Lernen von Assoziationsregeln

Gegeben:

R eine Menge von Objekten, die binäre Werte haben

t eine Transaktion, t $\subseteq$ R

r eine Menge von Transaktionen

$S_{min} \in [0,1]$ die minimale Unterstützung,

$Conf_{min} \in [0,1]$ die minimale Konfidenz

Finde alle Regeln c der Form X → Y, wobei X $\subseteq$ R, Y $\subseteq$ R, X $\cap$ Y = { }

$$s(r,c) = \frac{\left|\left\{t \in r \middle| X \cup Y \in t\right\}\right|}{|r|} \geq s_{\min} \qquad conf(r,c) = \frac{\left|\left\{t \in r \middle| X \cup Y \in t\right\}\right|}{\left|\left\{t \in r \middle| X \in r\right\}\right|} \geq conf_{\min}$$

Binäre Datenbanken

R eine Menge von Objekten, die binäre Werte haben

A, B, C

r eine Menge von Transaktionen

t eine Transaktion, $t \subseteq R$

B,C

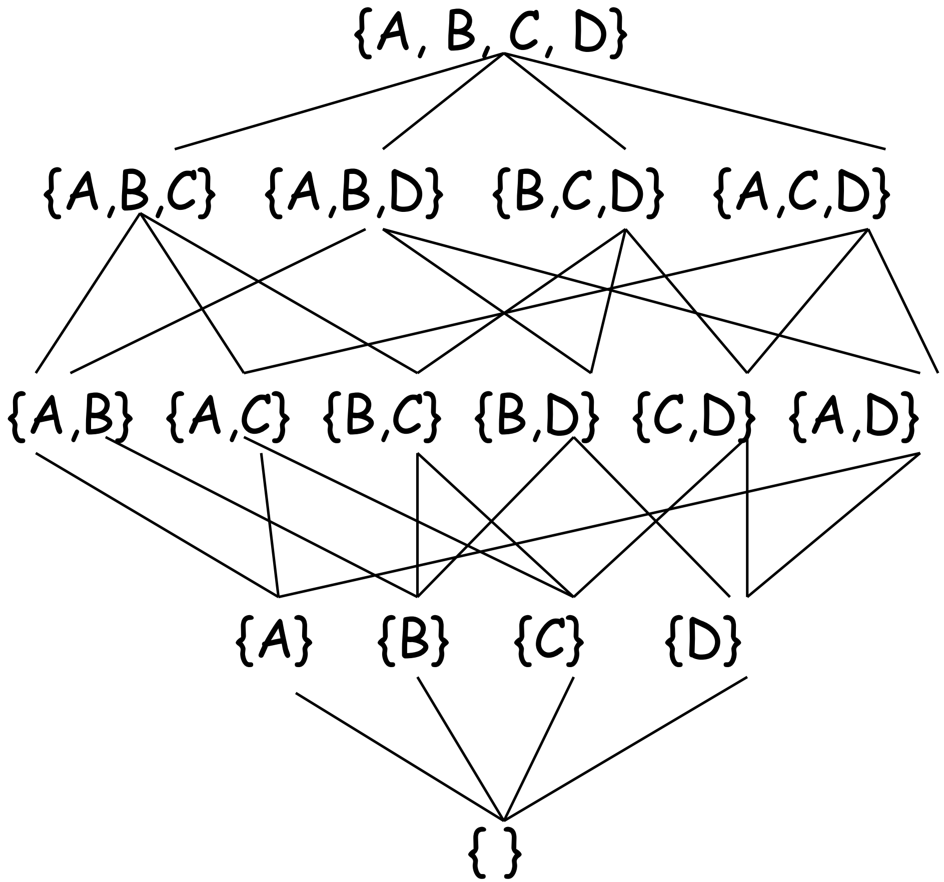
A	B	C	ID
0	1	1	1
1	1	0	2
0	1	1	3
1	0	0	4

Warenkorbanalyse

Aftershave	Bier	Chips	EinkaufsID
0	1	1	1
1	1	0	2
0	1	1	3
1	0	0	4

{Aftershave}→{Bier} $s = \frac{1}{4}$, conf = $\frac{1}{2}$
{Aftershave}→ {Chips} $s = 0$
{Bier} → {Chips} $s = \frac{1}{2}$, conf= 2/3 - zusammen anbieten?
{Chips}→{Aftershave} $s = 0$
{Aftershave}→{Bier,Chips} $s = 0$

Wieder ein Verband...



