

# Reglas de asociación

## Inteligencia Artificial

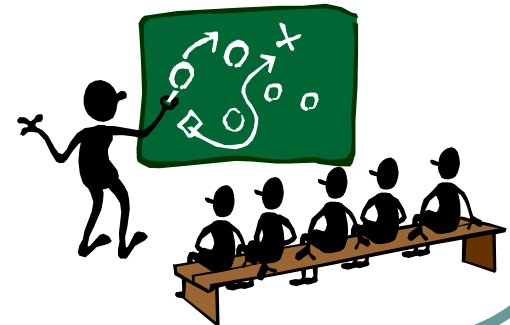
Prof. Dr. Ariel Monteserin

[ariel.monteserin@isistan.unicen.edu.ar](mailto:ariel.monteserin@isistan.unicen.edu.ar)

ISISTAN - UNICEN

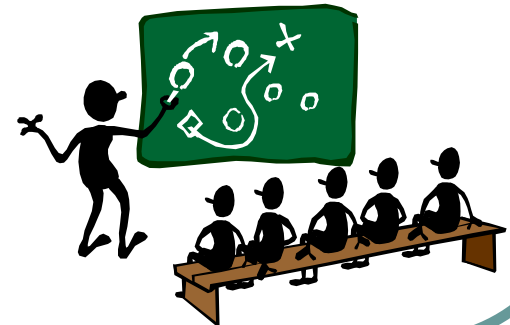
# Agenda

- Introducción ML y Conceptos
- Algoritmos de descubrimientos de reglas de asociación
- Post-procesamiento
- Actualización incremental de reglas
- Ejemplos de aplicación en IA
- Software



# Agenda

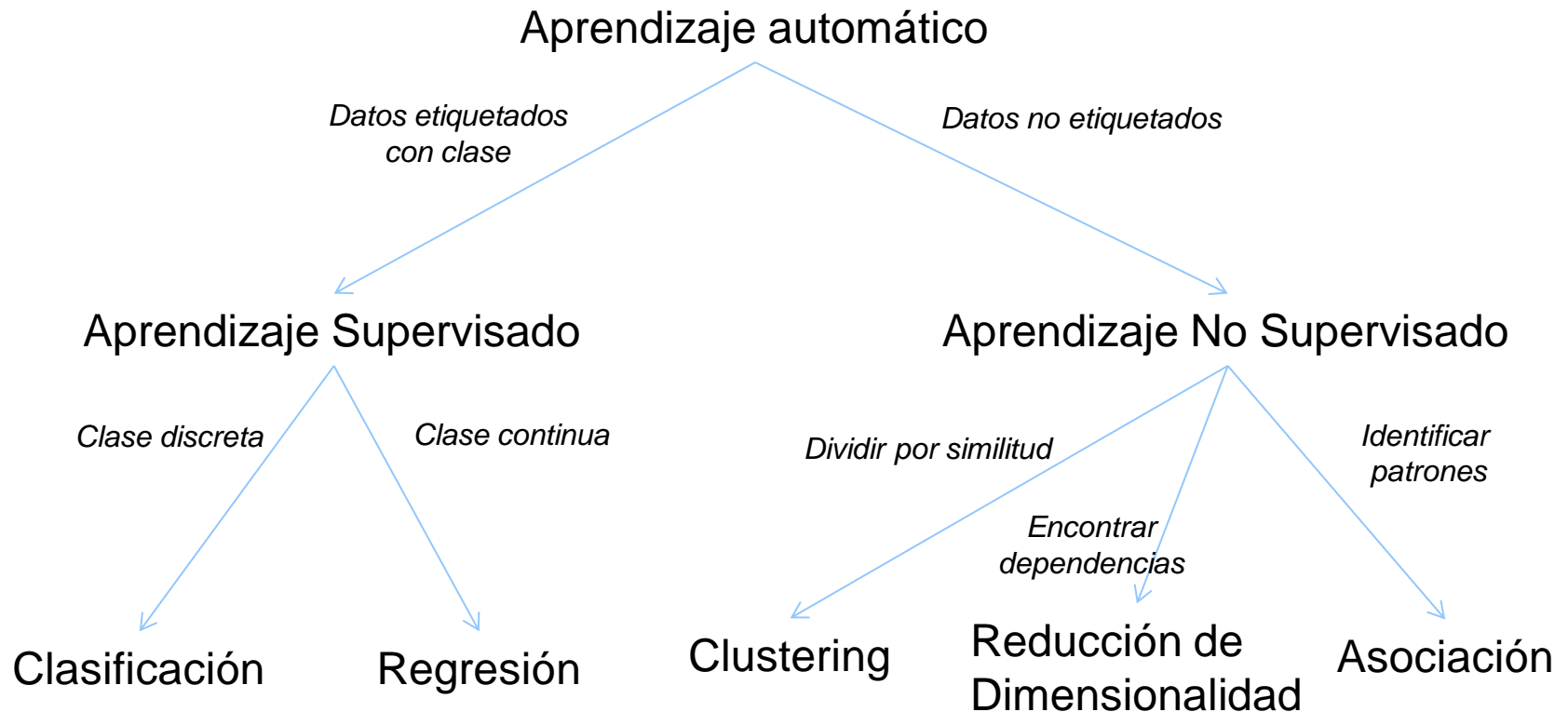
- **Introducción ML y Conceptos**
- Algoritmos de descubrimientos de reglas de asociación
- Post-procesamiento
- Actualización incremental de reglas
- Ejemplos de aplicación en IA
- Software



# Aprendizaje automático

- Formas principales de adquirir experiencia:
  - A partir de ejemplos ya clasificados o etiquetados
    - Aprendizaje Supervisado
  - Mediante la exploración autónoma de los datos disponibles
    - Aprendizaje No Supervisado

# Aprendizaje automático



# Introducción

- Reglas de asociación es una técnica de inteligencia artificial ampliamente utilizada en Data Mining.
- Data Mining es el proceso de descubrimiento de tendencias o patrones en grandes bases de dato con el objetivo de guiar futuras decisiones.



# Regla de asociación

- Regla de asociación
  - Describe una relación de asociación entre los elementos de un conjunto de datos relevantes.
  - Ejemplos
    - Estudiantes que cursan *Inteligencia Artificial* tienden a cursar *Taller de Sistemas Multiagentes*.
    - Clientes que adquieren un *producto lácteo* tienden a comprar un *producto de panificados*.
    - Artículos que referencian *Srikant & Agrawal (1997)* referencian también a *Agrawal et al. (1993)*

# Origen: *Market Basket Analysis*

- Se denomina Canasta de Mercado (*Market Basket*) a una colección de ítems que un cliente compra en una misma transacción.
- Problema: identificar el conjunto de ítems que son adquiridos en conjunto.
- Intenta identificar reglas de la forma:
  - {fideos, queso rallado} → {salsa}
  - Significado: si son comprados fideos y queso rallado en una transacción, es probable que también sea comprada salsa.





# Definición: regla de asociación

- Forma general:  $X \rightarrow Y$ , donde  $X$  e  $Y$  son conjuntos de ítems.
- $X$  es denominado el **antecedente** de la regla e  $Y$  su **consecuente**.
  - **Soporte:** El soporte para  $X \rightarrow Y$  es el porcentaje de las transacciones que contienen todos los ítems de  $X$  e  $Y$ .
  - **Confianza:** la confianza para  $X \rightarrow Y$  es el porcentaje de transacciones que contienen  $Y$ , entre las transacciones que contienen  $X$ .

# Soporte y confianza

- $\text{Soporte}(X \rightarrow Y) = \text{Prob}(X \cup Y) = \text{Soporte}(X \cup Y)$
- $\text{Confianza}(X \rightarrow Y) = \text{Prob}(Y / X) = \frac{\text{Soporte}(X \cup Y)}{\text{Soporte}(X)}$

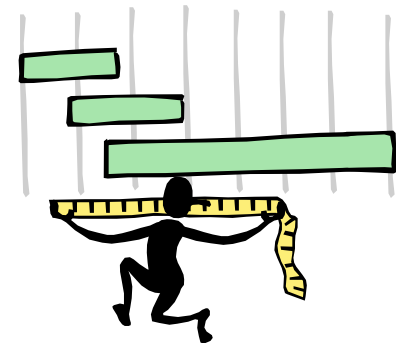
Transacciones
A B C
B C
A C
A C D

Soporte ( $A \rightarrow C$ ):

0,75

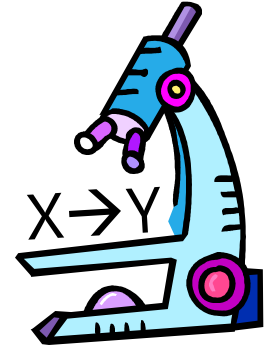
Confianza ( $A \rightarrow C$ ):

1



# Interpretación de las métricas

- Regla con bajo soporte
  - puede haber aparecido por casualidad.
- Regla con baja confianza
  - es probable que no exista relación entre antecedente y consecuente.
- ¿Qué diferencia a  $X \rightarrow Y$  de  $Y \rightarrow X$ ?
  - Tienen el mismo soporte pero distinta confianza.
  - $\{\text{Fernet}\} \rightarrow \{\text{Coca Cola}\}$  vs.  $\{\text{Coca Cola}\} \rightarrow \{\text{Fernet}\}$



# Diferentes tipos de reglas de asociación

- Reglas de asociación
  - Booleanas vs. Cuantitativas
  - Unidimensionales vs. Multidimensionales
  - Un nivel vs. Multinivel

