



ESCUELA DE INGENIERÍA
FACULTAD DE INGENIERÍA

IIC2433 Minería de Datos

Reglas de asociación: Apriori

Profesor: Mauricio Arriagada

REGLAS DE ASOCIACIÓN



OBJETIVO

- ▶ Analizar datos de las compras de clientes
- ▶ Buscar asociaciones entre los diferentes productos

Algunas aplicaciones

- ▶ Ordenamiento de productos
- ▶ Patrones de navegación
- ▶ Promoción de pares de productos
- ▶ Segmentación
 - ▶ Descuentos específicos por cliente

Método aplicado en reglas de asociación

- ▶ Algoritmo A priori (Agrawal, 1994)
- ▶ Este algoritmo permite encontrar reglas de asociación de manera automática desde los datos.

Definiciones

Itemset

- ▶ Colección de uno o más ítems
- ▶ Ejemplo:
 {Leche, Pañales, Cerveza}

Soporte

- ▶ Frecuencia relativa que un itemset aparece en la base de datos
- ▶ Esto se calcula como el número itemset que aparece en la base de datos de compras dividido por el número total de compras(transacciones)
- ▶ Ejemplo:

Si tenemos que Leche aparece en 3 compras entre un total de 10 compras, entonces el soporte es $3/10$

Itemset frecuente

- ▶ Un itemset que aparece en una frecuencia mayor a un umbral
- ▶ El umbral está determinado por uno o bien viene dado.

Regla de asociación

- ▶ Es una expresión de la forma $X \rightarrow Y$

Donde X e Y son itemsets

- ▶ Ejemplo:

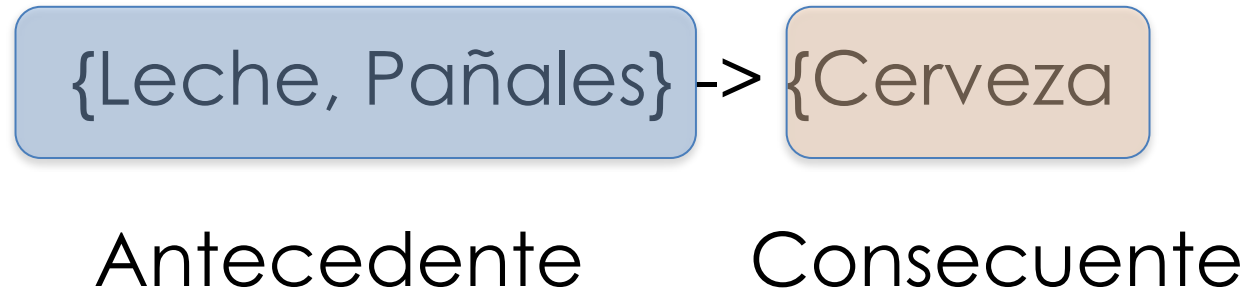
$\{\text{Leche, Pañales}\} \rightarrow \{\text{Cerveza}\}$

Análisis: se podría decir que si se compra leche y pañales es posible que se compre cerveza también.

(esto se obtiene desde datos de compras y depende de valores de confianza y soporte empírico)

Regla de asociación

- ▶ Otra defición para la regla $X \rightarrow Y$



Ejemplo de transacciones

T	ítems
1	Pan, leche
2	Pan, pañales, cerveza, huevos
3	Leche, pañales, cerveza, diario
4	Pan, leche, pañales, cerveza
5	Pan, leche, pañales, diario

T	ítems
1	Pan
1	Leche
2	Pan
2	Pañales
2	Cerveza
2	Huevos
3	Leche
3	Pañales
3	Cerveza
3	Diario

Ejemplo de reglas de asociación

- ▶ {leche, pañales} -> {cerveza}
- ▶ {leche, cerveza} -> {pañales}
- ▶ {pañales, cerveza} -> {leche}
- ▶ {cerveza} -> {leche, pañales}
- ▶ {pañales} -> {leche, cerveza}
- ▶ {cerveza} -> {pañales, cerveza}

Indicadores de rendimiento

Ejemplo de soporte

T	ítems
1	Pan, leche
2	Pan, pañales, cerveza, huevos
3	Leche, pañales, cerveza, diario
4	Pan, leche, pañales, cerveza
5	Pan, leche, pañales, diario

$\{\text{Leche, pañales}\} \rightarrow \{\text{cerveza}\}$

$X = \{\text{Leche, pañales, cerveza}\}$

$$s = \frac{\sigma(X)}{|T|}$$

$$s = \frac{\sigma(\{\text{Leche, pañales, cerveza}\})}{|T|}$$

Ejemplo de soporte

$\{\text{Leche, pañales}\} \rightarrow \{\text{cerveza}\}$

T	ítems
1	Pan, leche
2	Pan, pañales, cerveza, huevos
3	Leche, pañales, cerveza, diario
4	Pan, leche, pañales, cerveza
5	Pan, leche, pañales, diario

$X = \{\text{Leche, pañales, cerveza}\}$

$$s = \frac{\sigma(X)}{|T|}$$

$$s = \frac{2}{5} = 0.4$$

El 40% de las transacciones muestran que Leche, Pañales y Cerveza se compraron en juntos

Ejemplo de confianza

$$c(X \rightarrow Y) = \frac{\sigma(X \cup Y)}{\sigma(X)}$$

T	ítems
1	Pan, leche
2	Pan, pañales, cerveza, huevos
3	Leche, pañales, cerveza, diario
4	Pan, leche, pañales, cerveza
5	Pan, leche, pañales, diario

$X = \{\text{Leche, pañales, cerveza}\}$

$$c = \frac{\sigma(\{\text{Leche, pañales, cerveza}\})}{\sigma(\{\text{Leche, pañales}\})}$$

$$c = \frac{2}{3} = 0.67$$

$\{\text{Leche, pañales}\} \rightarrow \{\text{cerveza}\}$

Confianza (interpretación en probabilidad)

T	ítems
1	Pan, leche
2	Pan, pañales, cerveza, huevos
3	Leche, pañales, cerveza, diario
4	Pan, leche, pañales, cerveza
5	Pan, leche, pañales, diario

