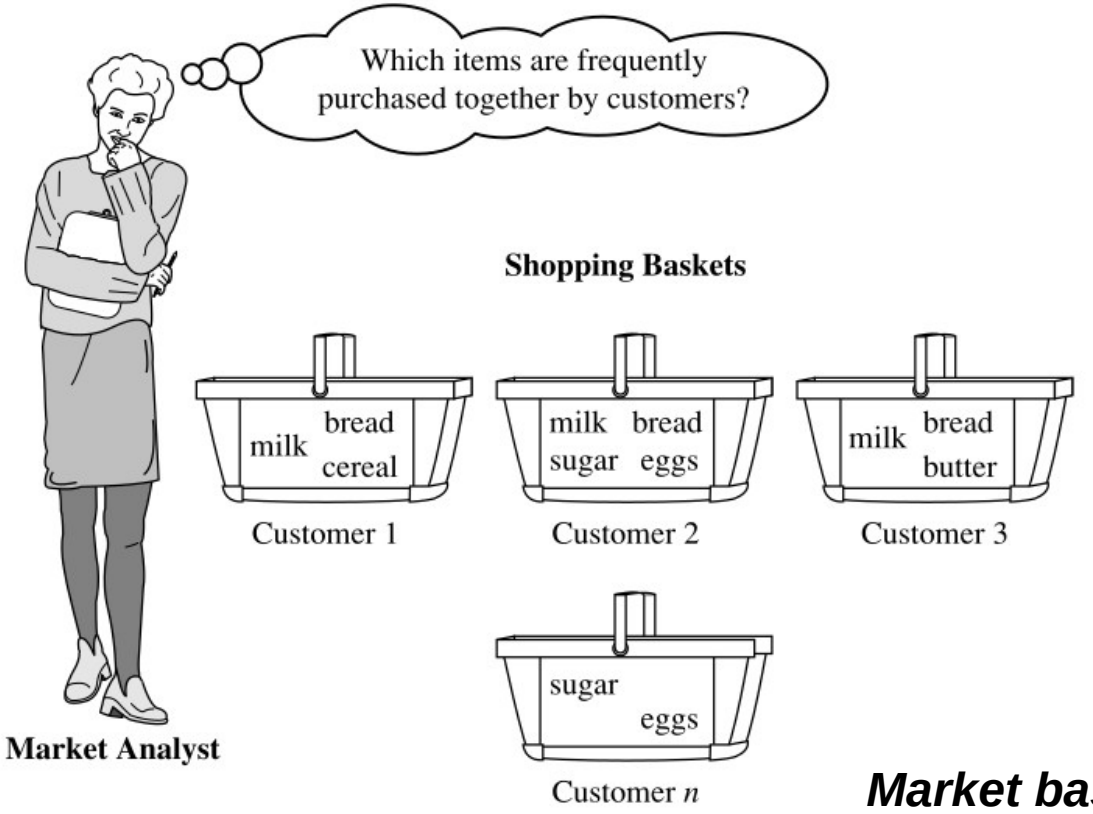
Análisis de Asociaciones

Oswaldo Solarte, Pd.D

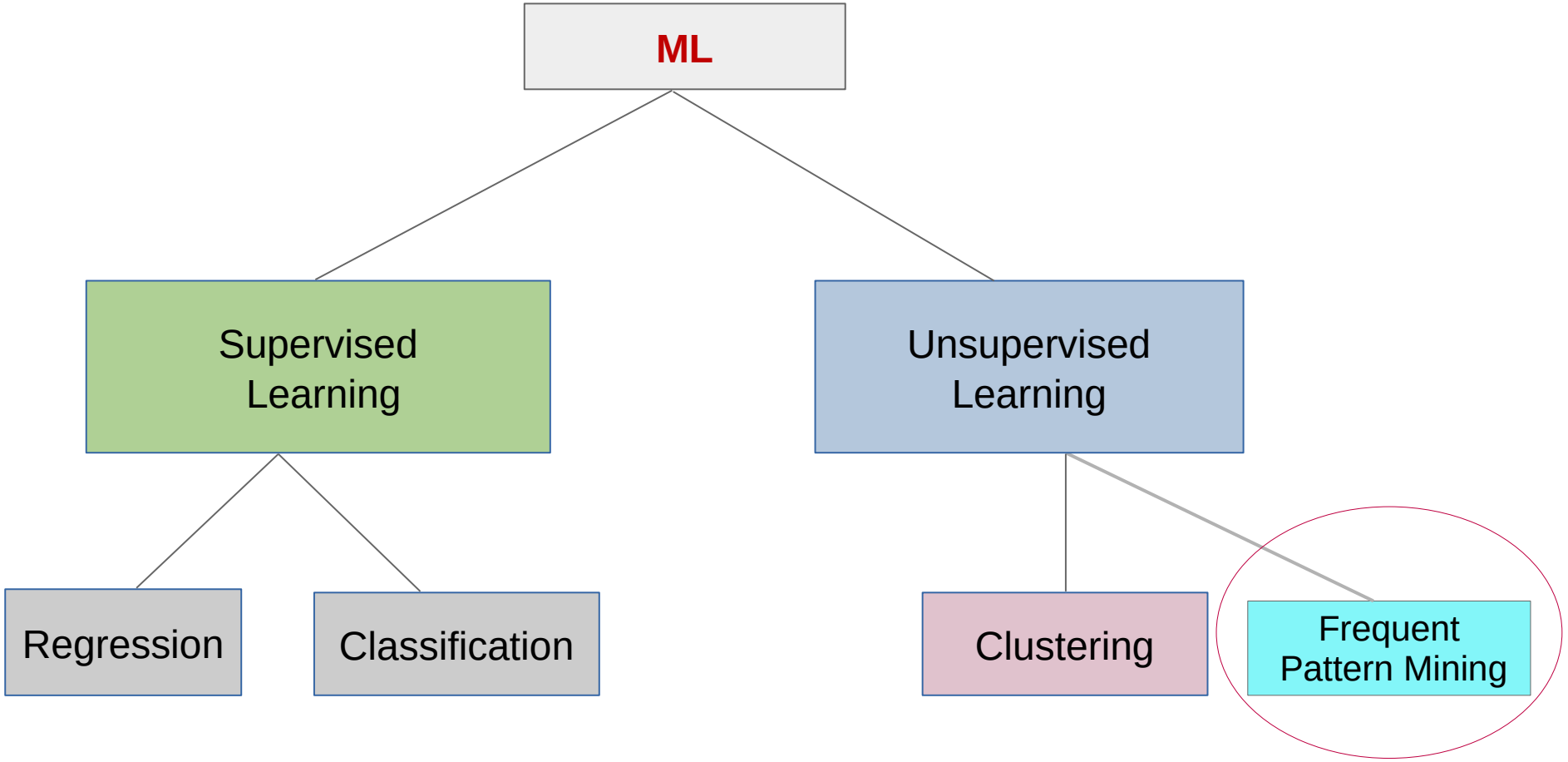
Frequent Pattern Mining

Frequent patterns are patterns (e.g., itemsets, subsequences, or substructures) that appear **frequently** in a data set.



Market basket analysis

Machine Learning



Análisis de Asociaciones

- Establecer vínculos entre los registros
- Asociaciones (productos que se compran juntos)
- Patrones secuenciales (si se compra algo en una fecha en x meses se adquiere otro producto)
- Secuencias similares. Detecta fenómenos con comportamientos similares

Reglas de asociación

Dado un conjunto de registros, encontrar reglas que predicen la **ocurrencia de un ítem, basándose en las ocurrencias de otros ítems** en el registro.