# Diferentes tipos de reglas de asociación

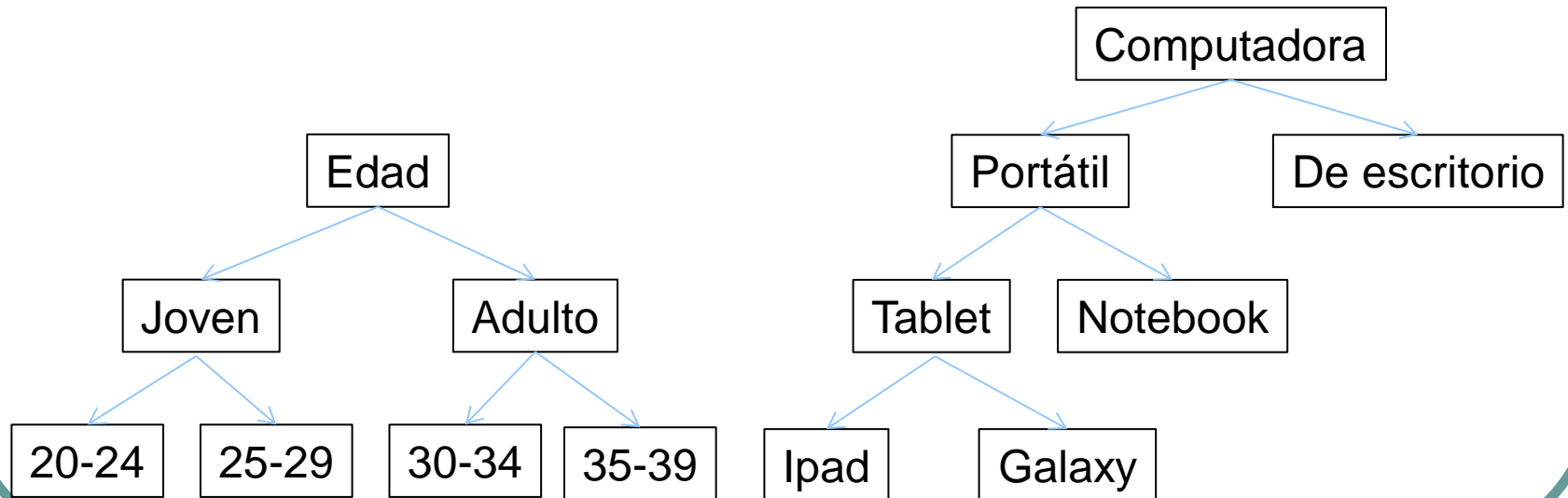
- Reglas de asociación booleanas vs. cuantitativas
  - Basadas en el tipo de valores manejados
    - $\{Compra(X, \text{"computadora"})\} \rightarrow Compra(X, \text{"impresora"})$
    - $\{Edad(X, \text{"30...34"}) \wedge Sueldo(X, \text{"5K...10K"})\} \rightarrow Compra(X, \text{"TV alta resolución"})$

# Diferentes tipos de reglas de asociación

- Reglas unidimensionales vs. multidimensionales
  - Según las *dimensiones* de los datos involucrados
    - $\{Compra(X, \text{"computadora"})\} \rightarrow Compra(X, \text{"impresora"})$
    - $\{Edad(X, \text{"30...34"}) \wedge Sueldo(X, \text{"5K...10K"})\} \rightarrow Compra(X, \text{"TV alta resolución"})$

# Diferentes tipos de reglas de asociación

- Reglas de un nivel vs. multinivel
  - Según el nivel de abstracción involucrado
    - Edad(X, "30-34") → Compra (X, "Galaxy")
    - Edad(X, "30-34") → Compra (X, "Tablet")



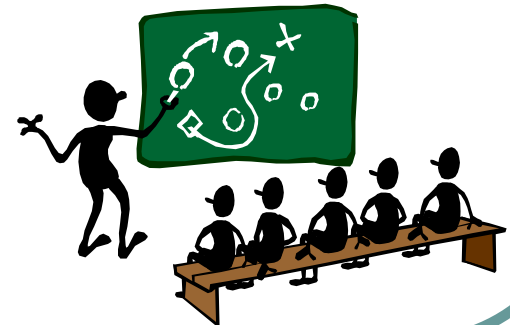
# Otros ejemplos

- Ambient Intelligent
  - Alarma despertador → Enciende cafetera
  - Temperatura baja → Enciende calefacción
- Recomendación
  - El Señor de los Anillos → El Hobbit
- Interacción usuario-computadora
  - Mensaje no importante, en reunión → No notificar
- Fútbol
  - Posición X, Muchos rivales → Remate

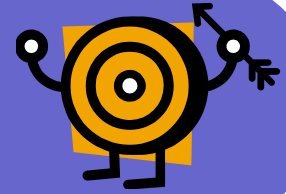


# Agenda

- Introducción ML y Conceptos
- **Algoritmos de descubrimientos de reglas de asociación**
- Post-procesamiento
- Actualización incremental de reglas
- Ejemplos de aplicación en IA
- Software



# Objetivo



- Encontrar **reglas de asociación** con valores de *soporte* y *confianza* altos.
  - Umbrales *minsup* y *minconf* definidos por el usuario.



**Recordar:** Encontrar dichas reglas no significa que deba existir una relación entre antecedente y consecuente. Por lo tanto, un experto debería evaluarlas.

# Definición formal del problema

- $I = \{ i_1, i_2, \dots, i_m \}$  es un conjunto de ítems.
- $D$  es un conjunto de transacciones  $T_j$ .
- Cada transacción  $T_j$  es un conjunto de ítems (subconjunto de  $I$ ).

# Ejemplo

TID	Transacciones
1	A C D
2	B C E
3	A B C E
4	B E
5	A B C E

- $I = \{A, B, C, D, E\}$
- $D = \{1, 2, 3, 4, 5\} = \{\{A, C, D\}, \{B, C, E\}, \{A, B, C, E\}, \{B, E\}, \{A, B, C, E\}\}$

# Itemsets

TID	Items
1	A C D
2	B C E
3	A B C E
4	B E
5	A B C E

- Un itemset es un conjunto de ítems.
- Si  $X$  es un **itemset**,  $X \subseteq I$ .
- Un **itemset** que contiene  $k$  ítems es llamado **k-itemset**.
- Ej.  $\{A,B\}$  es un 2-itemset



El soporte de un itemset  $X$  es el porcentaje de transacciones en  $D$  que contienen  $X$

$$\text{Soporte}(X) = \frac{|\{T \in D / X \subseteq T\}|}{|D|}$$

$$\text{Soporte}(\{A,B\}) = 2/5 = 0.4$$

# Descomposición del problema

- El descubrimiento de reglas de asociación puede ser descompuesto en dos sub-problemas:
  1. Encontrar todos los **itemsets** que tienen soporte mayor que el **soporte mínimo (itemsets frecuentes)**.
  2. Utilizar los itemsets frecuentes para generar las reglas deseadas.
- Por cada itemset frecuente  $L$  ( $k > 1$ ), encontrar todos los subconjuntos no vacíos, y para cada subconjunto  $\{a\}$  generar una regla  $\{a\} \rightarrow \{L-a\}$  si la confianza es mayor que la **confianza mínima**.
  - Para el itemset frecuente  $\{A, B, C\}$ :
    - $\{A\} \rightarrow \{BC\}$ ,  $\{AB\} \rightarrow \{C\}$ ,  $\{AC\} \rightarrow \{B\}$ ,  $\{B\} \rightarrow \{AC\}$ ,  $\{BC\} \rightarrow \{A\}$ ,  $\{C\} \rightarrow \{AB\}$



# Algoritmos

- Apriori y AprioriTid (Agrawal & Srikant, 1994)
- Opus (Webb, 1996)
- Direct Hasing and Pruning (DHP) (Adamo, 2001)
- Dynamic Set Counting (DIC) (Adamo, 2001)
- Charm (Zaki & Hsiao, 2002)
- FP-growth (Han, Pei & Yin, 1999)
- Closet (Pei, Han & Mao, 2000)
- .....



# Algoritmos

- Los diferentes algoritmos deben siempre generar el mismo conocimiento.
- ¿Qué los hace diferentes?
  - Forma en que los datos son cargados en memoria.
  - Tiempo de procesamiento.
  - Tipos de atributos (numéricos, categóricos).
  - Forma en que los itemsets son generados.
  - Estructura de datos utilizada.





# Ejemplo

TID	ítems
111	lapicera, tinta, agenda, jabón
112	lapicera, tinta, agenda
113	lapicera, agenda
114	lapicera, tinta, jabón, arroz

- Soporte mínimo = 0.7
- Nivel 1: Encontrar 1-itemsets frecuentes
  - {lapicera}, {tinta}, {agenda}, ~~{jabón}~~, ~~{arroz}~~

# Ejemplo

TID	ítems
111	lapicera, tinta, agenda, jabón
112	lapicera, tinta, agenda
113	lapicera, agenda
114	lapicera, tinta, jabón, arroz

- Nivel 2: Chequear cada uno de los 2-itemsets
  - {lapicera, tinta}, {lapicera, agenda}, ~~{lapicera, jabón},~~  
~~{lapicera, arroz}, {tinta, agenda}, {tinta, jabón},~~  
~~{tinta, arroz}, {agenda, jabón}, {agenda, arroz}...~~

# Ejemplo

TID	ítems
111	lapicera, tinta, agenda, jabón
112	lapicera, tinta, agenda
113	lapicera, agenda
114	lapicera, tinta, jabón, arroz

- Nivel 3: Chequear cada uno de los 3-itemsets
  - ~~{lapicera, tinta, agenda}, {lapicera, tinta, jabón}, {lapicera, tinta, arroz}, {lapicera, agenda, jabón}, {lapicera, agenda, arroz} ...~~
- Nivel 4: Chequear cada uno de los 4-itemsets
  - ~~{lapicera, tinta, agenda, jabón} {lapicera, tinta, agenda, arroz} {tinta, agenda, arroz, jabón}...~~

# Ejemplo

TID	ítems
111	lapicera, tinta, agenda, jabón
112	lapicera, tinta, agenda
113	lapicera, agenda
114	lapicera, tinta, jabón, arroz

- Los itemsets frecuentes son:
  - {lapicera}, {tinta}, {agenda},
  - {lapicera, tinta}, {lapicera, agenda}

# Refinamiento



- ¿Qué relación existe entre los  $n$ -itemsets y los  $n+1$ -itemsets eliminados?
  - Nivel 1: Encontrar 1-itemsets frecuentes
    - {lapicera}, {tinta}, {agenda}, {jabón}, {arroz}
  - Nivel 2: Chequear cada uno de los 2-itemsets
    - {lapicera, tinta}, {lapicera, agenda}, {lapicera, jabón}, {lapicera, arroz}, {tinta, agenda}, {tinta, jabón}, {tinta, arroz}, {agenda, jabón}, {agenda, arroz}
  - Nivel 3: Chequear cada uno de los 3-itemsets
    - {lapicera, tinta, agenda}, {lapicera, tinta, jabón}, {lapicera, tinta, arroz}, {lapicera, agenda, jabón}, {lapicera, agenda, arroz}
- **Refinamiento:** extender los itemsets frecuentes de una forma que asegure que todos sus subconjuntos son itemsets frecuentes.



# Propiedad Apriori

- **La propiedad Apriori:** cada subconjunto de un itemset frecuente debe ser también un itemset frecuente.
- Podemos crear itemsets frecuentes iterativamente, tomando los itemsets frecuentes de tamaño  $n$  y extendiéndolos a itemsets frecuentes de tamaño  $n+1$ .



# Algoritmo Apriori

1. Se calcula el soporte de cada ítem individual, y se determinan los 1-itemsets frecuentes.
2. En cada paso subsecuente, los *itemsets frecuentes* generados en los pasos anteriores se utilizan para generar los nuevos itemsets (*itemsets candidatos*).
3. Se calcula el *soporte* de cada *itemset candidato* y se determinan los *itemsets frecuentes*.
4. El proceso continúa hasta que no pueden ser encontrados nuevos *itemsets frecuentes*.

