

Reglas de Asociación



Valeria Rulloni, Georgina Flesia, Laura Alonso Alemany
Diplomatura en Ciencia de Datos,
Aprendizaje Automático y sus Aplicaciones
FaMAF-UNC
agosto 2019

Intuición

La probabilidad condicional hecha regla

¿Qué nos suma este formato?

- Más fácil de inspeccionar
 - Se pueden manipular distintamente componentes como antecedente, consecuente, representatividad,
 - Se pueden insertar métricas: novedad, sorpresa, valor económico, clase
- Más accionable!

De intuición a producción hay un buen trecho!

Contexto

- El algoritmo más popular es Apriori (Agrawal et al 1993)
- Todos los datos tienen que ser categóricos
- Inicialmente se usó para Análisis del Carrito de la Compra (Market Basket Analysis)

Pan → Leche [sop = 5%, conf = 100%]

Terminología

I = {i1, i2, ..., im}: un conjunto de **items**.

Transacción **t**:

t es un conjunto de items sin orden, y $t \subseteq I$.

Base de datos de transacciones: un conjunto de transacciones $T = \{t1, t2, \dots, tn\}$.

Ejemplo

Transacciones de compra de mercado:

```
t1: {pan, queso, leche}  
t2: {manzana, huevos, sal, yogur}  
...  
tn: {bizcocho, huevos, leche}
```

Definiciones:

- **item**: un item/artículo en el carrito de la compra
- **I**: todos los items que se venden en el negocio
- **transacción**: items comprados en un ticket (*basket*)

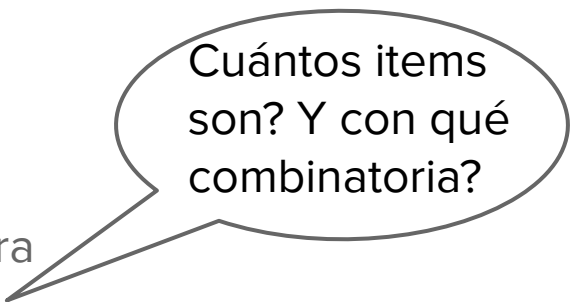
Ejemplo

Transacciones de compra de mercado:

```
t1: {pan, queso, leche}  
t2: {manzana, huevos, sal, yogur}  
...  
tn: {bizcocho, huevos, leche}
```

Definiciones:

- **item**: un item/artículo en el carrito de la compra
- **I**: todos los items que se venden en el negocio
- **transacción**: items comprados en un ticket (*basket*)



Cuántos items son? Y con qué combinatoria?

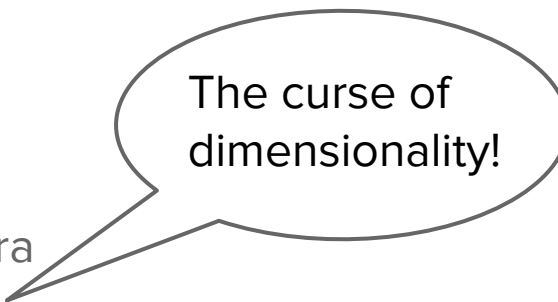
Ejemplo

Transacciones de compra de mercado:

```
t1: {pan, queso, leche}  
t2: {manzana, huevos, sal, yogur}  
...  
tn: {bizcocho, huevos, leche}
```

Definiciones:

- **item**: un item/artículo en el carrito de la compra
- **I**: todos los items que se venden en el negocio
- **transacción**: items comprados en un ticket (*basket*)



The curse of dimensionality!