Ordnungsrelation

- Hier ist die Ordnungsrelation die Teilmengenbeziehung.
- Eine Menge S_1 ist größer als eine Menge S_2, wenn $S_1 \supseteq S_2$.
- Eine kleinere Menge ist allgemeiner.

Assoziationsregeln

LH: Assoziationsregeln sind keine logischen Regeln!

- In der Konklusion können mehrere Attribute stehen
- Attribute sind immer nur binär.
- 0 wird nicht wie „falsch“ behandelt.
- Mehrere Assoziationsregeln zusammen ergeben kein Programm.

LE: Binärvektoren (Transaktionen)

- Attribute sind eindeutig geordnet.

Aufgabe:

- Aus häufigen Mengen Assoziationsregeln herstellen

Apriori Algorithmus

(Agrawal, Mannila, Srikant, Toivonen, Verkamo 1996)

LH des Zwischenschritts: Häufige Mengen $L_k = X \cup Y$
mit k Objekten (large itemsets, frequent sets)

Idee:

- **Wenn eine Menge häufig ist, so auch all ihre Teilmengen. (Anti-Monotonie)**
- **Wenn eine Menge selten ist, so auch all ihre Obermengen. (Monotonie)**
- Wenn X in L_{k+1} dann alle $S_i \subseteq X$ in L_k (Anti-Monotonie)
- Alle Mengen L_k, die k-1 Objekte gemeinsam haben, werden vereinigt zu C_{k+1}, d.h. sie bilden die Menge der Kandidaten für häufige Mengen in der nächsten Runde

Dies ist der Kern des Algorithmus‘, die **Kandidatengenerierung**.