$X \rightarrow Y$
 $\{\text{Leche, pañales}\} \rightarrow \{\text{cerveza}\}$

$$P(Y|X) = \frac{P(Y, X)}{P(X)}$$

$$c = \frac{2}{3} = 0.67$$

Por lo tanto, si una regla tiene una confianza de 0,67 podemos decir que de los consumidores que compraron leche y cerveza en conjunto, también compraron cerveza

Problema

Supongamos que la confianza es 0,7 para la siguiente regla
Leche -> Cerveza

$$c(\textit{Leche}, \textit{Cerveza}) = \frac{\sigma(\textit{Leche}, \textit{Cerveza})}{\sigma(\textit{Leche})} = 0.7$$

70% probabilidad empírica que el cliente compre cerveza si compra leche

¿Pero si la probabilidad apriori de comprar solamente cerveza ya es de 70%?

$$\sigma(\textit{Cerveza}) = 0.7$$

Lift

- ▶ Permite medir el incremento del lado derecho de la regla (consecuente) dada la compra del lado izquierdo (antecedente)
- ▶ Confianza de la regla dividido por el soporte del consecuente

$$Lift = \frac{c(X \rightarrow Y)}{s(Y)}$$

Ejemplo cálculo de lift

T	ítems
1	Pan, leche
2	Pan, pañales, cerveza, huevos
3	Leche, pañales, cerveza, diario
4	Pan, leche, pañales, cerveza
5	Pan, leche, pañales, diario

$\{\text{Leche, pañales}\} \rightarrow \{\text{Cerveza}\}$

$$\text{Lift} = \frac{c(\{ \text{Leche, pañales} \} \rightarrow \{ \text{Cerveza} \})}{s(\{ \text{Cerveza} \})}$$

Ejemplo cálculo de lift

T	ítems
1	Pan, leche
2	Pan, pañales, cerveza, huevos
3	Leche, pañales, cerveza, diario
4	Pan, leche, pañales, cerveza
5	Pan, leche, pañales, diario

$\{\text{Leche, pañales}\} \rightarrow \{\text{Cerveza}\}$

$$c(\{\text{Leche, pañales}\} \rightarrow \{\text{Cerveza}\}) = 0,67 \quad \checkmark$$

$$s(\{\text{Cerveza}\}) = \frac{3}{5} = 0,6$$

$$\text{Lift} = \frac{0,67}{0,6} \approx 1,117$$

La probabilidad aumenta de 0,6 a 0,67 cuando el cliente compra leche y pañales

Lift

$$> 1$$

La probabilidad del consecuente de la regla aumentó dado que el consumidor compró los ítems del antecedente

$$= 1$$

La probabilidad no se vio afectada, es decir, el consecuente no se ve influenciado por el antecedente

$$< 1$$

El antecedente tuvo un efecto negativo en la ocurrencia del consecuente, lo que baja su probabilidad

Algoritmo Apriori

Método aplicado en reglas de asociación

- ▶ Algoritmo A priori (Agrawal, 1994)
- ▶ Este algoritmo permite encontrar reglas de asociación de manera automática desde los datos.
- ▶ Algoritmo capaz de encontrar reglas de asociación que cumplan con un mínimo valor de soporte y confianza

Algoritmo

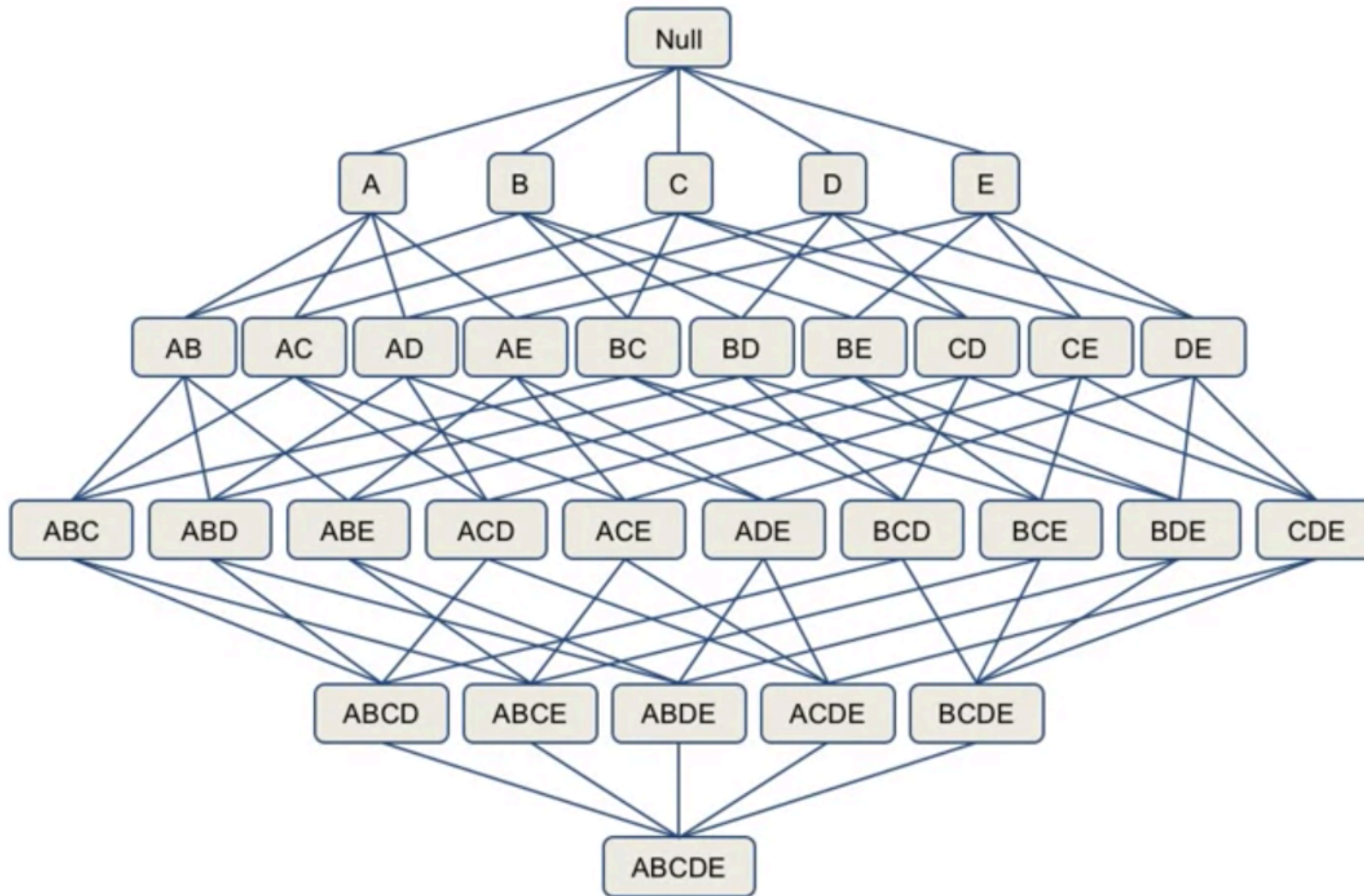
1. Se calcula el soporte de cada ítem individual, y se determinan los 1-itemsets frecuentes.
2. En cada paso subsecuente, los itemsets frecuentes generados en los pasos anteriores se utilizan para generar los nuevos itemsets (itemsets candidatos).
3. Se calcula el soporte de cada itemset candidato y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