Ejemplo

Un dataset de documentos de texto. Cada documento es una bolsa de palabras

doc1: Estudiante, Enseñar, Escuela

doc2: Estudiante, Escuela

doc3: Enseñar, Escuela, Ciudad, Partido

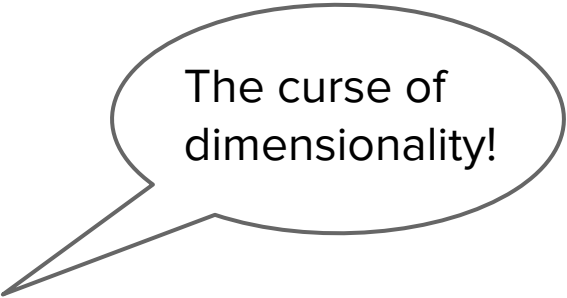
doc4: Beisbol, Basket

doc5: Basket, Player, Espectador

doc6: Beisbol, Entrenador, Partido, Equipo

doc7: Basket, Equipo, Ciudad, Partido

- **item**: una palabra en un documento
- **I**: todas las palabras del conjunto de documentos
- **transacción**: las palabras de un documento



The curse of dimensionality!

Ejemplo

Un dataset de documentos de texto. Cada documento es una bolsa de palabras

doc1: Estudiante, Enseñar, Escuela

doc2: Estudiante, Escuela

doc3: Enseñar, Escuela, Ciudad, Partido

doc4: Beisbol

doc5: Basket

doc6: Beisbol, Entrenador, Partido, Equipo

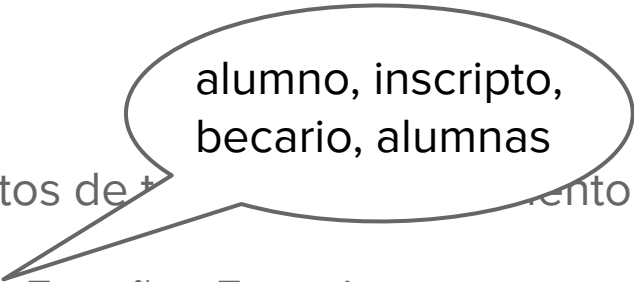
doc7: Basket, Equipo, Ciudad, Partido

Qué queremos saber?

- **item**: una palabra en un documento
- **I**: todas las palabras del conjunto de documentos
- **transacción**: las palabras de un documento

Ejemplo

Un dataset de documentos de texto es una bolsa de palabras



alumno, inscripto,
becario, alumnas

doc1: Estudiante, Enseñar, Escuela

doc2: Estudiante, Escuela

doc3: Enseñar, Escuela, Ciudad, Partido

doc4: Beisbol, Basket

doc5: Basket, Player, Espectador

doc6: Beisbol, Entrenador, Partido, Equipo

doc7: Basket, Equipo, Ciudad, Partido

- **item**: una palabra en un documento
- **I**: todas las palabras del conjunto de documentos
- **transacción**: las palabras de un documento

Ejemplo

Un dataset de documentos de texto es una bolsa de palabras

alumno, inscripto,
becario, alumnas

doc1: Estudiante, Enseñar, Escuela

doc2: Estudiante, Escuela

doc3: Enseñar, Escuela, Ciudad, Pa

doc4: Beisbol, Basket

doc5: Basket, Player, Espectador

doc6: Beisbol, Entrenador, Partido,

doc7: Basket, Equipo, Ciudad, Partido

- Pre-procesos
- Conocimiento de dominio (traductores, sinónimos)
- Embeddings!

- **item**: una palabra en un documento
- **I**: todas las palabras del conjunto de documentos
- **transacción**: las palabras de un documento

Ejemplo

Un conjunto de historias clínicas.

paciente1:

consulta1:deshidratación, fiebre38.5, ibuprofeno

consulta2:gastritis, protector_gástrico

Paciente2:

consulta1:dolor_articular, fiebre39, antibiótico, ibuprofeno

consulta2:dolor_articular, febrícula37.5, ibuprofeno

consulta3:gastritis, protector_gástrico

- **item**: una evento en una historia clínica
- **I**: todos los eventos en todas las historias clínicas
- **transacción**: Cada consulta? Cada historia clínica? Cada período de tiempo?

