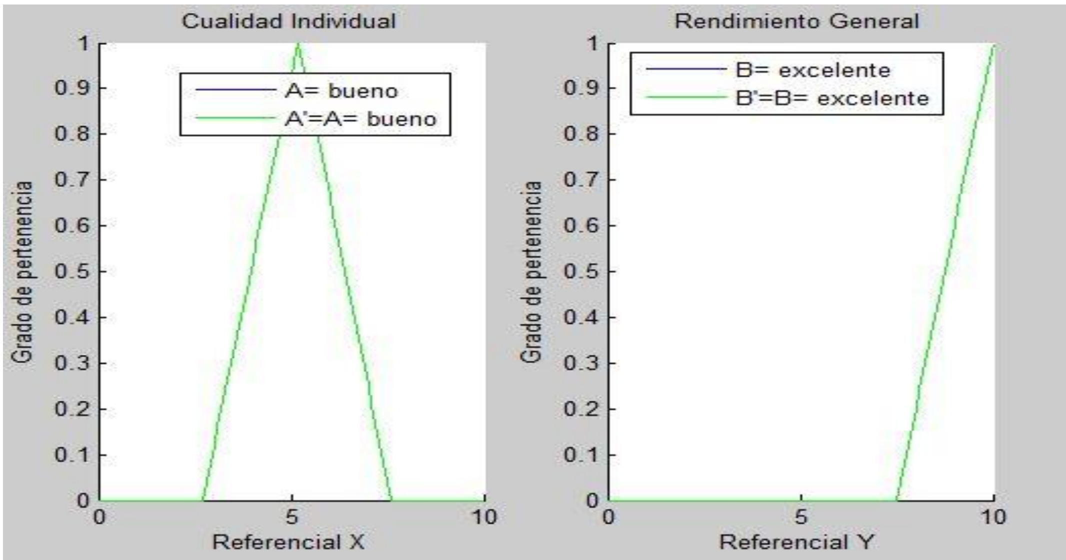


Fig. 2.5. Estudio del primer axioma de Fukami con índice de overlap (2)

Proposición 9: En este caso se puede observar que B' es igual a B luego

(F2) If $A'=A^2$ then B'=B

Si $A'=A^q$ con $q\geq1$ entonces B'=B

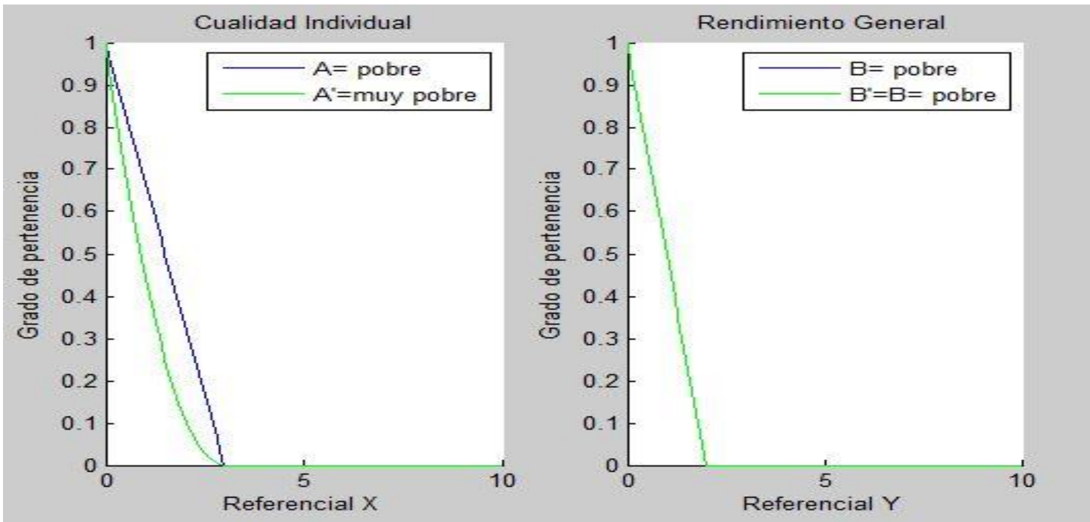
<u>Ejemplo 10</u>: Si A' es igual a A^2 luego el consecuente B' es:

Si cualidad individual = A entonces rendimiento general=B

cualidad individual = A^2

---

rendimiento general=B'



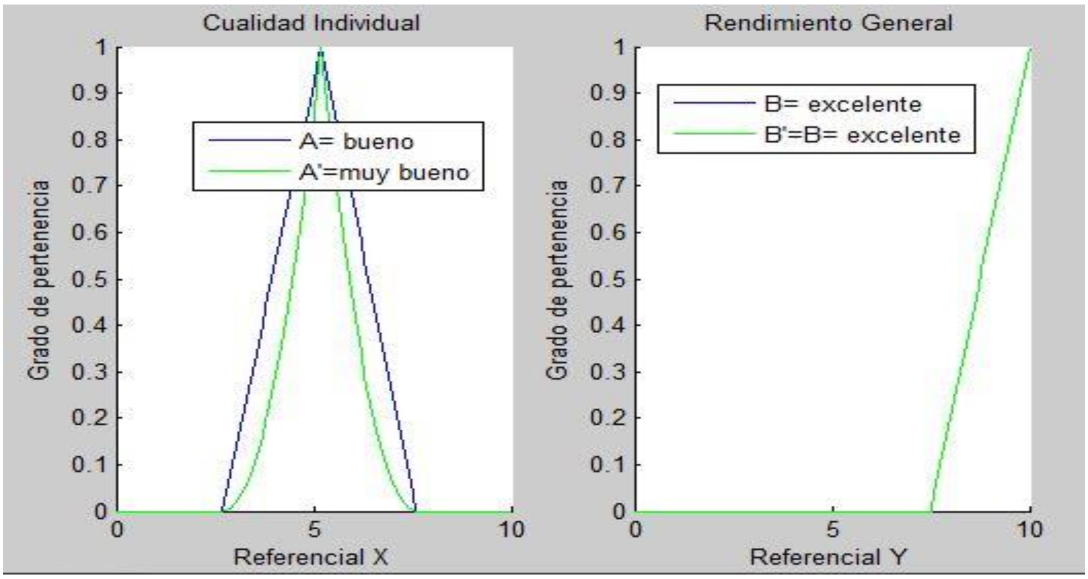


Fig. 2.6. Estudio del segundo axioma de Fukami con índice de overlap (2)