# Ejemplo

**D**

TID	Items
100	1 3 4
200	2 3 5
300	1 2 3 5
400	2 5

**minsup= 0.5**

**L<sub>2</sub>**

Itemset	Soporte
{1, 3}	2/4
{2, 3}	2/4
{2, 5}	3/4
{3, 5}	2/4

**L<sub>1</sub>**

Itemset	Soporte
{1}	2/4
{2}	3/4
{3}	3/4
{5}	3/4

**C<sub>3</sub>**

Itemset	Soporte
{2, 3, 5}	2/4

**C<sub>2</sub>**

Itemset	Soporte
{1, 2}	1/4
{1, 3}	2/4
{1, 5}	1/4
{2, 3}	2/4
{2, 5}	3/4
{3, 5}	2/4

**L<sub>3</sub>**

Itemset	Soporte
{2, 3, 5}	2/4



# Derivación de reglas de asociación

- Para cada itemset frecuente  $I$ , generamos todos los subconjuntos no vacíos de  $I$ .
- Para cada subconjunto  $a \subset I$ , generamos una regla de la forma  $a \rightarrow (I-a)$  si la tasa entre ***soporte***( $I$ ) y ***soporte***( $a$ ) es al menos la **confianza mínima** (minconf).



# Ejemplo

## Itemsets frecuentes

Itemset	Soporte
{1, 3}	2/4
{2, 3}	2/4
{2, 5}	3/4
{3, 5}	2/4
{2, 3, 5}	2/4

**L<sub>1</sub>**

Itemset	Soporte
{1}	2/4
{2}	3/4
{3}	3/4
{5}	3/4

## Reglas encontradas

Regla	Conf.	Regla	Conf
1 → 3		2,5 → 3	
2 → 3,5		2 → 5	
3 → 1		5 → 2,3	
2,3 → 5		5 → 2	
2 → 3		3 → 5	
3 → 2,5		5 → 3	
3 → 2		3,5 → 2	

$$\text{Confianza}(X \rightarrow Y) = \frac{\text{Soporte}(X, Y)}{\text{Soporte}(X)}$$



# Ejemplo

## Itemsets frecuentes

Itemset	Soporte
{1, 3}	2/4
{2, 3}	2/4
{2, 5}	3/4
{3, 5}	2/4
{2, 3, 5}	2/4

**L<sub>1</sub>**

Itemset	Soporte
{1}	2/4
{2}	3/4
{3}	3/4
{5}	3/4

## Reglas encontradas

Regla	Conf.	Regla	Conf
1 → 3	2/2	2,5 → 3	
2 → 3,5		2 → 5	
3 → 1		5 → 2,3	
2,3 → 5		5 → 2	
2 → 3		3 → 5	
3 → 2,5		5 → 3	
3 → 2		3,5 → 2	

$$\text{Confianza}(X \rightarrow Y) = \frac{\text{Soporte}(X, Y)}{\text{Soporte}(X)}$$



# Ejemplo

## Itemsets frecuentes

Itemset	Soporte
{1, 3}	2/4
{2, 3}	2/4
{2, 5}	3/4
{3, 5}	2/4
{2, 3, 5}	2/4

**L<sub>1</sub>**

Itemset	Soporte
{1}	2/4
{2}	3/4
{3}	3/4
{5}	3/4

## Reglas encontradas

Regla	Conf.	Regla	Conf
1 → 3	2/2	2,5 → 3	
2 → 3,5	2/3	2 → 5	
3 → 1		5 → 2,3	
2,3 → 5		5 → 2	
2 → 3		3 → 5	
3 → 2,5		5 → 3	
3 → 2		3,5 → 2	

$$\text{Confianza}(X \rightarrow Y) = \frac{\text{Soporte}(X, Y)}{\text{Soporte}(X)}$$



# Ejemplo

## Itemsets frecuentes

Itemset	Soporte
{1, 3}	2/4
{2, 3}	2/4
{2, 5}	3/4
{3, 5}	2/4
{2, 3, 5}	2/4

**L<sub>1</sub>**

Itemset	Soporte
{1}	2/4
{2}	3/4
{3}	3/4
{5}	3/4

## Reglas encontradas

Regla	Conf.	Regla	Conf
1 → 3	2/2	2,5 → 3	2/3
2 → 3,5	2/3	2 → 5	3/3
3 → 1	2/3	5 → 2,3	2/3
2,3 → 5	2/2	5 → 2	3/3
2 → 3	2/3	3 → 5	2/3
3 → 2,5	2/3	5 → 3	2/3
3 → 2	2/3	3,5 → 2	2/2

$$\text{Confianza}(X \rightarrow Y) = \frac{\text{Soporte}(X, Y)}{\text{Soporte}(X)}$$

minconf = 0.9



# Ejemplo

## Itemsets frecuentes

Itemset	Soporte
{1, 3}	2/4
{2, 3}	2/4
{2, 5}	3/4
{3, 5}	2/4
{2, 3, 5}	2/4

**L<sub>1</sub>**

Itemset	Soporte
{1}	2/4
{2}	3/4
{3}	3/4
{5}	3/4

## Reglas encontradas

Regla	Conf.	Regla	Conf
<b>1 → 3</b>	<b>2/2</b>	2,5 → 3	2/3
2 → 3,5	2/3	<b>2 → 5</b>	<b>3/3</b>
3 → 1	2/3	5 → 2,3	2/3
<b>2,3 → 5</b>	<b>2/2</b>	<b>5 → 2</b>	<b>3/3</b>
2 → 3	2/3	3 → 5	2/3
3 → 2,5	2/3	5 → 3	2/3
3 → 2	2/3	<b>3,5 → 2</b>	<b>2/2</b>

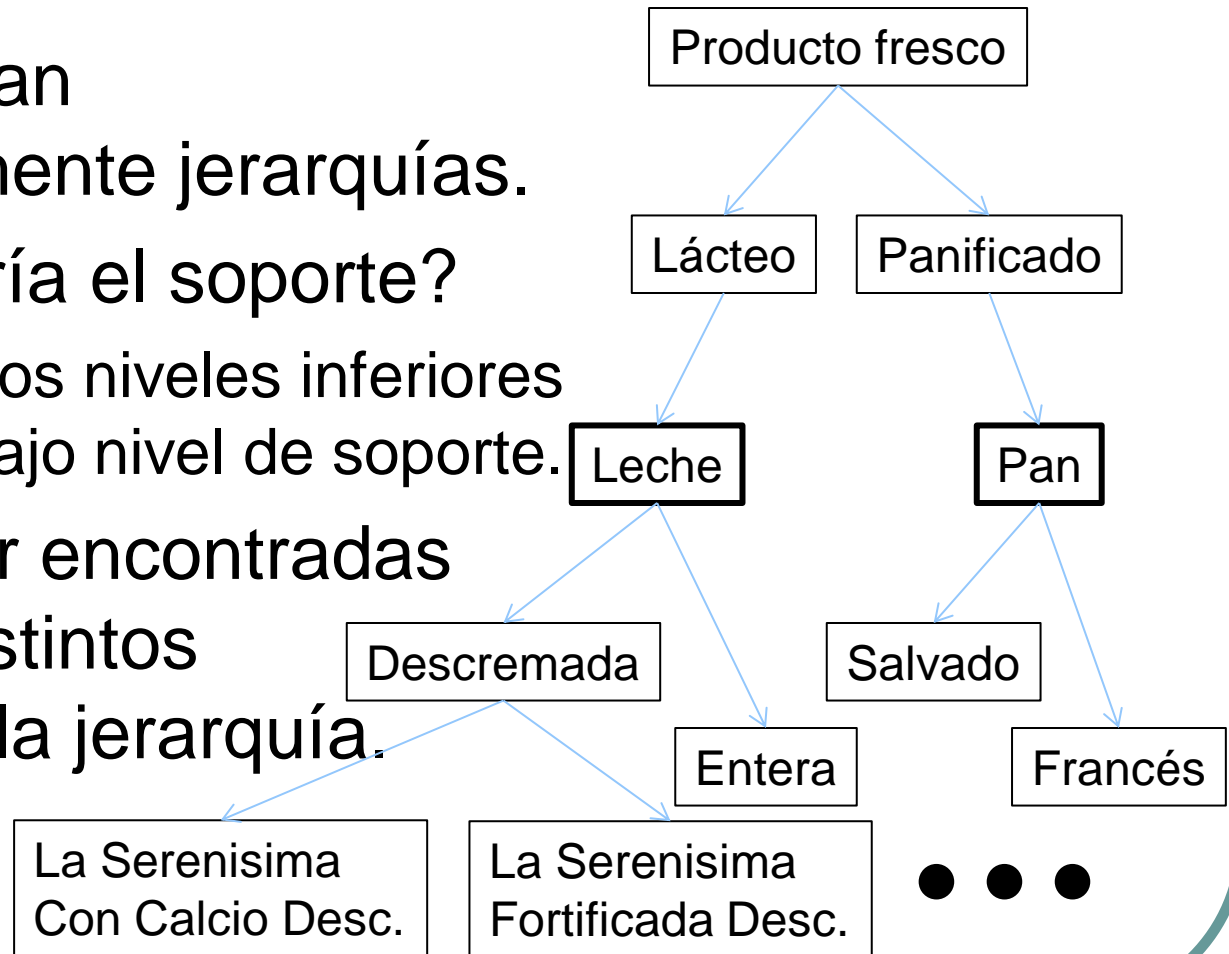
$$\text{Confianza}(X \rightarrow Y) = \frac{\text{Soporte}(X, Y)}{\text{Soporte}(X)}$$

$$\text{minconf} = 0.9$$



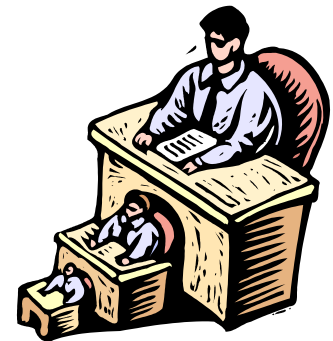
# Reglas de asociación multinivel (generalizadas)

- Ítems forman frecuentemente jerarquías.
- ¿Cómo varía el soporte?
  - Ítems de los niveles inferiores tendrán bajo nivel de soporte.
- Pueden ser encontradas reglas a distintos niveles de la jerarquía.