Ejemplo

Un conjunto de historias clínicas.

paciente1:

consulta1:deshidratación, fiebre 38.5, ibuprofeno

consulta2:gastritis, protector_gástrico

Paciente2:

consulta1:dolor_articular, fiebre 39, antibiótico, ibuprofeno

consulta2:dolor_articular, febrícula 37.5, ibuprofeno

consulta3:gastritis, protector_gástrico

- **item**: una evento en una historia clínica
- **I**: todos los eventos en todas las historias clínicas
- **transacción**: Cada consulta? Cada historia clínica? Cada período de tiempo?



discretizar

Ejemplo

Un conjunto de historias clínicas.

paciente1:

consulta1:deshidratación, fiebre38.5, ibuprofeno

consulta2:gastritis, protector_gástrico

Paciente2:

consulta1:dolor_articular, fiebre39, antibiótico, ibuprofeno

consulta2:dolor_articular, febrícula37.5, ibuprofeno

consulta3:gastritis, protector_gástrico

discretizar

clases de equivalencia
semántica

- **item**: una evento en una historia clínica
- **I**: todos los eventos en todas las historias clínicas
- **transacción**: Cada consulta? Cada historia clínica? Cada período de tiempo?

Ejemplo

- Patrones de navegación de usuarios en la web
- Patrones de aprendizaje en plataformas on-line
- Patrones de fallo de discos rígidos
- Esperanza de vida de animales
- ...

Una regla de asociación $X \rightarrow Y$ es un patrón que dice que cuando ocurre X , ocurre Y con una cierta probabilidad.

Una transacción t contiene X , un conjunto de items (itemset) en I , si $X \subseteq t$.

Una regla de asociación es una implicación:

$$\mathbf{X} \rightarrow \mathbf{Y}, \text{ donde } X, Y \subset I, \text{ y } X \cap Y = \emptyset$$

Un itemset es un conjunto de items.

$$X = \{\text{leche}, \text{ pan}, \text{ cereal}\}$$

Un k -itemset es un itemset con k items.

$$\{\text{leche}, \text{ pan}, \text{ cereal}\} \text{ es un 3-itemset}$$

Métricas

Soporte: La regla $X \rightarrow Y$ tiene Soporte sup en T (el dataset de transacciones) si $sup\%$ de las transacciones contienen $X \cup Y$.

$$sup = \Pr(X \cup Y).$$

Confianza: La regla $X \rightarrow Y$ tiene Confianza $conf$ en T si $conf\%$ de las transacciones que contienen X también contienen Y .

$$conf = \Pr(Y \mid X).$$

Lift: $lift = \Pr(X \cup Y) / (\Pr(X) * \Pr(Y))$

Convicción: $conv = (1 - sup(Y)) / (1 - conf(X \rightarrow Y)).$

Métricas

Soporte: La regla $X \rightarrow Y$ tiene Soporte sup en T (el dataset de transacciones) si $sup\%$ de las transacciones contienen $X \cup Y$.

$$sup = Pr(X \cup Y)$$

Confianza: La regla $X \rightarrow Y$ tiene Confianza $conf$ si $conf\%$ de las transacciones que contienen X también contienen Y .

$$conf = Pr(Y | X).$$

Lift: $lift = Pr(X \cup Y) / (Pr(X) * Pr(Y))$

Convicción: $conv = (1 - sup(Y)) / (1 - conf(X \rightarrow Y)).$

¿Qué van a priorizar estas métricas?
¿Responden a nuestras preguntas?
¿Nos aportan información valiosa?

transacciones

Métricas

más **soporte**: la regla se encuentra en más transacciones

más **confianza**: mayor probabilidad de que la regla sea cierta para una transacción

más **lift**: menor probabilidad de que la regla sea una casualidad

más **convicción**: mayor grado de implicación, va de 1 a infinito (si la confianza es 1, la convicción es infinita (no 0))

Lo veremos en la notebook

Objetivo de las reglas de asociación

Encontrar todas las reglas que satisfacen un soporte mínimo y confianza mínima

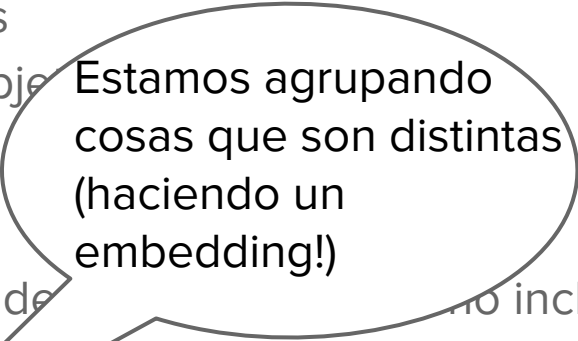
- Todas las reglas
- No hay items objetivo

