**La minería de datos** se refiere a la actividad de revisar grandes conjuntos de datos para buscar información relevante o pertinente. Este tipo de actividad es realmente un buen ejemplo del antiguo axioma "buscando una aguja en un pajar". La idea es que las empresas recopilen conjuntos masivos de datos que pueden ser homogéneos o recopilados automáticamente. Los tomadores de decisiones necesitan acceder a datos más pequeños y específicos de esos conjuntos grandes. Utilizan la minería de datos para descubrir las piezas de información que informarán al liderazgo y ayudarán a trazar el curso para un negocio.

Reglas de asociación

### La minería de asociación

La asociación de minería se usa comúnmente para hacer recomendaciones de productos al identificar productos que se compran con frecuencia. Pero, si no tiene cuidado, las reglas pueden dar resultados confusos en ciertos casos.

La minería de asociación generalmente se realiza con datos de transacciones de un mercado minorista o de una tienda de comercio electrónico en línea. Como la mayoría de los datos de transacciones son grandes, el a priori algoritmo facilita la búsqueda de estos patrones o *reglas* rápidamente.

Una regla es una notación que representa qué artículo / s se compra con frecuencia con qué artículo / s. Tiene una parte de *LHS* y una parte de *RHS* y se puede representar de la siguiente manera:

**conjunto de elementos A => conjunto de elementos B**

Esto significa que los artículos de la derecha fueron comprados frecuentemente junto con los artículos de la izquierda.

<http://r-statistics.co/Association-Mining-With-R.html>

* Encontrar patrones frecuentes, asociaciones, correlaciones, o estructuras causales entre conjuntos de elementos u objetos en bases de datos de transacciones, bases de datos relacionales, y otros repositorios de información.

Dado un conjunto de transacciones, la minería de reglas de asociación tiene como objetivo encontrar las reglas que nos permitan predecir la aparición de un artículo específico en función de las ocurrencias de los otros artículos en la transacción.

* aplicaciones
* Análisis de datos de la cesta.
* Marketing cruzado
* Diseño de catálogo
* …

La minería de reglas de asociación es el proceso de extracción de datos para encontrar las reglas que pueden regir las asociaciones y los objetos causales entre conjuntos de elementos.   
  
Entonces, en una transacción dada con múltiples artículos, trata de encontrar las reglas que gobiernan cómo o por qué tales artículos se compran a menudo juntos. Por ejemplo, la mantequilla de maní y la jalea a menudo se compran juntas porque a mucha gente le gusta hacer sándwiches PB&J. También sorprendentemente, los pañales y la cerveza se compran juntos porque, como resulta, los papás tienen la tarea de hacer las compras mientras las madres se quedan con el bebé.

<https://www.techopedia.com/definition/30306/association-rule-mining>

### Aplicaciones

* **Análisis de datos de la cesta:** cada transacción del cliente que contiene el conjunto de artículos comprados. El objetivo es encontrar grupos de elementos que sean frecuentemente comprados juntos. Uso: Re organizar los productos en el supermercado. Análisis de datos de la cesta: consiste en analizar la asociación de los artículos comprados en una sola cesta o en una sola compra según los ejemplos que se dieron anteriormente.
* **Telecomunicaciones:** Cada cliente es una transacción que contiene el conjunto de llamadas telefónicas o los productos que se contratan (teléfono, internet, tv) para ofrecer paquetes.
* **e-Bussines:** diseño web, con qué secuencias están asociadas comprando o dejando el proceso. Sugerencia de otros productos (Amazon).
* **Tarjetas de crédito / servicios bancarios :** cada tarjeta/cuenta es una transacción que tiene el conjunto de pagos del cliente.
* **Tratamientos médicos:** cada paciente está representado como una transacción. Contiene el conjunto ordenado de enfermedades.
* **Análisis del juego de baloncesto:** cada juego se representa como una transacción que contiene el conjunto ordenado de pases de balón.
* **Marketing cruzado:** es trabajar con otras empresas que complementen las suyas, no con la competencia. Por ejemplo, los concesionarios de vehículos y los fabricantes realizan campañas de mercado cruzadas con compañías de petróleo y gas por razones obvias.
* **Diseño del catálogo:** la selección de artículos en el catálogo de una empresa a menudo está diseñada para complementarse entre sí, de modo que comprar un articulo llevará a la compra de otro. Así que estos artículos suelen ser complementos o muy relacionados.