Encontrando Itemsets

Una idea es obtener todos los itemsets posibles encontrando junto a su frecuencia

Ejemplo: Si se quieren combinar 5 items A,B,C,D y E, esto se podría representar vizualizándolo en un lattice de itemsets

Encontrando Itemsets



Total de combinaciones
 2^5 menos el item vacío = 32

Encontrando Itemsets

Ejemplo:

Combinación de 3 productos.



{  } {  } {  }

{   } {   } {   }

{    }

En general $2^n - 1$

$$2^3 - 1 = 7$$

Encontrando Itemsets PROBLEMA

¿Qué pasa cuando n es grande?

2^n se convierte en un valor muuuuy grande

Ejemplo:

1000 productos de un supermercado

Solución: principio de Monotonicidad

Ayuda a reducir la cantidad de itemsets posibles a considerar dentro de nuestro set de datos

Si un itemset es frecuente, entonces todos los subgrupos de este itemset también son frecuentes.

Ejemplo: si el itemset frecuente es {pescado, mayonesa, bebida} entonces el itemset mayonesa, bebida también será frecuente

Solución: principio de Monotonicidad (análogamente)

Ejemplo:

Si el itemset {queso, galleta} no es frecuente, entonces si agregamos el item mayonesa a este itemset, quedando {queso, galleta, mayonesa}, la frecuencia empezará a bajar

Ejemplo principio de Monotonicidad

T	ítems
1	leche, manzana, naranja, pera
2	leche, naranja, pera
3	manzana, naranja, pera
4	naranja, pera, limon, plátano

$$s(\{Pera, Manzana\}) = \frac{2}{4}$$

Si el umbral estuviese dado por:

$$umbral\ mínimo = \frac{3}{4}$$



Pera, Manzana no supera el umbral

Ejemplo principio de Monotonicidad

T	ítems
1	leche, manzana, naranja, pera
2	leche, naranja, pera
3	manzana, naranja, pera
4	naranja, pera, limon, plátano

$$s(\{Pera, Manzana, leche\}) = \frac{1}{4}$$

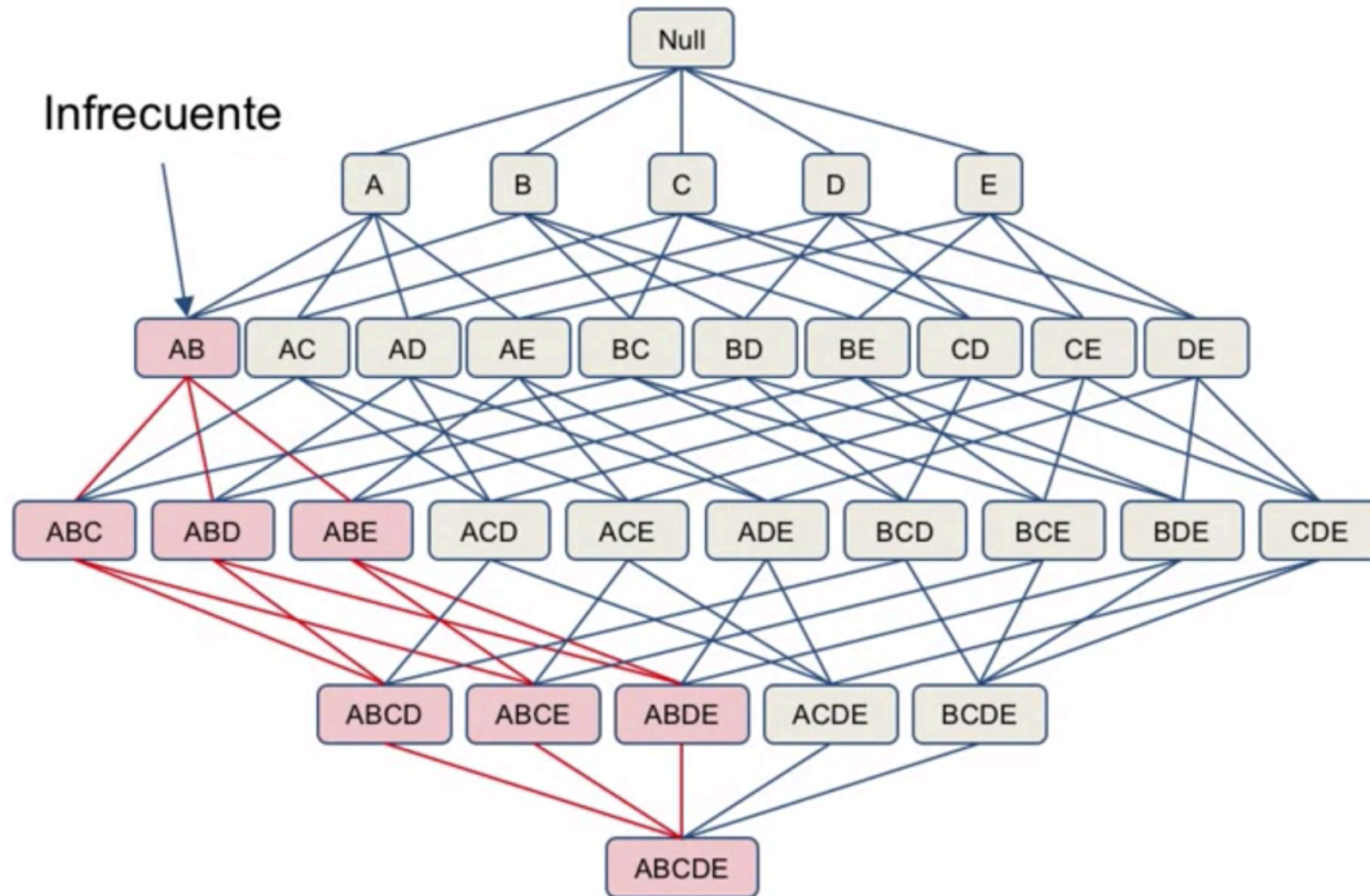
Si el umbral estuviese dado por:

$$umbral\ mínimo = \frac{3}{4}$$



Pera, Manzana no superaba el umbral, por lo que si se agrega leche, tampoco lo superará

principio de Monotonidad

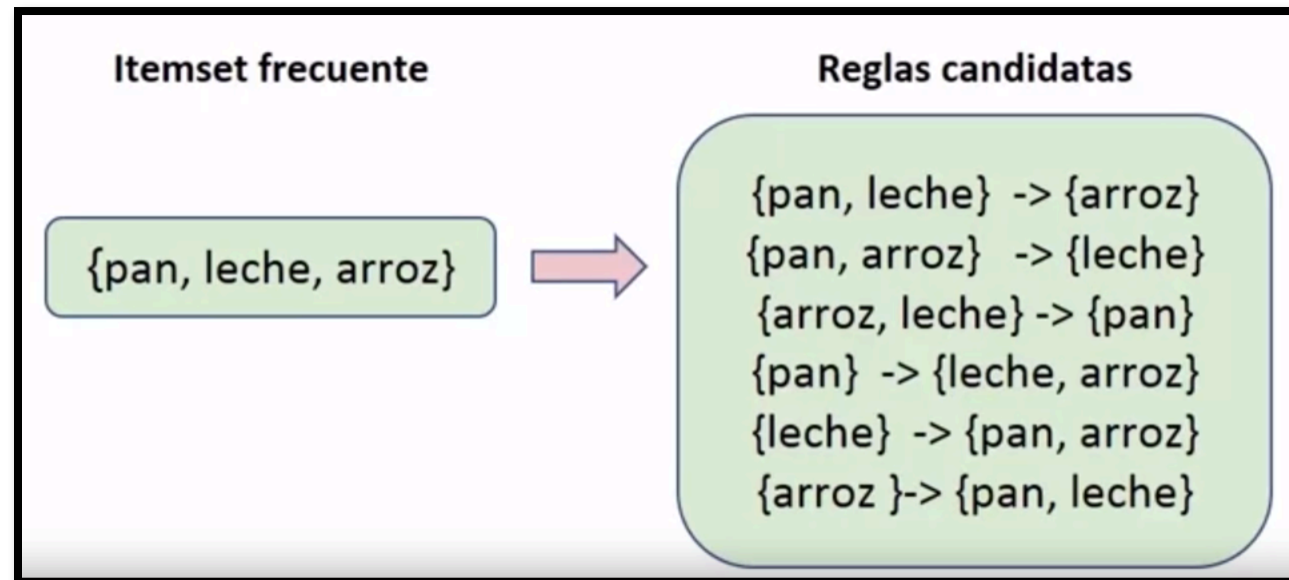


Generación de itemset candidatos

Inicio del algoritmo

El algoritmo primero obtiene los itemsets frecuentes y después calcula las reglas de asociación a partir de ellos.

Ejemplo:



Ejemplo: Inicio del algoritmo

Usemos el mismo set de datos que hemos estado viendo

T	ítems
1	leche, manzana, naranja, pera
2	naranja, pera
3	manzana, naranja, pera
4	naranja, pera, limon, plátano

Ejemplo: Inicio del algoritmo

Asumir que todos los productos son candidatos a ser item frecuente de manera individual

Limón
Manzana
Naranja
Plátano
Leche
Pera

Ejemplo: Inicio del algoritmo

Eliminar los itemsets que no superen un umbral determinado

$$\textit{umbral mínimo} = \frac{2}{4}$$

Limón
Manzana
Naranja
Plátano
Leche
Pera

Ejemplo: Inicio del algoritmo

Eliminar los itemsets que no superen un umbral determinado

$$\text{umbral mínimo} = \frac{2}{4}$$

T	ítems
1	leche, manzana, naranja, pera
2	naranja, pera
3	manzana, naranja, pera
4	naranja, pera, limon, plátano

1/4 Limón

2/4 Manzana 

4/4 Naranja  1/4

Plátano

1/4 Leche

4/4 Pera 

Ejemplo: Inicio del algoritmo

Ahora se deben generar los itemset que tengan 2 items tomando sólo en consideración los items seleccionados anteriormente

T	ítems
1	leche, manzana, naranja, pera
2	naranja, pera
3	manzana, naranja, pera
4	naranja, pera, limon, plátano

2/4 Manzana 

4/4 Naranja  4/4

Pera 

Ejemplo: Inicio del algoritmo

Manzana

Naranja

Pera



Manzana
Naranja

Manzana
Pera

Naranja
Pera

Candidatos a pasar a la siguiente ronda

Ejemplo: Inicio del algoritmo

Eliminar los itemsets que no superen un umbral determinado

$$\text{umbral mínimo} = \frac{2}{4}$$

T	ítems
1	leche, manzana, naranja, pera
2	naranja, pera
3	manzana, naranja, pera
4	naranja, pera, limon, plátano

Manzana
Naranja

Manzana
Pera


Naranja
Pera


Ejemplo: Inicio del algoritmo


Eliminar los itemsets que no superen un umbral determinado

$$\text{umbral mínimo} = \frac{2}{4}$$

T	ítems
1	leche, manzana, naranja, pera
2	naranja, pera
3	manzana, naranja, pera
4	naranja, pera, limon, plátano

2/4 Manzana
Naranja 

2/4 Manzana
Pera 

2/4 Naranja
Pera 

Ejemplo: Inicio del algoritmo

Manzana
Naranja

Manzana
Pera

Naranja
Pera



Manzana
Naranja
Pera

Candidatos a pasar a la siguiente ronda

Ejemplo: Inicio del algoritmo

Eliminar los itemsets que no superen un umbral determinado

$$\text{umbral mínimo} = \frac{2}{4}$$

T	ítems
1	leche, manzana, naranja, pera
2	naranja, pera
3	manzana, naranja, pera
4	naranja, pera, limon, plátano

Manzana
Naranja
Pera

Ejemplo: Inicio del algoritmo

Eliminar los itemsets que no superen un umbral determinado

$$\text{umbral mínimo} = \frac{2}{4}$$

T	ítems
1	leche, manzana, naranja, pera
2	naranja, pera
3	manzana, naranja, pera
4	naranja, pera, limon, plátano

2/4

Manzana
Naranja
Pera



Ejemplo: Inicio del algoritmo

Manzana
Naranja
Pera



No hay más elementos qué combinar
por lo que el algoritmo se detiene en la
búsqueda de itemset candidatos

En general

Itemsets de tamaño 1

Manzana

Naranja

Pera

Itemsets de tamaño 2

Manzana
Naranja

Manzana
Pera

Naranja
Pera

Itemsets de tamaño 3

Manzana
Naranja
Pera

Itemsets de
tamaño K

?