Transacciones

TID	Items
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

Reglas Asociación

Market-Basket transactions

TID	Items
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

Example:

TID	Bread	Milk	Diaper	Beer	Eggs	Coke
1	1	1	0	0	0	0
2	1	0	1	1	1	0
3	0	1	1	1	0	1
4	1	1	1	1	0	0
5	1	1	1	0	0	1

Reglas de asociación: Importancia

- Conocer **relación** entre un conjunto de datos, por ejemplo, la temperatura, el clima y la aparición de una enfermedad.
- **Tomar decisiones estratégicas** en un negocio: ubicación de productos en un supermercado.

Reglas de asociación

Estructura de las reglas

$$P \rightarrow C$$

P implica a C

$$P_1 \wedge P_2 \wedge P_3 \wedge \ldots \wedge P_m \rightarrow C_1 \wedge C_2 \wedge \ldots \wedge C_n$$

Donde cada P_i es una **premisa o antecedente** de la regla y cada C_j es una **consecuencia**

Reglas de asociación

Estructura de las reglas

$$computer \Rightarrow antivirus\_software \quad [support = 2\%, confidence = 60\%].$$

Edad (x,"30…39"), **salario** (x,"5-8 Millones") ⟶

Compra (x, Iphone)

Reglas de asociación

Medidas sobre reglas

Personal_computer → Printer

Laptop_computer → Digital_camera

¿Cuál de las reglas es más significativa?

¿Con qué certeza se puede asegurar la regla en un conjunto de datos?

Reglas de asociación: Evaluación

$$\text{Soporte}(A \rightarrow B) = P(A \wedge B)$$

$$\text{Confianza}(A \rightarrow B) = P(B|A)$$

Reglas de asociación

TID	Items
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

Association Rule: $X \overset{s,c}{\Rightarrow} y$

Support: $s = \frac{\sigma(X \cup y)}{|T|} (s = P(X, y))$

Confidence: $c = \frac{\sigma(X \cup y)}{\sigma(X)} (c = P(y \mid X))$

Goal:

Discover all rules having support ≥ *minsup* and confidence ≥ *minconf* thresholds.

Example: $\{Milk, Diaper\} \Rightarrow Beer$

Soporte y Confianza

TID	Items
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

Association Rule: $X \overset{s,c}{\Rightarrow} y$

Support: $s = \frac{\sigma(X \cup y)}{|T|} (s = P(X, y))$

Confidence: $c = \frac{\sigma(X \cup y)}{\sigma(X)} (c = P(y \mid X))$

Goal:

Discover all rules having support ≥ *minsup* and confidence ≥ *minconf* thresholds.

Example: $\{Milk, Diaper\} \Rightarrow Beer$

$$s = \frac{\sigma(Milk, Diaper, Beer)}{|T|} = \frac{2}{5} = 0.4$$

$$c = \frac{\sigma(Milk, Diaper, Beer)}{\sigma(Milk, Diaper)} = \frac{2}{3} = 0.67$$

Reglas de asociación

$$Soporte(X \rightarrow Y) = P(X \wedge Y)$$

$$Confianza(X \rightarrow Y) = P(Y|X)$$

X	Y
X	Y
X	Y
X	Z
X	Z
X	Z
W	X
W	Y

Reglas de asociación

$$Rule\ X \Longrightarrow Y$$

$$Support = \frac{Frequency\ (X,Y)}{N}$$

$$Confidence = \frac{Frequency\ (X,Y)}{Frequency(X)}$$

$$Lift = \frac{Support}{Support(X) * Support(Y)}$$

Reglas de asociación: Ejemplo

TID	Lista de Item
T100	milk, eggs, bread
T200	eggs, beer
T300	eggs, coke
T400	milk, eggs, beer
T500	milk, coke
T600	egs, coke
T700	milk, coke
T800	milk, eggs, coke, bread
T900	milk, eggs, coke

Milk=I1

Eggs=I2

Coke=I3

Beer=I4

Bread=I5

Se asignan identificadores a cada ítem

Reglas de asociación

Algoritmos para identificar reglas de asociación

1. Encontrar los **itemsets frecuentes**, aquellos conjuntos de ítems que aparecen cantidad dada de veces, conocida como minimum support count.