# Descubrimiento de reglas de asociación multinivel

- Enfoque de profundidad progresiva
  - Primero encontrar reglas en los niveles altos
    - Leche  $\rightarrow$  Pan [0.2, 0.6]
  - Luego, encontrar las reglas de bajo nivel correspondientes
    - Leche descremada  $\rightarrow$  Pan de salvado [0.06, 0.5]



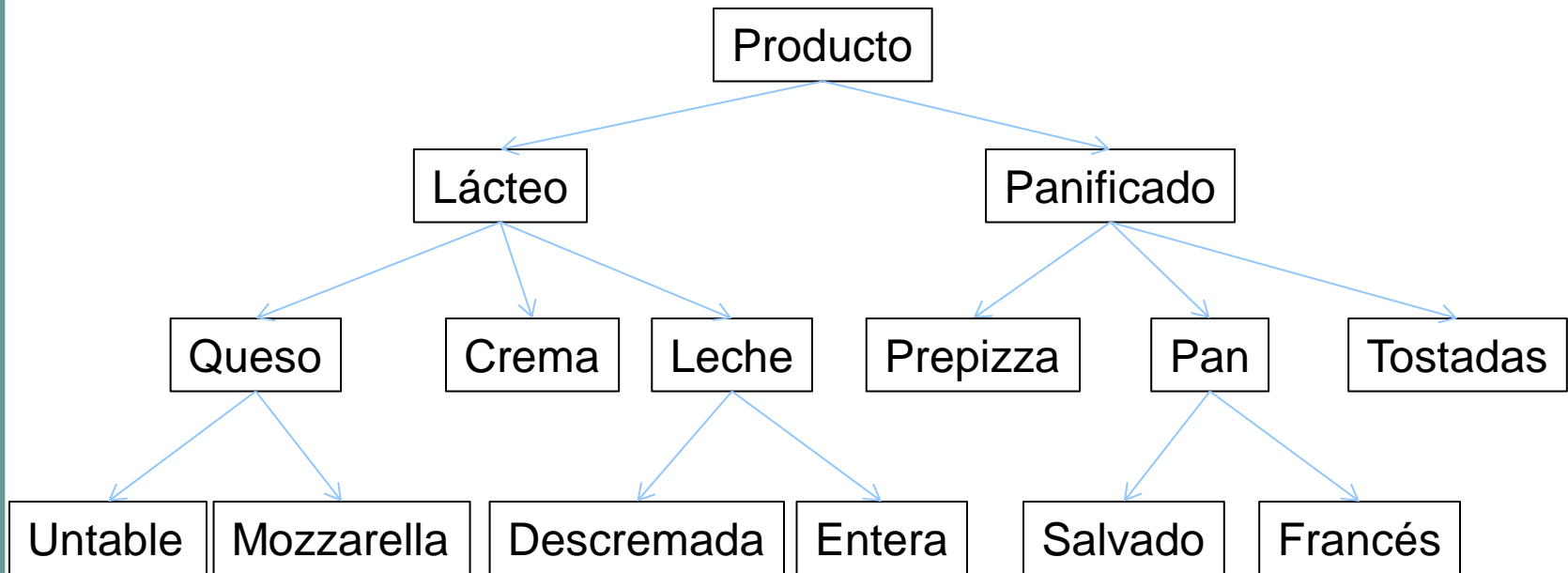


# Algoritmo *Basic* (Srikant y Agrawal, 1997)

- Extiende el algoritmo *Apriori*.
- Una regla de asociación generalizada  $X \Rightarrow Y$  es definida idénticamente a una regla de asociación regular, excepto que ningún ítem de  $Y$  puede ser ancestro de un ítem de  $X$ .
- Agregar todos los ancestros de cada ítem en  $t$  a  $t$ , removiendo duplicados.



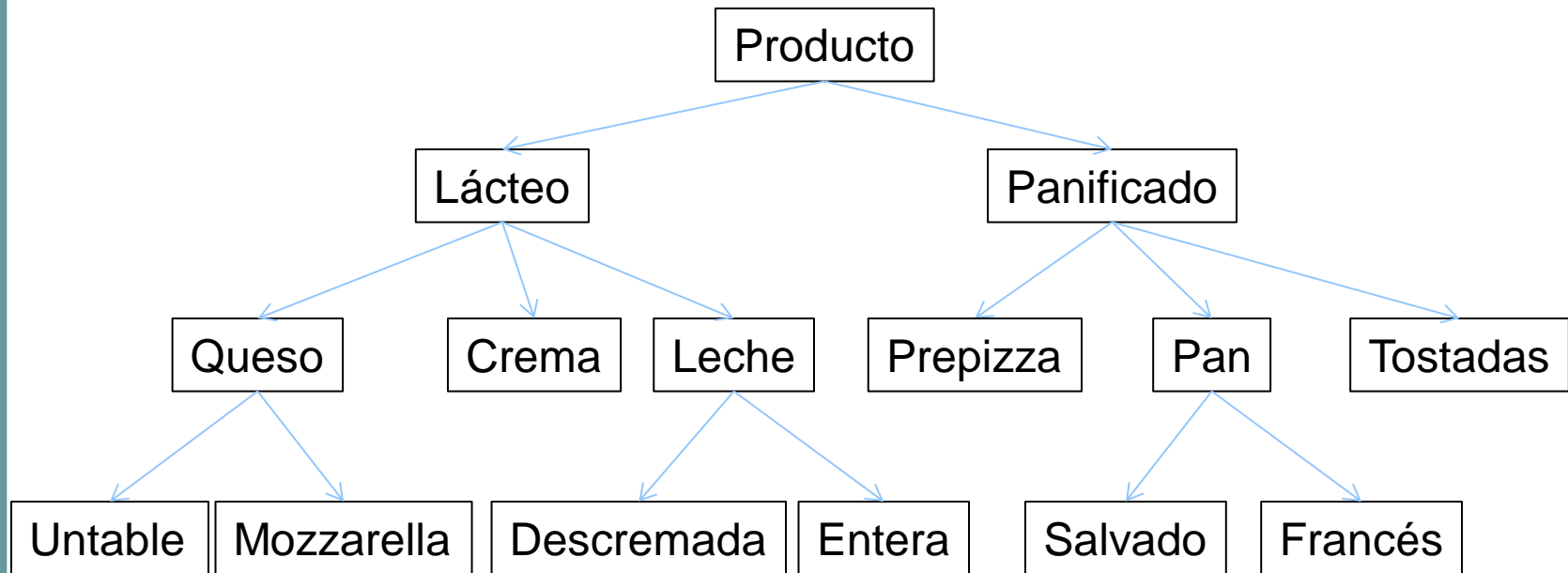
# Ejemplo multinivel



TID	Transacciones
111	untable, l. entera, francés, jabón
112	mozzarella, prepizza
113	l. descremada, salvado
114	francés, untable, mermelada



# Ejemplo multinivel



TID	Transacciones extendidas
111	untable, l. entera, francés, jabón, Queso, Lácteo, Leche, Pan, Panificado
112	mozzarella, prepizza, Queso, Lácteo, Panificado
113	l. descremada, salvado, Leche, Lácteo, Pan, Panificado
114	francés, untable, mermelada, Pan, Panificado, Queso, Lácteo

# Ejemplo multinivel

TID	Transacciones extendidas
111	untable, l. entera, francés, jabón, Queso, Lácteo, Leche, Pan, Panificado
112	mozzarella, prepizza, Queso, Lácteo, Panificado
113	l. descremada, salvado, Leche, Lácteo, Pan, Panificado
114	francés, untable, mermelada, Pan, Panificado, Queso, Lácteo

Itemset	Sop.	Itemset	Sop.	Itemset	Sop.
untable	2/4	prepizza	1/4	Crema	0/4
l. entera	1/4	l. descrem.	1/4	Leche	2/4
francés	2/4	salvado	1/4	Pan	3/4
jabón	1/4	mermelada	1/4	Lácteo	4/4
mozzarella	1/4	Queso	3/4	Panificado	4/4

**minsup = 0.7**

# Ejemplo multinivel

TID	Transacciones extendidas
111	untable, l. entera, francés, jabón, Queso, Lácteo, Leche, Pan, Panificado
112	mozzarella, prepizza, Queso, Lácteo, Panificado
113	l. descremada, salvado, Leche, Lácteo, Pan, Panificado
114	francés, untable, mermelada, Pan, Panificado, Queso, Lácteo

Itemset	Sop.	Itemset	Sop.	Itemset	Sop.
untable	2/4	prepizza	1/4	Crema	0/4
l. entera	1/4	l. descrem.	1/4	Leche	2/4
francés	2/4	salvado	1/4	<b>Pan</b>	<b>3/4</b>
jabón	1/4	mermelada	1/4	<b>Lácteo</b>	<b>4/4</b>
mozzarella	1/4	<b>Queso</b>	<b>3/4</b>	<b>Panificado</b>	<b>4/4</b>

minsup = 0.7

# Ejemplo multinivel

<b>C<sub>2</sub></b>	Itemset	Sop.
	Queso, Panificado	3/4
	Pan, Lácteo	3/4
	<del>Pan, Panificado</del>	3/4
	Lácteo, Panificado	4/4
	<del>Queso, Pan</del>	2/4
	<del>Queso, Lácteo</del>	2/4

TID	Transacciones extendidas
111	untable, l. entera, francés, jabón, Queso, Lácteo, Leche, Pan, Panificado
112	mozzarella, prepizza, Queso, Lácteo, Panificado
113	l. descremada, salvado, Leche, Lácteo, Pan, Panificado
114	francés, untable, mermelada, Pan, Panificado, Queso, Lácteo

# Ejemplo multinivel

**L<sub>2</sub>**

Itemset	Sop.
Queso, Panificado	3/4
Pan, Lácteo	3/4
Lácteo, Panificado	4/4

**C<sub>3</sub>**

Itemset	Sop.

