Einführung in Web- und Data-Science

Prof. Dr. Ralf Möller

Universität zu Lübeck

Institut für Informationssysteme

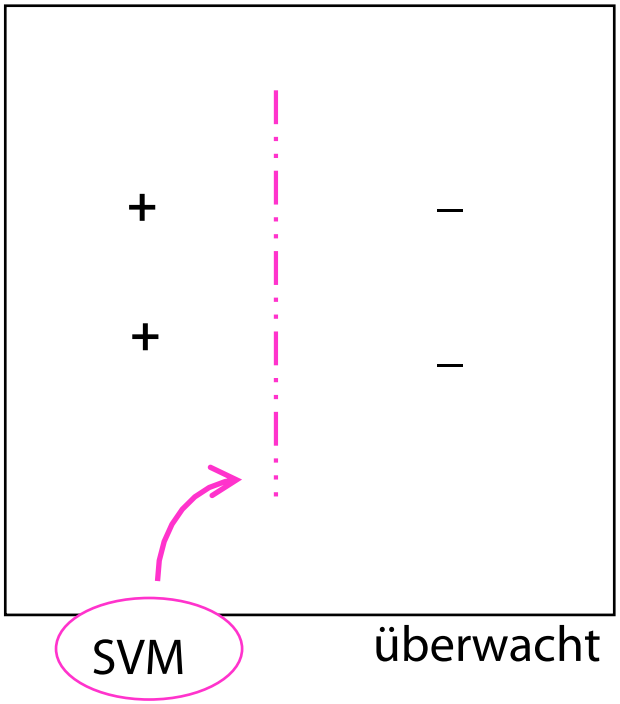
Tanya Braun (Übungen)

Wiederholung: Überwachtes Lernen

- Gegeben:
  - Tabellarische Daten,
  - Klassifikationsattribut vorhanden
    (Überwachung durch klassifizierte Daten)
- Gesucht: Klassifikator für neue Daten
- Klassifikator erstellbar z.B. durch
  - Netze mit Parametrierungs- und Kompositionsmöglichkeit
  - Support-Vektor-Maschinen mit Kernel-Operator

- Heute: Unüberwachtes Lernen
  (kein Klassifikationsattribut vorhanden)

Ausnutzung von Daten bei SVMs: Clusterbildung



Nur klassifizierte Daten

Clusterbildung z.B. realisierbar durch k-nächste-Nachbarn-Klassifikation
(also instanzbasiert, nicht modellbasiert)

Assoziationsregeln

Folgende 18 Präsentationen sind inspiriert von Präsentationen aus:
CAS CS 565, Data Mining, Fall 2012, Evimaria Terzi, Boston Univ.

- Gegeben eine Menge von Warenkörben, finde Regeln, die das Auftreten eines Artikels (oder mehrerer Artikel) vorhersagt
- Warenkorbeintragung in DB im Jargon Transaktion genannt (Daten aus Online-Transaction-Processing, OLTP)

Warenkorbtransaktionen

TID	*Artikel*
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

Beispiele für Assoziationsregeln

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

Rakesh Agrawal, Tomasz Imieliński, Arun Swami: Mining Association Rules between Sets of Items in Large Databases. In: Proc. 1993 ACM SIGMOD International Conference on Management of data, SIGMOD Record. Bd. 22, Nr. 2, Juni **1993**

Häufige Artikelmengen

- Gegeben eine Datenmenge D in Form von Warenkörben, finde Kombination von Artikeln, die häufig zusammen vorkommen

Warenkorbtransaktionen

TID	*Artikel*
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

Beispiele für häufige Artikelmengen

{Diaper, Beer},
{Milk, Bread}
{Beer, Bread, Milk},

Definition: Häufige Artikelmenge

- **Artikelmenge** $\subseteq$ Gesamt-Artikelmenge I
  - Z.B.: {Milk, Bread, Diaper}
- **Unterstützungszähler (σ)**
  - Anzahl $\sigma(w)$ des Auftretens der Artikelmenge w in den Daten (Anzahl der Warenkörbe, in denen Artikelmenge vorkommt)
  - Z.B.: $\sigma(\{Milk, Bread, Diaper\}) = 2$

TID	*Artikel*
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

- **Unterstützung (support)**
  - Anteil der Warenkörbe, in denen Artikelmenge vorkommt: $s(w) = \sigma(w) / |D|$
  - Z.B.: $s(\{Milk, Bread, Diaper\}) = 2/5$
- **Häufige Artikelmenge (frequent itemset)**
  - Artikelmenge mit Support $\geq$ ***minsup*** (Schwellwert)

Warum häufige Artikelmengen finden?

- Interessant für Platzierung von Artikeln
  (im Supermarkt, auf Webseiten, …)

- Häufige Artikelmengen kennzeichnen positive Kombinationen
  (seltene Artikelmengen kaum relevant),
  bieten also Zusammenfassung einer Datenmenge