Una visión simplista de los datos, porque no incluye:

- cantidad
- precio
- promociones

Objetivo de las reglas de asociación

Encontrar todas las reglas que satisfacen un soporte mínimo y confianza mínimo

- Todas las reglas
 - No hay items obsoletos
- 
- Estamos agrupando cosas que son distintas (haciendo un embedding!)

Una visión simplista de las reglas de asociación no incluye:

- cantidad
- precio
- promociones

Algoritmos de reglas

- Hay muchos!
- Usan diferentes estrategias y estructuras de datos
- Pero los conjuntos de reglas resultantes son todos los mismos: dado un dataset, un soporte mínimo y una confianza mínima, el conjunto de reglas de asociación en T es determinístico.

Vamos a ver Apriori (Agrawal et al. 1983)

Algoritmo Apriori

Apriori(T, ϵ)

$L_1 \leftarrow \{\text{large 1 - itemsets}\}$

$k \leftarrow 2$

while $L_{k-1} \neq \emptyset$

$C_k \leftarrow \{a \cup \{b\} \mid a \in L_{k-1} \wedge b \notin a\} - \{c \mid \{s \mid s \subseteq c \wedge |s| = k-1\} \not\subseteq L_{k-1}\}$

for transactions $t \in T$

$C_t \leftarrow \{c \mid c \in C_k \wedge c \subseteq t\}$

for candidates $c \in C_t$

$count[c] \leftarrow count[c] + 1$

$L_k \leftarrow \{c \mid c \in C_k \wedge count[c] \geq \epsilon\}$

$k \leftarrow k + 1$

return $\bigcup_k L_k$

Pasos

1. Encontrar todos los itemsets con soporte mínimo (itemsets frecuentes)

`{pollo, ropa, leche}` `[sop = 3/7]`

2. Usar los itemsets para generar reglas

`ropa → leche, pollo` `[sop = 3/7, conf = 3/3]`

Encontrar itemsets frecuentes

Iterativo (por niveles)

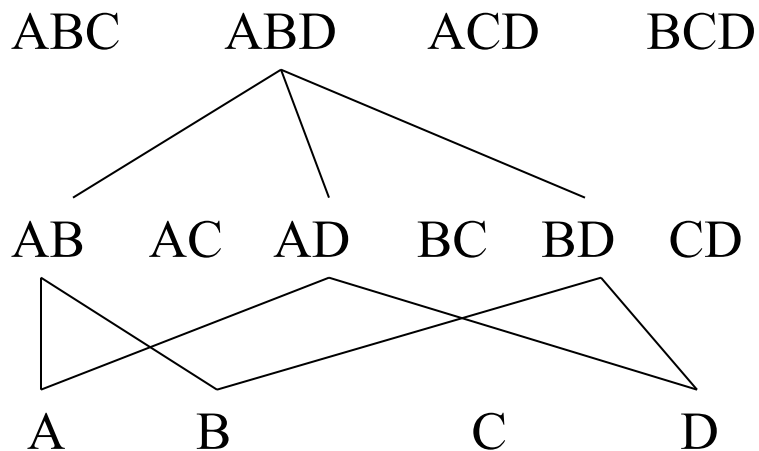
Encontrar todos los itemsets frecuentes de 1 item, entonces todos los itemsets frecuentes de 2 items, y así sucesivamente

- en cada iteración k , considerar solamente los itemsets que contienen un $(k-1)$ -itemset frecuente (descartar de entrada los itemsets que no contienen un $(k-1)$ -itemset frecuente)
- Los items están ordenados, para evitar repeticiones