### Datos

La transacción es un conjunto de elementos. T = {item1, item2, …, itemN}

* Transacciones = documentos; elementos (ítems) = palabras.
* Que palabras aparecen a menudo juntas en los documentos. Debe indicar la existencia de conceptos complejos.
* Transacciones = documentos, elementos = frases.
* Que frases aparecen a menudo juntas en los documentos. Debe indicar la existencia de plagio.
* Transacciones = pagina web; artículos = etiquetas HTML.
* Puede indicar la duplicación de la web (necesita investigación para la web 2.0, mucho más interesante de analizar).
* Transacciones = cursos; artículos = inscripciones.
* Que cursos son tomados frecuentemente juntos por los estudiantes. Lo Hacemos en la FME, útil para hacer previsiones de matriculación.

### Análisis de la cesta de mercado (MBA)

**Objetivo**: Generar reglas de asociación entre diferentes compras -> Encuentra grupos de artículos que frecuentemente se compran juntos. Encontrar reglas interesantes para tomar decisiones.

Descubrir asociaciones entre productos al buscar combinaciones de productos que con frecuencia ocurren en transacciones. En otras palabras, permite a los supermercados identificar las relaciones entre los productos que las personas compran. Por ejemplo, es probable que los clientes que compran un lápiz y papel compren una goma o una regla.

Dentro de la base de datos, hay miles y miles de reglas ...

* La mayoría de las reglas son triviales.
* Algunas son inexplicables (debido a la aleatoriedad de los datos).
* Solo un conjunto pequeño es de oro: digno de tomar decisiones reales.

Por lo tanto, es necesario realizar una minería de datos (a posteriori) sobre las reglas para encontrar los buenos. El MBA puede ser meramente exploratorio, veamos que se puede encontrar.

* Datos : Base de datos de transacciones de clientes
* Individual: transacciones del cliente, donde cada transacción es un conjunto de ítems.
* Transacción 1: {Pan, Nueces, Leche, Fruta, Mermelada}
* Transacción 2: {Pan, Mermelada, Soda, Patatas, Leche, Fruta}

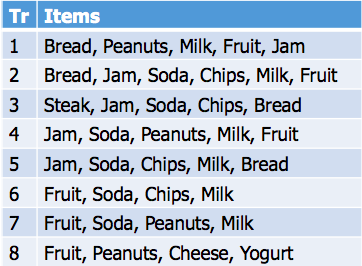
Cada transacción representa un grupo de artículos o productos que se han comprado juntos y, a menudo, se los denomina "conjunto de elementos". Las transacciones se analizan para identificar las reglas de asociación.

<https://select-statistics.co.uk/blog/market-basket-analysis-understanding-customer-behaviour/>

### Asociación de minería de reglas

Dado un conjunto de transacciones, encuentra reglas que predicen la aparición de un ítem en función de las apariciones de otros ítems en la transacción.

Ejemplo:

{Pan} 🡪 {Leche}

{Pan} 🡪 {Mermelada}

{Soda} 🡪 {Patatas}

### Notación y Definición

I : Conjunto de ítems .|I| es muy alto ( I > 100.000 )

I = {x1,x2,…,xn}

: Conjunto de transacciones. | | es enorme ( > 1.000.000)

Transacciones = { T | T I }

Itemset :

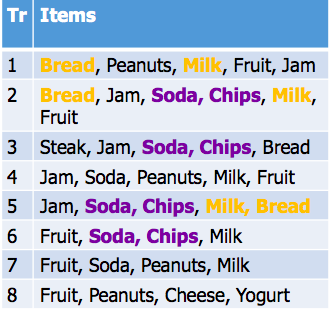
* Colección de uno o más ítems :
* Ejemplo : {Pan, Leche, Mermelada}
* K-itemset, un itemset que contiene k ítems:

k-itemset X I, k = |X|

Regla de asociación X Y | X, Y itemsets y XY = ø

**Definición :**

Itemset frecuente 🡪

* Recuento de apoyo
* Frecuencia de ocurrencia de un conjunto de ítems.
* ({Leche, Pan}) = 3
* ({Soda, Chips}) = 4
* = P(XY)
* Fracción de las transacciones que contienen el conjunto de ítems (itemset) X.
* Proporción de transacciones en la bd que contienen dicho conjunto de ítems.
* Support ({Leche, Pan}) = 3/8
* Support ({Soda, Chips}) = 4/8
* Itemset frecuente
* Un conjunto de ítems X cuyo support(X) >= umbral de soporte mínimo.