2. Generar **reglas de asociación** a partir de los itemsets frecuentes.

Reglas de asociación

Es el itemset {milk, coke} frecuente?.

Considere un *support count* de 2

TID	List of Item
T100	milk, eggs, bread
T200	eggs, beer
T300	eggs, coke
T400	milk, eggs, beer
T500	milk, coke
T600	egs, coke
T700	milk, eggs, coke

Algoritmo Apriori

Propiedad Apriori: ejemplo

Sea **{milk, beer, bread}** un *itemset frecuente*, entonces {milk, beer}, {milk, bread} y {beer, bread} deben ser también itemsets frecuentes

Algoritmo Apriori

- Busca itemsets frecuentes usando generación de candidatos
- Solo se generan aquellos itemsets candidatos que cumplan la propiedad apriori

Entrada: minimum support count + transacciones

Salida: itemsets frecuentes del tamaño más grande posible + frecuencia para cada itemset

Algoritmo Apriori [Agrawal '93]

Objetivo

Obtener **itemsets frecuentes** (conjuntos de valores que se repiten) de un determinado tamaño, para combinarlos en reglas

Algoritmo Apriori

- Algoritmo clásico **de reglas de asociación**
- Trabaja sobre bases de datos de transacciones
- El algoritmo busca subconjuntos de items que con comunes y que aparecen al menos un mínimo número de veces