TID	Transacciones extendidas
111	untable, l. entera, francés, jabón, Queso, Lácteo, Leche, Pan, Panificado
112	mozzarella, prepizza, Queso, Lácteo, Panificado
113	l. descremada, salvado, Leche, Lácteo, Pan, Panificado
114	francés, untable, mermelada, Pan, Panificado, Queso, Lácteo

# Ejemplo multinivel

Regla	C	Regla	C
<b>Queso → Panificado</b>	<b>3/3</b>	Lácteo → Pan	3/4
Panificado → Queso	3/4	<b>Lácteo → Panificado</b>	<b>4/4</b>
<b>Pan → Lácteo</b>	<b>3/3</b>	<b>Panificado → Lácteo</b>	<b>4/4</b>

**L<sub>1</sub>**

Itemset	Sop.
Pan	3/4
Lácteo	4/4
Panificado	4/4
Queso	3/4

**L<sub>2</sub>**

Itemset	Sop.
Queso, Panificado	3/4
Pan, Lácteo	3/4
Lácteo, Panificado	4/4

**minconf = 0.9**



# Atributos cuantitativos

- Ejemplos anteriores: booleanos
- Atributos cuantitativos
  - Transformar a categóricos
    - Crear intervalos discretos
  - Por ejemplo, edad, altura, ingresos.
    - Niño (0..10), Adolescente (11..18), Joven (19..30), Adulto (31..50)
    - Juan: 30, Pedro: 31 se convierte en Juan: Joven, Pedro: Adulto
- Problema de la frontera

# Problema de la frontera

Nombre	Edad	Estado civil	Viaja
Juan	30	Soltero	Si
Pedro	31	Casado	Si
Guillermo	31	Soltero	Si
Rodrigo	30	Casado	Si
Joaquín	18	Soltero	No

# Problema de la frontera

Nombre	Edad	Estado civil	Viaja
Juan	Joven (30)	Soltero	Si
Pedro	Adulto (31)	Casado	Si
Guillermo	Adulto (31)	Soltero	Si
Rodrigo	Joven (30)	Casado	Si
Joaquín	Adolescente	Soltero	No

Minsup = 0,5

**L1: {EC = Soltero}, {Viaja=Si}**

**L2: {}**

# Problema de la frontera

Nombre	Edad	Estado civil	Viaja
Juan	Joven (30)	Soltero	Si
Pedro	Joven (31)	Casado	Si
Guillermo	Joven (31)	Soltero	Si
Rodrigo	Joven (30)	Casado	Si
Joaquín	Adolescente	Soltero	No

Minsup = 0,5

**L1: {Edad = Joven },{EC = Soltero},{Viaja=Si}**

**L2: ({Edad=Joven}{Viaja=Si})**

# Solución al problema de la frontera

- Solapamiento de intervalos

- Niño (0..12), Adolescente (8..20), Joven (16,32), Adulto (28,50)
- Juan: 30, Pedro: 31 → Juan, Pedro: Joven y Adulto

- Conjuntos difusos (*fuzzy set*)

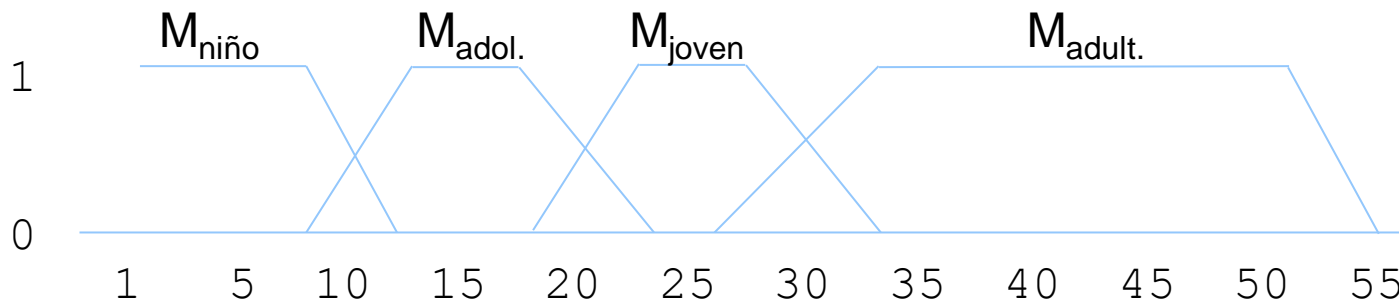
- Un elemento puede pertenecer a un conjunto con un valor de membresía en  $[0,1]$
- Función de membresía

$$m_{f_x}(x) = D_x \rightarrow [0,1]$$



# Reglas de asociación difusas

- Dada una base transaccional  $D = \{t_1, \dots, t_n\}$  y un conjunto de atributos o ítems  $i = \{i_1, \dots, i_m\}$  presentes en las transacciones almacenadas en  $D$ .
- Cada ítem  $i_j$  tendrá asociado una serie de regiones difusas  $F_{ij} = \{f_{ij}^1, \dots, f_{ij}^p\}$ 
  - $F_{\text{Edad}} = \{\text{niño}, \text{adolescente}, \text{joven}, \text{adulto}\}$



# Reglas de asociación difusas

- Entonces, un algoritmo de reglas de asociación difusas será capaz de encontrar reglas de la siguiente forma:
  - $\langle X, A \rangle \rightarrow \langle Y, B \rangle$  en donde
    - X e Y son conjuntos de atributos
    - A y B contienen las regiones difusas asociadas con los correspondientes atributos de X e Y .
      - Si **Edad** es **Joven**  $\rightarrow$  **Ingreso** es **Medio**

# Soporte difuso

- Soporte difuso de un itemset

$$FSop(\langle X, A \rangle) = \frac{\sum_{t_i \in D} \prod_{x_j \in X} \mu_{a_j}^i(x_j)}{|D|}$$

donde  $\mu_{a_j}^i(x_j)$  es el grado de membresía del atributo  $x_j$  en la transacción  $i$ -ésima a la región difusa  $a_j$



# Soporte y confianza de una regla difusa

- Soporte de una regla difusa

$$FSop(\langle X, A \rangle \rightarrow \langle Y, B \rangle) = FSop(X \cup Y, A \cup B)$$

- Confianza de una regla difusa

$$FConf(\langle X, A \rangle \rightarrow \langle Y, B \rangle) = \frac{FSop(X \cup Y, A \cup B)}{FSop(\langle X, A \rangle)}$$

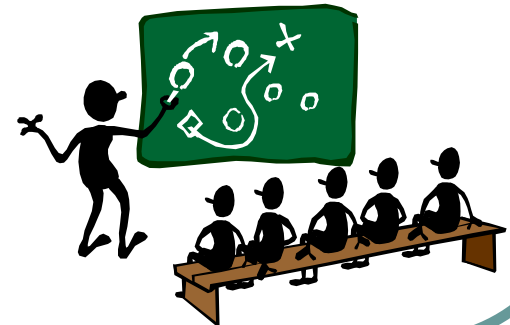
# Algoritmos de descubrimiento de reglas de asociación difusas

- Reglas de asociación difusas (Kuok et al., 1998)
- Reglas de asociación difusas y generalizadas (Hong et al., 2003)
- Survey (Delgado et al., 2003)



# Agenda

- Introducción ML y Conceptos
- Algoritmos de descubrimientos de reglas de asociación
- **Post-procesamiento**
- Actualización incremental de reglas
- Ejemplos de aplicación
- Software



# Post-procesamiento

- Numerosas reglas son generadas en el proceso de descubrimiento
  - Reglas triviales
    - “Preñada” → “Hembra” [0.2, 1.0]
    - “Fernet” → “Coca Cola” [0.15, 0.95]
  - Reglas redundantes
    - “Alta velocidad” → “Tuvo un accidente” [0.1, 0.4]
    - “Alta velocidad”, “Nacido en MDP” → “Tuvo un accidente” [0.09, 0.42]



# Tareas de post-procesamiento

- Filtrar reglas no interesantes.
- Filtrar reglas insignificantes.
- Eliminar reglas redundantes.

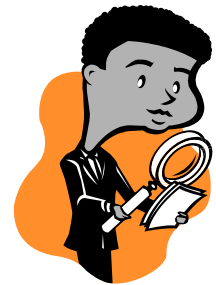
# Filtrado de reglas no interesantes

- El estudio del interés de las reglas de asociación apunta a presentar las reglas que son interesantes para el usuario.
- Relacionado al estudio de reglas de asociación *imprevisibles* o *inesperadas*.



# Reglas interesantes: dos enfoques

- Métricas objetivas (basada en los datos)
  - Reglas son ordenadas según un sistema de ranking predefinido.
  - Reglas son filtradas según un conjunto de reglas de podado predefinidas.
- Métricas subjetivas (basada en el usuario)
  - Se requiere a los usuarios especificar si las reglas son interesantes.
    - Es imposible hacerlo regla por regla.
    - Por lo tanto, las reglas deben ser procesadas conjuntamente.



# Métricas objetivas

- Lift (a.k.a interest and strength) (Brin et al., 1997; Dhar & Tuzhilin, 1993)
- Conviction (Brin et al., 1997)
- Gain (Fukuda et al., 1996)
- Chi-squared value (Morimoto et al., 1998)
- Laplace (Webb, 1995)
- Gini (Morimoto et al., 1998)
- Piatetsky-Shapiro (Piatetsky & Shapiro, 1991)
- Entropy gain (Morimoto et al., 1998)



# Problemas del framework soporte-confianza

- Definimos:
  - $P(X)$ : probabilidad de que una transacción  $T$  de la base  $D$  contenga el itemset  $X$ .
  - $P(X, Y)$ : probabilidad que ambos  $X$  e  $Y$  sean contenidos por  $T$  en  $D$ .
- Si suponemos que  $X$  e  $Y$  son estocásticamente independientes
  - $P(X/Y)=P(X)$
- Entonces la confianza de la regla  $X \rightarrow Y$  será:
  - $Conf(X \rightarrow Y) = P(Y/X) = P(Y)$
- Así, si el itemset  $Y$  ocurre frecuentemente en los datos, la regla  $X \rightarrow Y$  también tendrá un valor de confianza alto. Esto sugiere una dependencia de  $Y$  con  $X$  aunque, de hecho, ambos itemsets son estocásticamente independientes.

# Problemas del framework soporte-confianza

- Dominio: desperfectos en combis
  - Fallo el motor: si  $\rightarrow$  Color: blanco
    - Soporte(Color: blanco) = alto
    - Confianza regla = alto



# Lift

- **Lift** indica la proporción entre el soporte observado de un conjunto de ítems respecto del soporte teórico de ese conjunto dado el supuesto de independencia.
- Un valor de **lift = 1** indica que ese conjunto aparece una cantidad de veces acorde a lo esperado bajo condiciones de independencia.
- Un valor de **lift > 1** indica que ese conjunto aparece una cantidad de veces superior a lo esperado bajo condiciones de independencia (por lo que se puede intuir que existe una relación que hace que los ítems se encuentren en el conjunto más veces de lo normal).
- Un valor de **lift < 1** indica que ese conjunto aparece una cantidad de veces inferior a lo esperado bajo condiciones de independencia (por lo que se puede intuir que existe una relación que hace que los ítems no estén formando parte del mismo conjunto más veces de lo normal).

$$\text{lift}(X \rightarrow Y) = \frac{\text{conf}(X \rightarrow Y)}{P(Y)}$$
$$\text{conf}(X \rightarrow Y) / \text{sup}(Y)$$

# Conviction

- $P(\neg Y)$ : probabilidad de una transacción  $T$  en  $D$  con  $Y \notin T$ .
- $P(X, \neg Y)$ : probabilidad de una transacción  $T$  en  $D$  que contenga  $X$  pero no  $Y$ .
- **Conviction** expresa que tan independientes son las variables  $X$  e  $\neg Y$ .
- Altos valores para **conv**( $X \rightarrow Y$ ), cuando  $P(X, \neg Y)$  tiende a cero, afirman la convicción de que esta regla representa una causalidad.

$$\text{conviction}(X \rightarrow Y) = P(X).P(\neg Y) / P(X, \neg Y)$$

# Métricas subjetivas

- Reglas interesantes y no interesantes pueden ser especificadas con un template (Klementinen et al., 1994).
- Un template especifica que atributos deben figurar en el antecedente y consecuente de una regla.
  - Ej. Cualquier regla de la forma “Preñada” & (otras condiciones) → “Hembra” no es interesante.