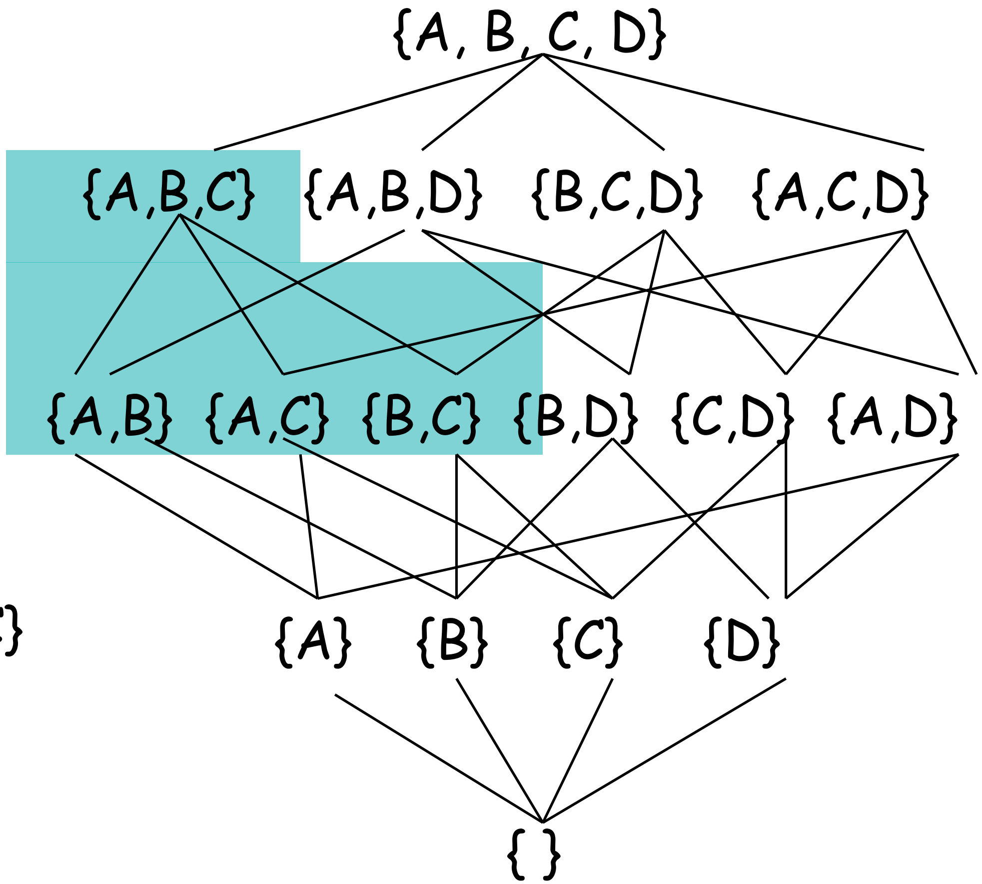
Beispiel

Wenn häufig

dann häufig

Generiere aus
{A,B},{A,C},{B,C}

{A,B,C}



$k+1=3$

$k=2$

Häufige Mengen L_k
ergeben
Kandidaten C_{k+1}

Beispiel

Gesucht werden Kandidaten mit k+1 = 5

L_4= { {ABCD}, {ABCE}, {ABDE}, {ACDE}, {BCDE} }
k-1 Stellen gemeinsam,
vereinigen zu:
I = { ABCDE }
Sind alle k langen Teilmengen von I in L_4?
{ABCD} {ABCE} {ABDE} {ACDE} {BCDE} – ja!
Dann wird I Kandidat C_5.

L_4= { {ABCD}, {ABCE} }
I = { ABCDE }
Sind alle Teilmengen von I in L_4?
{ABCD} {ABCE} {ABDE} {ACDE} {BCDE} – nein!
Dann wird I nicht zum Kandidaten.

Kandidatengenerierung

Erzeuge-Kandidaten(L_k)

$C_{k+1} := \{\}$

Forall I_1, I_2 in L_k , so dass $I_1 = \{i_1, ..., i_{k-1}, i_k\}$,

$I_2 = \{i_1, ..., i_{k-1}, i'_k\}$, $i_k < i'_k$

$I := \{i_1, ..., i_{k-1}, i_k, i'_k\}$

if alle k-elementigen Teilmengen von I sind in L_k

then $C_{k+1} := C_{k+1} \cup \{I\}$

Return C_{k+1}

Prune(C_{k+1}, r) vergleicht Häufigkeit von Kandidaten mit s_{min} und liefert tatsächliche Mengen L_{k+1}.

Häufige Mengen

Häufige-Mengen(R, r, smin)

$C_1 := \bigcup_{i \in R} \{i\}$, $k=1$,

$L_1 := \text{Prune}(C_1)$

while $L_k \neq \{\,\}$

$\quad C_{k+1} := \text{Erzeuge-Kandidaten}(L_k)$

$\quad L_{k+1} := \text{Prune}(C_{k+1}, r)$

$\quad k := k+1$

Return $\bigcup_{j=2}^{k} L_j$

APRIORI

Apriori(R, r, smin, confmin)

L:= Häufige-Mengen(R, r, smin)

c:= Regeln (L, confmin)

Return c

Regelgenerierung

Aus den häufigen Mengen werden Regeln geformt.

Wenn die Konklusion länger wird, kann die Konfidenz sinken.

Die Ordnung der Attribute wird ausgenutzt:

$I_1 = \{i_1, \ldots, i_{k-1}, i_k\}$ $c_1 = \{i_1, \ldots, i_{k-1}\} \rightarrow \{ i_k \}$ $conf_1$

$I_1 = \{i_1, \ldots, i_{k-1}, i_k\}$ $c_2 = \{i_1, \ldots \} \rightarrow \{i_{k-1}, i_k\}$ $conf_2$