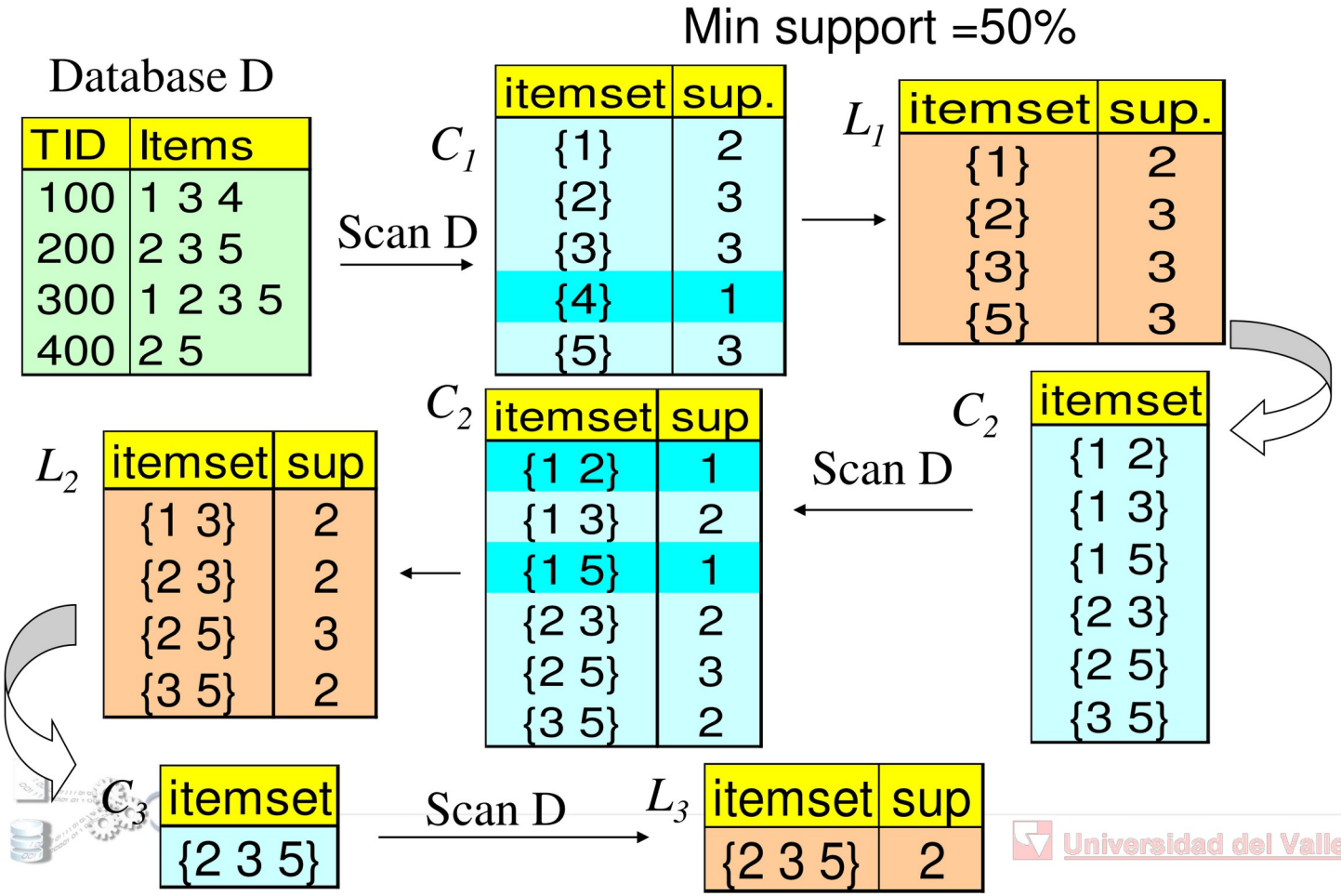
Algoritmo Apriori

- Apriori usa enfoque "bottom up”
  - subconjuntos de items frecuentes se extienden cada vez (generación de candidatos)
  - Grupos de candidatos se prueban con respecto a los datos

- Si no hay extensiones nuevas el algoritmo termina

- Apriori usa **busqueda en amplitud primero** y una estructura de **árbol hash** para contar itemsets eficientemente.

Ejemplo de A priori



e Han et al.)

Algoritmo Apriori (Agrawal et a.l , IBM Almaden Research Centre)

Apriori Algorithm: can be used to generate all frequent itemset
Pass 1 Generate the candidate itemsets in C_1
Save the frequent itemsets in L_1
Pass *k* Generate the candidate itemsets in C_k from the frequent itemsets in L_{k-1}

Join L_{k-1} *p* with L_{k-1}q, as follows:
insert into C_k
select *p*.item$_1$, *p*.item$_2$, . . . , *p*.item$_{k-1}$, *q*.item$_{k-1}$
from L_{k-1} *p*, L_{k-1}q
where *p*.item$_1$ = *q*.item$_1$, . . . *p*.item$_{k-2}$ = *q*.item$_{k-2}$, *p*.item$_{k-1}$ < *q*.item$_{k-1}$
Generate all (*k*-1)-subsets from the candidate itemsets in C_k
Prune all candidate itemsets from C_k where some (*k*-1)-subset of the candidate itemset is not in the frequent itemset L_{k-1}

Scan the transaction database to determine the support for each candidate itemset in C_k
Save the frequent itemsets in L_k

Algoritmo Apriori

TID	List of Item
T100	I1, I2, I5
T200	I2, I4
T300	I2, I3
T400	I1, I2, I4
T500	I1, I3
T600	I2, I3
T700	I1, I3
T800	I1, I2, I3, I5
T900	I1, I2, I3

