Minería de Reglas de Asociación

Felipe Bravo
(Basado en una versión previa de
Bárbara Poblete)

Minería de reglas de asociación

- Métodos para encontrar relaciones entre atributos en grandes volúmenes de datos.

- Estas relaciones se presentan como **reglas de asociación** o **conjuntos de ítems frecuentes.**

- Es una tarea no-supervisada.

- A diferencia de clustering aquí encontramos asociaciones entre atributos en vez de agrupar instancias.

Minería de reglas de asociación

- Criterio principal para evaluar un algoritmo de reglas de asociación: eficiencia computacional.

- Primer algoritmo eficiente fue presentado en 1993 en SIGMOD (congreso importante en bases de datos).

- Idealmente trabajamos sobre **datos categóricos y binarios.**

- Existen adaptaciones para trabajar con secuencias, grafos, y otros tipos de entradas estructuradas.

¿Para qué?

- Muchos negocios acumulan grandes cantidades de datos sobre sus operaciones diarias.
  - Ejemplo: transacciones en una tienda de retail.
- Aprender de estos datos permiten entender el comportamiento adquisitivo/comercial de los clientes
- Esta información se puede utilizar para la toma de decisiones en un negocio (manejo de inventario, promoción de productos para venta cruzada)

Ejemplo

- Tenemos las siguientes transacciones en una canasta de compra:

TID	Items
1	{Bread, Milk}
2	{Bread, Diapers, Beer, Eggs}
3	{Milk, Diapers, Beer, Cola}
4	{Bread, Milk, Diapers, Beer}
5	{Bread, Milk, Diapers, Cola}

- La siguiente regla se podría extraer de estos datos:

$$\{\texttt{Diapers}\} \longrightarrow \{\texttt{Beer}\}$$

- Esta regla sugiere una relación entre la venta de pañales y cerveza porque varios clientes que compraron pañales también compraron cerveza.
- Un retailer puede usar este tipo de reglas para encontrar oportunidades de venta cruzada de productos.

Dominios de Aplicación

- Además del análisis de canasta de compra, las reglas de asociación pueden ser útiles en varios dominios: bioinformática, diagnóstico médico, web mining, análisis de datos científicos.
- En ciencias de la tierra: las asociaciones pueden revelar conexiones entre el océano, la tierra, y procesos atmosféricos.

Definiciones

- **Itemset**
  - Un conjunto de uno o más elementos
    - Ej: {Milk, Bread, Diaper}
  - k-itemset
    - ítemset que contiene k ítems
- **Support count (σ)**
  - Frecuencia con que ocurre un ítemset
  - Ej. σ({Milk, Bread, Diaper}) = 2
- **Support**
  - Fracción de las transacciones que contiene un ítemset
  - Ej. s({Milk, Bread, Diaper}) = 2/5
- **Itemset frecuente**
  - Un ítemset cuyo support es mayor o igual a un parámetro *minsup*

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

Definición: Regla de Asociación

- Regla de Asociación
  - Expresión de implicancia de la forma $X \rightarrow Y$, donde X (antecedente) e Y (consecuente) son ítemsets.
  - Ejemplo: {Milk, Diaper} → {Beer}
- Métricas para evaluar las reglas
  - Soporte (s) o Support en inglés
    - Fracción de las transacciones que contienen a ambos X e Y

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

  - Confianza (C) o confianza en inglés.
    - Mide qué tan frecuentemente los ítems en Y aparecen en transacciones que contienen X

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

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

Ejemplo:

$$\{\text{Milk}, \text{Diaper}\} \Rightarrow \text{Beer}$$

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

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

Minería de Reglas de Asociación

- Dado un conjunto de transacciones T, el objetivo de la minería de reglas de asociación es encontrar todas las reglas que tengan
  - support ≥ minsup
  - confidence ≥ minconf

Transacciones comerciales

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

Ejemplo de Reglas de Asociación

{Diaper} → {Beer},
{Milk, Bread} → {Eggs,Coke},
{Beer, Bread} → {Milk},

¡¡La implicancia significa co-ocurrencia y no causalidad!!

Para tener en cuenta

- A diferencia de clasificación y clustering, aquí buscamos una solución exacta.
- Siempre pueden ocurrir asociaciones aleatorias de carácter espontáneo sin valor.
- El resultado debe ser interpretado con precaución .
- Una fuerte correlación no necesariamente implica causalidad.

¿Por qué usamos support y confidence?

- Si el soporte es muy bajo
  - X e Y pueden haber co-ocurrido por azar
  - También es poco interesante del punto de vista del negocio
  - Sirve para eliminar reglas poco interesantes

¿Por qué usamos support y confidence?

- La confianza, mide qué tanto podemos "confiar" en la inferencia hecha por la regla.
- Mientras mayor sea la confianza, mayor será la probabilidad de observar Y en transacciones que tengan X.
- La confianza estima la probabilidad condicional: P(Y|X).
- Co-ocurrencia no es causalidad (causalidad implica una relación temporal entre las variables)

Minería de Reglas de Asociación

- Aproximación por Fuerza-Bruta:
  - Listar todas las reglas de asociación posibles (R)
  - Calcular el support y confidence para cada regla
  - filtrar las reglas que no cumplan con las restricciones de *minsup* y *minconf*
  - Computacionalmente prohibitivo! Para d ítems

$$R = 3^d - 2^{d+1} + 1$$

Minería de 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

Ejemplo de Reglas:

{Milk,Diaper} → {Beer} (s=0.4, c=0.67)
{Milk,Beer} → {Diaper} (s=0.4, c=1.0)
{Diaper,Beer} → {Milk} (s=0.4, c=0.67)
{Beer} → {Milk,Diaper} (s=0.4, c=0.67)
{Diaper} → {Milk,Beer} (s=0.4, c=0.5)
{Milk} → {Diaper,Beer} (s=0.4, c=0.5)

Observaciones:

- Todas las reglas listadas vienen del mismo itemset:
  {Milk, Diaper, Beer}
- Las reglas que originan del mismo itemset tienen idéntico support pero pueden tener diferente confidence
- El cálculo del support sólo depende de $X \cup Y$

Generación de Reglas de Asociación

- Cómo todas las posibles reglas provenientes del mismo itemset tienen el mismo soporte podemos dividir la tarea en 2 pasos:

1. Generación de ítemsets frecuentes
   - Generar todos los ítemsets cuyo support ≥ minsup

2. Generación de Reglas
   - Generar reglas de alto confidence para cada itemset, donde cada regla es una partición binaria de un ítemset frecuente

Aún así, la generación de patrones frecuentes es muy costosa

Generación de ítemsets frecuentes

- Estrategia fuerza bruta: generar todos los itemsets y descartar todos los que no cumplan minsup.
  - Eso requiere comparaciones del orden O(NMw) con N el número de transacciones, $M = 2^k - 1$ el número de itemsets de tamaño k, y w es el largo de la transacciones con más ítems.
- Estrategia inteligente: reducir el número de itemsets candidatos a ser evaluados.