Proposición 10: Bajo las condiciones de algoritmo y un índice de overlap no normal sí que se da el siguiente resultado:

(F3) If $A'=A^{½}$ se obtiene que $B'=B^{½}$

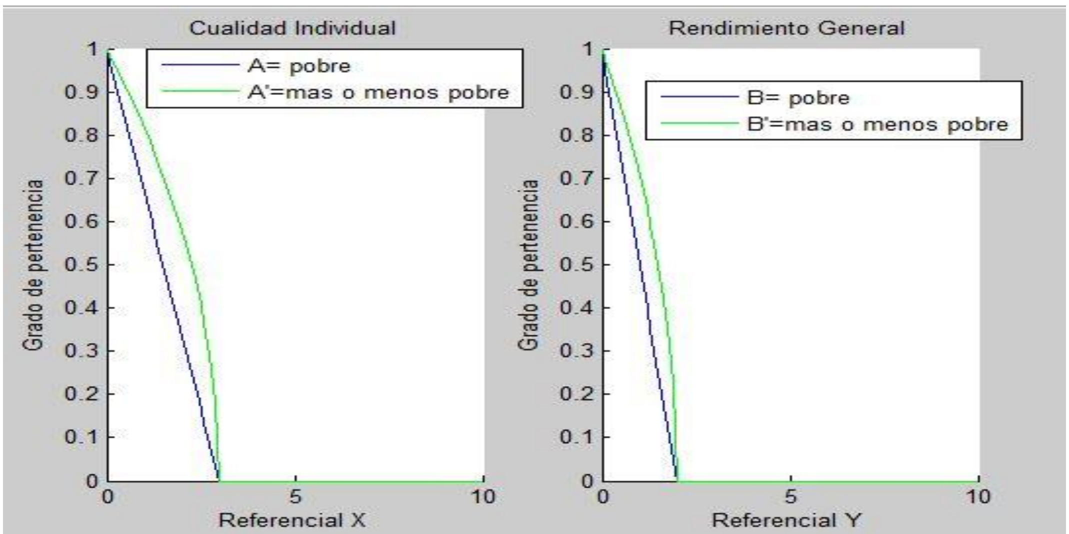
Ya que el valor de γ es igual a 0.5, calculado en el paso (2.2)

Ejemplo 11: En este ejemplo vamos a calcular el consecuente B' aplicándole el modificador más o menos $A' = A^{½}$,

Si cualidad individual = A entonces rendimiento general=B
cualidad individual = $A^{½}$

---

rendimiento general=B'



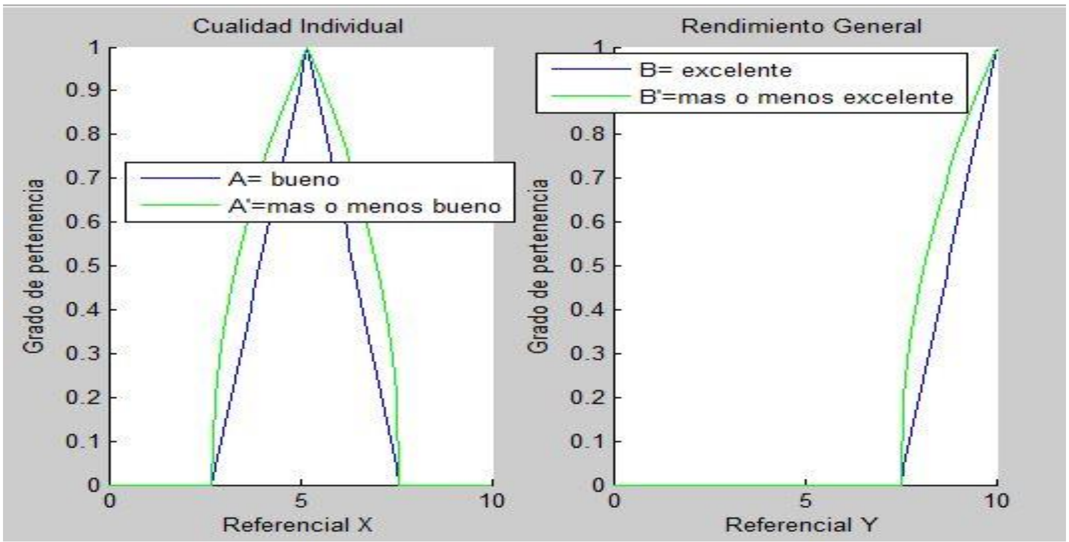


Fig. 2.7. Estudio del tercer axioma de Fukami con índice de overlap (2)

Proposición 11: Satisface la siguiente característica:

(F4) If $A'=A_C$ then $B'=Y$

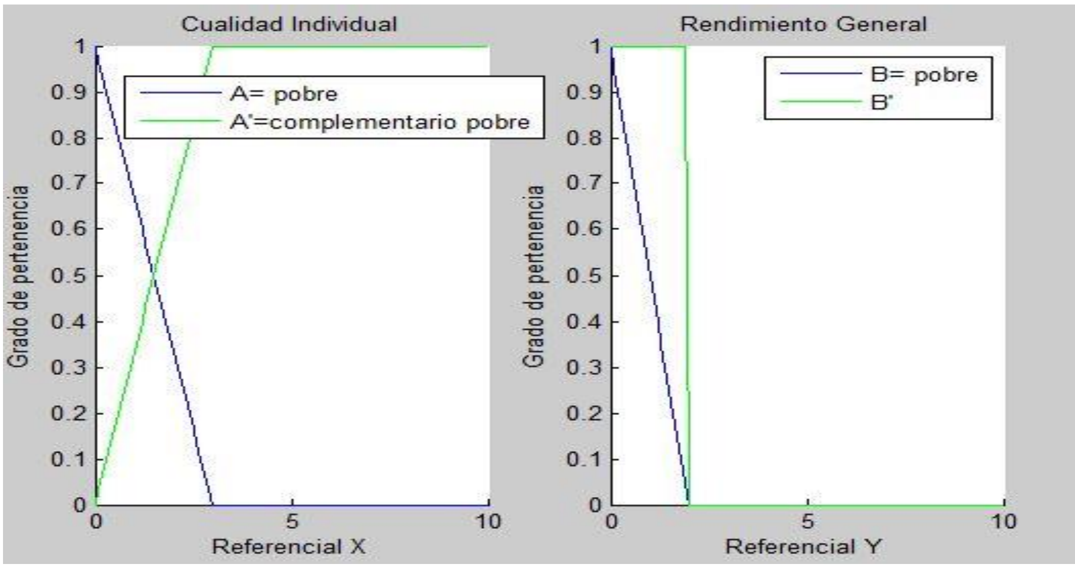
(B3) If $A'=A_C$ then $B'=Y$

<u>Ejemplo 12</u>: En este ejemplo vamos a calcular el consecuente B’ siendo A’ el complementario de A.