- …

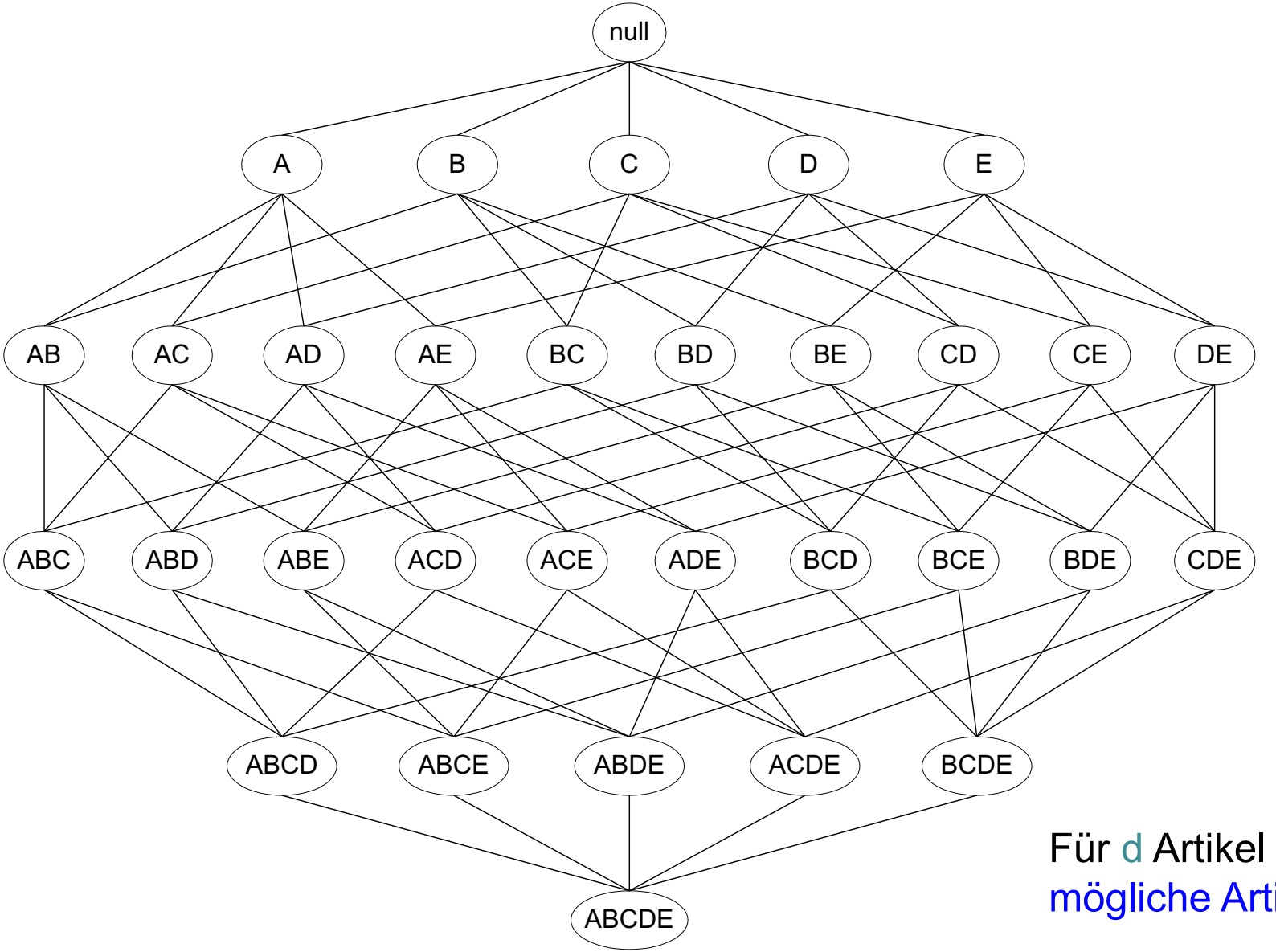
Suche nach häufigen Artikelmengen

- Aufgabe:
  - Gegeben eine Transaktionsdatenbasis D (Warenkörbe) und ein Schwellwert minsup
  - Finde alle häufigen Artikelmengen (und deren jeweilige Anzahl in den Daten)

- Anders gesagt: Zähle die jeweiligen Vorkommen von Kombinationen von Artikeln in den Daten über einem Schwellwert minsup

- Annahme: Gesamt-Artikelmenge I bekannt

Wie viele Artikelmengen gibt es?



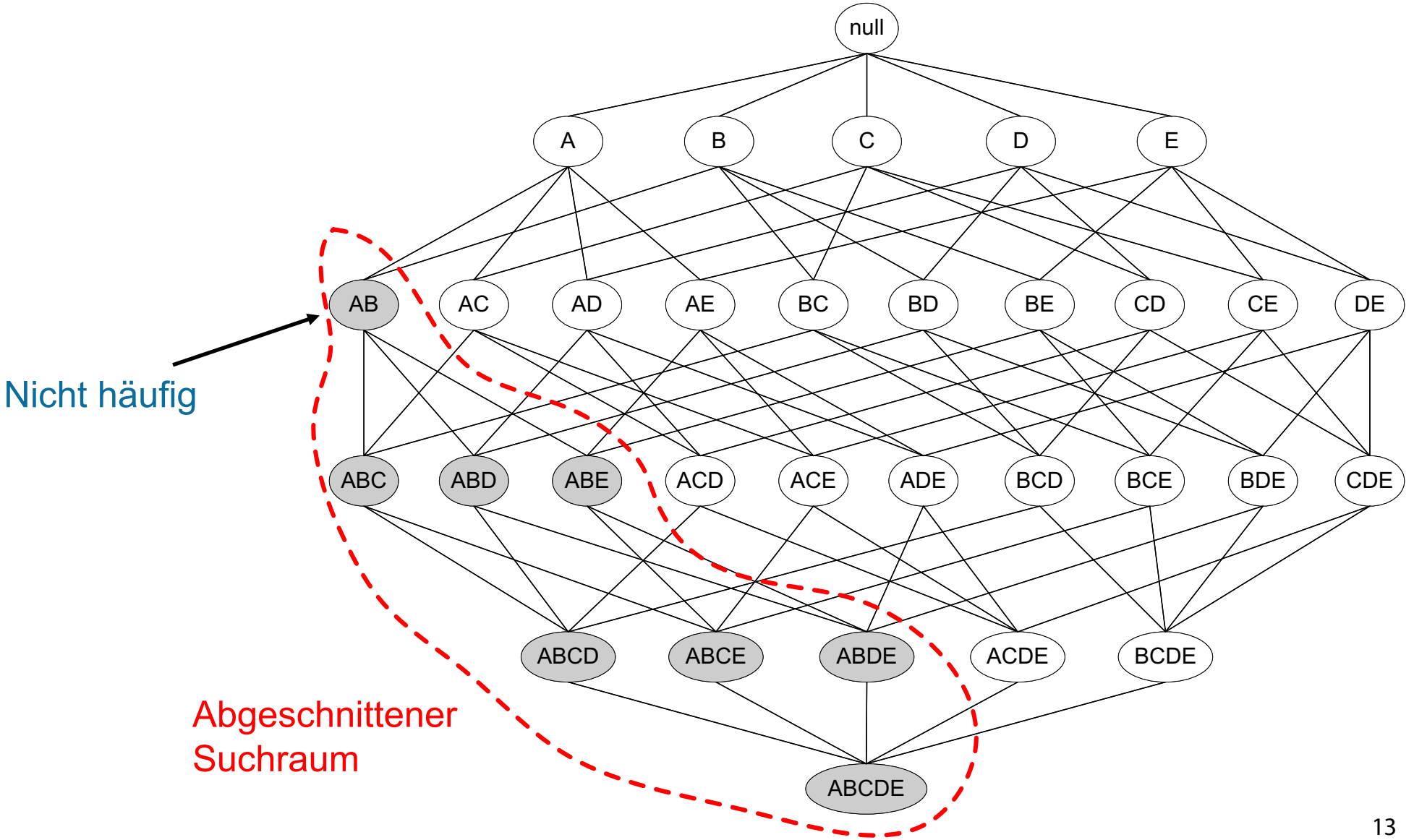
Für d Artikel gibt es 2^d mögliche Artikelmengen