El Principio Apriori

- Si un itemset es frecuente, entonces todos sus subconjuntos son frecuentes.

- De manera análoga, si un itemset es infrecuente todos sus superconjuntos son infrecuentes.

Todos los Itemsets posibles

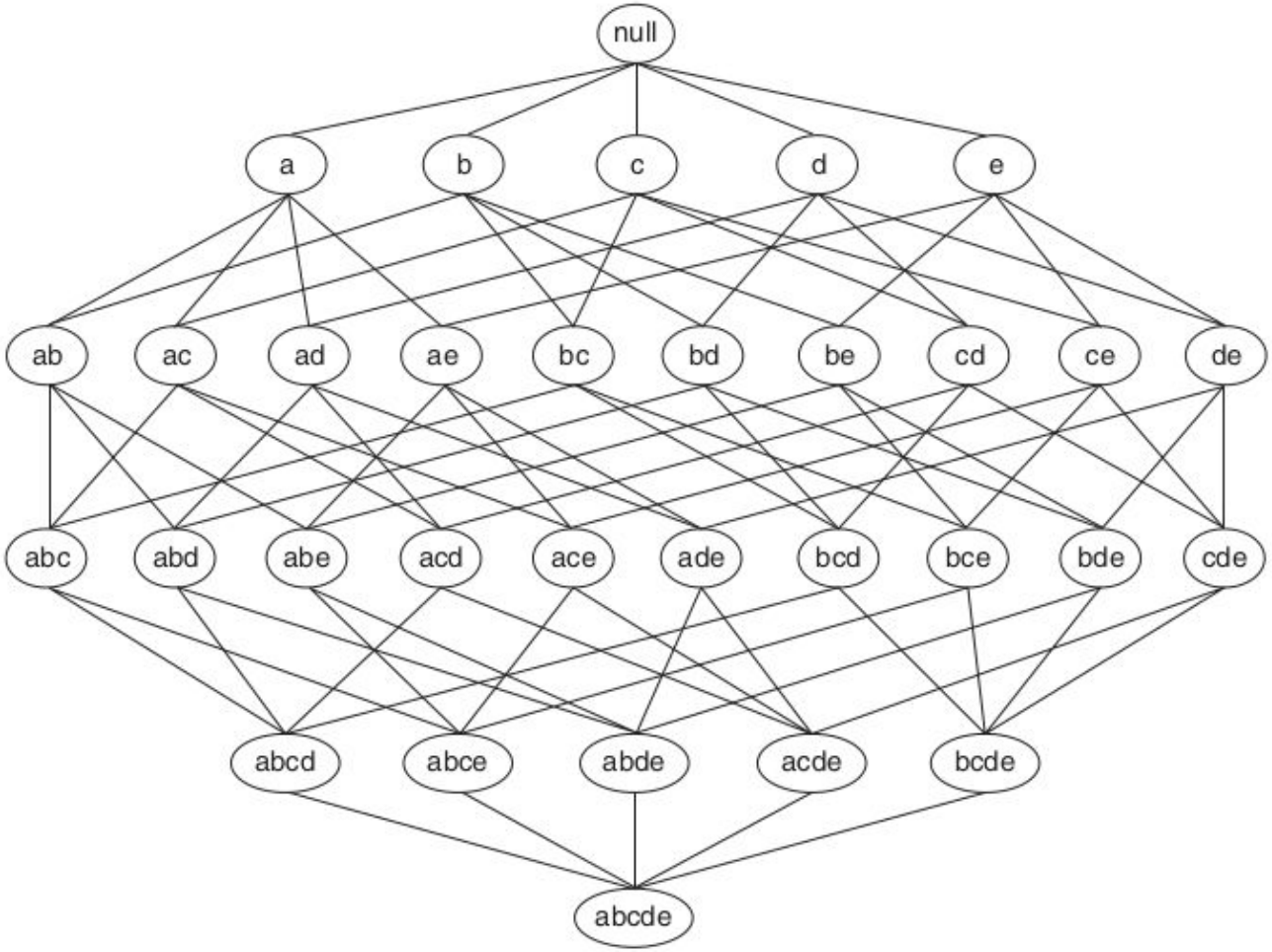
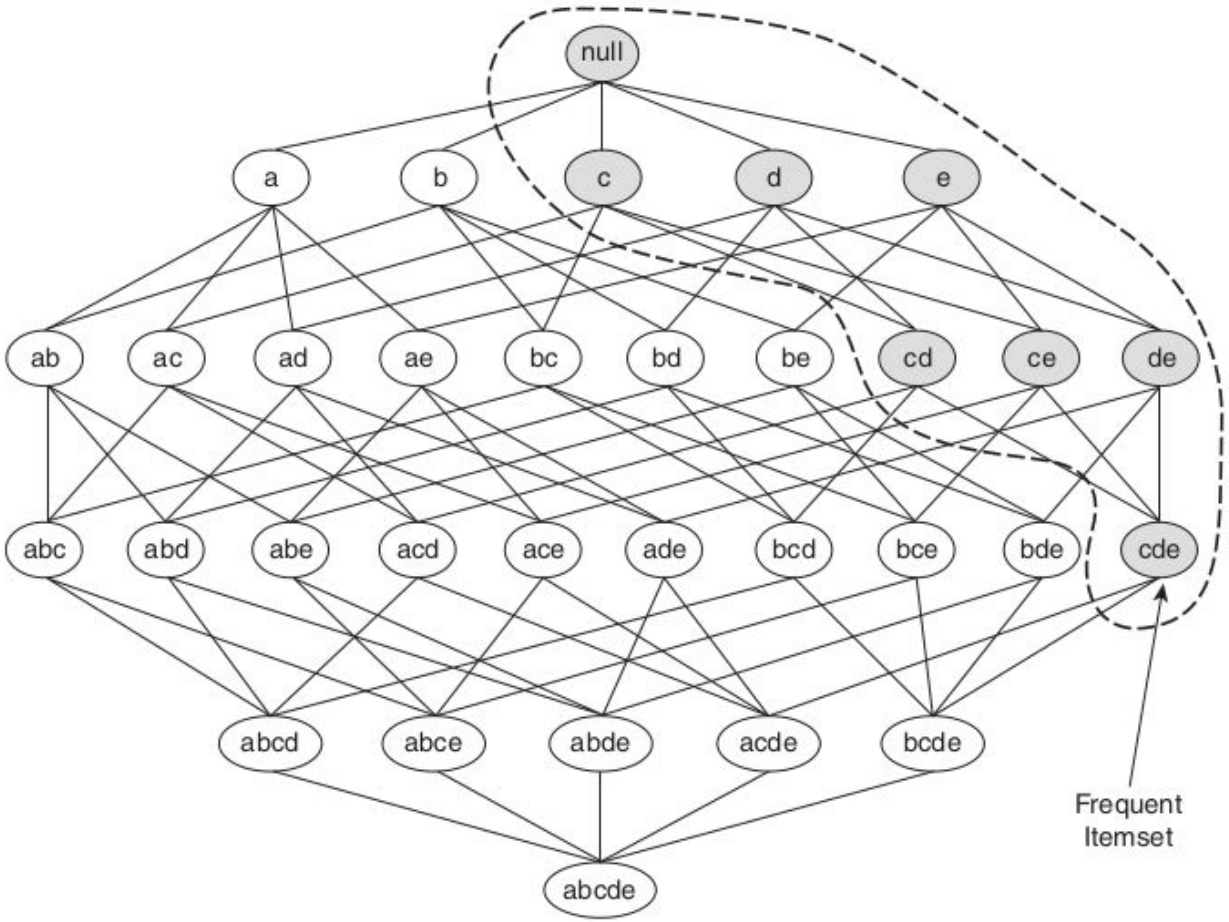


