



“Data Mining is a non-trivial process of identifying valid, novel, potentially useful and ultimately understandable patterns in data.”

— (Fayyad et al. 1996)



Apriori

Ranking von Web-Seiten



Anwendung: Optimierung von Supermärkten

Ziel: häufig gemeinsam gekaufte Dinge gruppieren

- ▶ Datengrundlage:
 - ▶ Logfiles von Registrierkassen
- ▶ Häufig zitiertes Beispiel:
 - ▶ Windeln und Bier
 - ▶ wahrscheinlich ein Mythos...
- ▶ Populäre Anwendung im Netz
 - ▶ Recommender-Systeme
 - ▶ Kunden, die A kauften, kauften auch B



Gegeben:

- ▶ Eine Menge von Einkäufen, z.B.
 - ▶ Nudeln, Tomaten, Basilikum, Tageszeitung
 - ▶ Brötchen, Tageszeitung
 - ▶ Nudeln, Tomaten, Hackfleisch, Basilikum, Zigaretten
 - ▶ ...

Finde:

- ▶ Häufige Muster in Form von Regeln, z.B.
 - ▶ Wenn Nudeln, dann auch Tomaten
 - ▶ Wenn Hackfleisch und Basilikum, dann auch Nudeln und Tomaten
 - ▶ Wenn Brötchen, dann auch Tageszeitung
 - ▶ ...



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 $X \rightarrow Y$, wobei $X \subseteq R$, $Y \subseteq R$, $X \cap Y = \{\}$

$$s(r, c) = \frac{|\{t \in r \mid X \cup Y \subseteq t\}|}{|r|} \geq s_{min} \quad (1)$$

$$conf(r, c) = \frac{|\{t \in r \mid X \cup Y \subseteq t\}|}{|\{t \in r \mid X \subseteq t\}|} \geq conf_{min} \quad (2)$$



Sei R eine Menge von Objekten, die binäre Werte haben, und r eine Menge von Transaktionen, dann ist $t \in r$ eine Transaktion und die Objekte mit dem Wert 1 sind eine Teilmenge aller Objekte.

$$R = \{A, B, C\}$$

$$t = \{B, C\} \subseteq R$$

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



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

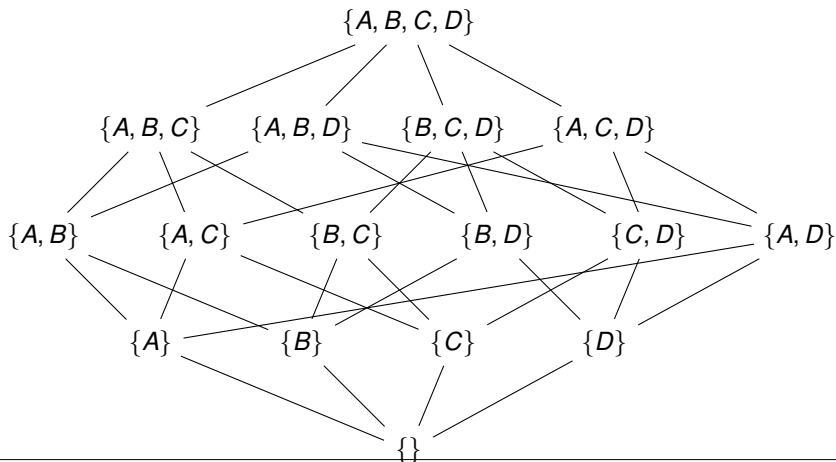
$$\{\text{Aftershave}\} \rightarrow \{\text{Bier}\} \quad s = \frac{1}{4}, \text{conf} = \frac{1}{2}$$

$$\{\text{Aftershave}\} \rightarrow \{\text{Chips}\} \quad s = 0$$

$$\{\text{Bier}\} \rightarrow \{\text{Chips}\} \quad s = \frac{1}{2}, \text{conf} = \frac{2}{3} \text{ (zusammen anbieten?)}$$

$$\{\text{Chips}\} \rightarrow \{\text{Aftershave}\} \quad s = 0$$

$$\{\text{Aftershave}\} \rightarrow \{\text{Bier, Chips}\} \quad s = 0$$





- ▶ 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.



LH: Assoziationsregeln sind keine logischen Regeln!

- ▶ In der Konklusion können mehrere Attribute stehen
- ▶ Attribute sind immer nur binär.
- ▶ Mehrere Assoziationsregeln zusammen ergeben kein Programm.

LE: Binärvektoren (Transaktionen)

- ▶ Attribute sind eindeutig geordnet.