Monotonie vom Support

TID	*Artikel*
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

s(Bread) > s(Bread, Beer)
s(Milk) > s(Bread, Milk)
s(Diaper, Beer) > s(Diaper, Beer, Coke)

Apriori-Verfahren (Idee)



Apriori-Verfahren (Prinzip)

1er-Artikelmengen

Artikel	Anzahl
Bread	**4**
Coke	**2**
Milk	**4**
Beer	**3**
Diaper	**4**
Eggs	**1**

minsup = 3/5

Artikelmenge	Anzahl
{Bread,Milk}	**3**
{Bread,Beer}	**2**
{Bread,Diaper}	**3**
{Milk,Beer}	**2**
{Milk,Diaper}	**3**
{Beer,Diaper}	**3**

2er-Artikelmengen

(Cola und Eier nicht mehr berücksichtigt)

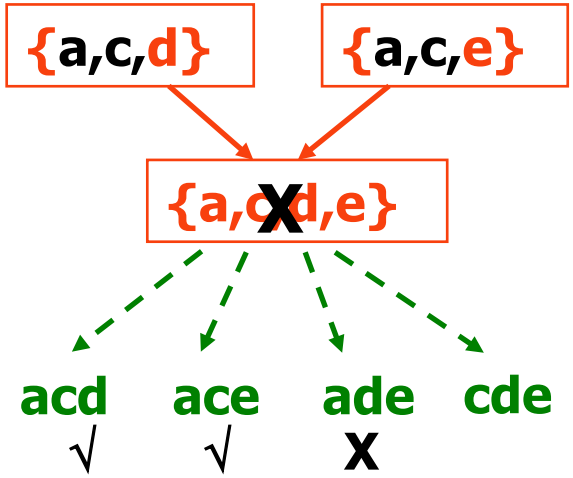
3er-Artikelmengen

Artikelmenge	Anzahl
{Bread,Milk,Diaper}	**3**

Rakesh Agrawal, Ramakrishnan Srikant: *Fast Algorithms for Mining Association Rules.* In: Proc. 20th International Conference on Very Large Data Bases. Morgan Kaufmann Publishers Inc., **1994**

Apriori-Verfahren (Prinzip)

- L_3={*abc, abd, acd, ace, bcd*}
- ***Self-Join***: $L_3 \bowtie L_3$
  - ***abcd*** aus ***abc*** und ***abd***
  - ***acde*** und ***acd*** und ***ace***
- ***Beschneidung:***
  - ***acde*** entfernt, weil ***ade*** nicht in L_3
- C_4={*abcd*}



Definition: Assoziationsregel

Sei D eine Datenbasis von Transaktionen

Transaction ID	Artikel
2000	A, B, C
1000	A, C
4000	A, D
5000	B, E, F

- Sei I die Artikelmenge in der DB, z.B.: $I=\{A,B,C,D,E,F\}$
- Eine Regel ist definiert durch $X \rightarrow Y$, wobei $X \subset I$, $Y \subset I$, $X \neq \emptyset$, $Y \neq \emptyset$, and $X \cap Y = \emptyset$
  - Beispiel: {B,C} → {A} ist eine Regel

Definition: Bewertungsmaße für Regel

- Unterstützung/Support s(.)
  - Anteil der Transaktionen, die X und Y enthalten
- Konfidenz c(.)
  - Maß, wie oft Artikel Y in Transaktionen vorkommen, die auch X enthalten

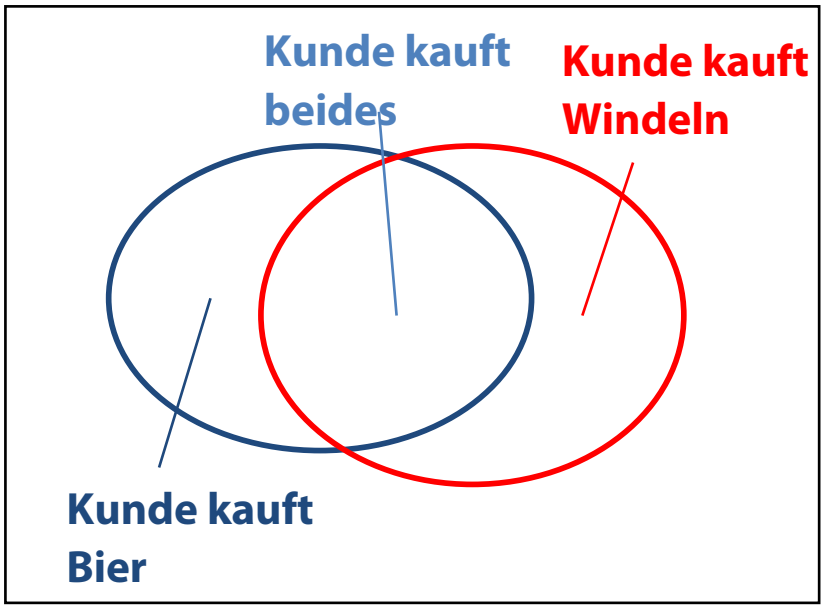
TID	Items
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