### Regla de asociación

Grupo de ítems que ocurren juntos (itemsets) 🡪 conocimientos representados por reglas.

se utilizan para descubrir hechos que ocurren en común dentro de un determinado conjunto de datos.

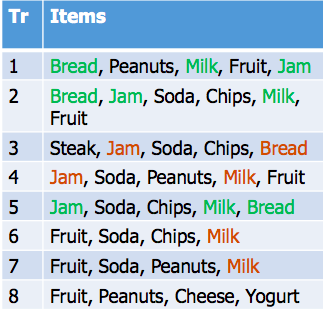
**Regla:** Condición 🡪 Resultado

* Condición: (lista de) ítems (itemset)
* Resultado: (lista de) ítems (itemset)
* Ex: {Leche, Cacao} 🡪 {Cereales}

### Medidas de Reglas de Asociación

* Con qué frecuencia aparece el conjunto de itemset (X U Y) en el conjunto de datos.
* Cuanto más alto sea el soporte más frecuente es la regla en el conjunto de datos.
* el porcentaje de transacciones que contienen todos los elementos en un conjunto de elementos. Cuanto mayor sea el soporte, más frecuentemente se produce el conjunto de elementos. Se prefieren las reglas con un alto soporte, ya que es probable que sean aplicables a un gran número de transacciones futuras.
* Con qué frecuencia se ha encontrado que la regla es verdadera.
* Cuanta mayor confianza mayor proporción de transacciones que contienen X que también contienen Y.
* La probabilidad de que una transacción que contenga los elementos en el lado izquierdo de la regla también contenga el elemento en el lado derecho. Cuanto mayor sea la confianza, mayor será la probabilidad de que se compre el artículo del lado derecho
* Relación entre el soporte observado y esperado si X e Y fueran independientes.
* Si Lift (Elevación) = 1 la probabilidad de ocurrencia de X e Y es independiente → No se puede dibujar regla
* Si Lift > 1 grado de dependencia → Regla potencialmente útil para predecir
* Si Lift < 1 los ítems se sustituyen entre sí → La presencia de X tiene efecto negativo sobre la presencia de Y y viceversa
* La elevación considera tanto la confianza de la regla como el conjunto de datos general
* La probabilidad de que todos los elementos de una regla se produzcan juntos divididos por el producto de las probabilidades de los elementos del lado izquierdo y derecho que se producen como si no hubiera asociación entre ellos. En general, lift resume la fuerza de asociación entre los productos del lado izquierdo y derecho de la regla; cuanto mayor sea la elevación mayor será el enlace entre los dos productos.

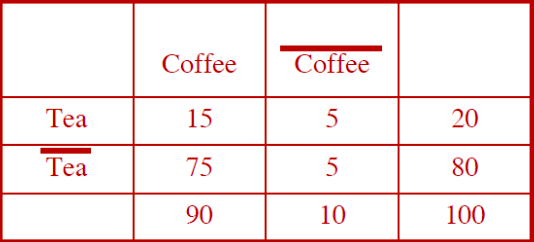
**Ejemplo:**



1. {Bread, Jam} → {Milk} s=3/8 c=0.75 l=1
2. {Milk, Jam} → {Bread} s=3/8 c=0.75 l=1.5
3. {Bread} → {Milk, Jam} s=3/8 c=0.75 l=1.5
4. {Jam} → {Bread, Milk} s=3/8 c=0.6 l=1.2
5. {Milk} → {Bread, Jam} s=3/8 c=0.5 l=1

* Todas las reglas anteriores son particiones binarias del mismo Itemlist {Milk, Bread, Jam}
* Todas las reglas tienen un soporte idéntico, pero pueden tener una confianza o elevamiento diferente
* Podemos desacoplar los requisitos de soporte, confianza y elevación.