Itemsets de
tamaño K+1



Regla

Definir un orden lexicográfico (arbitrario) entre todos los productos que estén en el set de datos de transacciones.

Ejemplo:

naranja	pera	platano	limón	pan	manzana	guinda	leche
leche	limón	manzana	pera	pan	naranja	plátano	guinda
1	2	3	4	5	6	7	8

¿cuándo se pueden combinar 2 itemsets?

Itemset A

manzana

naranja

pera



Itemset B

plátano

manzana

pera

¿cuándo se pueden combinar 2 itemsets?

Primero se deben ordenar usando el criterio arbitrario inicial

Itemset A

manzana	naranja	pera
---------	---------	------



manzana	pera	naranja
---------	------	---------

Itemset B

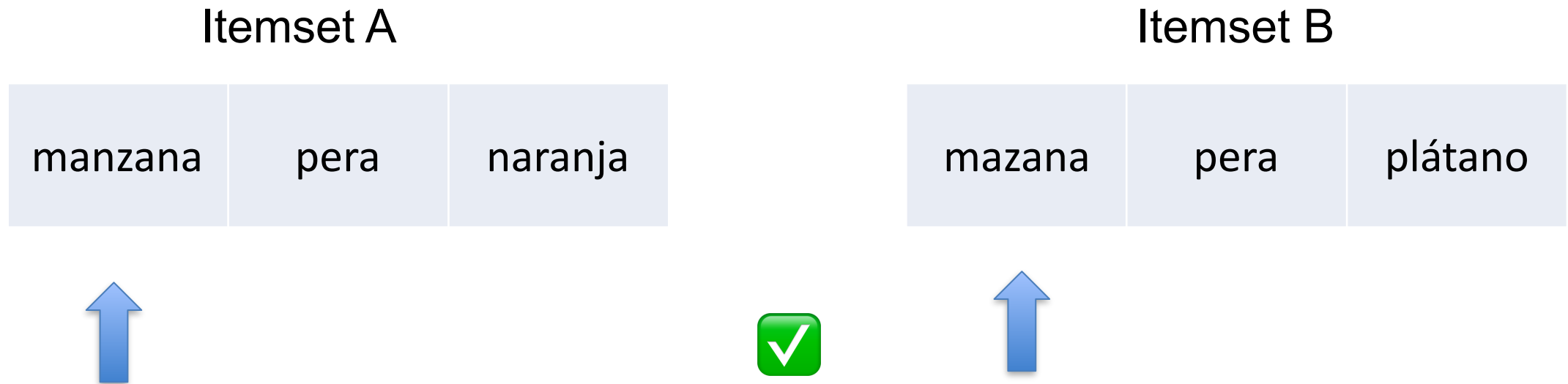
plátano	manzana	pera
---------	---------	------



mazana	pera	plátano
--------	------	---------

¿cuándo se pueden combinar 2 itemsets?

Se debe comparar item por item, de izquierda a derecha que los elementos sean iguales, menos en la última posición



¿cuándo se pueden combinar 2 itemsets?

Itemset A

manzana	pera	naranja
---------	------	---------



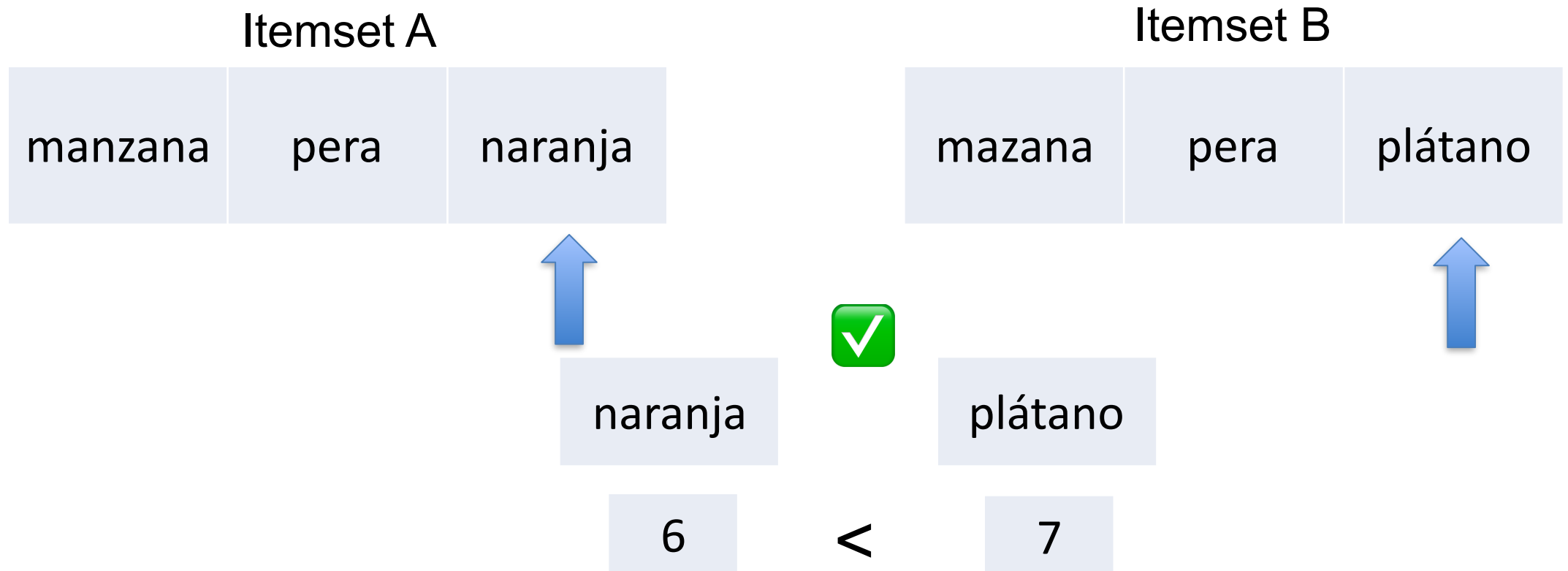
Itemset B

mazana	pera	plátano
--------	------	---------



¿cuándo se pueden combinar 2 itemsets?