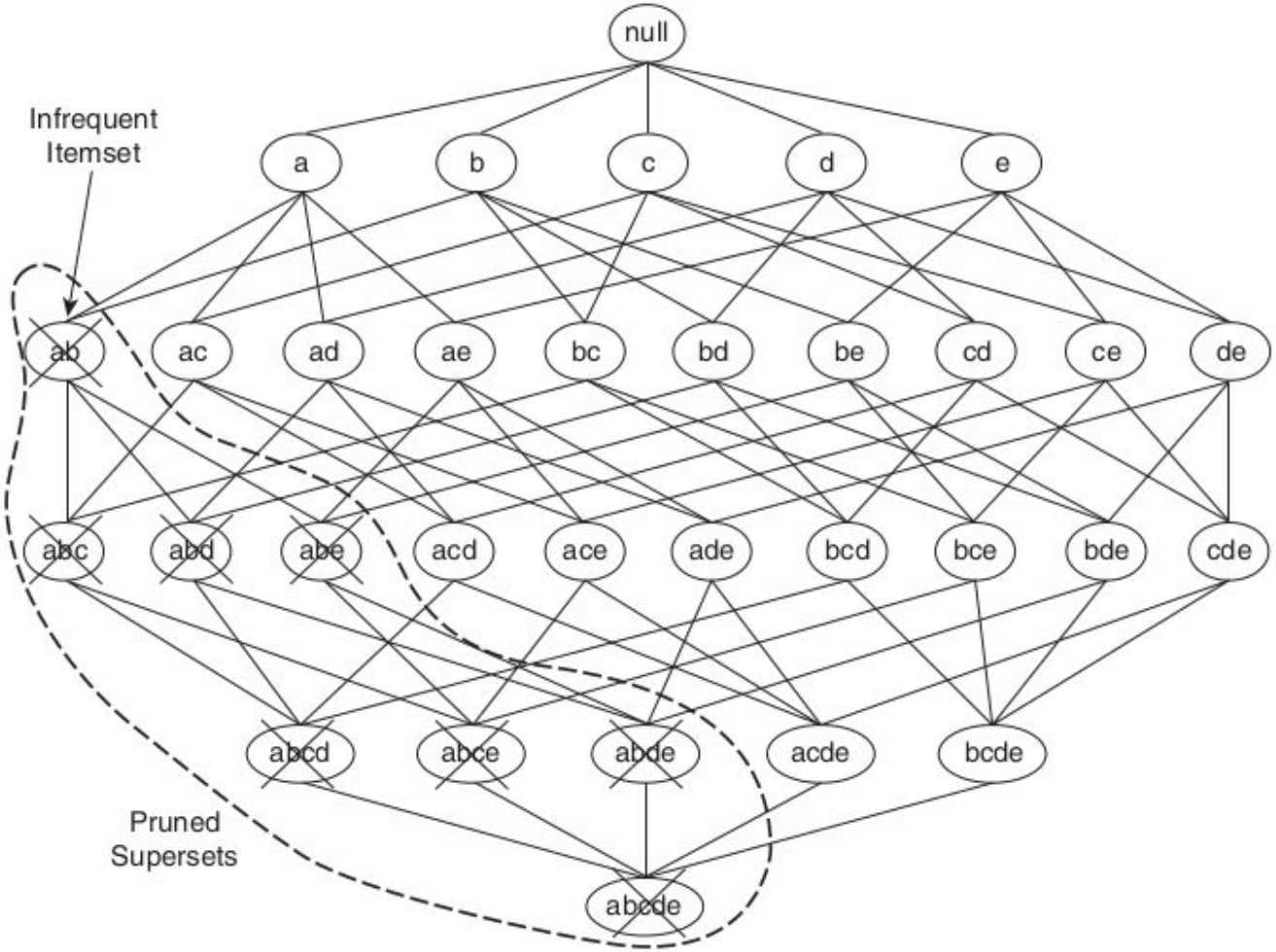
Figure 5.1. An itemset lattice.

Principio Apriori



Si {c,d,e} es frecuente, todos sus subconjuntos son frecuentes.

Principio Apriori



Si {a,b} es infrecuente, todos sus superconjuntos son infrecuentes.

Generación de Itemsets Frecuentes usando Apriori

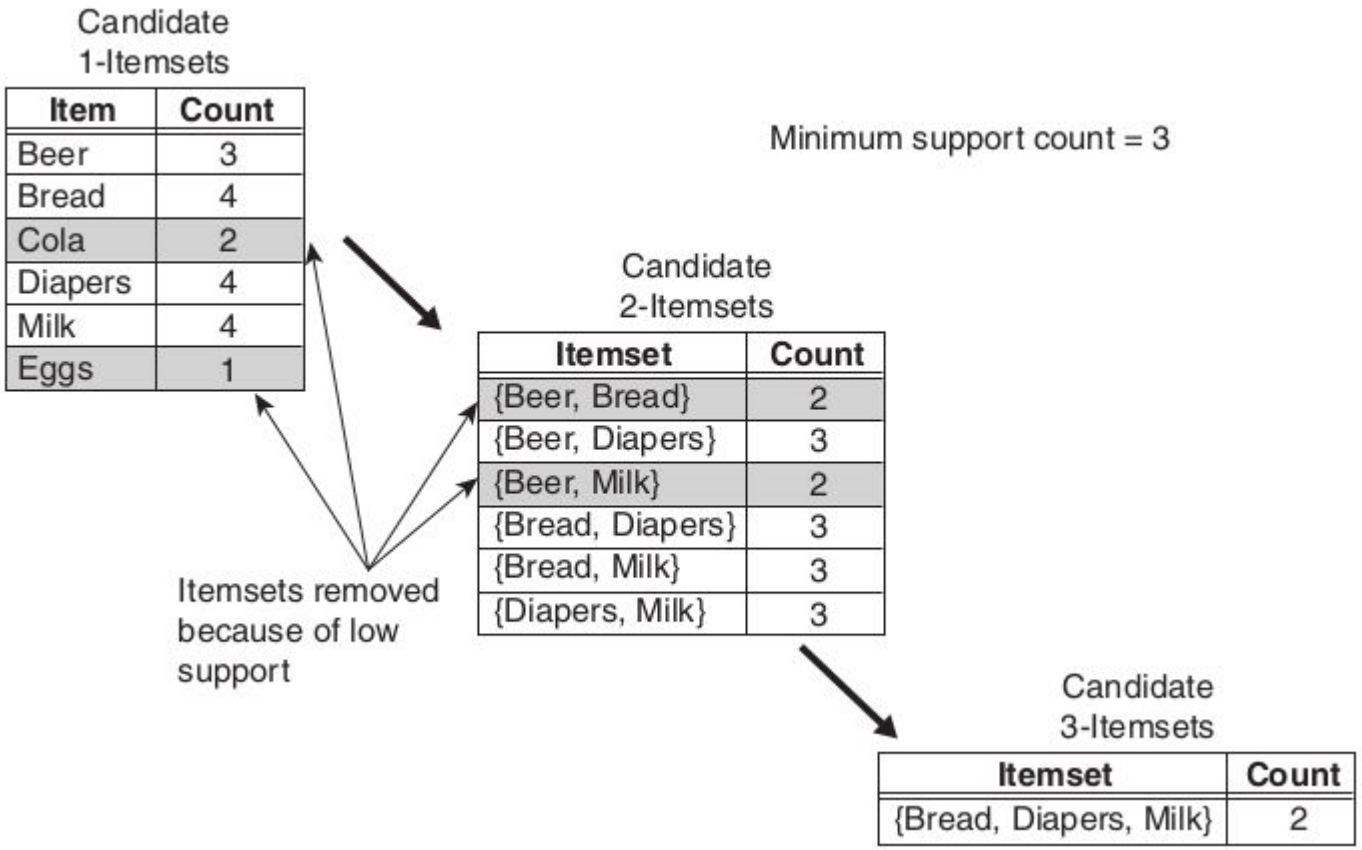
- Apriori es el primer algoritmo para encontrar reglas de asociación que usa **poda** basada en soporte para mitigar el crecimiento exponencial de los itemsets candidatos.