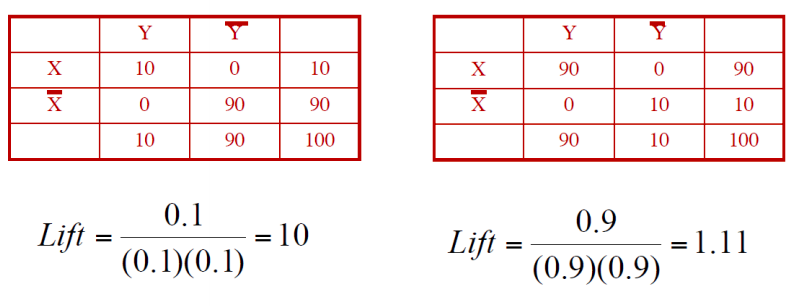
### Inconveniente de la confianza



* Regla de asociación: Tea → Coffee
* Confianza ({Coffee, Tea}) = 0.75 pero la confianza ({Coffee}) = 0.9
* Aunque la confianza es alta, la regla es engañosa
* Confianza ({Coffee, No Tea}) = 0.9375
* Elevación ({Coffee, Tea}) = 0.75/0.9 = 0.8333 (< 1 por lo tanto es asociación negativa)

### Inconveniente de la elevación

### 



<https://es.wikipedia.org/wiki/Reglas_de_asociaci%C3%B3n>

### Encontrar reglas de asociación

* Dado un conjunto de transacciones, el objetivo de la minería de reglas es encontrar todas las reglas que tengan:
* Soporte >= umbral de soporte mínimo.
* Confianza >= umbral de confianza mínimo.
* Elevación -> usado en nuevas versiones de los algoritmos.
* Enfoque de fuerza bruta:
* Listar todas las reglas de asociación posibles.
* Calcular el soporte y confianza de cada regla.
* Las reglas de poda que fallan el soporte y umbrales de confianza mínimos.
* Este enfoque es computacionalmente prohibitivo!!!!!!!

### Macintosh HD:Users:Sofia:Desktop:Captura de pantalla 2018-10-20 a las 0.25.11.pngGeneración de itemsets

* Todos los posibles itemsets:
* N ítems 🡪 2N itemsets posibles.
* 5 ítems iniciales: A, B, C, D, E

### Estrategias de generación de itemsets frecuentes

* Reducir el número de candidatos (M)
* Búsqueda completa M = 2N
* Usar técnicas de poda para reducir M
* Reducir el número de transacciones (T)
* Reducir el tamaño de T a medida que aumenta el tamaño del itemset
* Reducir el número de comparaciones (MT)
* Utilizar estructuras de datos eficientes para almacenar los candidatos o transacciones.
* No es necesario que cada candidato coincida con cada transacción.

### Algoritmos

* Encontrar itemsets frecuentes
* Generar todos los itemsets con soporte >= soporte mínimo.
* Computacionalmente más caro.
* Algoritmo de Eclat (Transformación de clase de equivalencia) [Zaki, Parthasarathy, Li (1997)]
* FP-growth Algorithm
* Buscar reglas de asociación
* Generar todas las reglas con confianza >= confianza mínima y/o soporte >= soporte mínimo.
* Apriori [Agrawal (1993)]
* Berry-Larrubia [PFC, Tomas Aluja, 2014]

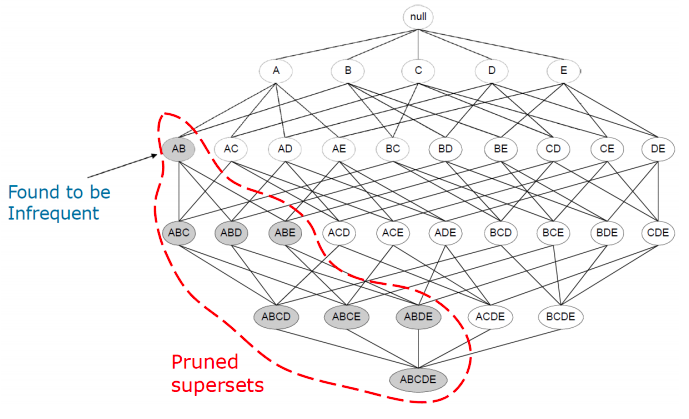
Apriori

Estrategia de búsqueda en primer lugar para contar el soporte de itemsets.