Minimum support =2

Algoritmo Apriori

Scan DB for count of each candidate

C1

Itemset	Sup count
{I1}	
{I2}	
{I3}	
{I4}	
{I5}	

Compare support

Algoritmo Apriori

C1

Itemset	Sup count
{I1}	6
{I2}	7
{I3}	6
{I4}	2
{I5}	2

Algoritmo Apriori

Itemset	Sup count
{I1}	6
{I2}	7
{I3}	6
{I4}	2
{I5}	2

C1: candidatos de tamaño 1

Algoritmo Apriori

Scan DB for count of each candidate

Itemset	Sup count
{I1}	6
{I2}	7
{I3}	6
{I4}	2
{I5}	2

Compare support

C1: candidatos de tamaño 1

Los itemset que no pasen el soporte mínimo se eliminan

Algoritmo Apriori

L1

Itemset	Sup count
{I1}	6
{I2}	7
{I3}	6
{I4}	2
{I5}	2

Generate C2

Algoritmo Apriori

Generate C2

Itemset

Scan DB for count of each candidate

Para generar C2 se hace el join de L1 con L1

C2

Algoritmo Apriori

Generate C2

Itemset
{I1 ,I2}
{I1 .I3}
{I1 ,I4}
{I1 ,I5}
{I2,I3}
{I2,I4}
{I2,I5}
{I3,I4}
{I3,I5}
{I4,I5}

Algoritmo Apriori

TID	List of Item
T100	I1, I2, I5
T200	I2, I4
T300	I2, I3
T400	I1, I2, I4
T500	I1, I3
T600	I2, I3
T700	I1, I3
T800	I1, I2, I3, I5
T900	I1, I2, I3

Algoritmo Apriori

Scan DB for count of each candidte

C2

Itemset	Sup count
{I1,I2}	4
{I1,I3}	4
{I1,I4}	1
{I1,I5}	2
{I2,I3}	4
{I2,I4}	2
{I2,I5}	2
{I3,I4}	0
{I3,I5}	1
{I4,I5}	0

Compare support

Algoritmo Apriori

L2

Itemset	Sup count
{I1,I2}	4
{I1,I3}	4
{I1,I5}	2
{I2,I3}	4
{I2,I4}	2
{I2,I5}	2

Compare support

Generate C3

Algoritmo Apriori

Generate C3

C3

Itemset
{I1,I2, I3}
{I1.I2, I5}

Scan DB for count of each candidate

Apply Apriori principle

Algoritmo Apriori

Scan DB for count of each candidate

C3

Itemset	Sup count
{I1,I2, I3}	2
{I1.I2, I5}	2

Compare support

L3

Itemset	Sup count
{I1,I2, I3}	2
{I1.I2, I5}	2

Algoritmo Apriori

C4

Itemset
{I1,I2, I3, I5}

El itemset {I2,I3,I5} no es frecuente

Generación de reglas

Para cada itemset frecuente *l*, generar todos los subconjuntos no vacíos de *l*

Para cada subconjunto *s* de *l* genere la regla $s \rightarrow (l\text{-}s)$, si:

$$\text{Soporte(l)/soporte (s)} \geq \text{min\_conf}$$

donde, min_conf es la confianza mínima determinada.

Reglas de asociación

Considere el itemset frecuente {I1,I2,I5}
Cuáles son las reglas que se pueden generar?

$I1 \wedge I2 \rightarrow I5$
$I1 \wedge I5 \rightarrow I2$
$I2 \wedge I5 \rightarrow I1$
$I1 \rightarrow I2 \wedge I5$
$I2 \rightarrow I1 \wedge I5$
$I5 \rightarrow I1 \wedge I2$

Reglas de asociación

$I1 \wedge I2 \rightarrow I5$, confianza=2/4=50%

$I1 \wedge I5 \rightarrow I2$, confianza=2/2=100%

$I2 \wedge I5 \rightarrow I1$, confianza=2/2=100%

$I1 \rightarrow I2 \wedge I5$, confianza=2/6=33%

$I2 \rightarrow I1 \wedge I5$, confianza=2/7=29%

$I5 \rightarrow I1 \wedge I2$, confianza=2/2=100%

Análisis de la cesta de la compra

- Se utiliza la información de lo que compran los clientes para intentar descubrir **¿Quién?** Y **¿Cómo ?** Se compran esos productos.