Si cualidad individual = A entonces rendimiento general=B
cualidad individual = A_C

---

rendimiento general=B’



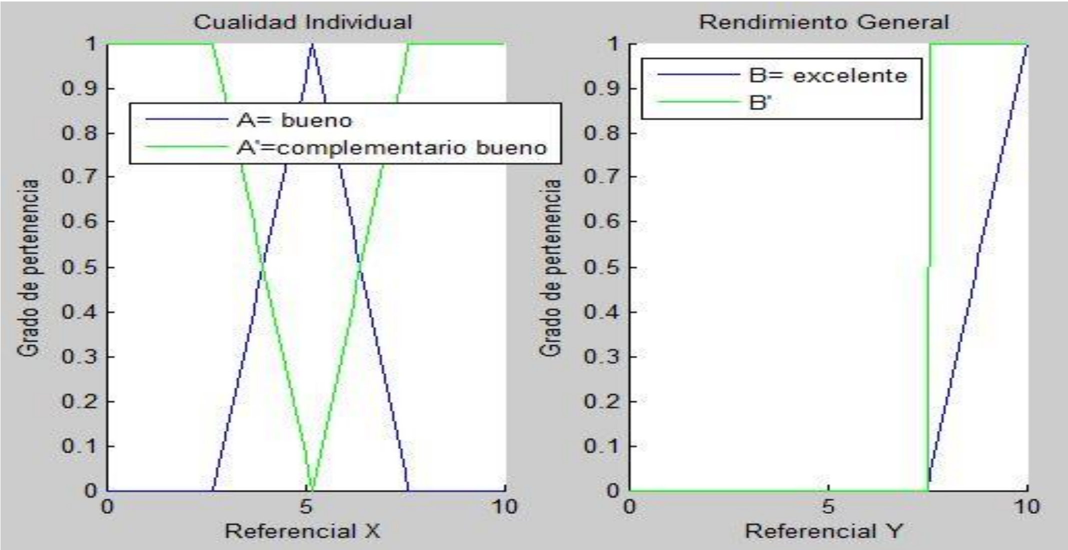


Fig. 2.8. Estudio del cuarto axioma de Fukami con índice de overlap (2)

Reducción de un sistema de reglas a una sola aplicando funciones penalty

Construcción de R para cada regla

Cada regla, como ya se ha nombrado anteriormente, tiene la siguiente estructura:

- Si X_1=A_1 y X_2=A_2 y X_3=A_3 y X_4=A_4 entonces Y=B.

Debido a que la aplicación consta de cuatro antecedentes, lo primero que se debe hacer es unirlos y obtener uno único, para ello vamos a utilizar una t-norma:

Ejemplo:

A_1=A_2= A_3= A_4=$\{x_1, \dots, x_n\}$

A(x_i)=min($A_1(x_i), A_2(x_i), A_3(x_i), A_4(x_i)$) para cada i ϵ {1,…, n}

Después de esto ya se puede construir la relación R, por ejemplo utilizando un operador de implicación I_L=min(1, 1-x+y).

Entonces para cada regla j ϵ {1,..,8} tenemos:

$A_j = \{x_1, \dots, x_n\}$
$B_j = \{y_1, \dots, y_m\}$

$$R_j = \begin{pmatrix} I_L(x_1, y_1) & \cdots & I_L(x_1, y_m) \\ \vdots & \ddots & \vdots \\ I_L(x_n, y_1) & \cdots & I_L(x_n, y_m) \end{pmatrix}$$

Finalmente obtenemos 8 matrices R, una para cada regla.

Funciones penalty

Elegir el conjunto de funciones de agregación para agregar los datos de entrada al sistema, los cuales están formados por muchas componente, y con la finalidad de que escoger el que presenta menos disimilitud respecto a dichos datos de entrada.

Definición 1: Una función penalty es una función P: [a, b]$^{n+1}$ -> $\mathbb{R}^+$ = [0, ∞] que satisface las siguientes propiedades:

1. P($\mathbf{x}$, y)>=0 para todo x ϵ [a, b]^n, y ϵ [a, b]
2. P($\mathbf{x}$, y)=0 si x_i = y para todo i=1,..., n
3. P($\mathbf{x}$, y) es quasiconvexa en y para cualquier $\mathbf{x}$, esto es,
   P($\mathbf{x}$, λ · y_1 + (1- λ) · y_2) <= max(P($\mathbf{x}$, y_1), P($\mathbf{x}$, y_2))

Denominaremos función basada en la función penalty P a la función

f(x)=arg $\min_{y} P(\boldsymbol{x}, y)$.

si y es el único mínimo, e $y = \frac{c+d}{2}$ si el conjunto de mínimos es el intervalo [c, d]

Teorema: Cualquier función de agregación promedio puede ser representada como una función basada en una función penalty escrita en la Definición 1.

Ejemplos:

1. La media aritmética es la función basada en la función penalty
   P($\mathbf{x}$, y)=$\sum_{i=1}^{n}(x_i - y)^2$

   Y por tanto es la solución de: minimizar $\sum_{i=1}^{n}(x_i - y)^2$

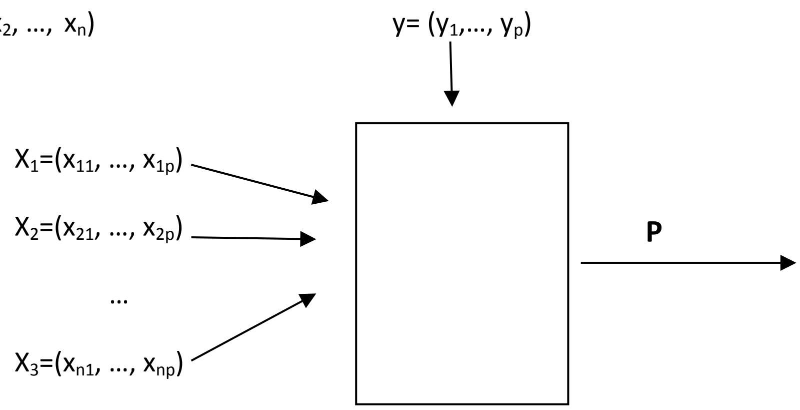
2. La mediana es la función basada en la función penalty

   P($\mathbf{x}$, y)=$\sum_{i=1}^{n}|x_i - y|$

   Luego es la solución de: minimizar $\sum_{i=1}^{n}|x_i - y|$

Función penalty en producto cartesiano de retículos

X:(x_1, x_2, ..., x_n)



Corolario: Sean a, b ϵ L_m ={C_1 x … x C_m, ≤, Λ, V }. Entonces, todas las cadenas maximales que unen Λ(a, b) y V(a, b) tienen la misma longitud.