Aufgabe:

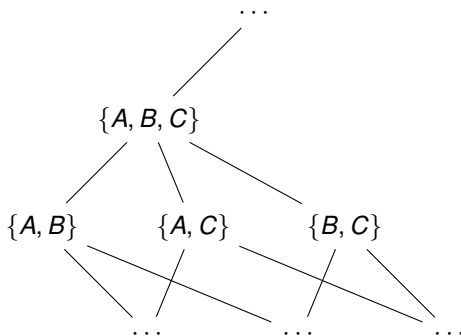
- ▶ Aus häufigen Mengen Assoziationsregeln herstellen



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

- ▶ 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 L_{k+1} .

Dies ist der Kern des Algorithmus, die Kandidatengenerierung.



- ▶ Wenn $\{A, B, C\}$ häufig ist, dann sind auch $\{A, B\}$, $\{A, C\}$, $\{B, C\}$ häufig.
- ▶ Das bedeutet, daß $\{A, B\}$, $\{A, C\}$, $\{B, C\}$ ($k = 2$) häufig sein müssen, damit $\{A, B, C\}$ ($k + 1 = 3$) häufig sein *kann*.
- ▶ Also ergeben die häufigen Mengen aus L_k die Kandidaten C_{k+1}

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.



► Erzeuge-Kandidaten(L_k)

- $C_{k+1} := \{\}$
- For all l_1, l_2 in L_k , sodass

$$l_1 = \{i_1, \dots, i_{k-1}, i_k\} \text{ und}$$

$$l_2 = \{i_1, \dots, i_{k-1}, i'_k\} i'_k < i_k$$

- $l := \{i_1, \dots, i_{k-1}, i_k, i'_k\}$
- if alle k -elementigen Teilmengen von l in L_k sind, then

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

- return C_{k+1}

► Prune(C_{k+1}, r) vergleicht Häufigkeit von Kandidaten mit s_{min} .



- ▶ Häufige-Mengen(R, r, s_{min})
 - ▶ $C_1 := \cup_{i \in R} i, k = 1$
 - ▶ $L_1 := \text{Prune}(C_1)$
 - ▶ while $L_k \neq \{\}$
 - ▶ $C_{k+1} := \text{Erzeuge-Kandidaten}(L_k)$
 - ▶ $L_{k+1} := \text{Prune}(C_{k+1}, r)$
 - ▶ $k := k + 1$
 - ▶ return $\cup_{j=2}^k L_j$



- ▶ Apriori($R, r, s_{min}, conf_{min}$)
 - ▶ $L := \text{Häufige-Mengen}(R, r, s_{min})$
 - ▶ $c := \text{Regeln}(L, conf_{min})$
 - ▶ return c

Aus den häufigen Mengen werden Regeln geformt. Wenn die Konklusion länger wird, kann die Konfidenz sinken. Die Ordnung der Attribute wird ausgenutzt:

$$l_1 = \{i_1, \dots, i_{k-1}, i_k\}$$

$$l_1 = \{i_1, \dots, i_{k-1}, i_k\}$$

...

$$l_1 = \{i_1, \dots, i_{k-1}, i_k\}$$

$$c_1 = \{i_1, \dots, i_{k-1}\} \rightarrow \{i_k\}$$

$$c_2 = \{i_1, \dots\} \rightarrow \{i_{k-1}, i_k\}$$

...

$$c_k = \{i_1\} \rightarrow \{\dots, i_{k-1}, i_k\}$$

*conf*₁

*conf*₂

...

*conf*_k

$$conf_1 \geq conf_2 \geq \dots \geq conf_k$$



- ▶ 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.



- ▶ 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 a priori Wahrscheinlichkeit nicht berücksichtigen. Wenn sowieso alle Cornflakes essen, dann essen auch hinreichend viele Fußballer Cornflakes.



1. $RI(A \rightarrow B) = 0$, wenn $|A \rightarrow B| = \frac{(|A||B|)}{|r|}$
 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 $|A \rightarrow B| > \frac{(|A||B|)}{|r|}$, d.h. wenn A positiv mit B korreliert ist.
- ▶ $RI < 0$, wenn $|A \rightarrow B| < \frac{(|A||B|)}{|r|}$, d.h. wenn A negativ mit B korreliert ist.

Wir wissen, dass immer $|A \rightarrow B| \leq |A| \leq |B|$ gilt, also

- ▶ RI_{min} , wenn $|A \rightarrow B| = |A|$ oder $|A| = |B|$
- ▶ RI_{max} , wenn $|A \rightarrow B| = |A| = |B|$

Piatetsky-Shapiro 1991

- ▶ 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 \rightarrow kein-Dienst-in-Vietnam

mit 90% Konfidenz. Tatsächlich ist $s(\text{kein-Dienst-in-Vietnam}) = 95\%$ 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 \rightarrow Cornflakes

hat Konfidenz 66% Tatsächlich senkt aber Basketball die Cornflakes Häufigkeit!



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

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

- Die Signifikanz der Korrelation zwischen A und B ist:

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

Was wissen Sie jetzt?