- Objetivo: reducir la cantidad de candidatos a itemsets frecuentes aprovechando el principio apriori.

- Encontrar 1-itemsets es fácil: se escanea la base de datos y se cuenta la frecuencia de cada ítem.

- Idea: mezclar pares de 1-itemsets frecuentes para encontrar candidatos a 2-itemsets frecuentes, luego repetir usando pares de 2-itemsets frecuentes para encontrar candidatos a 3-itemsets, y así sucesivamente.

Ejemplo: Generación de Itemsets Frecuentes usando Apriori



Generación de Itemsets Frecuentes usando Apriori

- Por el principio Apriori sabemos que si X es un k-itemset frecuente, entonces todos sus (k-1)-item subsets son frecuentes también.

- Estrategia: encontrar k-itemsets mezclando (k-1)-itemsets frecuentes.
- Los ítems dentro de un itemset se ordenan lexicográficamente y así sólo mezclamos pares de itemsets que difieren en su último ítem.
  - Esto nos asegura que no generamos dos veces el mismo k-itemset combinando (k-1)-itemsets.
  - Ejemplo: {b,c,a} y {a,c,b} se transforman a {a,b,c}.

- Para encontrar candidatos de k-itemsets frecuentes, sólo mezclamos (k-1)-itemsets que tengan los mismos k-2 primeros ítems.

Ejemplo

- Tenemos cinco 3-itemsets frecuentes

  (A B C), (A B D), (A C D), (A C E), (B C D)

  - Sólo mezclamos pares de ítemsets que difieren en su último ítem: (A B C) con (A B D) y (A C D) con (A C E).

- Candidatos a 4-itemsets:

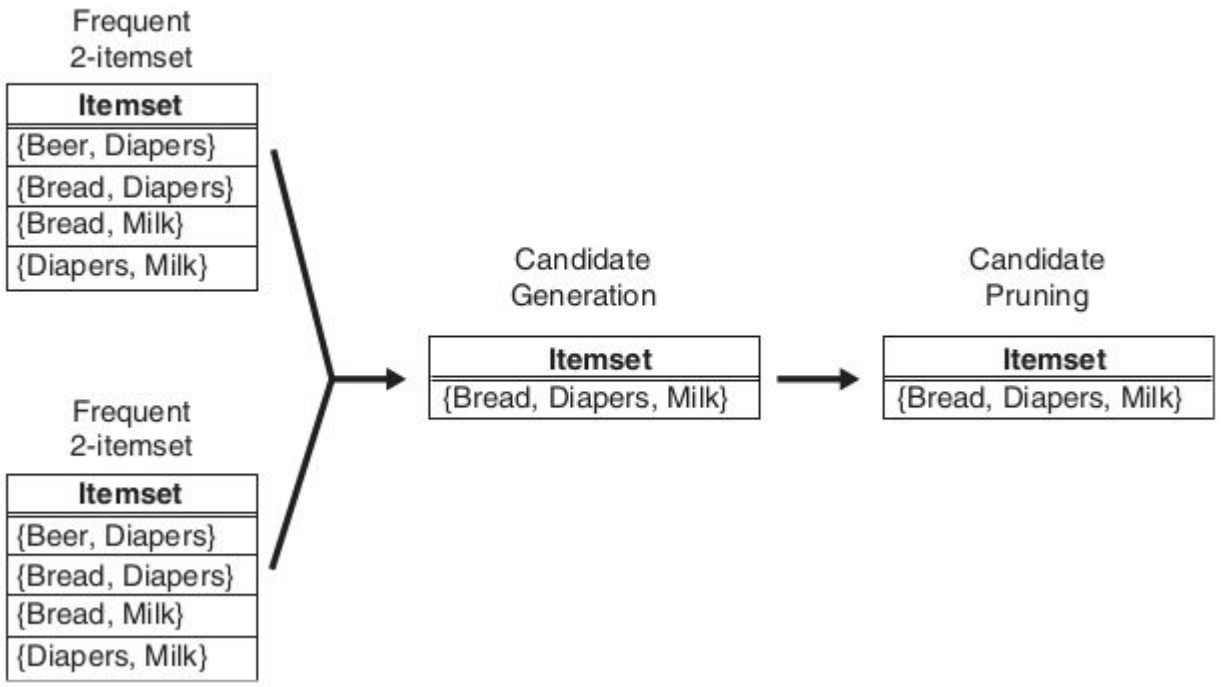
  - Chequeo que todos los sub (k-1)-itemsets del k-itemset candidato son frecuentes.

  (A B C D) es un candidato válido porque todos sus subconjuntos son frecuentes (A C D) (B C D)

  (A C D E) No es candidato porque (C D E) no es frecuente

  - Al final se deben contar todas las transacciones que contengan el k-itemset candidato. Un k-itemset puede ser infrecuente incluso si todos sus subconjuntos son frecuentes (es una condición necesaria pero no suficiente).