En la última posición se debe tener en consideración el orden arbitrario inicial

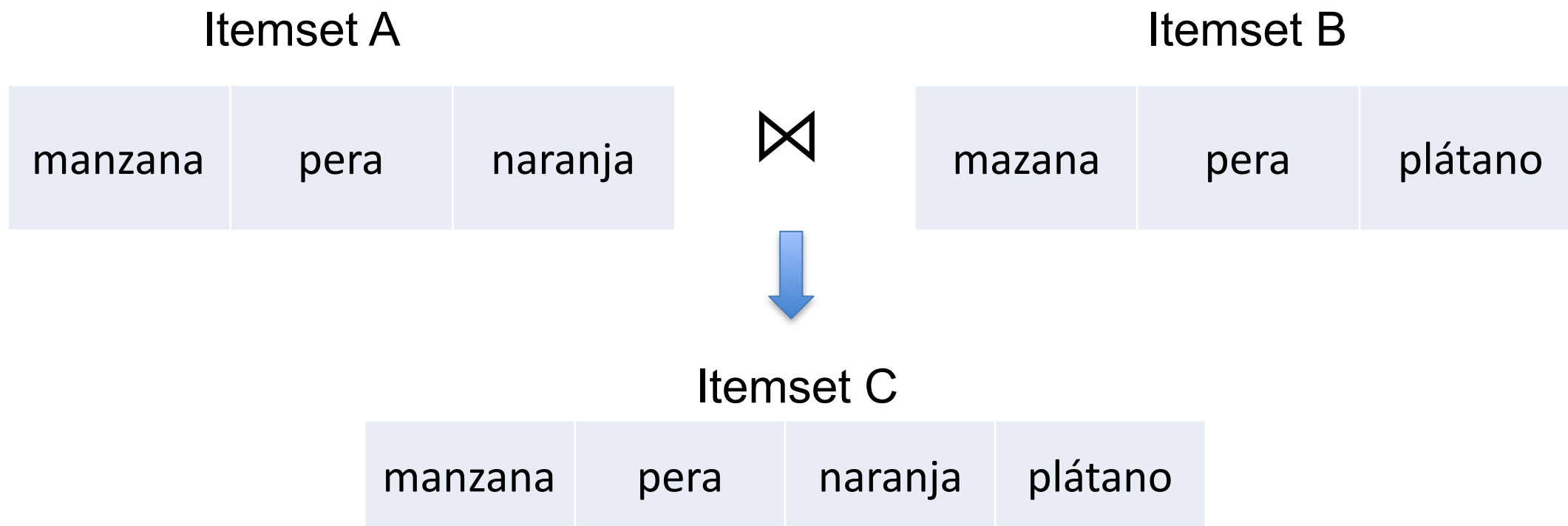


Resumen de itemsets combinables

- ▶ Dos itemsets son combinables si es que todos sus productos son iguales menos el último y, además,
- ▶ El último producto del itemset A debe ser menor que el último producto del itemset B, según el orden lexicográfico.

Combinación de 2 itemsets

- ▶ La combinación de 2 itemsets se realiza a través de la operación join (\bowtie)
- ▶ Ejemplo

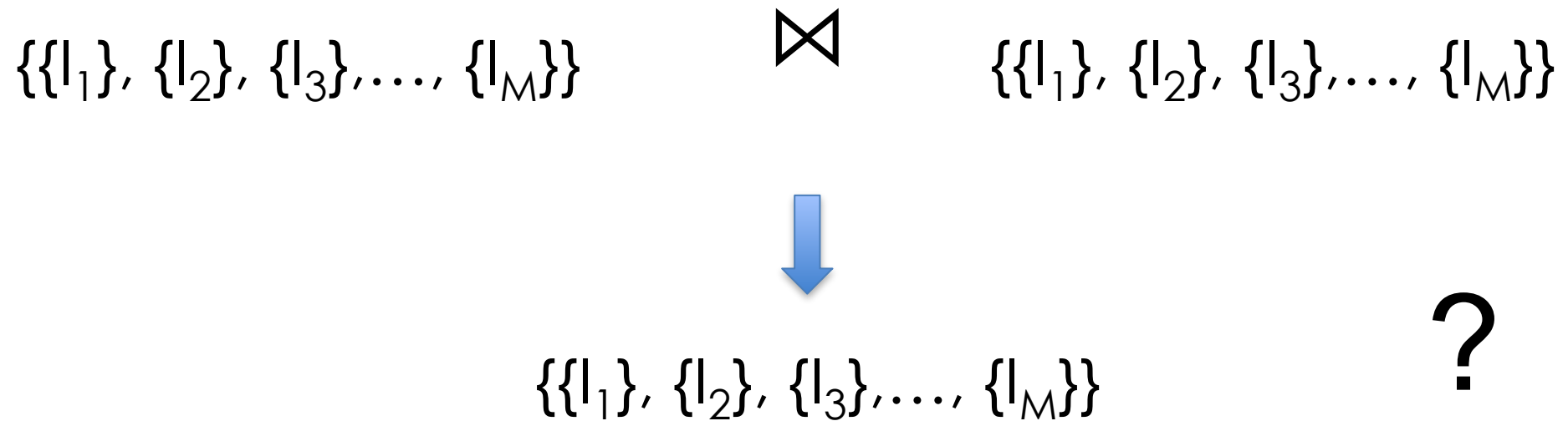


¿Cómo combinar M itemsets?

- ▶ Se debe realizar el Join del conjunto de itemsets consigo mismo.
- ▶ Se chequean todos los pares posibles y se combinan aquellos que cumplen con la condición de combinación.