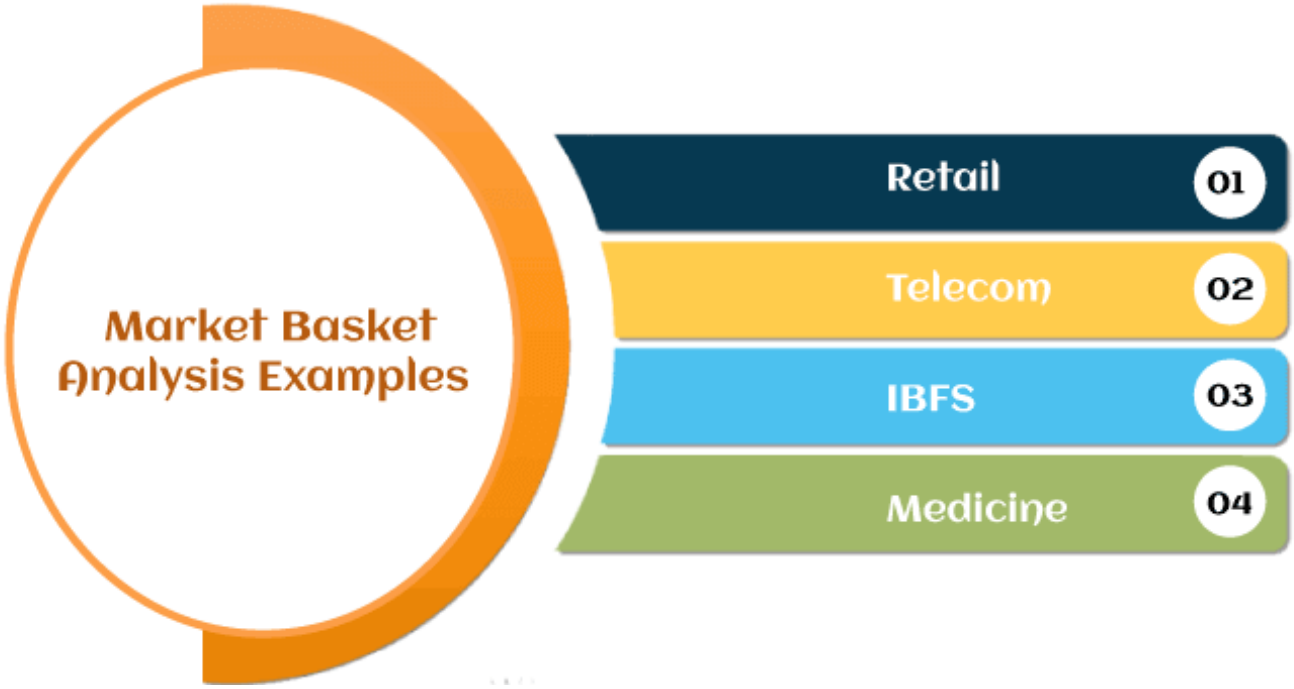
Beneficios ???

Análisis de la cesta de la compra

- Se utiliza la información de lo que compran los clientes para intentar descubrir ¿Quién? Y ¿Cómo ? Se compran esos productos.

Análisis de la cesta de la compra

- Se utiliza la información de lo que compran los clientes para intentar descubrir ¿Quién? Y ¿Cómo ? Se compran esos productos.



Referencias

- Agrawal R, Imielinski T, Swami AN. "Mining Association Rules between Sets of Items in Large Databases.“SIGMOD. June 1993, 22(2):207-16
- Agrawal R, Srikant R. "Fast Algorithms for Mining Association Rules", VLDBSep 12-15 1994, Chile, 487-99,.
- Mannila H, Toivonen H, Verkamo AI. "Efficient algorithms for discovering association rules." AAAI *Workshop on Knowledge Discovery in Databases (SIGKDD)*. July 1994, Seattle, 181-92,