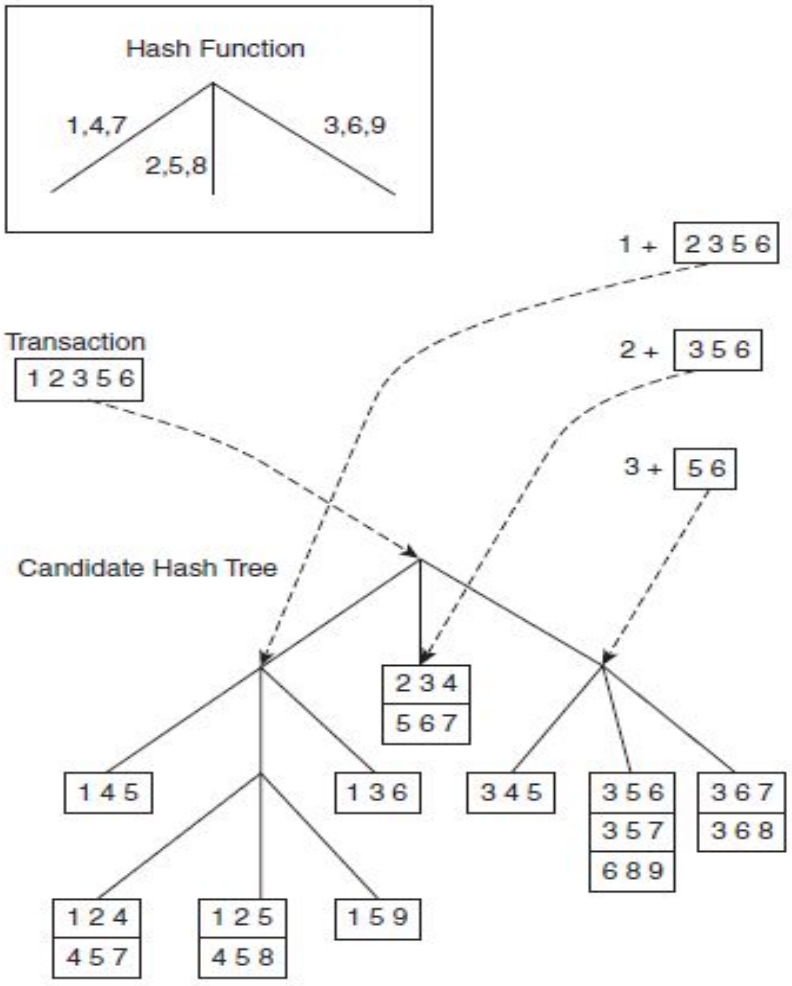
Generar y podar candidatos a k-itemsets mezclando pares de (k-1)-itemsets frecuentes.



Algoritmo Apriori

1. Encuentro los 1-itemset frecuentes escaneando la base de datos
2. **Mezcla:** Encuentro candidatos a k-itemsets frecuentes combinando pares de (k-1)-itemsets frecuentes que sólo difieran en su último elemento. (Los itemsets deben estar ordenados lexicográficamente)
3. **Poda:** chequeo que los sub-itemsets del candidato sean frecuentes. Si encuentro algún sub-itemset no frecuente descarto el candidato por principio Apriori.
4. **Conteo de suporte**: cuento el soporte del itemset candidato y chequeo si cumple el criterio minsup. Uso un árbol hash (hash tree) para hacer el conteo de manera eficiente.

Usar un Hash Tree para mantener los conteos de soporte



Algoritmo Apriori para generación de itemsets frecuentes

Algorithm 5.1 Frequent itemset generation of the *Apriori* algorithm.

1: $k = 1$.
2: $F_k = \{ i \mid i \in I \wedge \sigma(\{i\}) \geq N \times minsup \}$. {Find all frequent 1-itemsets}
3: **repeat**
4: $k = k + 1$.
5: $C_k = \text{candidate-gen}(F_{k-1})$. {Generate candidate itemsets.}
6: $C_k = \text{candidate-prune}(C_k, F_{k-1})$. {Prune candidate itemsets.}
7: **for** each transaction $t \in T$ **do**
8: $C_t = \text{subset}(C_k, t)$. {Identify all candidates that belong to t.}
9: **for** each candidate itemset $c \in C_t$ **do**
10: $\sigma(c) = \sigma(c) + 1$. {Increment support count.}
11: **end for**
12: **end for**
13: $F_k = \{ c \mid c \in C_k \wedge \sigma(c) \geq N \times minsup \}$. {Extract the frequent k-itemsets.}
14: **until** $F_k = \emptyset$
15: Result $= \bigcup F_k$.

Generación de Reglas a partir de un Itemset

- Una vez encontrados todos los itemsets que satisfacen la restricción de minsup, podemos usarlos para generar reglas.

- Podemos particionar el itemset Y en dos subconjuntos no vacios X e $Y - X$ para formar la regla $X \rightarrow Y - X$

- Donde $X \rightarrow Y - X$, tiene que satisfacer la restricción de minconf.

- Ejemplo: Y = {a,b,c}, X = {a,b}, $Y - X$ = {c} produce la regla {a,b} → {c}.

Generación de Reglas a partir de un Itemset

- Para el itemset Y = {a, b, c}
- Se pueden generar 6 reglas 2^k-2, ignorando las que tienen antecedente o consecuente vacío.
- {a, b} → {c}, {a, c}→{b}, {b, c} → {a}, {a} → {b, c}, {b}→ {a, c} y {c} → {a, b}.
- Como el soporte de cada regla es igual al de X, todas estas reglas satisfacen minsup.

Generación eficiente de reglas

- Queremos encontrar todas las reglas que satisfacen minconf.
  - Confianza = soporte del itemset dividido por el soporte del antecedente

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

  - Los valores de soporte ya fueron calculados en la fase previa y se encuentran guardados en el hash tree => no tenemos que escanear la base de datos nuevamente.
  - Ejemplo: Considere la regla {1, 2}→ {3} generada partir del itemset X = {1, 2, 3}.
  - La confianza de esta regla es σ({1, 2, 3})/σ({1, 2}).
  - Como {1, 2, 3} es frecuente, el principio apriori nos asegura que {1, 2} es frecuente también y por ende el soporte del {1,2} se encuentra guardado en el hash tree.

Poda basada en confianza

- Una estrategia fuerza bruta para generar reglas sería generar todas las reglas posibles a partir de todos los itemsets frecuentes y ver si cumplen con **minconf**.

- Eso equivale a evaluar 2^k-2 reglas (con k el tamaño del itemset).

- Estrategia eficiente: poda basada en confianza.
- Teorema: Sea Y un itemset y X un subconjunto de Y .
  - Si una regla $X \rightarrow Y - X$ no satisface **minconf**, entonces cualquier regla $\tilde{X} \rightarrow Y - \tilde{X}$, con $\tilde{X}$ subconjunto de X, tampoco va a satisfacer la regla de **minconf**.
  - Si muevo itemsets del antecedente al consecuente no puedo subir el nivel confianza.