# Templates

- Un template es una expresión de la forma:
  - $A_1, \dots, A_k \rightarrow A_{k+1}, \dots, A_n$

donde cada  $A_i$  es un nombre de atributo, un nombre de clase, o una expresión C+ y C\*, que corresponden a uno o mas y cero o mas instancias de la clase C, respectivamente. Una regla  $B_1, \dots, B_h \rightarrow B_{h+1}, \dots, B_m$  corresponde a un patrón si la regla puede ser considerada una instancia de dicho patrón.

# Restricciones de ítems

- Las restricciones son expresiones booleanas sobre la presencia o ausencia de ítems en las reglas.
  - Si existe una taxonomía, los elementos de las expresiones booleanas pueden ser ítems ancestros o descendientes.
- Restricciones son embebidas en el algoritmo de reglas de asociación.
  - Más eficiente que en el post-procesamiento.

# Eliminar reglas redundantes

R1: “Alta velocidad” → “Tuvo un accidente” [0.1, 0.4]

R2: “Alta velocidad”, “Nacido en MDP” → “Tuvo un accidente” [0.09, 0.42]

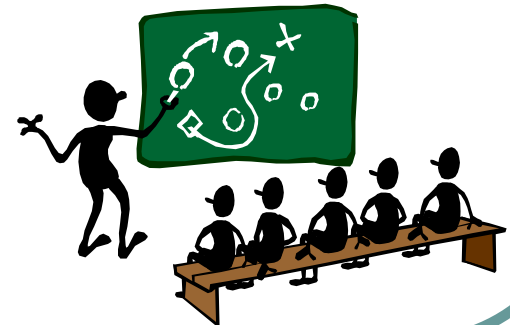
- Si conocemos R1, entonces R2 es insignificante dado que da muy poca información extra. La escasa diferencia entre los valores de confianza es más probable que sea producto de una casualidad que de una correlación verdadera.
- Reglas generales y simples son preferidas
  - R2 debería ser filtrada dado que R1 es más general y simple.





# Agenda

- Conceptos
- Algoritmos de descubrimientos de reglas de asociación
- Post-procesamiento
- **Actualización incremental de reglas**
- Ejemplos de aplicación
- Software



# Actualización incremental de reglas de asociación

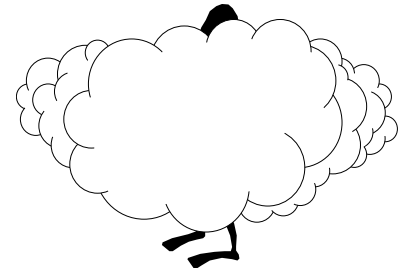
- Las reglas descubiertas a partir de una base de datos, sólo reflejan el estado actual de la misma. Para lograr que las reglas descubiertas sean estables y confiables, una gran cantidad de datos debería ser recolectada durante un período sustancial de tiempo.
- Por ejemplo, en una base de datos de ventas, nuevos registros son agregados frecuentemente, y registros viejos pueden ser eliminados.
- Las actualizaciones en una base de datos puede introducir nuevas reglas de asociación e invalidar las existentes. Por lo tanto, es importante el estudio de algoritmos eficientes para la actualización incremental de dichas reglas.

# Solución

- El problema de la actualización puede ser reducido a encontrar el nuevo conjunto de itemsets frecuentes. Luego, las nuevas reglas de asociación pueden ser generadas a partir de estos nuevos itemsets.
- Un itemset frecuente anterior a una actualización de la base de datos puede convertirse en un itemset no frecuente luego de que se realice la actualización.
- Un itemset no frecuente anterior a una actualización podría convertirse en frecuente luego de que la misma se realice.

# Enfoque trivial

- Ejecutar nuevamente el algoritmo de descubrimiento de reglas de asociación sobre toda la base.
- Desventaja: todo el computo realizado inicialmente para encontrar los itemsets frecuentes es desaprovechado y todos los itemsets deben ser computados desde cero.



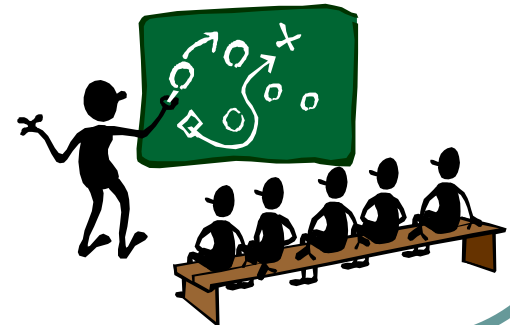
# Algoritmos

- FUP (Fast Update) (Cheung et al., 1996) actualiza las reglas de asociación cuando nuevas transacciones son agregadas a la base de datos.
- FUP2 (Cheung et al., 1997) actualiza las reglas de asociación cuando transacciones son agregadas a, eliminadas de o actualizadas en la base de datos.
- UWEP [Ayan99] (Update With Early Pruning)
- IDIC\_1 y IDIC\_M [Ng99]
- GSP+ y MFS+ [Zhang02]
- ...



# Agenda

- Conceptos
- Ejemplos
- Algoritmos de descubrimientos de reglas de asociación
- Post-procesamiento
- Introducción ML y Actualización incremental de reglas
- **Ejemplos de aplicación en IA**
- Software



# Negociación basada en argumentación

- Agentes deben negociar para resolver conflictos.
- Negociación
  - Intercambio de propuestas
  - Argumentos
    - Justificar una propuesta propia.
    - Influnciar la postura de negociación de un oponente.
    - Argumentos retóricos
      - Apelaciones
      - Recompensas
      - Amenazas



# Problema

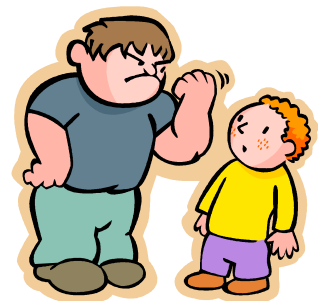
- Queremos que un agente negociador aprenda como generar argumentos a partir de la observación de otros agentes en el contexto de una negociación.
  - Aprender las reglas de generación de argumentos.





# Argumentos

- `appeal(Ai, Aj , do(Aj , Action), [Just])`
  - `appeal(x, y, do(y, actionA), [imply(do(y, actionA), goal1), hasgoal( y, goal(goal1))])`
- `reward(Ai, Aj , do(Aj , Actionh), [do(Ai, Actionk)])`
- `threat(Ai, Aj , do(Aj , Actionh), [do(Ai, Actionk)])`



# Generación de argumentos

- Reglas explícitas

- Especifican las condiciones que el contexto de la negociación debe cumplir.

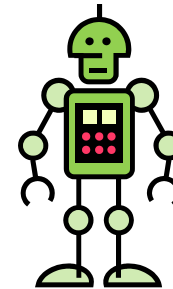
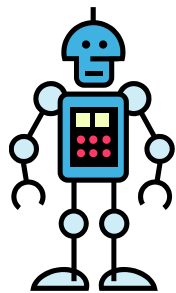
- Apelación de interés propio:

- Condiciones: `hasgoal(Y, goal(Goal)), believe(X, imply(do(Y, Action), Goal))`
      - Argumento: `appeal(X, Y, do(Y, Action), [imply(do(Y, Action), Goal), hasgoal(Y, goal(Goal))])`

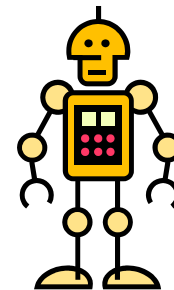
- Recompensa:

- Condiciones: `hasgoal(Y, goal(Goal)), believe(Y, imply(do(X, ActionR), Goal)), hasgoal(X, goal(Goal2)), believe(X, imply(do(Y, ActionP), Goal2))`
      - Argumento: `reward(X, Y, do(Y, ActionP), [do(X, ActionR)])`

# Contexto de la negociación



Reglas



Reglas

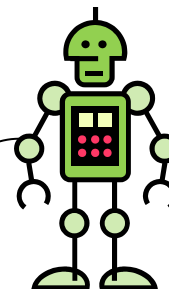
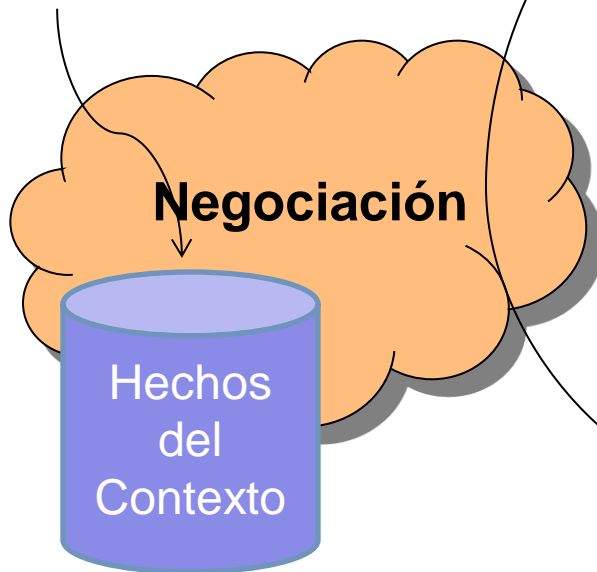
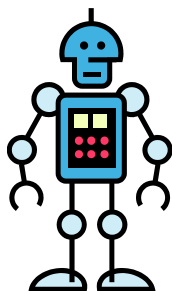
# Hechos del contexto

- **goal(G)**: G es un objetivo.
- **hasgoal(A, goal(G))**: el agente A tiene el objetivo G.
- **believe(A, B)**: el usuario A cree B, en otras palabras A tiene B entre sus creencias.
- **prefer(A, goal(G1), goal(G2))**: el agente A prefiere cumplir el objetivo G1 en lugar de cumplir el objetivo G2.
- **accept(P)**: representa la aceptación de la propuesta P.
- **reject(P)**: representa el rechazo de la propuesta P.
- **pastpromise(Ai, Aj, P)**: el usuario Ai ha prometido al usuario Aj cumplir la propuesta P.
- **do(A, Action)**: representa la ejecución de la acción Action por parte de A. Action puede ser instanciada con **accept(P)** o **reject(P)**.
- **wasgoal(A, goal(G))**: A persiguió el objetivo G en el pasado.
- **did(A, Action)**: A ejecutó la acción Action anteriormente.