Si L_m es el producto cartesiano de m cadenas, entonces la distancia entre x, y ϵ L_m puede ser definida como la longitud de la cadena C con elemento minimal a= Λ(x, y) y el elemento maximal b=V(x, y). Es decir

d(x, y)= longitud(C)-1

Es equivalente a:

d(x, y)= $\sum_{i=1}^{m} d_i(x_i, y_i) = \sum_{i=1}^{m} |x_i - y_i|$

<u>Método de construcción de FPSPCR</u>

Funciones de disimilitud restringidas reticulares fieles:

δ_R(x, y)=K(d(x, y))=K($\sum_{i=1}^{m} |x_i - y_i|$)

Siendo K: C -> C una función convexa con un único mínimo en K(0)=0

Teorema: Sean K_i : ℝ-> $\mathbb{R}^+$ funciones convexas con un único mínimo K_i(0)=0 (i=1, …, m), y sea la distancia entre conjuntos difusos definida por:

D(A,B)= $\sum_{i=1}^{m} |A(u_i) - B(u_i)|$

Donde A, B ϵ FS(U) y Cardinal(U)=n. Entonces la aplicación

P_∇: $FS(U)^m$ x L^*_m -> $\mathbb{R}^+$ dada por

P_∇(**A**, Y)=D(**A, B**$_Y$)=$\sum_{q=1}^{m} K_q\left(D\left(A_q, B_{y_q}\right)\right) = \sum_{q=1}^{m} K_q(\sum_{p=1}^{n} |A_q(u_p) - y_q|)$

satisface

1. P_∇(**A**, Y)≥0
2. P_∇(**A**, Y)=0 si y solo si A_q = y_q para cada q=1,…,m
3. Es convexa en y_q para cada q=1,..,m

Ejemplo.

1. Si tomamos $K_q(x)=x^2$ para todo $q \in \{1,\ldots,m\}$ entonces

$$P_\nabla(\mathbf{A}, Y)=\sum_{q=1}^{m}(\sum_{p=1}^{n}|A_q(u_p)-y_q|)^2$$

2. Si $K_q(x)=x$ para todo $q \in \{1, .., m\}$ entonces

$$P_\nabla(\mathbf{A}, Y)=\sum_{q=1}^{m}\sum_{p=1}^{n}|A_q\left(u_p\right)-y_q|$$

Uso de funciones penalty en nuestra aplicación

En nuestro caso tenemos 8 matrices las cuales queremos reducir a una sola.

A=$\{x_1, \ldots, x_{101}\}$

B=$\{y_1, \ldots, y_{101}\}$

Luego cada relación R, construida utilizando un operador de implicación, tiene 101 filas y 101 columnas:

$$R_1=\begin{pmatrix} r_1(1,1) & \cdots & r_1(1,101) \\ \vdots & \ddots & \vdots \\ r_1(101,1) & \cdots & r_1(101,101) \end{pmatrix}$$

$$\vdots \qquad\qquad \vdots$$

$$R_8=\begin{pmatrix} r_8(1,1) & \cdots & r_8(1,101) \\ \vdots & \ddots & \vdots \\ r_8(101,1) & \cdots & r_8(101,101) \end{pmatrix}$$

Las funciones de agregación utilizadas son: mínimo, máximo y la media geométrica(M.G).

El proceso que se utilizará para obtener la relación resultante de unir las ocho, será el siguiente:

1. Hasta que no lleguemos al final de la matriz

   1.1. Selecciona los mismos tres elementos consecutivos para cada relación.

   Ej.

$$\begin{pmatrix} (R_{(1,1)})_1 \\ \vdots \\ (R_{(1,1)})_8 \end{pmatrix}, \begin{pmatrix} (R_{(1,2)})_1 \\ \vdots \\ (R_{(1,2)})_8 \end{pmatrix}, \begin{pmatrix} (R_{(1,3)})_1 \\ \vdots \\ (R_{(1,3)})_8 \end{pmatrix}$$

1.2. Generamos todas las posibles maneras de agregarlos

VR_3^3=3^3=27 posibles maneras

Ej.

$$\text{agregación}(1,1{:}3)=\min\begin{pmatrix}(R_{(1,1)_1})\\ \vdots \\ (R_{(1,1)_8}\end{pmatrix},\min\begin{pmatrix}(R_{(1,2)_1})\\ \vdots \\ (R_{(1,2)_8}\end{pmatrix},\min\begin{pmatrix}(R_{(1,3)_1})\\ \vdots \\ (R_{(1,3)_8}\end{pmatrix}$$

$$\text{agregación}(2,\ 1{:}3)=\min\begin{pmatrix}(R_{(1,1)_1})\\ \vdots \\ (R_{(1,1)_8}\end{pmatrix},\min\begin{pmatrix}(R_{(1,2)_1})\\ \vdots \\ (R_{(1,2)_8}\end{pmatrix},\max\begin{pmatrix}(R_{(1,3)_1})\\ \vdots \\ (R_{(1,3)_8}\end{pmatrix}$$

$\vdots \qquad\qquad\qquad \vdots$

$$\text{agregación}(27,\ 1{:}3)=\text{M.G}\begin{pmatrix}(R_{(1,1)_1})\\ \vdots \\ (R_{(1,1)_8}\end{pmatrix},\text{M.G}\begin{pmatrix}(R_{(1,2)_1})\\ \vdots \\ (R_{(1,2)_8}\end{pmatrix},\text{M.G}\begin{pmatrix}(R_{(1,3)_1})\\ \vdots \\ (R_{(1,3)_8}\end{pmatrix}$$

1.3. Aplicamos la función penalty a cada una de las agregaciones del paso 2 y seleccionamos el mínimo.

$P_\nabla(\mathbf{A}, Y)=\sum_{q=1}^{m}\sum_{p=1}^{n}(A_q(u_p)-y_q)^2$

Ej.