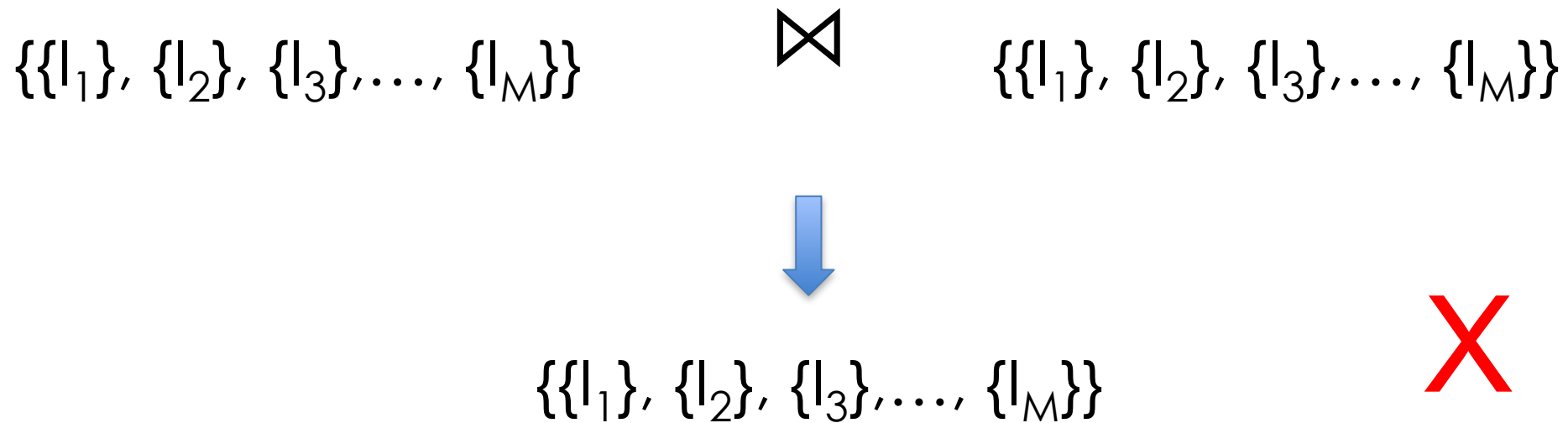
Ejemplo

- Revisemos $\{I_1\}$ con si mismo



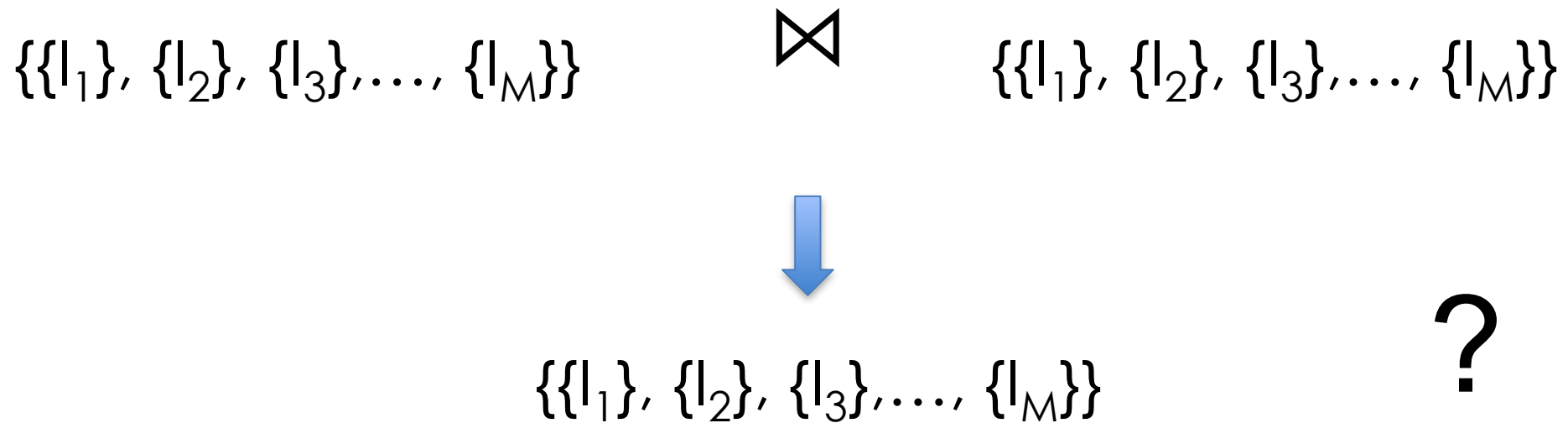
Ejemplo

- ▶ $\{l_1\}$ con es combinable consigo mismo porque no cumple con la condición que el último elemento sea diferente



Ejemplo

- El siguiente paso es comparar el segundo par posible $\{l_1\}$ con $\{l_2\}$ y evaluar si son combinables o no.



Ejemplo

- Proceso largo e iterativo que prueba todos los posibles pares de ítemets que cumplan con la condición

$\{\{I_1\}, \{I_2\}, \{I_3\}, \dots, \{I_M\}\}$



$\{\{I_1\}, \{I_2\}, \{I_3\}, \dots, \{I_M\}\}$

Ejemplo algoritmo Apriori

Ejemplo

- Consideremos el siguiente set de datos de transacciones

T	ítems
1	I1 , I2 , I4
2	I2 , I4 , I5
3	I1 , I3
4	I1 , I2 , I4
5	I1 , I2 , I3
6	I2 , I4
7	I1 , I3
8	I1 , I2 , I4 , I5
9	I1 , I2 , I3