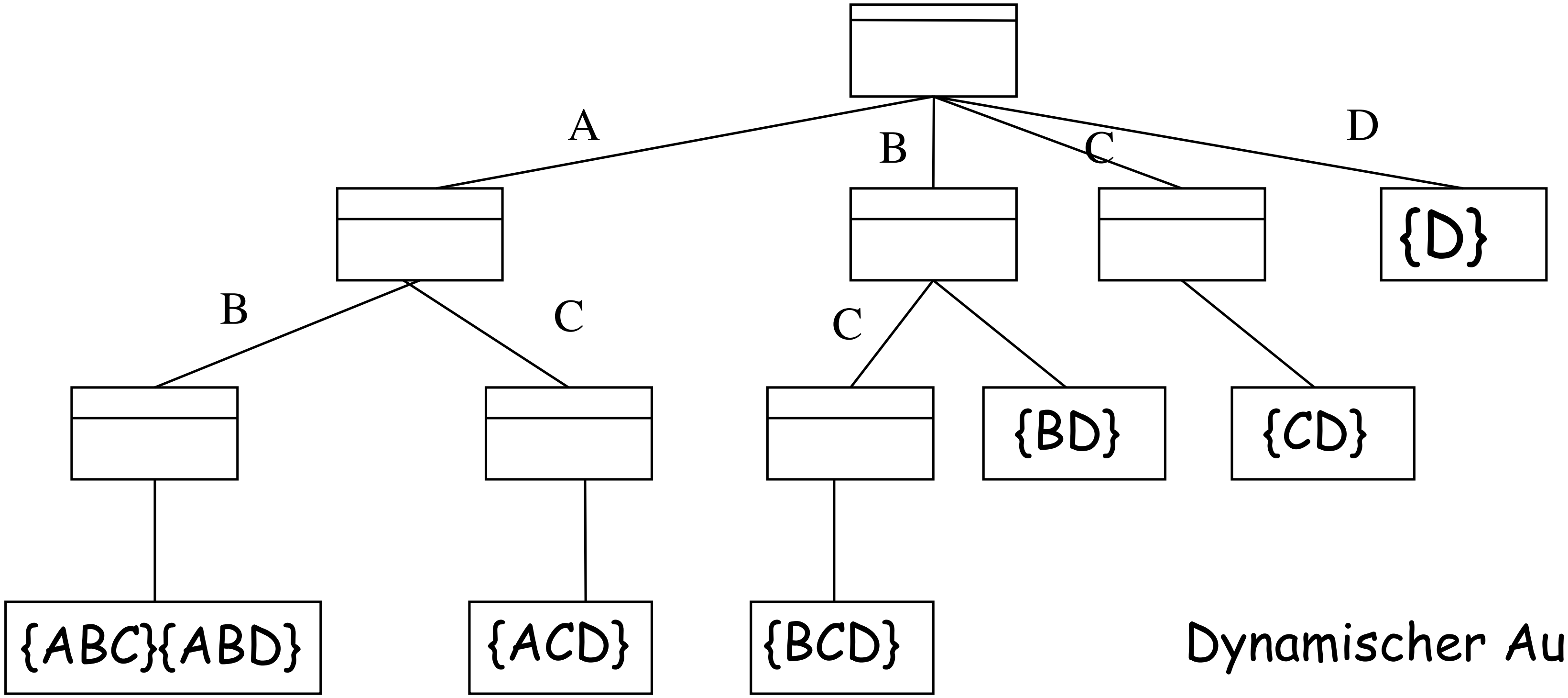
...

$I_1 = \{i_1, \ldots, i_{k-1}, i_k\}$ $c_k = \{i_1\} \rightarrow \{\ldots, i_{k-1}, i_k\}$ $conf_k$

$conf_1 \geq conf_2 \geq \ldots \geq conf_k$

Implementierung

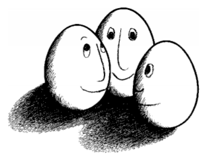
- Hash-Tree für den Präfixbaum, der sich aus der Ordnung der Elemente in den Mengen ergibt.
- An jedem Knoten werden Schlüssel und Häufigkeit gespeichert.



Dynamischer Aufbau

Was wissen Sie jetzt?

- Assoziationsregeln sind keine logischen Regeln.
- Anti-Monotonie der Häufigkeit: Wenn eine Menge häufig ist, so auch all ihre Teilmengen.
- Man erzeugt häufige Mengen, indem man häufige Teilmengen zu einer Menge hinzufügt und diese Mengen dann auf Häufigkeit testet
(Bottom-up Suche im Verband der Mengen).
- Monotonie der Seltenheit: Wenn eine Teilmenge selten ist, so auch jede Menge, die sie enthält.
- Man beschneidet die Suche, indem Mengen mit einer seltenen Teilmenge nicht weiter betrachtet werden.