Beispiel: $\{\text{Milk}, \text{Diaper}\} \rightarrow \text{Beer}$

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

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

Mining von Assoziationsregeln



Finde alle Regeln $r = X \rightarrow Y$ mit

- $s(r) \geq$ minsup und
- $c(r) \geq$ minconf
- Support *s*: relative Häufigkeit (in %), von Transaktionen, die $X \cup Y$ enthalten
- Konfidenz *c:* Bedingte relative Häufigkeit (in %) von Transaktionen Y enthalten, wenn sie auch X enthalten

TID	Items
100	A,B,C
200	A,C
300	A,D
400	B,E,F

Sei der minimale Support 50% und die minimale Konfidenz 50%:

- $A \rightarrow C$ (50%, 66.6%)
- $C \rightarrow A$ (50%, 100%)

Brute-Force-Verfahren

- Betrachte alle möglichen Assoziationsregeln
- Berechne Support und Konfidenz für jede Regel
- Eliminiere Regeln, deren Support oder Konfidenz kleiner als minsup und minconf Schwellwerte

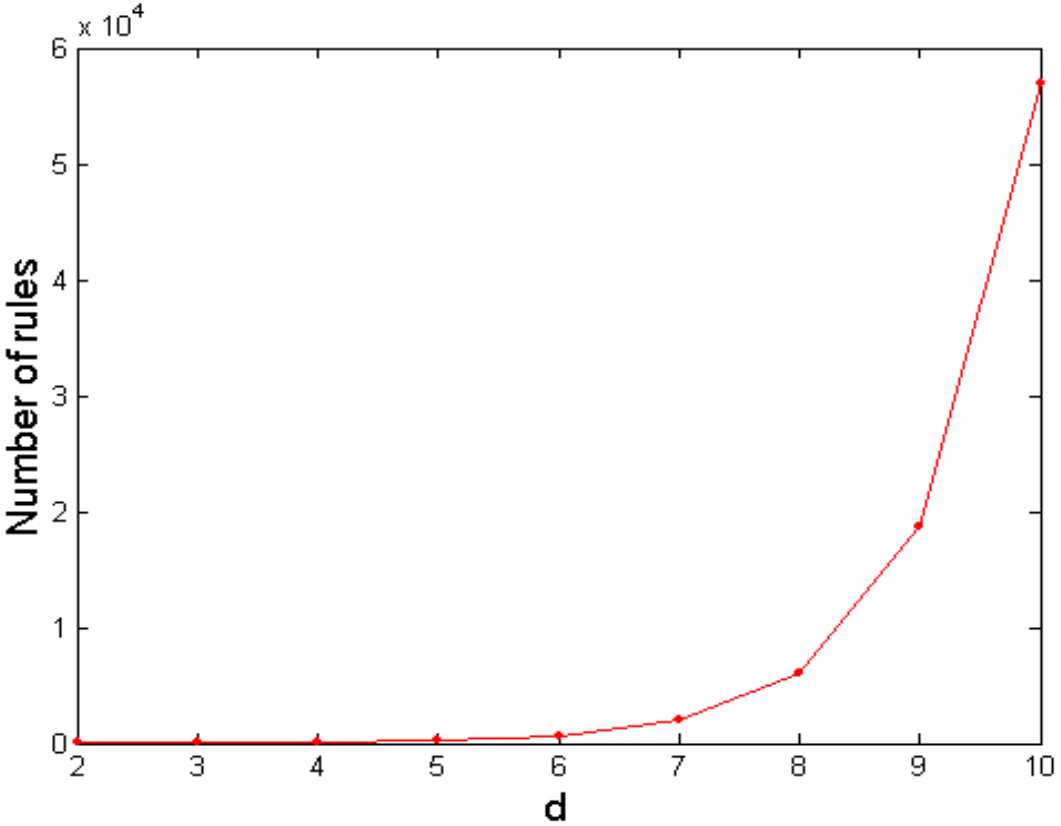
- $\Rightarrow$ Zu aufwendig! Kombinatorische Explosion

Berechnungsaufwand

- Gegeben d Artikel in I:
  - Anzahl der Artikelmengen: 2^d
  - Anzahl der Assoziationsregeln:

$$\sum_{k=1}^{d-1}\left[\binom{d}{k}\times\sum_{j=1}^{d-k}\binom{d-k}{j}\right]$$

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



Wenn d=6, R = 602 Regeln

Mining von Assoziationsregeln

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

Beispiele für Regeln:

{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)

Beobachtungen:

Regeln sind binäre Partitionen der gleichen Artikelmenge:
{Milk, Diaper, Beer}

Regeln von der gleichen Artikelmenge haben gleichen Support aber verschiedene Konfidenz

Entkopplung von Support und Konfidenz

Zweischrittiger Ansatz

- Generiere häufige Artikelmengen mit

  support $\geq$ minsup

- Generiere Assoziationsregeln

  durch binäre Partitionierung
  von häufigen Artikelmengen, so dass

  confidence $\geq$ minconf

Regelgenerierung – Einfacher Ansatz

- Gegeben die häufige Artikelmenge X, finde alle nichtleeren Teilmengen $y \subset X$, so dass $y \rightarrow X - y$ die Konfidenzanforderung erfüllt

  Beispiel: $\{A,B,C,D\}$ sei eine häufige Artikelmenge:

  $ABC \rightarrow D$, $ABD \rightarrow C$, $ACD \rightarrow B$, $BCD \rightarrow A$,
  $A \rightarrow BCD$, $B \rightarrow ACD$, $C \rightarrow ABD$, $D \rightarrow ABC$
  $AB \rightarrow CD$, $AC \rightarrow BD$, $AD \rightarrow BC$, $BC \rightarrow AD$,
  $BD \rightarrow AC$, $CD \rightarrow AB$

- Falls $|X| = k$, dann gibt es $2^k - 2$ Kandidatenregeln (ohne $L \rightarrow \emptyset$ und $\emptyset \rightarrow L$)

Anwendungen der Warenkorbanalyse

- "Verstehen" von Transaktionsdaten
- Data Mining (offline)
  - Systematische Anwendung statistischer Methoden auf große Datenbestände, mit dem Ziel, ...
  - ... neue Querverbindungen und Trends zu erkennen
- Empfehlungsgenerierung (online)
  - Vorhersage treffen, die quantifiziert wie stark das Interesse eines Benutzers an einem Objekt ist, ...
  - ... um dem Benutzer genau die Objekte aus der Menge aller vorhandenen Objekte zu empfehlen, für die er sich vermutlich am meisten interessiert

Wikipedia

Anwendung: Empfehlungsgenerierung

Suche

Artikel +
Empfehlungen

Artikel

- Beschränktheit der Ressource "Platz"
- Was sind "gute" Empfehlungen
  - Steigerung der Kundenzufriedenheit
  - Steigerung des Umsatzes des Anbieters
- Techniken
  - Verwendung von häufigen Artikelmengen
  - Verwendung von Assoziationsregeln

Produkte, Websites, Blogs, Schlagzeilen, …

Verfeinerung der Empfehlungsgenerierung

Personalisierung: Kundenspezifische Empfehlung

Schätzung der Kundenzufriedenheit über "Nützlichkeit"

- C: Menge von Kunden
- S: Menge von Artikeln
- Nützlichkeitsmaß: $u : C \times S \longrightarrow R$
  - R: Menge von Bewertungen (total geordnete Menge)
  - Beispiele: 0-5 Sterne, reelle Zahlen aus [0,1]
- Nützlichkeit = engl. Utility

Maximierung der Nützlichkeitsschätzung

- Für jeden Nutzer $c \in C$
  bestimme diejenigen Artikel s' aus dem Sortiment S,
  die die Nützlichkeiten für den Nutzer c maximieren

$$\forall c \in C, \quad s'_c = \arg\max_{s \in S} u(c, s).$$

- Kundenspezifische Nützlichkeit s'_c eines Artikels
  definiert Rang der Artikel

Zentrales Problem

- Nützlichkeit nur partiell definiert, also nicht für alle Elemente aus dem *CxS* Raum bekannt
- Nützlichkeit u muss extrapoliert werden

	King Kong	LOTR	Matrix	Nacho Libre
Alice	1		0.2	
Bob		0.5		0.3
Carol	0.2		1	
David				0.4

Woher kommen die Einträge?