$P_i=\sum_{q=1}^{3}\sum_{p=1}^{8}(R_{(p,q)_p}-agregacion(i,q))^2$ para i ϵ {1,.., 27}

1.4. Insertamos en la matriz final el resultado del paso 3.

R_{Final}=agregación(min(P_i), 1:3) siendo $1\leq i\leq 27$

2. Tenemos nuestra relación final R_{final}(101x101)

Nota: debido a que las matrices están compuestas por 10201 elementos, y no es múltiplo de tres, entonces en la última iteración en vez de coger tres se seleccionarán cuatro elementos.

Reconstrucción de una regla con el algoritmo del gradiente

Algoritmo del gradiente

Dada la relación R se trata de encontrar la familia de conjuntos $\{A\}_i$ y $\{B\}_i$:

1. Creamos las matrices de la iteración 0 $\{A^0\}_i$ y $\{B^0\}_i$ aleatoriamente.

2. Construimos la matriz $\tilde{R}$

$$\tilde{R}(x,y) = \bigvee_{i=1}^{c} A(x) \wedge B(y)$$

   Es decir, para calcular el elemento situado en la posición (x, y) de la nueva matriz debemos coger la columna x de la matriz A y la columna y de la matriz B, hacer el mínimo elemento a elemento y finalmente, calcular el máximo de los mínimos obtenidos.

3. Iterar

   a. Construimos la función costo(fitness), que nos servirá para calcular el error

$$Q = \sum_{(x,y)\epsilon X,Y} (R - \tilde{R})^2 = \sum_{(x,y)\epsilon X,Y} (R(x,y) - \bigvee_{i=1}^{c} A_i(x) \wedge B_i(y))^2$$

$$Q^{iter} = \sum_{(x,y)\epsilon X.Y} (R(x,y) - \bigvee_{i=1}^{c} A_i^{iter}(x) \wedge B_i^{iter}(y))^2$$

   b. Calculamos los valores de A y B para la determinada iteración

$$A_i^{iter+1}(x) = A_i^{iter}(x) - \alpha \frac{dQ^{iter}}{dA_i^{iter}(x)}$$

$$B_i^{iter+1}(y) = B_i^{iter}(y) - \alpha \frac{dQ^{iter}}{dB_i^{iter}(y)}$$

Condición de parada:

- Alcanzar el número máximo de iteraciones
- $Q^{iter-1} - Q^{iter} \ < umbral$

Las ecuaciones del paso 3.b equivalen:

- Actualización del elemento situado en la fila r y en la columna t de la matriz A sería:

$$A_r^{nuevo}(t) = A_r(t) + 2 * \alpha * \sum_{y=1}^{Cols} \left(R(t,y) - \bigvee_{i=1}^{c} A_i(t) \wedge B_i(y) \right)$$

$$* \left(A_r(t) \wedge B_i(y) \geq \bigvee_{\substack{i=1 \\ i \neq r}}^{c} A_i(t) \wedge B_i(y) \right) * (A_r(t) \leq B_r(y))$$

- Actualización del elemento situado en la fila r y en la columna t de la matriz B sería:

$$B_r^{nuevo}(t) = B_r(t) + 2 * \alpha * \sum_{x=1}^{Filas} \left(R(x,t) - \bigvee_{i=1}^{c} A_i(x) \wedge B_i(t) \right)$$

$$* \left(A_i(t) \wedge B_r(y) \geq \bigvee_{\substack{i=1 \\ i \neq r}}^{c} A_i(x) \wedge B_i(t) \right) * (B_r(t) \leq A_r(x))$$

En este caso tenemos que encontrar cuatro conjuntos de A: A_1, A_2, A_3, A_4 debido a que tenemos cuatro antecedentes y luego uno para B.

Después de aplicar el algoritmo del gradiente sobre la R_{final} obtenemos la siguiente familia del conjunto de A (A_1, A_2, A_3, A_4):