Poda basada en confianza

Ejemplo:

- Sea Y = {a,b,c}, X={a,b}, $\tilde{X}$ = {a}, con $\tilde{X} \subset X$
- $X \rightarrow Y - X$ = {a,b} → {c}
- $\tilde{X} \rightarrow Y - \tilde{X}$ = {a} → {b,c} // moví un item del antecedente al consecuente.
- $c(X \rightarrow Y\text{-}X) = \sigma(Y)/\sigma(X) = \sigma(\{a,b,c\})/\sigma(\{a,b\})$
- $c(\tilde{X} \rightarrow Y\text{-}\tilde{X}) = \sigma(Y)/\sigma(\tilde{X}) = \sigma(\{a,b,c\})/\sigma(\{a\})$

- Por principio apriori: $\sigma(\{a\}) \geq \sigma(\{a,b\})$, $\sigma(\tilde{X}) \geq \sigma(X)$
- c({a} → {b,c}) requiere dividir por un número más grande que en c({a,b} → {c})
- Entonces c({a,b} → {c}) ≥ c({a} → {b,c})

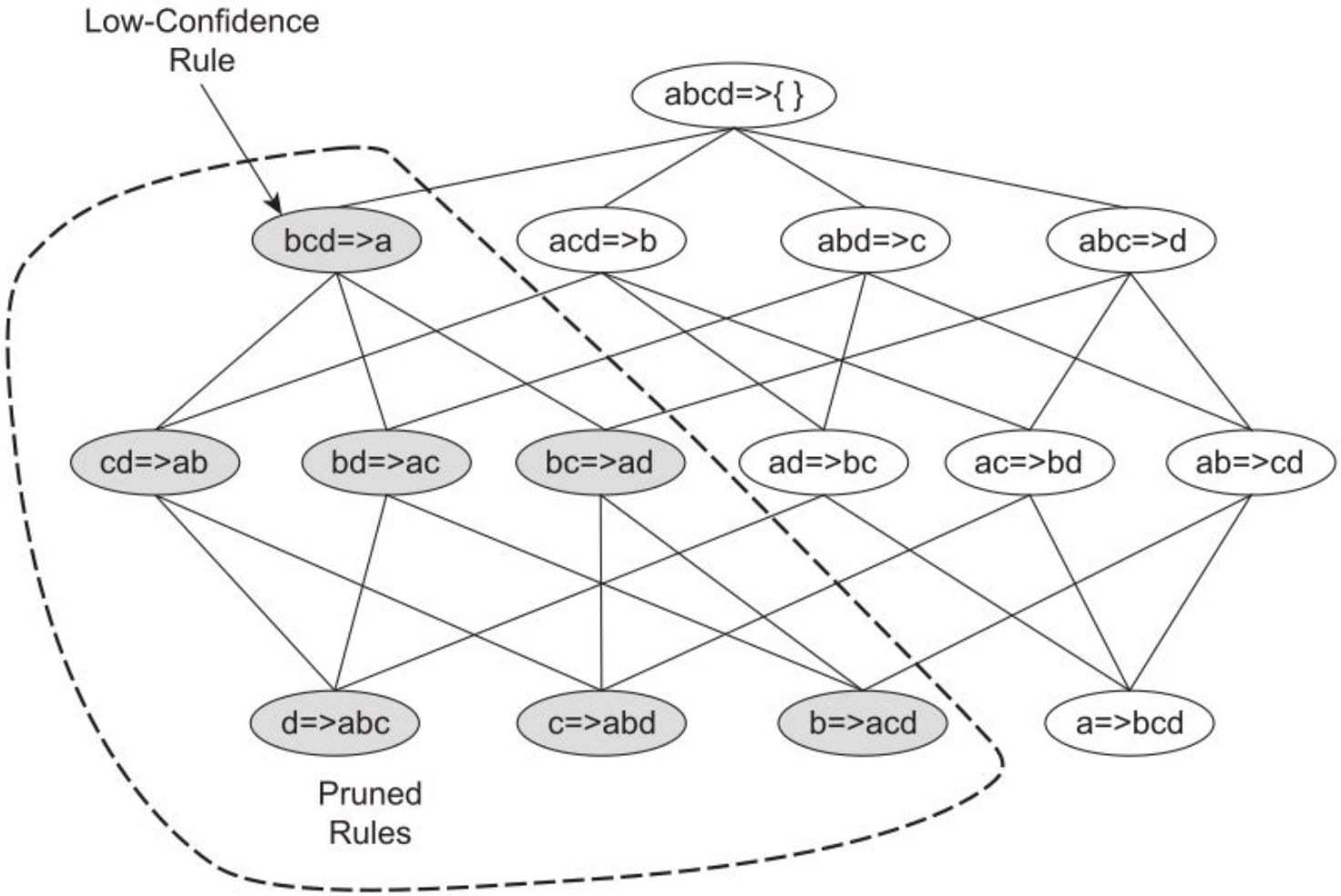
¡La confianza no puede crecer si muevo itemsets del antecedente consecuente!

Poda basada en confianza

- La confianza no puede crecer si muevo itemsets del antecedente al consecuente.

- Demostración:
  - Sean dos reglas $\tilde{X} \rightarrow Y - \tilde{X}$, $X \rightarrow Y - X$, donde $\tilde{X} \subset X$.
  - La confianza de estas reglas es $\sigma(Y)/\sigma(\tilde{X})$ y $\sigma(Y)/\sigma(X)$ respectivamente.
  - Como $\tilde{X}$ es un subconjunto de X, $\sigma(\tilde{X}) \geq \sigma(X)$.
  - Por consecuencia, la primera regla no puede tener una confianza mayor que la segunda.

Poda de Reglas basada en Confianza



Generación de reglas eficientes en Apriori

Algorithm 5.2 Rule generation of the *Apriori* algorithm.

1: for each frequent k-itemset f_k, $k \geq 2$ do
2: $H_1 = \{i \mid i \in f_k\}$ {1-item consequents of the rule.}
3: call ap-genrules$(f_k, H_1.)$
4: end for

Algorithm 5.3 Procedure ap-genrules(f_k, H_m).

1: $k = |f_k|$ {size of frequent itemset.}
2: $m = |H_m|$ {size of rule consequent.}
3: if $k > m + 1$ then
4: H_{m+1} = candidate-gen(H_m).
5: H_{m+1} = candidate-prune(H_{m+1}, H_m).
6: for each $h_{m+1} \in H_{m+1}$ do
7: $conf = \sigma(f_k)/\sigma(f_k - h_{m+1})$.
8: if $conf \geq minconf$ then
9: output the rule $(f_k - h_{m+1}) \longrightarrow h_{m+1}$.
10: else
11: delete h_{m+1} from H_{m+1}.
12: end if
13: end for
14: call ap-genrules$(f_k, H_{m+1}.)$
15: end if

Ejemplo

- El siguiente ejemplo muestra los votos del congreso Estadounidense en 1984.
- Cada transacción indica el partido político del congresista y su voto respecto a varios temas.