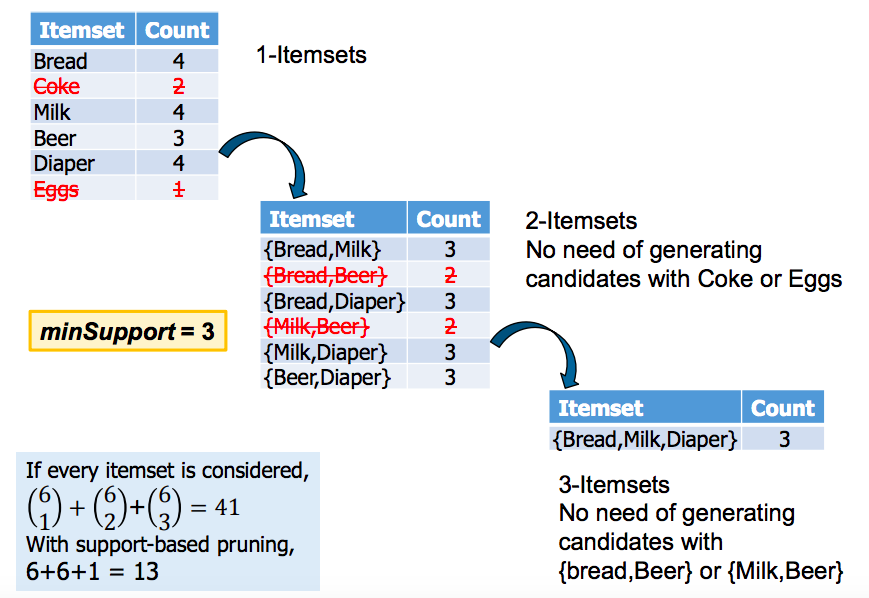
* Utiliza una función de generación candidata que explota la propiedad anti-monótona del soporte.
* Probablemente el más usado.
* Varias mejoras en la literatura.
* Procede identificando los ítems individuales frecuentes en la base y extendiéndolos a conjuntos de mayor tamaño siempre y cuando esos conjuntos de datos aparezcan suficientemente seguidos en dicha base de datos
* Pasos:
* Sistemáticamente identificar itemsets que se producen con frecuencia en el conjunto de datos con un soporte mayor que un umbral pre-especificado.
* Calcular la confianza de todas las reglas posibles dados los itemsets frecuentes y mantener solo aquellos con una confianza mayor que un umbral pre-especificado.

Propiedad de soporte 🡪 Propiedad anti-monótona de soporte:

∀𝑋, 𝑌: 𝑋 ⊆ 𝑌 → 𝑠 X ≥ 𝑠 Y



* Sea k = 1
* Generar itemsets frecuentes de longitud 1 (Lk)
* Repetir hasta que no se identifiquen nuevos itemsets frecuentes
* Generar Ck + 1: itemsets candidatos de longitud k+1 utilizando el itemsets frecuente de LK
* Pese a los itemsets que eran infrecuentes en CK: Todos los itemsets que incluyen un conjunto de ítems eliminados cuando se creo LK desde CK
* Contar el soporte de cada candidato en CK +1 usando todas las transacciones
* Generar LK +1 incluyendo todos los conjuntos de elementos candidatos de CK +1 que son frecuentes (soporte >= mínimo soporte)
* K = k+1
* Devolver la unión de todos los LK.