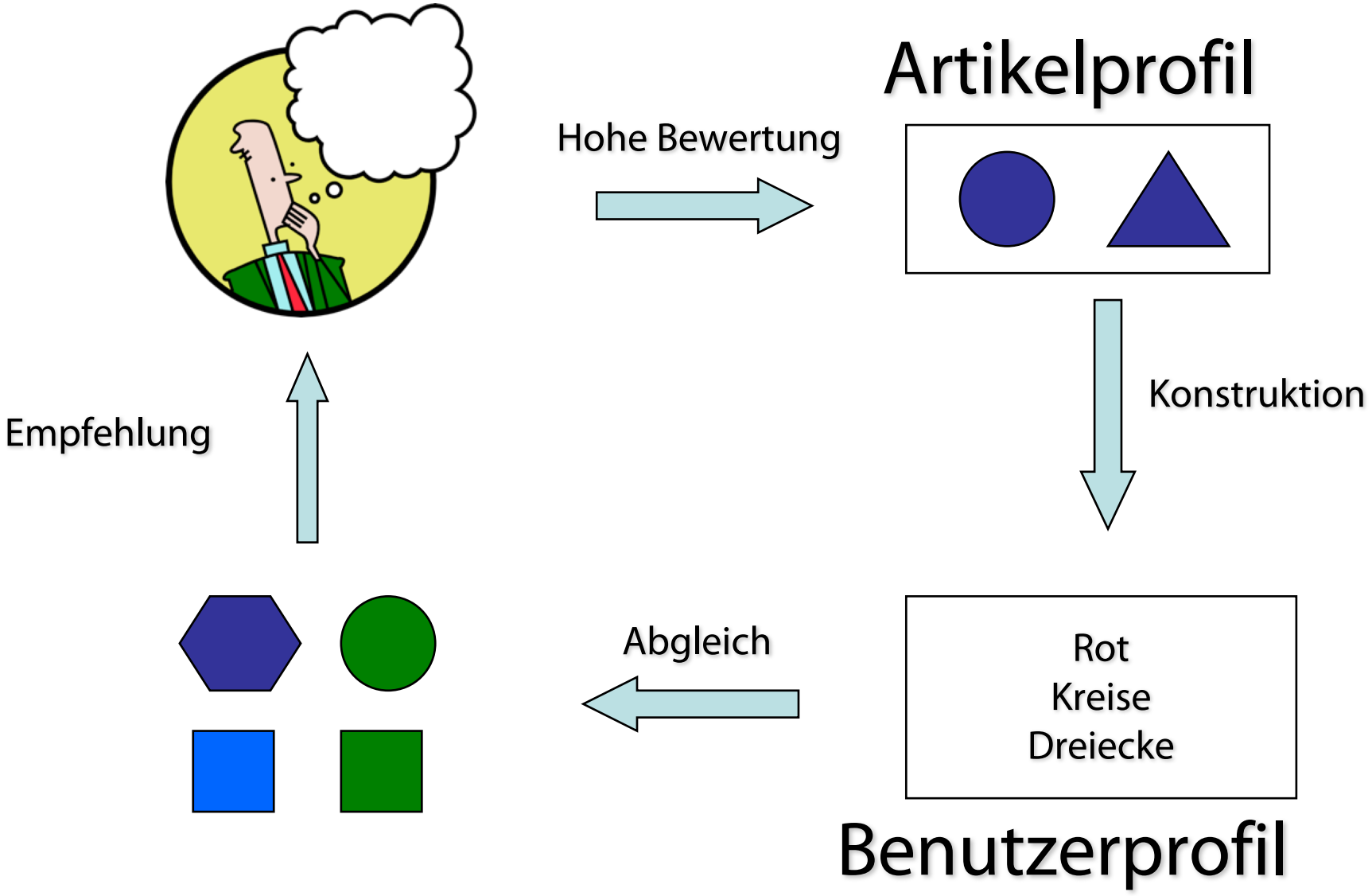
Erfassung von Nützlichkeitsmaßen

- Explizit
  - Nutzer bewerten Artikel
  - Funktioniert nicht in der Praxis, Nutzer werden gestört
- Implizit
  - Erfasse Maße aus Nutzeraktionen
    - Kauf eines Artikels ergibt gute Bewertung
    - Was ist mit schlechten Bewertungen?

Extrapolierung der Nützlichkeiten

- Schlüsselproblem: Matrix U ist dünn besetzt
  - Die meisten Leute habe die meisten Artikel nicht bewertet
  - Extrapolation nötig (Filterung)
- Ansätze
  - Inhaltsbasierte Filterung
    - Empfehlung von Artikeln, die "ähnlich" zu den schon hoch bewerteten sind: Wähle u(c,s) wie u(c, s') mit sim(s, s')
  - Kollaborative Filterung
    - Empfehlung von Artikeln, die von "ähnlichen" Benutzern hoch bewertet werden: u(c,s) wie u(c', s) mit sim(c, c')

Übersicht


Artikelprofil
Hohe Bewertung
Konstruktion
Empfehlung
Abgleich
Rot
Kreise
Dreiecke
Benutzerprofil

Artikelmerkmale

- Für jeden Artikel s generiere Artikelprofil content(s)
- Profil ist Menge von Merkmalswerten
  - Text: Menge von (Wort, Gewicht)-Paaren
    - Kann als Vektor gedeutet werden
  - Auch für Filme:
    - Text extrahieren aus Angaben zu
      Titel, Schauspieler, Regisseur, usw.
- Wie findet man Gewichtsangaben?
  - Standardansatz: TF.IDF
    (Term Frequency by Inverse Doc Frequency)

TF.IDF

f_{ij} = relative Anzahl der Terme t_i im Document d_j

$$TF_{ij} = \frac{f_{ij}}{\underset{k}{\text{argmax}}\ f_{kj}}$$

n_i = Anzahl der Dokumente in denen Term i vorkommt

N = Gesamtanzahl der Dokumente

$$IDF_i = \log \frac{N}{n_i}$$

TF.IDF-Maß $w_{ij} = TF_{ij} \cdot IDF_i$

Inhaltsbasierte Nützlichkeitsschätzung (Filterung)

- Für Nutzer c nehme zugeordnete Artikel items(c)…
- ... und bestimme content(s) für alle s ∈ items(c)
- Definiere profile(c) als
  - Mittel der content(s) für alle s ∈ items(c) oder
  - Mittel der Abstände vom Mittelwert von content(s) mit s ∈ items(c) (weitere Definitionen sind möglich)
- Wir erhalten: Menge von (Term, Gewicht)-Paaren
  - Kann als Vektor w gedeutet werden
- Nützlichkeits-funktion u(c, s):
- Profil auch Bewertung genannt

$$u(c,s) = \cos(\vec{w}_c, \vec{w}_s) = \frac{\vec{w}_c \cdot \vec{w}_s}{\|\vec{w}_c\|_2 \times \|\vec{w}_s\|_2}$$

$$= \frac{\sum_{i=1}^{K} w_{i,c} w_{i,s}}{\sqrt{\sum_{i=1}^{K} w_{i,c}^2} \sqrt{\sum_{i=1}^{K} w_{i,s}^2}},$$

$\|\cdot\|_2$ denotiert Euklidische Norm (Länge des Vektors)

Begrenzungen der inhaltsbasierten Filterung

- Merkmale nicht immer einfach zu definieren
  - Bilder?, Musik?
  - Meist umgebender oder zugeordneter Text verwendet
- Überspezialisierung
  - Artikel außerhalb des Profils werden nicht empfohlen
  - Menschen haben verschiedene Interessen
- Empfehlungen für neue Benutzer
  - Wie ist das Profil definiert?
  - Rückgriff auf:
    - Häufige Artikelmengen (nutzerunspezifisch)
    - Assoziationsregeln (nutzerunspezifisch)

Nutzer-Nutzer kolloborative Filterung

- Betrachte Nutzer c
- Bestimme Menge D von Nutzern, deren Bewertungen "ähnlich" zu denen von c sind
- Schätze $profile(c)$ aus den Angaben $profile(d)$ für $d \in D$

- Was sind "ähnliche" Nutzer?