---

1. Republican
2. Democrat
3. handicapped-infants = yes
4. handicapped-infants = no
5. water project cost sharing = yes
6. water project cost sharing = no
7. budget-resolution = yes
8. budget-resolution = no
9. physician fee freeze = yes
10. physician fee freeze = no
11. aid to El Salvador = yes
12. aid to El Salvador = no
13. religious groups in schools = yes
14. religious groups in schools = no
15. anti-satellite test ban = yes
16. anti-satellite test ban = no
17. aid to Nicaragua = yes
18. aid to Nicaragua = no
19. MX-missile = yes
20. MX-missile = no
21. immigration = yes
22. immigration = no
23. synfuel corporation cutback = yes
24. synfuel corporation cutback = no
25. education spending = yes
26. education spending = no
27. right-to-sue = yes
28. right-to-sue = no
29. crime = yes
30. crime = no
31. duty-free-exports = yes
32. duty-free-exports = no
33. export administration act = yes
34. export administration act = no

Ejemplo

Association Rule	Confidence
{budget resolution = no, MX-missile=no, aid to El Salvador = yes } ⟶ {Republican}	91.0%
{budget resolution = yes, MX-missile=yes, aid to El Salvador = no } ⟶ {Democrat}	97.5%
{crime = yes, right-to-sue = yes, physician fee freeze = yes} ⟶ {Republican}	93.5%
{crime = no, right-to-sue = no, physician fee freeze = no} ⟶ {Democrat}	100%

- Reglas generadas usando Apriori con minsup 30% y minconf = 90%.
- Las primeras dos reglas sugieren que la mayoría de los votantes que votaron por ayudar al Salvador, no para resolución de presupuesto y no para MX-missile son republicanos.
- Estas reglas muestran los temas que más dividen a los miembros de los dos partidos.

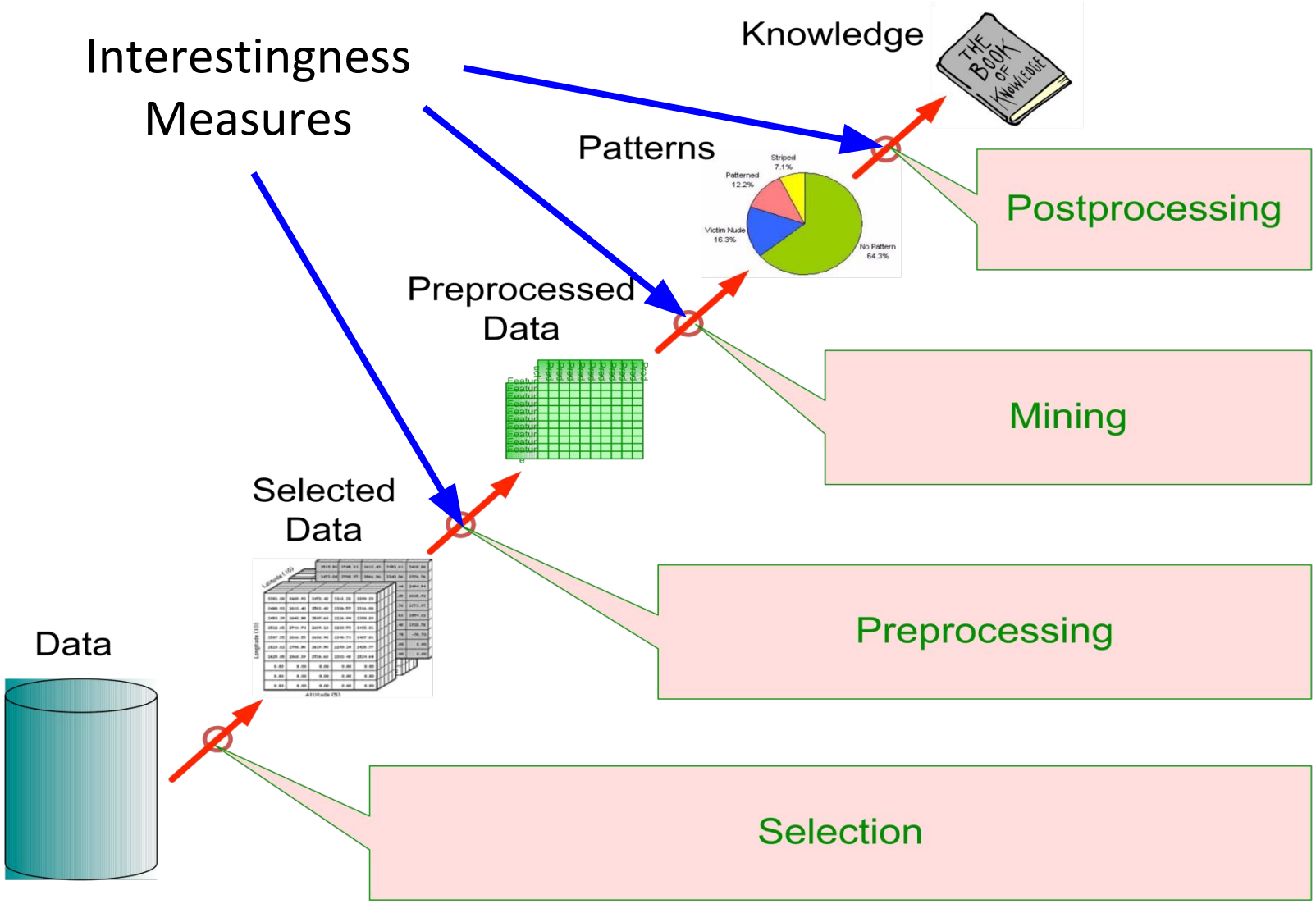
Evaluación de Patrones

- Los algoritmos de reglas de asociación tienden a producir demasiadas reglas
  - Muchas son redundantes o poco interesantes
  - Redundante si $\{A,B,C\} \rightarrow \{D\}$ y $\{A,B\} \rightarrow \{D\}$ tienen el mismo support y confidence
- Se pueden usar medidas de interés para podar/rankear los patrones derivados
- En la formulación original de reglas de asociación, support y confidence son las únicas medidas

Medidas objetivas de interés

- Aplicamos medidas objetivas de interés.
  - Soporte y confianza son ejemplos de medidas objetivas de interés.
- Se busca descartar reglas que asocian itemsets que son independientes entre sí (estadísticamente independientes)

Uso de medidas de interés



Calculando el interés

- Dada una regla $X \rightarrow Y$, la información requerida para calcular su medida de interés se puede obtener de la tabla de contingencia

Tabla de contingencia para $X \rightarrow Y$

	Y	$\overline{Y}$	
X	f_{11}	f_{10}	f_{1+}
$\overline{X}$	f_{01}	f_{00}	f_{o+}
	f_{+1}	f_{+0}	$\|T\|$

f_{11}: support de X e Y
f_{10}: support de $\overline{X}$ e $\overline{Y}$
f_{01}: support de $\overline{X}$ e Y
f_{00}: support de $\overline{X}$ e $\overline{Y}$

Usado para definir varias medidas

- support, confidence, lift, Gini, J-measure, etc.