TECHNISCHE
UNIVERSITÄT
DARMSTADT

- ▶ Sie haben drei Prinzipien für die Regelbewertung kennengelernt:
 - ▶ 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 Maße kennen gelernt, die den Prinzipien genügen:
 - ▶ Einfaches Maß und
 - ▶ statistisches Maß



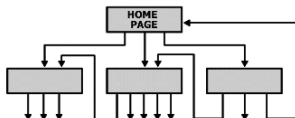
Das WWW hat zu einer Menge interessanter Forschungsaufgaben geführt. Unter anderem gibt es:

- ▶ Indexieren von Web-Seiten für die Suche – *machen wir hier nicht*
- ▶ Analysieren von Klick-Strömen – *web usage mining kommt später*
- ▶ Co-Citation networks – *machen wir hier nicht*
- ▶ Finden häufiger Muster in vernetzten Informationsquellen
- ▶ Ranking von Web-Seiten

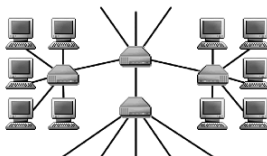
Das Web und das Internet als Graph

Webseiten sind Knoten, verbunden durch Verweise. Router und andere Rechner sind Knoten, physikalisch verbunden.

WORLD-WIDE WEB



INTERNET





Was sind besonders *wichtige* Seiten?

- ▶ Eine Seite, von der besonders viele Links ausgehen, heißt *expansiv*.
- ▶ Eine Seite, auf die besonders viele links zeigen, heißt *beliebt*.
- ▶ Wie oft würde ein zufälliger Besucher auf eine Seite i kommen? Zufällige Besuche von einer beliebigen Startseite aus:
 - ▶ Mit der Wahrscheinlichkeit α folgt man einer Kante der aktuellen Seite (Übergangswahrscheinlichkeit).
 - ▶ Mit der Wahrscheinlichkeit $1 - \alpha$ springt man auf eine zufällige Seite, unter der Annahme, dass die Seiten gleich verteilt sind (Sprungwahrscheinlichkeit).

Der Rang einer Seite $PageRank(i)$ ist der Anteil von i an den besuchten Knoten.



Matrix M_{ij} für Kanten von Knoten j zu Knoten i ; $n(j)$ ist die Anzahl der von j ausgehenden Kanten; N Knoten insgesamt.

$$\begin{pmatrix} & 1 & \dots & N \\ 1 & 0 & \dots & M_{1N} \\ \vdots & \dots & M_{ij} = 1/n(j) & \dots \\ N & \dots & \dots & 0 \end{pmatrix}$$

Matrix $N \times N$ mit den Einträgen $1/N$ gibt die Gleichverteilung der Knoten an (Sprungwahrscheinlichkeit).

Die Wahrscheinlichkeit, die Seite zu besuchen, ist die Summe von Sprung- und Übergangswahrscheinlichkeit, angegeben in $N \times N$ Matrix M' :

$$M' = (1 - \alpha) \left[\frac{1}{N} \right] + \alpha M \quad (3)$$



Eigenvektoren von M' geben den Rang der Knoten an.
Man kann das Gleichungssystem für $\alpha < 1$ lösen:

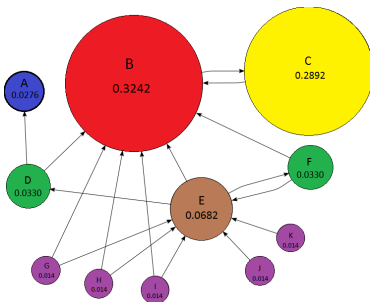
$$\text{Rang}_i = (1 - \alpha) \left[\frac{1}{N} \right] + \alpha \sum_j M^{-1} ij$$

PageRank ist der rekursive Algorithmus:

$$\text{Rang}_i = \frac{1 - \alpha}{N} + \alpha \sum_{\forall j \in \{(i,j)\}} \frac{\text{Rang}_j}{n(j)} \quad (4)$$

PageRank Beispiel

Mit $\alpha = 0,85$ hier ein kleines Beispiel (wikipedia). Die Größe der Kreise entspricht der Wahrscheinlichkeit, mit der ein Surfer auf die Seite kommt. Seite C wird nur von einer einzigen, aber gewichtigeren Seite verlinkt und hat also einen höheren PageRank als Seite E, obwohl E von sechs Seiten verlinkt wird.



Was wissen Sie jetzt?



TECHNISCHE
UNIVERSITÄT
DARMSTADT

- ▶ Sie kennen jetzt die Grundlage des Ranking von Web-Seiten und einige Probleme.
- ▶ PageRank schätzt die Wahrscheinlichkeit ab, auf die Seite zu kommen, indem es Kanten folgt und zufällig auf Knoten springt. Dabei verwendet es die Wahrscheinlichkeit α als Gewicht der Übergangswahrscheinlichkeiten und $1 - \alpha$ als Gewicht der Sprungwahrscheinlichkeit.