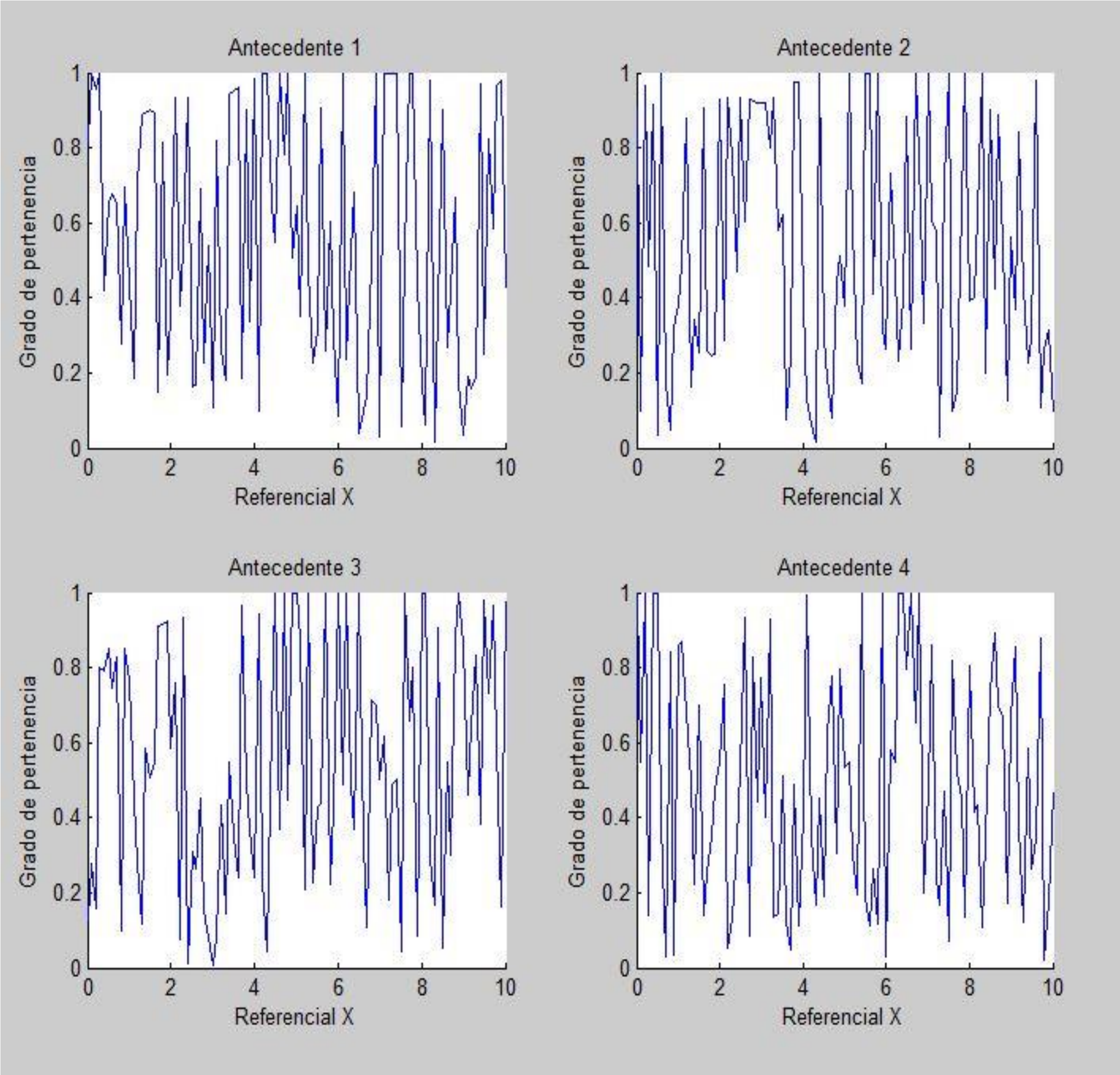


Fig.3.1. Los conjuntos de los antecedentes (A_1, A_2, A_3, A_4)

Y como consecuente B:

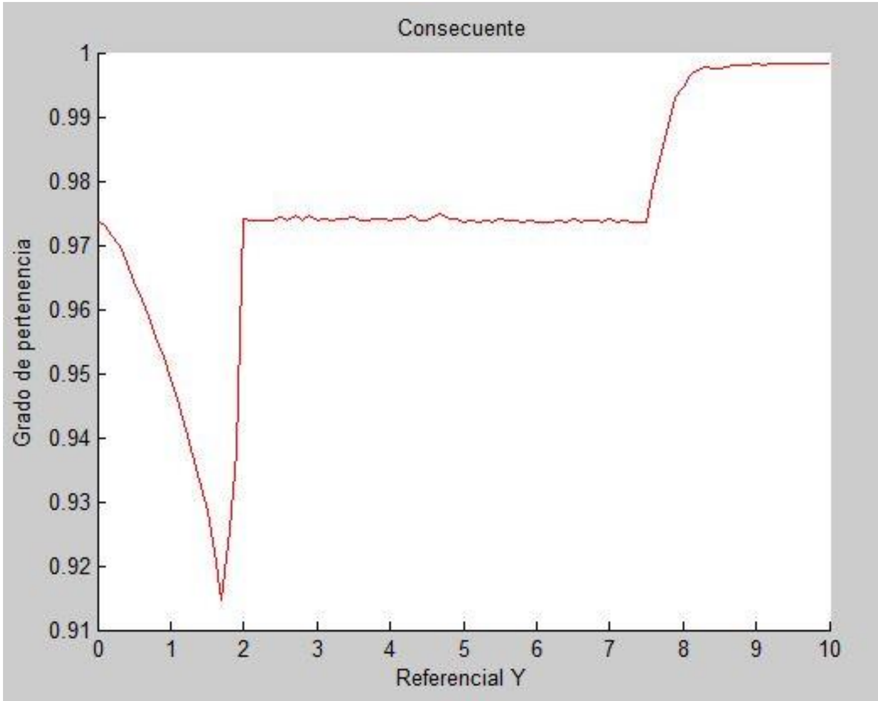


Fig.3.2. El conjunto del consecuente (B)

El resultado es la obtención de una única regla:

- Si $X_1=A_1$ y $X_2=A_2$ y $X_3=A_3$ y $X_4=A_4$ entonces Y=B.

Aplicación del nuevo algoritmo GMP utilizando índices de igualdad e overlap

Este método trabaja con reglas que solamente tiene un antecedente (X=A), asique debemos agregar los cuatro antecedentes de la "Fig.2.6." utilizando una t-norma, como por ejemplo el mínimo.

El conjunto difuso A es:

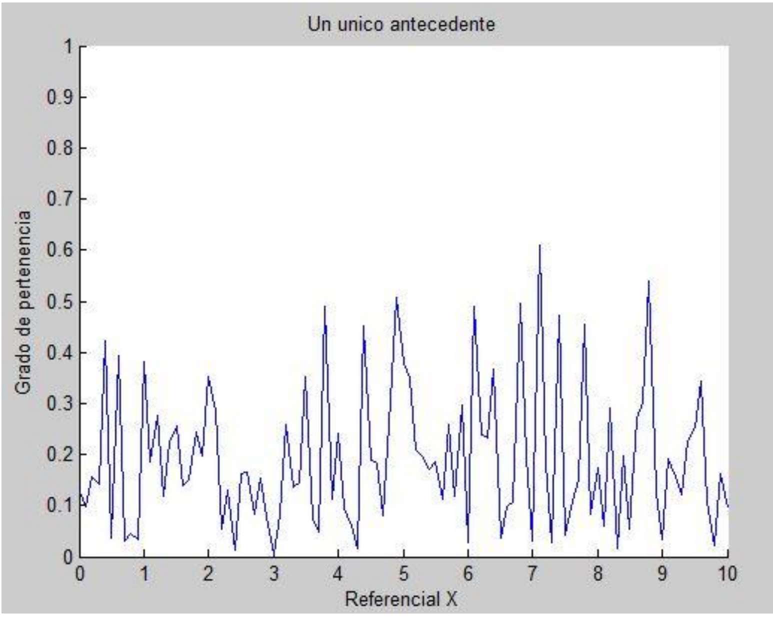


Fig.4.1. Agregación de los cuatro antecedentes

Hemos transformado una regla formada por cuatro antecedentes a otra con uno solo:

- Si X=A entonces Y=B.

Estudio de los axiomas de Fukami

Escogemos los siguientes índices de igualdad y overlap:

$$EQ_{DI}(A,B) = \wedge\{\frac{1}{n}\sum_{i=1}^{n}\wedge(1,1-A(x)+B(x)),\frac{1}{n}\sum_{i=1}^{n}\wedge(1,1-B(x)+A(x))\}$$

O(A,B)=$\vee\, A(x_i)\wedge B(x_i)$

Vamos a comprobar sus axiomas a partir de realizar una serie de ejemplos.

Proposición 1: Bajo la condiciones de algoritmo, se cumple.

(F1) If A'=A then B'=B

Se obtiene el valor de $\gamma = 1$.