Ähnliche Nutzer: Distanzmaße

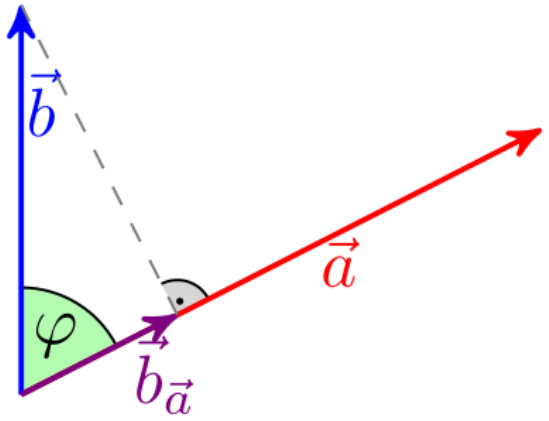
Sei eine Nutzerbewertung r_c = profile(c) gegeben, dann definiere Ähnlichkeit sim der Nutzer c_1 und c_2 als

1. Kosinusähnlichkeit

$sim(c_1,c_2) = cos(r_{c_1}, r_{c_2})$

$$\mathbf{a} \cdot \mathbf{b} = \|\mathbf{a}\| \, \|\mathbf{b}\| \cos\theta$$

$$\cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \sqrt{\sum_{i=1}^{n} B_i^2}}$$



Orthogonale Projektion $\vec{b}_{\vec{a}}$ des Vektors $\vec{b}$ auf die durch $\vec{a}$ bestimmte Richtung

Wikipedia

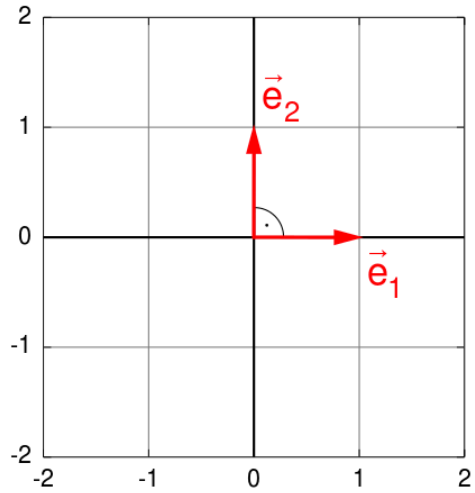
$$\vec{a} \cdot \vec{b} = a_1\, b_1 + a_2\, b_2 + a_3\, b_3\,.$$

Für die kanonischen Einheitsvektoren $\vec{e}_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ und $\vec{e}_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$ gilt nämlich:

$$\vec{e}_1 \cdot \vec{e}_1 = 1,\ \vec{e}_1 \cdot \vec{e}_2 = \vec{e}_2 \cdot \vec{e}_1 = 0 \text{ und } \vec{e}_2 \cdot \vec{e}_2 = 1$$

Daraus folgt (unter Vorwegnahme der weiter unten erläuterten Eigenschaften des Skalarproduktes):

$$\begin{aligned} \vec{a} \cdot \vec{b} &= (a_1\, \vec{e}_1 + a_2\, \vec{e}_2) \cdot (b_1\, \vec{e}_1 + b_2\, \vec{e}_2) \\ &= a_1 b_1\, \vec{e}_1 \cdot \vec{e}_1 + a_1 b_2\, \vec{e}_1 \cdot \vec{e}_2 + a_2 b_1\, \vec{e}_2 \cdot \vec{e}_1 + a_2 b_2\, \vec{e}_2 \cdot \vec{e}_2 \\ &= a_1 b_1 + a_2 b_2 \end{aligned}$$



Kanonische Einheitsvektoren in der euklidischen Ebene

Wikipedia

Ähnliche Nutzer: Distanzmaße

Sei eine Nutzerbewertung $r_c = profile(c)$ gegeben, dann definiere Ähnlichkeit sim der Nutzer c_1 und c_2 als

1. Kosinusähnlichkeit
   - $sim(c_1,c_2) = cos(r_{c_1}, r_{c_2})$ oder als
2. Funktion über Bewertungen $x=r_{c_1}$ and $y= r_{c_2}$, so dass
   - falls c_1 und c_2 gleiche Bewertungen vergeben → max
   - Normalisierung von x und y nötig
   - Allgemein bekannt als:
     - Pearson Korrelationskoeffizient oder
     - empirischer Korrelationskoeffizient

Korrelationskoeffizient

Normalisierte Werte (z-Transformation)

$$r = \frac{1}{n-1} \sum_{i=1}^{n} \frac{(x_i - \bar{x})}{s_x} \frac{(y_i - \bar{y})}{s_y}$$

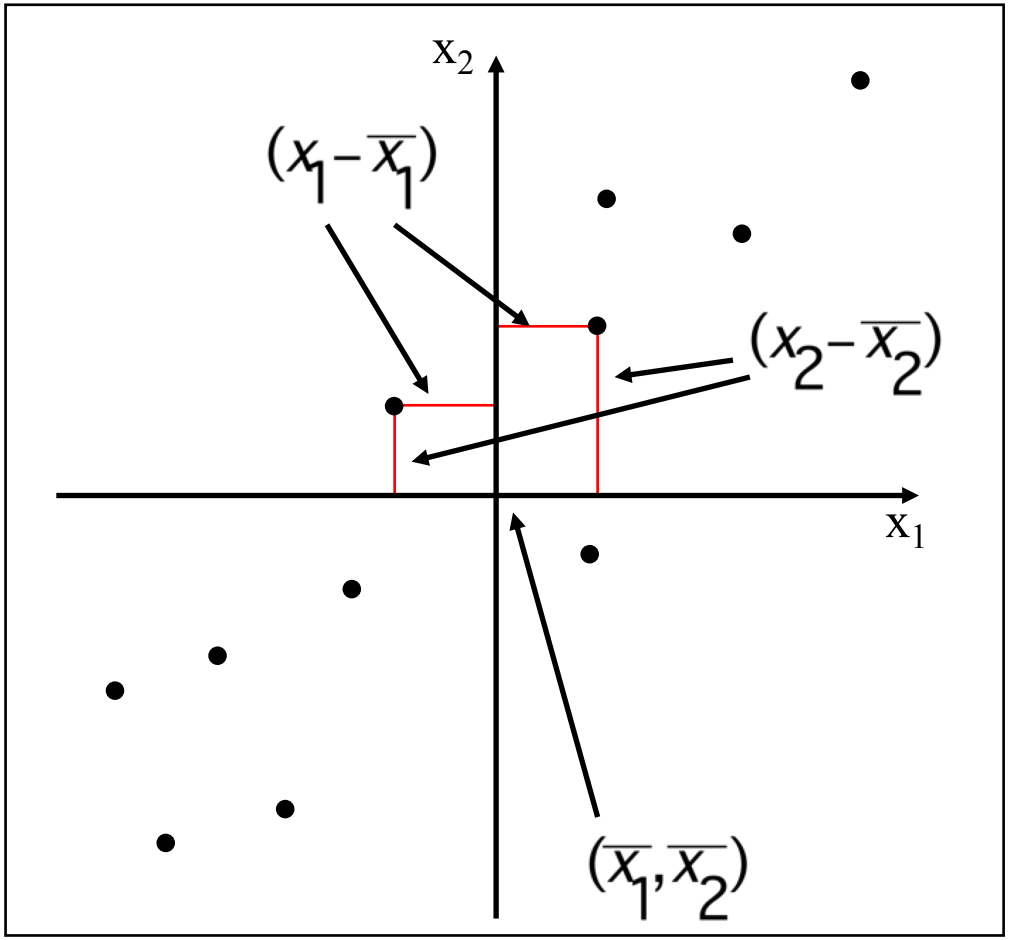
n = Anzahl der Artikel		
x_i = Bewertungskomp. x		y_i = Bewertungskomp. y
$\bar{x}$ = Mittel von x		$\bar{y}$ = Mittel von y
s_x= Standardabweichung x		s_y= Standardabweichung y