Encontrar itemsets frecuentes

Itemset frecuente \rightarrow Soporte \geq minsup

propiedad apriori (downward closure): todos los subconjuntos de un itemset frecuente también son itemsets frecuentes



Encontrar itemsets frecuentes

Itemset frecuente \rightarrow Soporte \geq minsup

propiedad apriori (downward closure): todos los subconjuntos de un itemset frecuente también son itemsets frecuentes

ABC ABD ACD BCD

AB AC AD BC BD CD

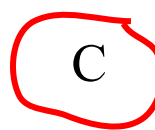
A

B

C

D

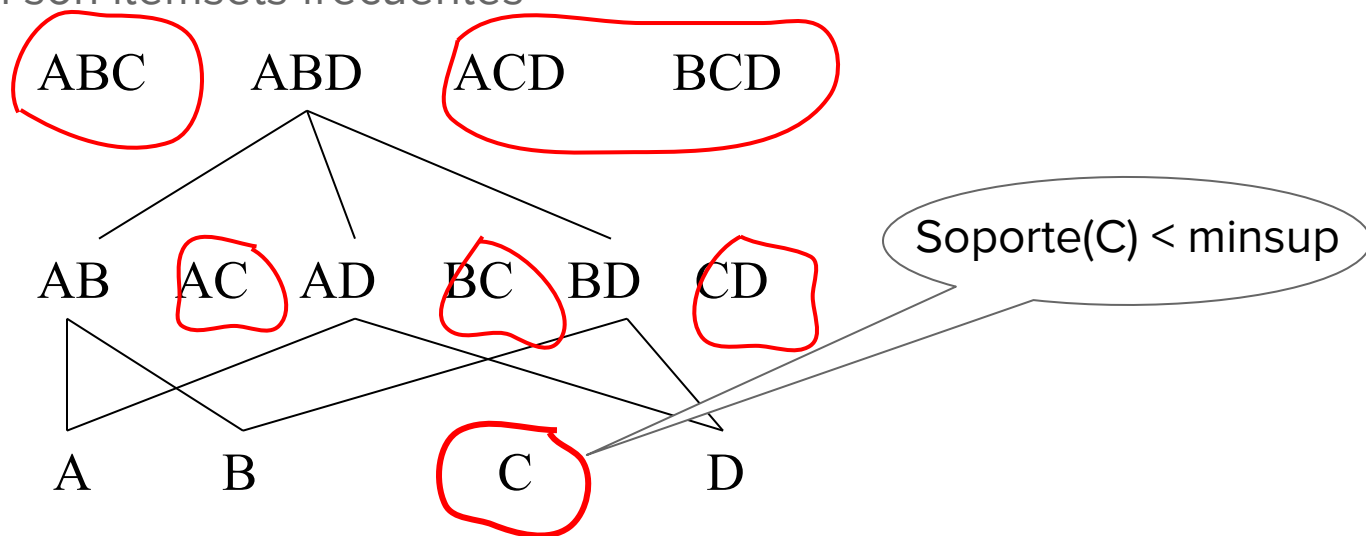
Soporte(C) < minsup



Encontrar itemsets frecuentes

Itemset frecuente \rightarrow Soporte \geq minsup

propiedad apriori (downward closure): todos los subconjuntos de un itemset frecuente también son itemsets frecuentes



Encontrar confianza

Para cada itemset frecuente X ,

Para cada subconjunto no vacío A de X ,

Sea $B = X - A$

$\text{Soporte}(A \rightarrow B) = \text{Soporte}(A \cup B) = \text{Soporte}(X)$

$\text{Confianza}(A \rightarrow B) = \text{Soporte}(A \cup B) / \text{Soporte}(A)$

$A \rightarrow B$ es una regla de asociación si

$\text{Confianza}(A \rightarrow B) \geq \text{minconf}$

Esta información ya se obtuvo en el momento de generación de itemsets, no hay que recorrer el dataset de vuelta