Un umbral mínimo de
 $\frac{2}{9}$

Ejemplo

1) Generar el conjunto de itemsets candidatos de 1 item

$$C_1 = \{I_1, I_2, I_3, I_4, I_5\}$$

2) Evaluar el soporte de cada candidato

En este paso, se eliminarán los que no cumplen con el mínimo requerido

T	ítems
1	I1 , I2 , I4
2	I2 , I4 , I5
3	I1 , I3
4	I1 , I2 , I4
5	I1 , I2 , I3
6	I2 , I4
7	I1 , I3
8	I1 , I2 , I4 , I5
9	I1 , I2 , I3

$$s(\{I1\}) = \frac{7}{9} \quad \checkmark$$

$$s(\{I2\}) = \frac{7}{9} \quad \checkmark$$

$$s(\{I3\}) = \frac{4}{9} \quad \checkmark$$

$$s(\{I4\}) = \frac{5}{9} \quad \checkmark$$

$$s(\{I5\}) = \frac{2}{9} \quad \checkmark$$

Un umbral mínimo de
2/9

3) Se forma así el conjunto de itemset frecuentes de tamaño 1

$$L_1 = \{I_1, I_2, I_3, I_4, I_5\}$$

4) Se generan los nuevos itemsets de tamaño 2 a partir de L_1

$$L_1 = \{I_1, I_2, I_3, I_4, I_5\} \quad \bowtie \quad L_1 = \{I_1, I_2, I_3, I_4, I_5\}$$

$$C_2 = \{\{I_1, I_2\}, \{I_1, I_3\}, \{I_1, I_4\}, \{I_1, I_5\}, \\ \{I_2, I_3\}, \{I_2, I_4\}, \{I_2, I_5\}, \\ \{I_3, I_4\}, \{I_3, I_5\}, \\ \{I_4, I_5\}\}$$

T	ítems
1	I1 , I2 , I4
2	I2 , I4 , I5
3	I1 , I3
4	I1 , I2 , I4
5	I1 , I2 , I3
6	I2 , I4
7	I1 , I3
8	I1 , I2 , I4 , I5
9	I1 , I2 , I3

$$s(\{I1, I2\}) = \frac{5}{9} \quad \checkmark$$

$$s(\{I1, I3\}) = \frac{4}{9} \quad \checkmark$$

$$s(\{I1, I4\}) = \frac{3}{9} \quad \checkmark$$

$$s(\{I1, I5\}) = \frac{1}{9} \quad \times$$

$$s(\{I2, I3\}) = \frac{2}{9} \quad \checkmark$$

$$s(\{I2, I4\}) = \frac{5}{9} \quad \checkmark$$

$$s(\{I2, I5\}) = \frac{2}{9} \quad \checkmark$$

$$s(\{I3, I4\}) = \frac{0}{9} \quad \times$$

$$s(\{I3, I5\}) = \frac{0}{9} \quad \times$$

$$s(\{I4, I5\}) = \frac{2}{9} \quad \checkmark$$

5) Se forma así el conjunto de itemset frecuentes de tamaño 2 en la segunda iteración

$$L_2 = \{\{I1, I2\}, \{I1, I3\}, \{I1, I4\}, \\ \{I2, I3\}, \{I2, I4\}, \{I2, I5\}, \{I4, I5\}\}$$

6) Se generan los nuevos itemsets de tamaño 3 a partir de L_2

$$L_2 = \{\{I1, I2\}, \{I1, I3\}, \{I1, I4\}, \\ \{I2, I3\}, \{I2, I4\}, \{I2, I5\}, \{I4, I5\}\}$$



$$L_2 = \{\{I1, I2\}, \{I1, I3\}, \{I1, I4\}, \\ \{I2, I3\}, \{I2, I4\}, \{I2, I5\}, \{I4, I5\}\}$$

$$C_3 = \{\{I1, I2, I3\}, \{I1, I2, I4\}, \{I1, I3, I4\}, \\ \{I2, I3, I4\}, \{I2, I3, I5\}, \{I2, I4, I5\}\}$$

T	items
1	I1 , I2 , I4
2	I2 , I4 , I5
3	I1 , I3
4	I1 , I2 , I4
5	I1 , I2 , I3
6	I2 , I4
7	I1 , I3
8	I1 , I2 , I4 , I5
9	I1 , I2 , I3

$$s(\{I1, I2, I3\}) = \frac{2}{9} \quad \checkmark$$

$$s(\{I2, I3, I4\}) = \frac{0}{9} \quad \times$$

$$s(\{I1, I2, I4\}) = \frac{3}{9} \quad \checkmark$$

$$s(\{I2, I3, I5\}) = \frac{0}{9} \quad \times$$

$$s(\{I1, I3, I4\}) = \frac{0}{9} \quad \times$$

$$s(\{I2, I4, I5\}) = \frac{2}{9} \quad \checkmark$$

7) Se forma así el conjunto de itemset frecuentes de tamaño 3 en la tercera iteración

$$L_3 = \{\{I1, I2, I3\}, \{I1, I2, I4\}, \\ \{I2, I4, I5\}\}$$

8) Se generan los nuevos itemsets de tamaño 4 a partir de L_3

$$L_3 = \{\{I1, I2, I3\}, \{I1, I2, I4\}, \{I2, I4, I5\}\} \quad \boxtimes \quad L_3 = \{\{I1, I2, I3\}, \{I1, I2, I4\}, \{I2, I4, I5\}\}$$

$$C_4 = \{ \{I1, I2, I3, I4\} \}$$

T	ítems
1	I1 , I2 , I4
2	I2 , I4 , I5
3	I1 , I3
4	I1 , I2 , I4
5	I1 , I2 , I3
6	I2 , I4
7	I1 , I3
8	I1 , I2 , I4 , I5
9	I1 , I2 , I3

$$s(\{I1, I2, I3, I4\}) = \frac{0}{9} \quad \text{X}$$

9) No se cumple con el criterio de soporte por lo que tenemos el conjunto vacío

$$L_4 = \{ \}$$

10) Al no haber candidatos, se detienen las iteraciones

- Finalmente los itemsets frecuentes que quedan, y que tienen soporte mayor al umbral $2/9$ son:

$$L_1 = \{I_1, I_2, I_3, I_4, I_5\}$$

$$L_2 = \{\{I_1, I_2\}, \{I_1, I_3\}, \{I_1, I_4\}, \\ \{I_2, I_3\}, \{I_2, I_4\}, \{I_2, I_5\}, \{I_4, I_5\}\}$$

$$L_3 = \{\{I_1, I_2, I_3\}, \{I_1, I_2, I_4\}, \\ \{I_2, I_4, I_5\}\}$$

REFERENCIAS

- ▶ Han, J., Pei, J., & Kamber, M. (2011). *Data mining: concepts and techniques*. Elsevier.
- ▶ Witten, I. H., Frank, E., Hall, M. A., & Pal, C. J. (2016). *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann.
- ▶ Material del curso Minería de Datos IIC25433 profesor Karim Pichara
- ▶ Hand, D. J. (2006). Data Mining. *Encyclopedia of Environmetrics*, 2.