Probleme von Apriori

- Im schlimmsten Fall ist Apriori exponentiell in R, weil womöglich alle Teilmengen gebildet würden.

  In der Praxis sind die Transaktionen aber spärlich besetzt. Die Beschneidung durch s_{min} und $conf_{min}$ reicht bei der Warenkorbanalyse meist aus.

- Apriori liefert unglaublich viele Regeln.

- Die Regeln sind höchst redundant.

- Die Regeln sind irreführend, weil die Kriterien die apriori Wahrscheinlichkeit nicht berücksichtigen.
  Wenn sowieso alle Cornflakes essen, dann essen auch hinreichend viele Fußballer Cornflakes.

Prinzipien für Regelbewertungen

1. $RI(A \rightarrow B) = 0$, wenn $|A \rightarrow B| = (|A|\,|B|)/|r|$
   A und B sind unabhängig.

2. $RI(A \rightarrow B)$ steigt monoton mit $|A \rightarrow B|$.

3. $RI(A \rightarrow B)$ fällt monoton mit $|A|$ oder $|B|$.

Also: $RI > 0$, wenn $|A \rightarrow B| > (|A|\,|B|)/|r|$
d.h., wenn A positiv mit B korreliert ist.
$RI < 0$, wenn $|A \rightarrow B| < (|A|\,|B|)/|r|$
d.h., wenn A negativ mit B korreliert ist.

Wir wissen, dass immer $|A \rightarrow B| \leq |A| \leq |B|$ gilt, also
lokal: RI_{max} wenn $|A \rightarrow B| = |A|$ oder $|A| = |B|$
global: RI_{max} wenn $|A \rightarrow B| = |A| = |B|$

(Piatetsky-Shapiro 1991)

Konfidenz

- Die Konfidenz erfüllt die Prinzipien nicht! (Nur das 2.)
  Auch unabhängige Mengen A und B werden als hoch-konfident bewertet.

- Die USA-Census-Daten liefern die Regel
  aktiv-militär → kein-Dienst-in-Vietnam mit 90% Konfidenz.
  Tatsächlich ist s(kein-Dienst-in-Vietnam)=95%
  Es wird also wahrscheinlicher, wenn aktiv-militär gegeben ist!

- Gegeben eine Umfrage unter 2000 Schülern, von denen 60% Basketball spielen, 75% Cornflakes essen. Die Regel
  Basketball → Cornflakes hat Konfidenz 66%
  Tatsächlich senkt aber Basketball die Cornflakes Häufigkeit!

Signifikanztest

- Ein einfaches Maß, das die Prinzipien erfüllt, ist:

$$|A \rightarrow B| - \frac{|A||B|}{|r|}$$

- Die Signifikanz der Korrelation zwischen A und B ist:

$$\frac{|A \rightarrow B| - \frac{|A||B|}{|r|}}{\sqrt{|A||B|\left(1 - \frac{|A|}{|r|}\right)\left(1 - \frac{|B|}{|r|}\right)}}$$

Sicherheitsmaß

Shortliffe, Buchanan 1990 führten ein Sicherheitsmaß CF ein (für Regeln in Wissensbasen).

- Wenn conf(A → B) > s(B)
  CF(A→B)= conf(A→B) – s(B)/(1-s(B))
- Wenn conf(A→B) < s(B)
  CF(A→B)= conf(A→B)
- Sonst
  CF(A→B)= 0.

Das Sicherheitsmaß befolgt die Prinzipien für Regelbewertung.

Wendet man Signifikanztest oder Sicherheitsmaß an, erhält man weniger (irrelevante, irreführende) Assoziationsregeln.

Was wissen Sie jetzt?

- Sie haben drei Prinzipien für die Regelbewertung kennen gelernt:
  - Unabhängige Mengen sollen mit 0 bewertet werden.
  - Der Wert soll höher werden, wenn die Regel mehr Belege hat.
  - Der Wert soll niedriger werden, wenn die Mengen weniger Belege haben.
- Sie haben drei Maße kennen gelernt, die den Prinzipien genügen:
  - Einfaches Maß,
  - statistisches Maß und
  - Sicherheitsmaß.