Ejemplo 1: En este ejemplo vamos a calcular el consecuente B', en los casos en los que A' = A.

$$\begin{array}{l} \text{Si } X = A \text{ entonces } Y = B \\ \underline{\quad X = A \qquad\qquad\qquad} \\ \qquad\qquad\qquad Y = B' \end{array}$$

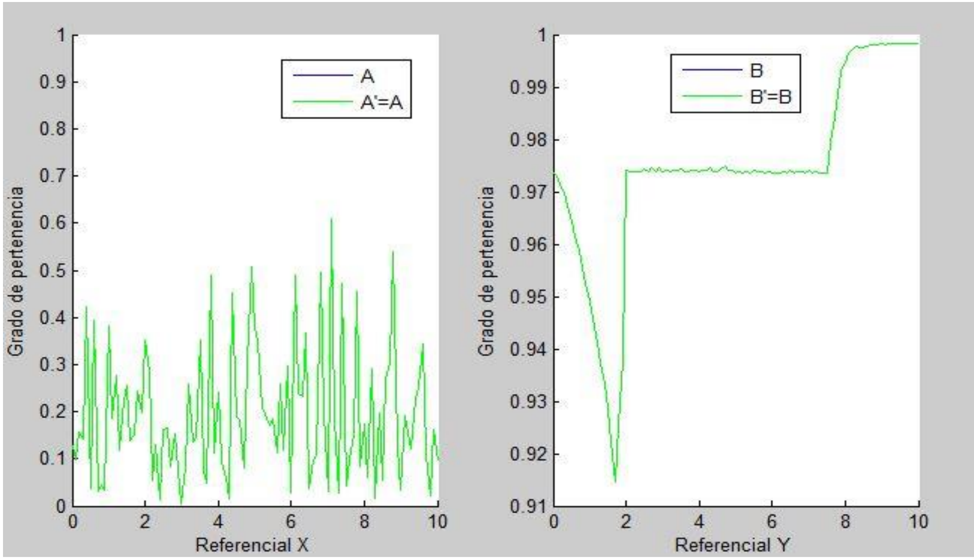


Fig.4.2. Estudio del primer axioma de Fukami

Proposición 2: Bajo la condiciones de algoritmo, se puede concluir a que satisface el siguiente axioma.

(F2) If A'=A^2 then B'=B

Si A'=A^q con q≥1 entonces B'=B

Ejemplo 2: En este ejemplo vamos a calcular el consecuente B', aplicando el modificador "muy", es decir, A'=A^2

$$\begin{array}{l} \text{Si } X = A \text{ entonces } Y = B \\ \underline{\quad X = A^2 \qquad\qquad\qquad} \\ \qquad\qquad\qquad Y = B' \end{array}$$

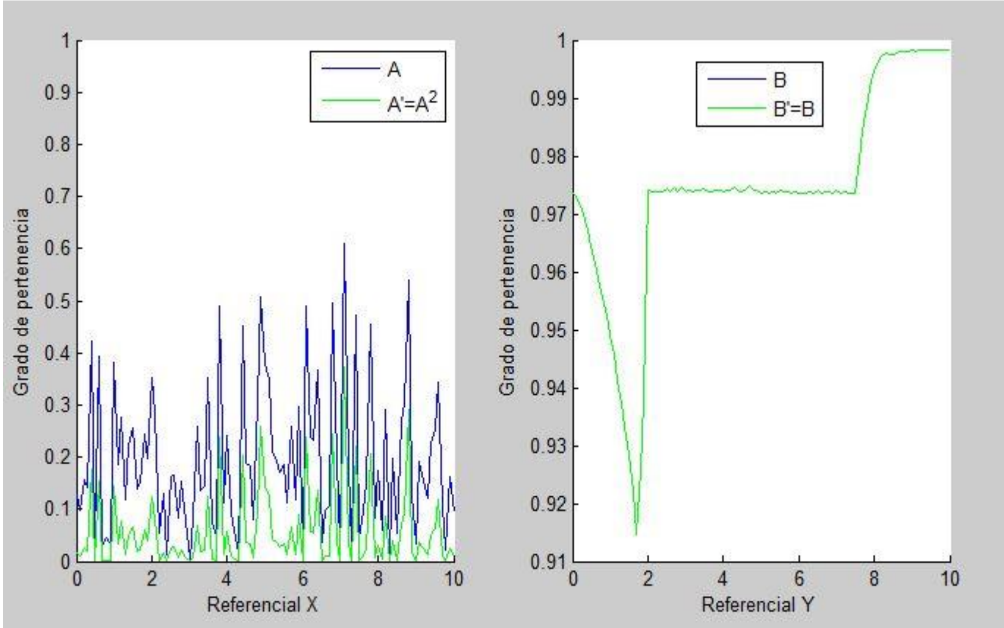


Fig.4.3. Estudio del segundo axioma de Fukami

Proposición 3: Bajo la condiciones de algoritmo satisface el tercer axioma de Fukami:

(F3) If $A'=A^{1/2}$ then $B'=B^{1/2}$

Si $A'=A^q$ con $0<q<1$ y $A(x_i)A'(x_i) \neq 1$ para todo $x_i \in X$ entonces $B'=B^q$

<u>Ejemplo 3</u>: En este ejemplo vamos a calcular el consecuente B', aplicando el modificador "mas o menos", es decir, $A'=A^{1/2}$

$$\begin{array}{l} \text{Si } X = A \text{ entonces } Y = B \\ \quad X = A^{1/2} \\ \hline \qquad\qquad\qquad Y = B' \end{array}$$