Hechos  
del  
Contexto

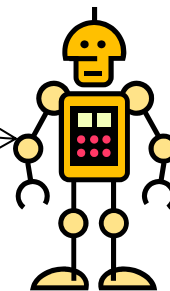
# Contexto de la negociación

hasgoal(a, goal(money)), believe(a, imply(do(? , pay a ?), money)), hasgoal(v, goal(relax)), hasgoal(v, goal(result x)), believe(v, imply(do(? , run x), result x)), did(r, run y), pastpromise(a, o, pay a 20)



Reglas

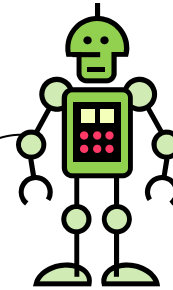
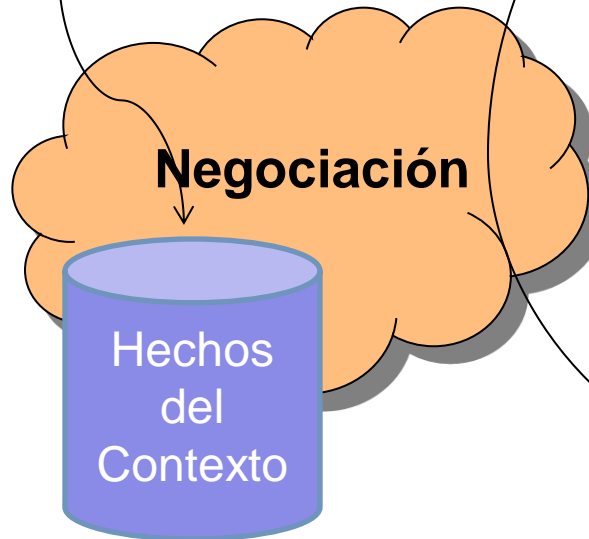
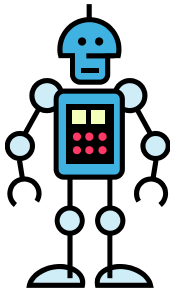
reward(v, a, do(a, run x), [do(v, pay a 10)])



Reglas

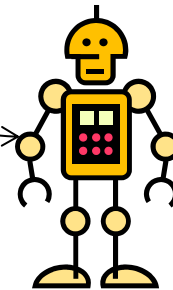
# Contexto de la negociación

hasgoal(a, goal(money)), believe(a,  
imply(do(? , pay a ?), money)), hasgoal(v,  
goal(relax)), hasgoal(v, goal(result x)),  
believe(v, imply(do(? , run x), result x)),  
did(r, run y), pastpromise(a, o, pay a 20)



Reglas

reward(v, a, do(a, run  
x), [do(v, pay a 10)])



Reglas

# Contexto de la negociación

- Incertidumbre

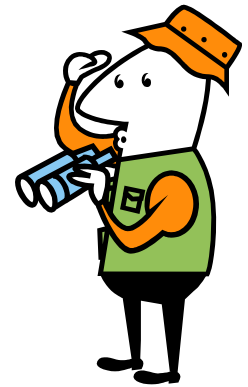
- Los agentes mantienen las reglas que utilizan para la generación de argumentos privadas.
  - No se tiene certeza de que hechos del contexto observan, aunque sí hay relaciones semánticas.

- Especificidad

- La información observada está instanciada en un contexto específico.
  - Las reglas obtenidas deben poder aplicarse en distintos contextos.

# Observaciones

- Base de observaciones  $O$ 
  - $o = (H, a)$
  - $H = \{h_1 \dots h_n\}$





# Observaciones (ejemplo)

- Hechos

- `hasgoal(i1,goal(date(thursday)))`.
- `hasgoal(i1,goal(discusstopic(topic1)))`.
- `hasgoal(i1,goal(not(discusstopic(topic4))))`.
- `prefer(i1,goal(not(time(night))),goal(not(inplace(room1))))`.
- `prefer(i1,goal(date(wednesday)),goal(date(tuesday)))`.
- `believe(i2, imply(do(_,accept(discusstopic(topic2))),inplace(lab)))`.
- `believe(i2, imply(do(_,accept(discusstopic(topic3))),inplace(room1)))`.
- `believe(i2, imply(do(_,accept(discusstopic(topic3))),inplace(room2)))`.
- `hasgoal(i3,goal(inplace(room1)))`.
- `pastpromise(i3,i1,do(i3,accept(discusstopic(topic3))))`.
- `pastpromise(i1,i2,do(i1,accept(inplace(room1))))`.
- `pastpromise(i4,i1,do(i4,accept(discusstopic(topic2))))`.
- `did(i2,accept(time(night)))`.
- `did(i1,accept(inplace(lab)))`.
- `did(i3,accept(inplace(room2)))`.
- `wasgoal(i3,goal(not(discusstopic(topic3))))`.
- `wasgoal(i4,goal(not(inplace(cafe))))`.
- .....

- Argumento

- `reward(a1, a2, do(a2, accept(discusstopic(topic1))), [do(a1, accept(discusstopic(topic2))])]`

# Definición de transacciones

- Base de observaciones  $O$ 
  - $o = (H, a)$
  - $H = \{h_1 \dots h_n\}$
- Reglas deseadas
  - $C \rightarrow a$ 
    - $C = \{c_1 \dots c_n\}$
    - $C \subseteq H$
- Función de membresía
  - Determina la relación probable entre el argumento generado  $a$  y cada hecho observado  $h_i$
  - $m_a(h_i) : H \rightarrow [0; 1]$
- Transacción
  - Para cada observación  $o^j = (\{h_1^j, \dots, h_n^j\}, a^j)$  construimos una transacción  $t^j = (h_1^j, \dots, h_n^j, a^j)$  donde  $m_a^j > 0$ .

# Función de membresía

## ● Basada en reglas

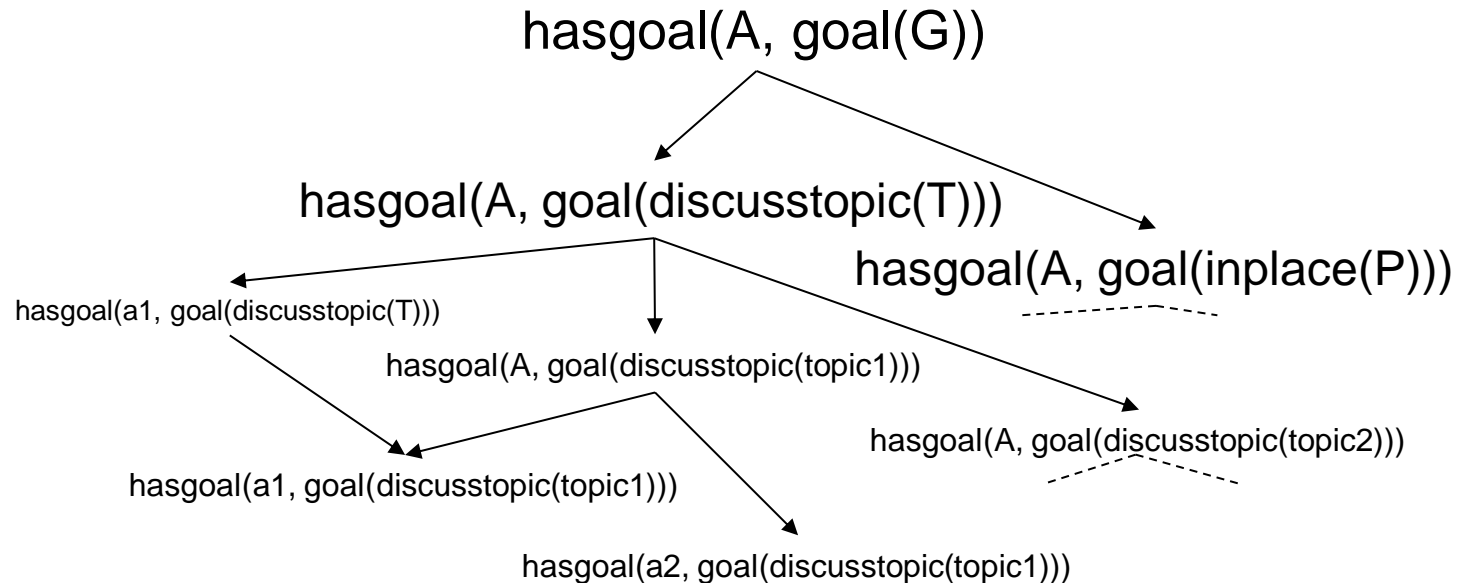
- relatedTo(Emisor, Receptor, Hecho\_observado, Hecho\_relacionado, Index)

## ● Ejemplo

- relatedTo(\_, \_, do(U, Action), hasgoal(U, goal(Goal)), Index)
  - believe(U, imply(do(U, Action), Goal)),  
hasgoal(U, goal(Goal)),  
Index = 0.8.

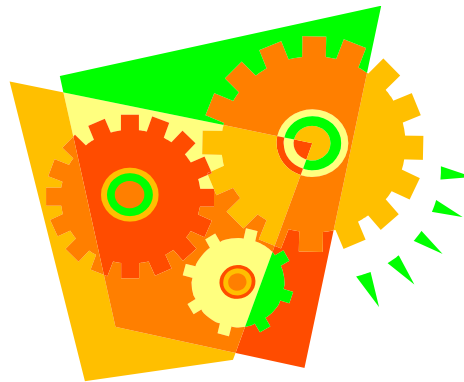
# Taxonomía de condiciones y argumentos

- Relación de inclusión
- Parte de la taxonomía



# Ejecución del algoritmo

- Algoritmo de descubrimiento difuso de reglas de asociación generalizadas.