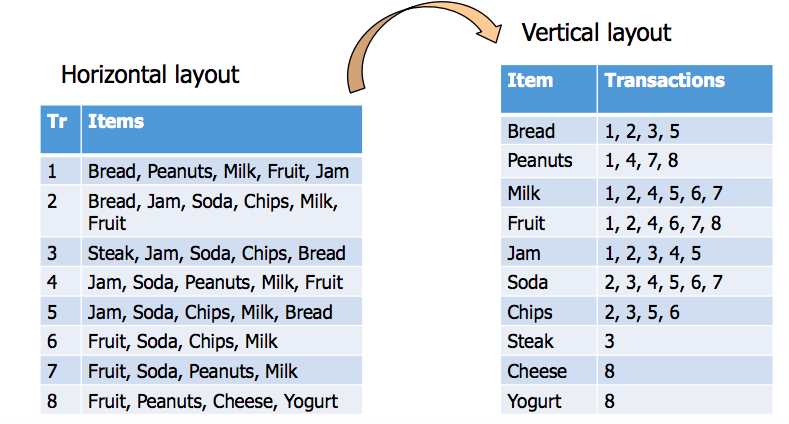


### Bottlenecks of Apriori

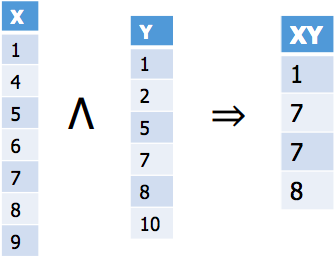
* La generación de candidatos puede parecer enorme
* 1-Itemset 104 generará 2-itemsets candidatos 107.
* Para descubrir un 100-itemset frecuente 🡪 generar 2100 ~ 1030 candidatos.
* Escaneo múltiple de todas las transacciones:
* Análisis de necesidades (k+1) donde k es la longitud del itemset frecuente más largo.

Eclat (Equivalence Class Transformation)

* Otro metro para la generación de itemsets frecuentes.
* Diseño de datos vertical: para cada ítem, almacenar una lista de identificadores de transacción.



* Determina el soporte de cualquier conjunto de elementos k mediante la intersección de la lista de transacciones de dos de sus subconjuntos k-1.



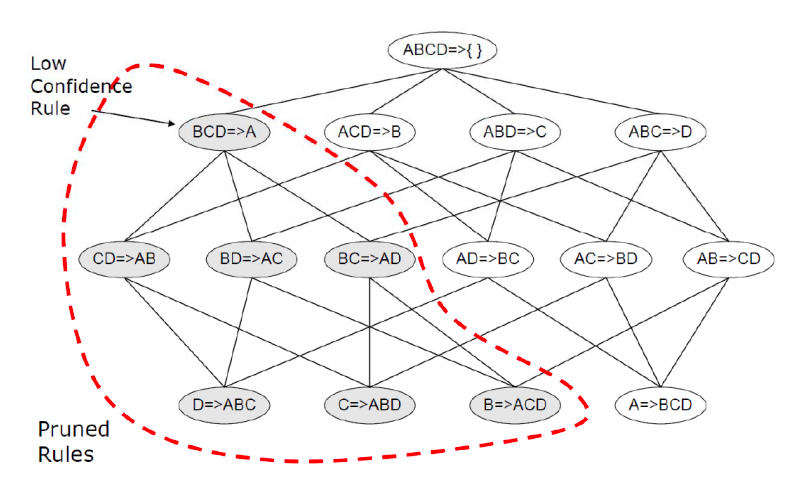
* Primer algoritmo de búsqueda basado en la intersección de conjuntos.
* 3 aproximaciones transversales.
* De arriba abajo, de abajo a arriba, hibrido.
* Ventajas: conteo de soporte muy rápido.
* Desventaja: la lista intermedia de transacciones puede ser demasiado grande para la memoria.

### Generación de Reglas

* Dado un conjunto de elementos (itemset) L frecuente, encontrar todos los subconjuntos no vacíos f ⊂ L de manera que f → L - f satisfaga el requisito de confianza mínimo.
* Si {A,B,C,D} es un frecuente itemset, las reglas candidatas serán:
* ABC→D, ABD→C, ACD→B, BCD→A,
* AB→CD, AC→BD, AD→BC, BC→AD, BD→AC, CD→AB,
* A→BCD, B→ACD, C→ABD, D→ABC
* Si |L| = k, entonces hay 2k – 2 reglas de asociación candidatas (ignorando L → ø y ø → L).