Ejemplo

Supongamos $\{2,3,4\}$ es frecuente, con $\text{sop}=50\%$

Subconjuntos propios no vacíos: $\{2,3\}$, $\{2,4\}$, $\{3,4\}$, $\{2\}$, $\{3\}$, $\{4\}$, con $\text{sop}=50\%$, 50% , 75% , 75% , 75% , 75% respectivamente

Generan estas reglas de asociación:

$2,3 \rightarrow 4$, Confianza= 100%

$2,4 \rightarrow 3$, Confianza= 100%

$3,4 \rightarrow 2$, Confianza= 67%

$2 \rightarrow 3,4$, Confianza= 67%

$3 \rightarrow 2,4$, Confianza= 67%

notebook

Consideraciones sobre Apriori

Parece muy caro pero...

- Búsqueda por niveles, explotando la propiedad de downward closure
 - El parámetro k (tamaño del itemset más grande) limita el coste
 - Escalable!
-
- El espacio de todas las reglas de asociación es exponencial, $O(2^m)$, donde m es el número de items en I .
 - Explota la sparseness de los datos, los valores altos de Soporte y Confianza.
 - Igualmente: un número enorme de reglas!!!

Diferentes soportes mínimos

Diferentes soportes mínimos

- El soporte mínimo genérico asume que todos los items se distribuyen igual
- En muchas aplicaciones, algunos items son muy frecuentes y otros no
- Si el soporte mínimo es muy alto, no encontramos reglas para items poco frecuentes
- Si el soporte mínimo es muy bajo, hay demasiadas reglas

Solución:

- Especificar diferentes soportes mínimos para diferentes items
- Para cada regla, inspeccionamos todos los items que se encuentran en la regla, vemos los soportes mínimos asociados a cada item, nos quedamos con el menor soporte mínimo y determinamos que ese es el soporte mínimo que va a tener que superar la regla

Ejemplo

pan, zapatos, ropa

Los valores MIS especificados por el usuario son:

$$\text{MIS}(\text{pan}) = 2\%$$

$$\text{MIS}(\text{zapatos}) = 0.1\%$$

$$\text{MIS}(\text{ropa}) = 0.2\%$$

Ejemplo

pan, zapatos, ropa

Los valores MIS especificados por el usuario son:

$$\text{MIS}(\text{pan}) = 2\% \quad \text{MIS}(\text{zapatos}) = 0.1\% \quad \text{MIS}(\text{ropa}) = 0.2\%$$

El soporte mínimo de esta regla es el mínimo soporte mínimo:

$$\text{ropa} \rightarrow \text{pan} \rightarrow \text{MIS}(\text{ropa} \rightarrow \text{pan}) = \mathbf{0.2\%}$$

Ejemplo

pan, zapatos, ropa

Los valores MIS especificados por el usuario son:

$MIS(\text{pan}) = 2\%$ $MIS(\text{zapatos}) = 0.1\%$ $MIS(\text{ropa}) = 0.2\%$

Esta regla no supera el soporte mínimo:

$\text{ropa} \rightarrow \text{pan} [\text{sup}=0.15\%, \text{conf}=70\%]$

Ejemplo

pan, zapatos, ropa

Los valores MIS especificados por el usuario son:

$MIS(\text{pan}) = 2\%$ $MIS(\text{zapatos}) = 0.1\%$ $MIS(\text{ropa}) = 0.2\%$

Esta regla no supera el soporte mínimo:

$\text{ropa} \rightarrow \text{pan} [\text{sup}=0.15\%, \text{conf}=70\%]$

Esta regla sí supera el soporte mínimo:

$\text{ropa} \rightarrow \text{zapatos} [\text{sup}=0.15\%, \text{conf}=70\%]$

Ejemplo

pan, zapatos, ropa

Los valores MIS especificados por el usuario son:

$$\text{MIS}(\text{pan}) = 2\% \quad \text{MIS}(\text{zapatos}) = 0.1\% \quad \text{MIS}(\text{ropa}) = 0.2\%$$

Esta regla no supera el soporte mínimo:

$$\text{ropa} \rightarrow \text{pan} [\text{sup}=0.15\%, \text{conf}=70\%] \text{ -- MIS}(\text{ropa} \rightarrow \text{pan}) = \mathbf{0.2\%}$$