# Reglas descubiertas

- $\text{believe}(\text{AGENT1}, \text{imply}(\text{do}(\text{AGENT0}, \text{ACTION0}), \text{GOAL0})) \rightarrow [\text{appeal}(\text{AGENT1}, \text{AGENT0}, \text{do}(\text{AGENT0}, \text{ACTION0}), [\text{imply}(\text{do}(\text{AGENT0}, \text{ACTION0}), \text{GOAL0}), \text{hasgoal}(\text{AGENT0}, \text{GOAL0})])]$
- $\text{hasgoal}(\text{AGENT0}, \text{goal}(\text{GOAL0})), \text{believe}(\text{AGENT1}, \text{imply}(\text{do}(\text{AGENT0}, \text{ACTION0}), \text{GOAL0})) \rightarrow [\text{appeal}(\text{AGENT1}, \text{AGENT0}, \text{do}(\text{AGENT0}, \text{ACTION0}), [\text{imply}(\text{do}(\text{AGENT0}, \text{ACTION0}), \text{GOAL0}), \text{hasgoal}(\text{AGENT0}, \text{GOAL0})])]$
- $\text{believe}(\text{AGENT1}, \text{imply}(\text{do}(\text{AGENT0}, \text{ACTION0}), \text{GOAL0})) \rightarrow \text{believe}(\text{AGENT0}, \text{imply}(\text{do}(\text{AGENT0}, \text{ACTION0}), \text{GOAL0}))$
- $\text{hasgoal}(\text{AGENT0}, \text{goal}(\text{discusstopic}(\text{T}))), \text{believe}(\text{AGENT1}, \text{imply}(\text{do}(\text{AGENT0}, \text{ACTION0}), \text{discusstopic}(\text{T}))) \rightarrow [\text{appeal}(\text{AGENT1}, \text{AGENT0}, \text{do}(\text{AGENT0}, \text{ACTION0}), [\text{imply}(\text{do}(\text{AGENT0}, \text{ACTION0}), \text{discusstopic}(\text{T})), \text{hasgoal}(\text{AGENT0}, \text{discusstopic}(\text{T}))])]$

# Post-procesamiento

- Filtrado por *template*

- $\{c_i, \dots, c_p\} \rightarrow a_j$

- Suficiencia

- $$\text{Suf}(r) = \frac{\text{condTotales}(r)}{\text{condPromedio}(\text{transaccionesSoportando}(r))}.$$

- Reglas ancestros

- Conservar las reglas con mayor grado de generalidad

# Reglas descubiertas

- $\text{believe}(\text{AGENT1}, \text{imply}(\text{do}(\text{AGENT0}, \text{ACTION0}), \text{GOAL0})) \rightarrow [\text{appeal}(\text{AGENT1}, \text{AGENT0}, \text{do}(\text{AGENT0}, \text{ACTION0}), [\text{imply}(\text{do}(\text{AGENT0}, \text{ACTION0}), \text{GOAL0}), \text{hasgoal}(\text{AGENT0}, \text{GOAL0})])]$
- $\text{hasgoal}(\text{AGENT0}, \text{goal}(\text{GOAL0})), \text{believe}(\text{AGENT1}, \text{imply}(\text{do}(\text{AGENT0}, \text{ACTION0}), \text{GOAL0})) \rightarrow [\text{appeal}(\text{AGENT1}, \text{AGENT0}, \text{do}(\text{AGENT0}, \text{ACTION0}), [\text{imply}(\text{do}(\text{AGENT0}, \text{ACTION0}), \text{GOAL0}), \text{hasgoal}(\text{AGENT0}, \text{GOAL0})])]$
- $\text{believe}(\text{AGENT1}, \text{imply}(\text{do}(\text{AGENT0}, \text{ACTION0}), \text{GOAL0})) \rightarrow \text{believe}(\text{AGENT0}, \text{imply}(\text{do}(\text{AGENT0}, \text{ACTION0}), \text{GOAL0}))$
- $\text{hasgoal}(\text{AGENT0}, \text{goal}(\text{discusstopic}(\text{T}))), \text{believe}(\text{AGENT1}, \text{imply}(\text{do}(\text{AGENT0}, \text{ACTION0}), \text{discusstopic}(\text{T}))) \rightarrow [\text{appeal}(\text{AGENT1}, \text{AGENT0}, \text{do}(\text{AGENT0}, \text{ACTION0}), [\text{imply}(\text{do}(\text{AGENT0}, \text{ACTION0}), \text{discusstopic}(\text{T})), \text{hasgoal}(\text{AGENT0}, \text{discusstopic}(\text{T}))])]$

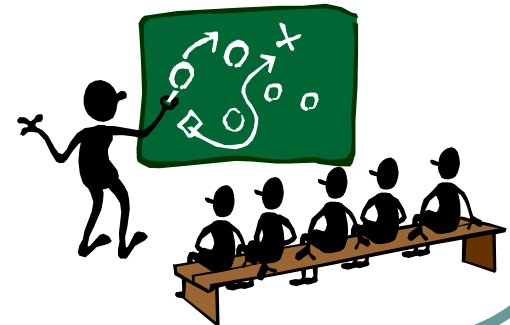


# Reglas descubiertas

- `believe(AGENT1, imply(do(AGENT0, ACTION0), GOAL0)) → [appeal(AGENT1, AGENT0, do(AGENT0, ACTION0), [imply(do(AGENT0, ACTION0), GOAL0), hasgoal(AGENT0, GOAL0)])]`
- `hasgoal(AGENT0, goal(GOAL0)), believe(AGENT1, imply(do(AGENT0, ACTION0), GOAL0)) → [appeal(AGENT1, AGENT0, do(AGENT0, ACTION0), [imply(do(AGENT0, ACTION0), GOAL0), hasgoal(AGENT0, GOAL0)])]`
- `believe(AGENT1, imply(do(AGENT0, ACTION0), GOAL0)) → believe(AGENT0, imply(do(AGENT0, ACTION0), GOAL0))`
- `hasgoal(AGENT0, goal(discusstopic(T))), believe(AGENT1, imply(do(AGENT0, ACTION0), discusstopic(T))) → [appeal(AGENT1, AGENT0, do(AGENT0, ACTION0), [imply(do(AGENT0, ACTION0), discusstopic(T)), hasgoal(AGENT0, discusstopic(T))])]`

# Agenda

- Introducción ML y Conceptos
- Ejemplos
- Algoritmos de descubrimientos de reglas de asociación
- Post-procesamiento
- Actualización incremental de reglas
- Ejemplos de aplicación en agentes
- **Software**



# Software

- MLxtend: <http://rasbt.github.io/mlxtend/>
- Weka (<http://www.cs.waikato.ac.nz/~ml/weka/>)
- Knime: <http://www.knime.org/>
- RapidMiner: <http://rapid-i.com/content/view/181/190/>
- AIAS: Association Interestingness Analysis System  
(<http://www.comp.nus.edu.sg/~dm2>)
- Gnome Data Mine (linux):  
(<http://www.togaware.com/datamining/gdatamine/gdmapriori.html>)
- FIMI, Frequent Itemset Mining Implementations repository, incluye software y datasets.
- Useful links: <http://www.kdnuggets.com/software/associations.html>



# Conclusiones

- Las reglas de asociación son útiles para descubrir asociaciones entre conjuntos de ítems en una base de datos de transacciones.
- Puede ser utilizado en múltiples dominios: análisis de canasta de mercado, datos de censos, recomendación, aprendizaje en sistemas multiagentes, etc..
- Varios algoritmos de descubrimiento de reglas de asociación, descubren el mismo conocimiento.
- Son necesarias diversas tareas de post-procesamiento para eliminar reglas no interesantes, podar reglas redundantes, etc.
- Extensiones: Reglas de asociación generalizadas, difusas temporales, etc.



# Preguntas



# Referencias

- R. Agrawal, R. Srikant - Fast Algorithms for Mining Association Rules - Proc. of the 20th International Conference on Very Large Databases, Santiago, Chile, Sept. 1994.
- M. Klemetinen, H. Mannila, P. Ronkainen, H. Toivonen, and A. Verkamo. Finding interesting rules from large sets of discovered association rules. Proc. CIKM, 1994.
- D. Shah, L. V. S. Lakshmanan, K. Ramamritham, and S. Sudarshan. Interestingness and pruning of mined patterns. In 1999 ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery, 1999.
- Geoffrey I. Webb. OPUS: An efficient admissible algorithm for unordered search. Journal of Artificial Intelligence Research, 3:45--83, 1996.
- Jean-Marc Adamo - Data Mining For Association Rules and Sequential Patterns Sequential and Parallel Algorithms (2001) - Springer
- Mohammed J. Zaki, Ching-Jui Hsiao - CHARM: An Efficient Algorithm for Closed Itemset Mining (2002)
- J. Han, J. Pei, and Y. Yin. Mining frequent patterns without candidate generation. Technical Report TR-99-12, Computing Science Technical Report, Simon Fraser University, October 1999
- J. Pei, J. Han, and R. Mao. CLOSET: An efficient algorithm for mining frequent closed itemsets. In DMKD 2000, pp. 11--20
- C.M. Kuok, A.W. Fu, and M.H. Wong. Mining fuzzy association rules in databases. SIGMOD Record, 27(1):41-46, 1998.
- T. Hong, K. Lin, and S. Wang. Fuzzy data mining for interesting generalized association rules. Fuzzy Sets Syst., 138(2):255-269, 2003.
- M. Delgado, N. Marn, D. Sanchez, and M.A. Vila. Mining fuzzy association rules: An overview. BISC International Workshop on Soft computing for Inter- net and Bioinformatics, pages 351-374, 2003.

# Referencias

- S Brin, R. Motwani, J. D. Ullman and Shalom Tsur - Dynamic itemset counting and implication rules for market basket data – Proc. of the 1997 ACM SIGMOD International conference on Management of data – 1997 – pp 255 - 264
- V. Dhar and A. Tuzhilin - Abstract-Driven Pattern Discovery in Databases - In: IEEE Transactions on Knowledge and Data Engineering 5 (6) (1993).
- T. Fukuda, Y.Morimoto, S. Morishita and T. Tokuyama - Data mining using two-dimensional optimized association rules: scheme, algorithms, and visualization – Proc.of the 1996 ACM SIGMOD international conference on Management of data – 1996 – pp 13 - 23
- Y. Morimoto and T. Fukuda and H. Matsuzawa and T.Tokuyama and K. Yoda - "Algorithms for Mining Association Rules for Binary Segmentations of Huge Categorical Databases" - In: "Proceedings of the 24th Very Large Data Bases Conference", pp. 380--391
- G. Webb - "OPUS: An Efficient Admisible Algorithm for Unordered Search" - In: Journal of Artificial Intelligence Research 3 pp. 431--465 (1995).
- G. Piatetsky-Shapiro - "Discovery, Analysis, and Presentation of Strong Rules" - Published by: AAAI/MIT Press, 1991. Chapter 13.
- D. Cheung, S.D. Lee, Benjamin Kao – A General Incremental Technique for Maintaining Discovered Association Rules – Proc. 5th International Conference on Database Systems for Advanced Applications – 1997
- D. Cheung, J. Han, V. Ng, C. Wong – Maintenance of Discovered Association Rules in Large Databases: An Incremental Updating Technique – Proc. 1996 International Conference on Data Engineering – pp. 106 -114 – 1996
- S.D.Lee anad D. Cheung – Maintenance of Discovered Association Rules: When to update? – Proc. Of SIGMOD Wrkshop on Research Issues in DM and KD - 1997