Desventaja de Confidence

	Coffee	$\overline{\text{Coffee}}$	
Tea	15	5	20
$\overline{\text{Tea}}$	75	5	80
	90	10	100

Regla de Asociación: Tea → Coffee

Support(Tea → Coffee) = 0.15

Confidence (Tea → Coffee) = P(Coffee|Tea) = 0.75

pero P(Coffee) = 0.9

La fracción de personas que toman café independientemente si toman té es 0.9.

Saber que alguien toma té baja su probabilidad de tomar café de 0.9 a 0.75

- Aunque confidence es alto, la regla es engañosa

¡La confianza ignora el soporte del consecuente de una regla!

Independencia estadística

- Población de 1000 estudiantes
  - 600 students know how to swim (S)
  - 700 students know how to bike (B)
  - 420 students know how to swim and bike (S,B)

  - $P(S \wedge B) = 420/1000 = 0.42$
  - $P(S) \cdot P(B) = 0.6 \cdot 0.7 = 0.42$

  - $P(S \wedge B) = P(S) \cdot P(B)$ => Independencia estadística
  - $P(S \wedge B) > P(S) \cdot P(B)$ => Correlación positiva
  - $P(S \wedge B) < P(S) \cdot P(B)$ => Correlación negativa

Medidas estadísticas

- Consideran dependencia estadística
  - Interest Factor o Lift

$$I(A,B) = \frac{s(A,B)}{s(A) \times s(B)} = \frac{N f_{11}}{f_{1+} f_{+1}}.$$

Se interpreta como:

$$I(A,B) \begin{cases} = 1, & \text{if } A \text{ and } B \text{ are independent;} \\ > 1, & \text{if } A \text{ and } B \text{ are positively related;} \\ < 1, & \text{if } A \text{ and } B \text{ are negatively related.} \end{cases}$$

Ejemplo: Lift/Interest

	Coffee	$\overline{\text{Coffee}}$	
Tea	15	5	20
$\overline{\text{Tea}}$	75	5	80
	90	10	100

Regla de Asociación: Tea → Coffee

Confidence (Tea → Coffee) = P(Coffee|Tea) = 0.75

pero P(Coffee) = 0.9

Lift = 0.75/0.9= 0.8333 (< 1, por lo tanto está asociado negativamente)

Muchas medidas propuestas en la literatura

Algunas medidas funcionan bien en algunas aplicaciones, pero en otras no

Table 5.9. Examples of objective measures for the itemset $\{A, B\}$.

Measure (Symbol)	Definition
Correlation (ϕ)	$\frac{N f_{11} - f_{1+} f_{+1}}{\sqrt{f_{1+} f_{+1} f_{0+} f_{+0}}}$
Odds ratio (α)	$\left(f_{11} f_{00}\right) / \left(f_{10} f_{01}\right)$
Kappa (κ)	$\frac{N f_{11} + N f_{00} - f_{1+} f_{+1} - f_{0+} f_{+0}}{N^2 - f_{1+} f_{+1} - f_{0+} f_{+0}}$
Interest (I)	$\left(N f_{11}\right) / \left(f_{1+} f_{+1}\right)$
Cosine (IS)	$\left(f_{11}\right) / \left(\sqrt{f_{1+} f_{+1}}\right)$
Piatetsky-Shapiro (PS)	$\frac{f_{11}}{N} - \frac{f_{1+} f_{+1}}{N^2}$
Collective strength (S)	$\frac{f_{11} + f_{00}}{f_{1+} f_{+1} + f_{0+} f_{+0}} \times \frac{N - f_{1+} f_{+1} - f_{0+} f_{+0}}{N - f_{11} - f_{00}}$
Jaccard (ζ)	$f_{11} / \left(f_{1+} + f_{+1} - f_{11}\right)$
All-confidence (h)	$\min \left[\frac{f_{11}}{f_{1+}}, \frac{f_{11}}{f_{+1}}\right]$

Propiedades de una buena medida

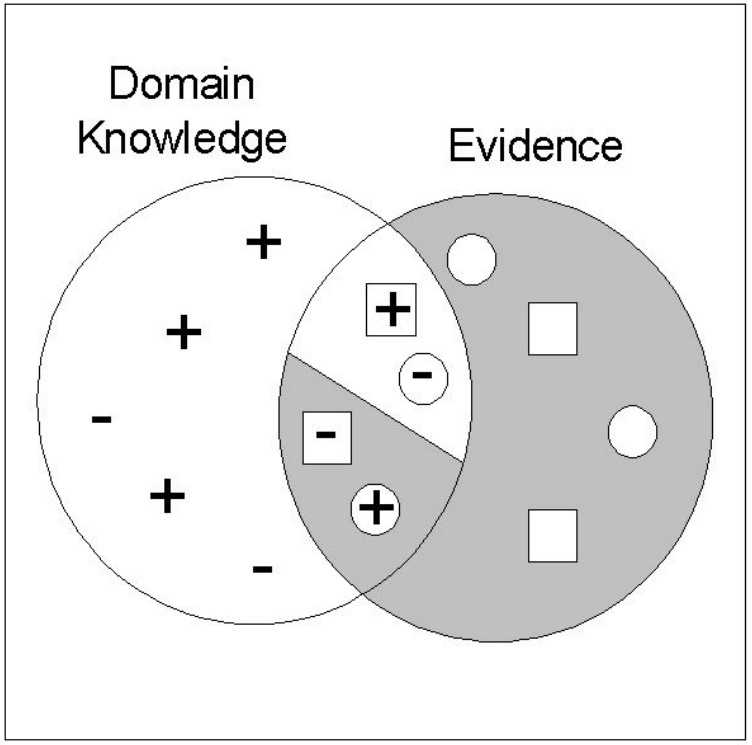
- Piatetsky-Shapiro:
  Tres propiedades que una buena medida M debe cumplir:
  - M(A,B) = 0 si A y B son estadísticamente independientes
  - M(A,B) aumenta monótonamente con P(A,B) cuando P(A) y P(B) se mantienen sin cambios
  - M(A,B) disminuye monótonamente con P(A) [o P(B)] cuando P(A,B) y P(B) [o P(A)] se mantienen sin cambios

Medida de interés subjetiva

- Medida objetiva:
  - Rankear patrones basado en estadísticas calculadas a partir de los datos
  - e.g., 21 medidas de asociación (support, confidence, Laplace, Gini, mutual information, Jaccard, etc).
- Medida subjetiva:
  - Rankear patrones de acuerdo a interpretación del usuario
    - Un patrón es subjetivamente interesante si contradice las expectativas de un usuario (Silberschatz & Tuzhilin)
    - Un patrón es subjetivamente interesante si es “actionable” (se puede usar como razón para hacer algo) (Silberschatz & Tuzhilin)

Interestingness via Unexpectedness

- Necesidad de modelar expectativas de usuario (conocimiento del dominio)



- Necesidad de combinar expectativas de los usuarios con evidencia de los datos (i.e., patrones extraídos)