Esta regla sí supera el soporte mínimo:

$$\text{ropa} \rightarrow \text{zapatos} [\text{sup}=0.15\%, \text{conf}=70\%] \text{ -- MIS}(\text{ropa} \rightarrow \text{zapatos}) = \mathbf{0.1\%}$$

Para qué es adecuado el soporte mínimo

- Cuando algo es muy caro: *caviar*
- Cuando algo es muy costoso: *cáncer*
- Cuando algo es nuevo: *estudiantes nuevos*
- Para hacer seguimientos específicos
- Para diseñar estrategias con objetivos específicos

Downward closure

Este modelo no preserva downward closure!

Ejemplo: consideramos los cuatro items 1, 2, 3 y 4 en una base de datos. Sus soportes mínimos son

$$\text{MIS}(1) = 10\% \quad \text{MIS}(2) = 20\%$$

$$\text{MIS}(3) = 5\% \quad \text{MIS}(4) = 6\%$$

$\{1, 2\}$ con Soporte 9% es infrecuente, pero $\{1, 2, 3\}$ y $\{1, 2, 4\}$ podrían ser frecuentes.

Valoración diferentes soportes mínimos

- Contiene al modelo con soporte mínimo genérico
- Es un modelo más realista para aplicaciones prácticas
- Ayuda a encontrar reglas para items raros sin producir un montón de reglas inútiles con items frecuentes
- Podemos forzar a hacer reglas solamente con esos items

Pero...

- Hay que asignar soporte mínimo a cada item, manualmente!

Reglas de asociación con clase

Reglas de asociación con clase

- Las reglas de asociación no tienen objetivo: encuentran todas las reglas que existen en los datos, cualquier item puede aparecer como consecuente o condición de una regla
- En algunas aplicaciones nos interesan algunos objetivos concretos

Ejemplo: encontrar palabras asociadas a algún tema

Reglas de asociación con clase

Sea un dataset de transacciones T con n transacciones.

Cada transacción también se etiqueta con una clase y .

Sea I el conjunto de todos los items en T , Y las etiquetas de clase y $I \cap Y = \emptyset$.

Una regla de asociación con clase es una implicación de la forma

$$X \rightarrow y, \text{ donde } X \subseteq I, y \in Y.$$

Las definiciones de Soporte y Confianza son igual que en las reglas de asociación normales.

Ejemplo

doc 1: Estudiante, Enseñar, Escuela : Educación
doc 2: Estudiante, Escuela : Educación
doc 3: Enseñar, Escuela, Ciudad, Partido : Educación
doc 4: Beisbol, Basket : Deporte
doc 5: Basket, Player, Espectador : Deporte
doc 6: Beisbol, Entrenador, Partido, Equipo : Deporte
doc 7: Basket, Equipo, Ciudad, Partido : Deporte

minsup = 20% y minconf = 60%

Estudiante, Escuela → Educación [sup= 2/7, conf = 2/2]

Partido → Deporte [sup= 2/7, conf = 2/3]

Algoritmo

Encontrar todos los items que tienen soporte $>$ minsup, con forma:

$(\text{condset}, y)$, y representa una regla $\text{condset} \rightarrow y$

Donde condset es un conjunto de items de I (i.e., $\text{condset} \subseteq I$), $y \in Y$ es una etiqueta de clase.

El algoritmo apriori se puede modificar para generar reglas con clase

Clase y diferentes soportes mínimos

El usuario puede especificar diferentes soportes mínimos para diferentes clases

Ejemplo:

- tenemos la clase Sí y la clase No
- Queremos soporte 5% para la clase Sí y Soporte 10% para la clase No

Si especificamos soporte mínimo de 100% para una clase, no se generan reglas para esa clase

Tarea

Obtener reglas de asociación entre películas en el dataset movielens
(como si fuera recomendación!) (ah! Pero recomendación es no supervisado?)

Aplicar diferentes métricas de ordenamiento

Hacer un pequeño informe

<https://rpubs.com/vitidN/203264>