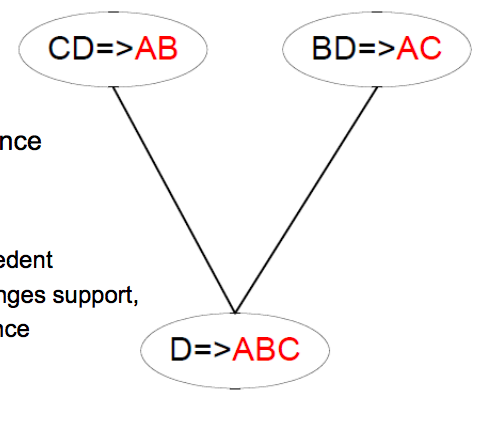
Estrategia de Confianza (Confidence Strategy)

* Como generar reglas de manera eficiente a partir de itemsets frecuentes?
* La confianza no tiene una propiedad anti-monótona
* c (ABC → D) puede ser mayor o menor que c (AB → D)
* Pero la confianza en las reglas generadas a partir del mismo conjunto de elementos tiene una propiedad anti-monótona.
* Ex, L = {A, B, C, D}: c (ABC → D) ≥ c (AB → CD) ≥ c (A → BCD)
* La confianza es anti-monótona con respecto al número de ítems en el lado derecho de la regla.



Generación de regla para el algoritmo a priori

* La regla candidata se genera al combinar dos reglas que comparten el mismo prefijo en la regla consecuente.
* JOIN (CD => AB, BD => AC) producirá la regla candidata D => ABC.
* Podar la regla D => ABC si su subconjunto AD => BC no tiene confianza alta.
* Dato clave: mover ítems del antecedente al consecuente nunca cambia el soporte y nunca aumenta la confianza



### Cómo configurar un apropiado soporte mínimo?

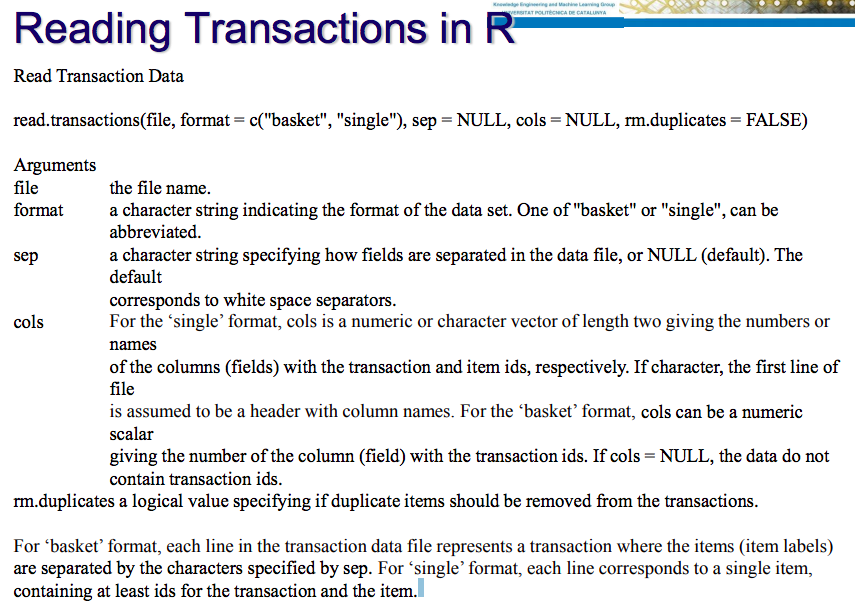
### Si el soporte mínimo es demasiado alto:

* perder los conjuntos de elementos que implican elementos raros interesantes.
* ej. productos caros
* asociación entre cervezas y pañales
* ...

### Si el soporte mínimo es demasiado bajo:

* Computacionalmente caro
* Numero de itemsets es muy grande
* Solución no global: un umbral de soporte mínimo único puede no ser efectivo.

### R 🡪



### Eclat in R

* Eclat algorithm
* mbaeclat<-eclat(Groceries)
* inspect(mbaeclat)
* mbaeclat<-eclat(Groceries, parameter = list(support=0.01))
* Eclat only generates itemsets, use ruleInduction to transform into rules.
* mbaeclatrules<-ruleInduction(mbaeclat,Groceries,confidence=0.1)

<https://select-statistics.co.uk/blog/market-basket-analysis-understanding-customer-behaviour/> 🡪 R

<http://r-statistics.co/Association-Mining-With-R.html> 🡪 R

<https://en.wikipedia.org/wiki/Apriori_algorithm> 🡪 Algoritmo a priori

<https://es.wikipedia.org/wiki/Algoritmo_apriori> 🡪 Algoritmo a priori