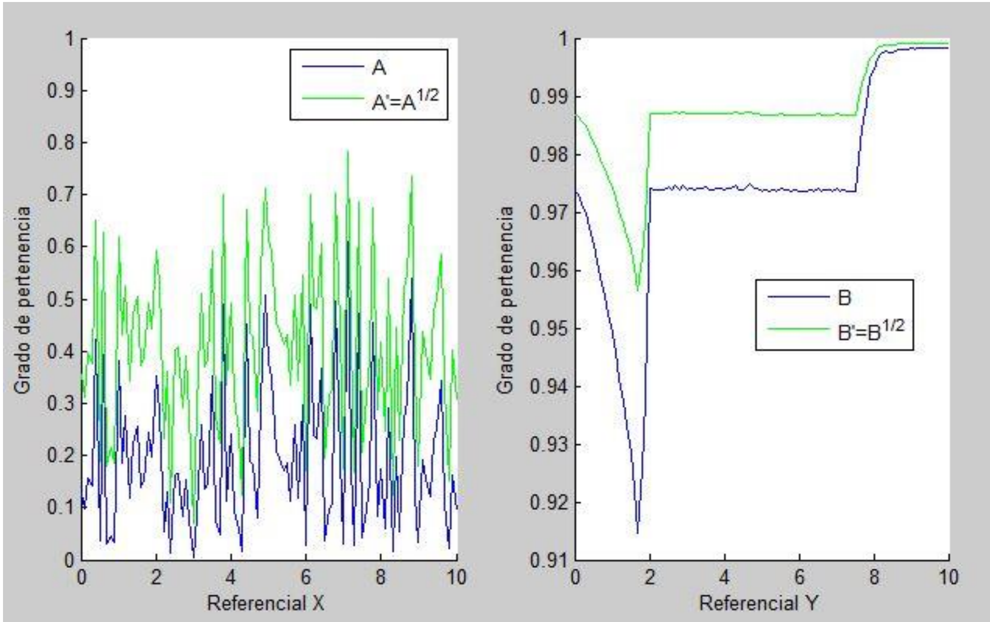


Fig.4.4. Estudio del tercer axioma de Fukami

Proposición 4: Bajo la condiciones de algoritmo satisface:

(F4) If $A' = A_C$ then $B' = Y$

Ejemplo 4: En este ejemplo vamos a calcular el consecuente B' siendo A' el complementario de A, esto es, $A' = A_C$

$$\frac{\begin{array}{l}\text{Si } X = A \text{ entonces } Y = B \\ X = A_C\end{array}}{Y = B'}$$

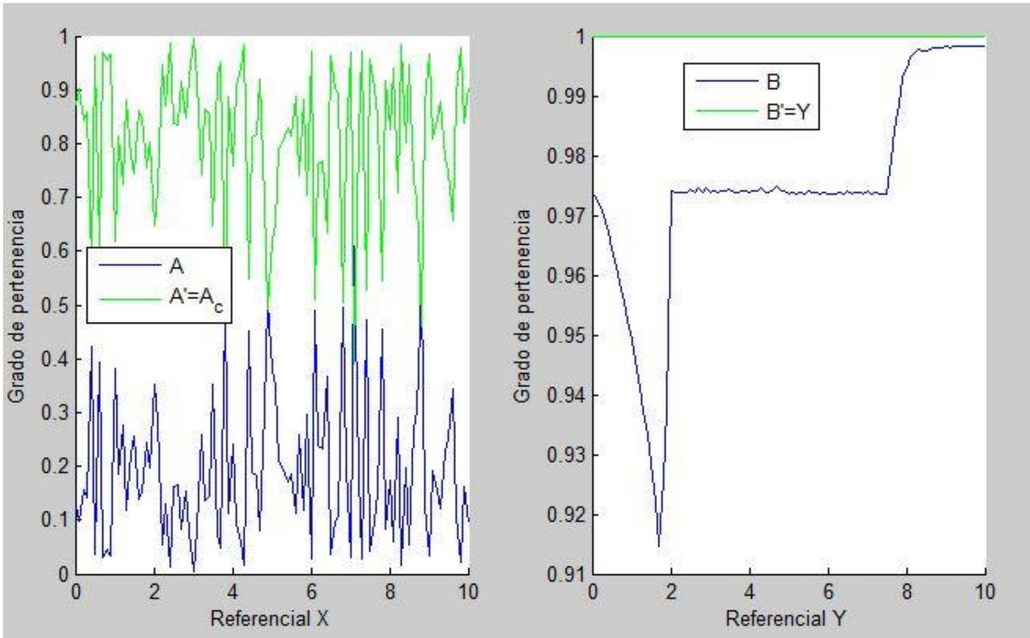


Fig.4.5. Estudio del cuarto axioma de Fukami

Conclusiones y líneas futuras

En este trabajo hemos partido de un sistema de reglas y finalmente hemos acabado con una única regla resultante del sistema, la cual hemos utilizado para clasificar nuevos datos. Como línea futura podría incorporar al proyecto el algoritmo gravitacional, en el apartado de reconstrucción de una regla.

Referencias

[1] G.Beliakov, A.Pradera, T. Calvo, Aggregation Functions: A Guide for Practitioners, Studies in Fuzziness and Soft Computing, 221, Springer, 2007.

[2] T. Calvo, A. Kolesárová, M. Komorníková and R. Mesiar: Aggregation operators: properties, classes and construction methods. In T. Calvo, G. Mayor and R.Mesiar (Eds.): *Aggregation Operators New Trends and Applications* (Physica-Verlag, Heidelberg); Pages 3-104, 2002

[3] H. Bustince, J. Fernández, R. Mesiar, J. Montero, R. Orduna, Overlap functions, *Nonlinear Analysis*, *Theory*, *Methods and Applications*, 72, pages 945-952, 2010.

[4] L. A. Zadeh, Fuzzy sets as a basis for a theory of possibility, Fuzzy Sets and Systems, 1, Pages 3-28, 1978

[5] S. Fukami, M. Mizumoto and K. Tanaka, Some considerations on fuzzy conditional inference, Fuzzy Sets and Systems, 4 pp. 243-273

[6] J.F. Baldwin and B.W. Pilsworth, Axiomatic approach to implication for approximate reasoning with fuzzy logic, *Fuzzy Sets and Systems*, 3, pp. 193-219, 1980.

[7] W. Bandler and L.J. Kohout, The four modes of inference in expert systems in R. Trappl (Ed.): *Cybernetics and Systems Research 2*, (Elsevier Science Publishers (North-Holland), Amsterdam); Pages 581-586, 1984.

[8] A. Mamdani, Applications of fuzzy set theory to control systems: a survey, in: M. Gupta and E. Sanchez (Eds.) *Fuzzy Information and Decision Processes,* (North-Holland, Amsterdam), 1977.

[9] I.B. Turksen, Approximate reasoning for production planning, *Fuzzy Sets and Systems,* 26, 23-37, 1988.

[10] L. A. Zadeh, A theory of approximate reasoning, Memorandum No, UCB/ERLM77/58, 1977.

[11] D. Dubois, W. Ostasiewicz, H. Prade, Fuzzy Sets: History and Basic Notions, in D. Dubois, H. Prade (Eds), *Fundamentals of Fuzzy Sets*, Kluwer, Boston, MA, Pages 21-124, 2000.

[12] L. A. Zadeh, Fuzzy sets, *Information Control*, 8, Pages 338-353, 1965.