Korrelationskoeffizient

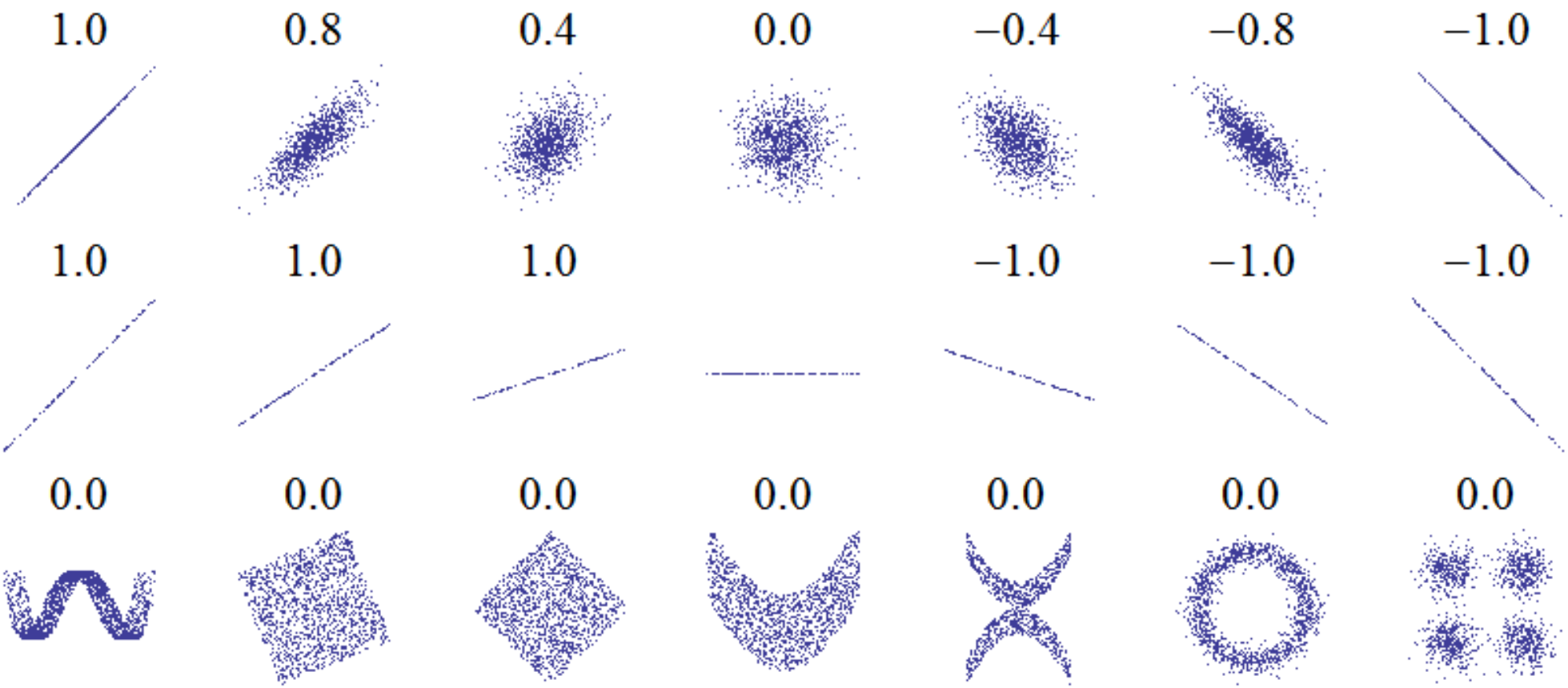
$$s_x = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})^2}$$

$$r = \frac{\sum_{i=1}^{n}(x_{i1} - \overline{x_1})(x_{i2} - \overline{x_2})}{\sqrt{\sum_{i=1}^{n}(x_{i1} - \overline{x_1})^2}\sqrt{\sum_{i=1}^{n}(x_{i2} - \overline{x_2})^2}}$$

$-1 \le r \le +1$



Korrelationskoeffizient: Veranschaulichung



Wikipedia

Bewertungsschätzungen

- Sei D die Menge der k ähnlichsten Nutzer zu c, die Artikel s bewertet haben
- Definiere Schätzfunktion für Bewertung von s:
  - $r_{cs} = 1/k \sum_{d \in D} r_{ds}$ oder
  - $r_{cs} = (\sum_{d \in D} sim(c,d) \cdot r_{ds})/(\sum_{d \in D} sim(c,d))$
  - …

Aufwand?

- Aufwendige Suche nach k ähnlichsten Nutzern
  - Betrachtung aller Nutzer
- Kann kaum zur "Laufzeit" erfolgen
  - Vorausberechnung nötig

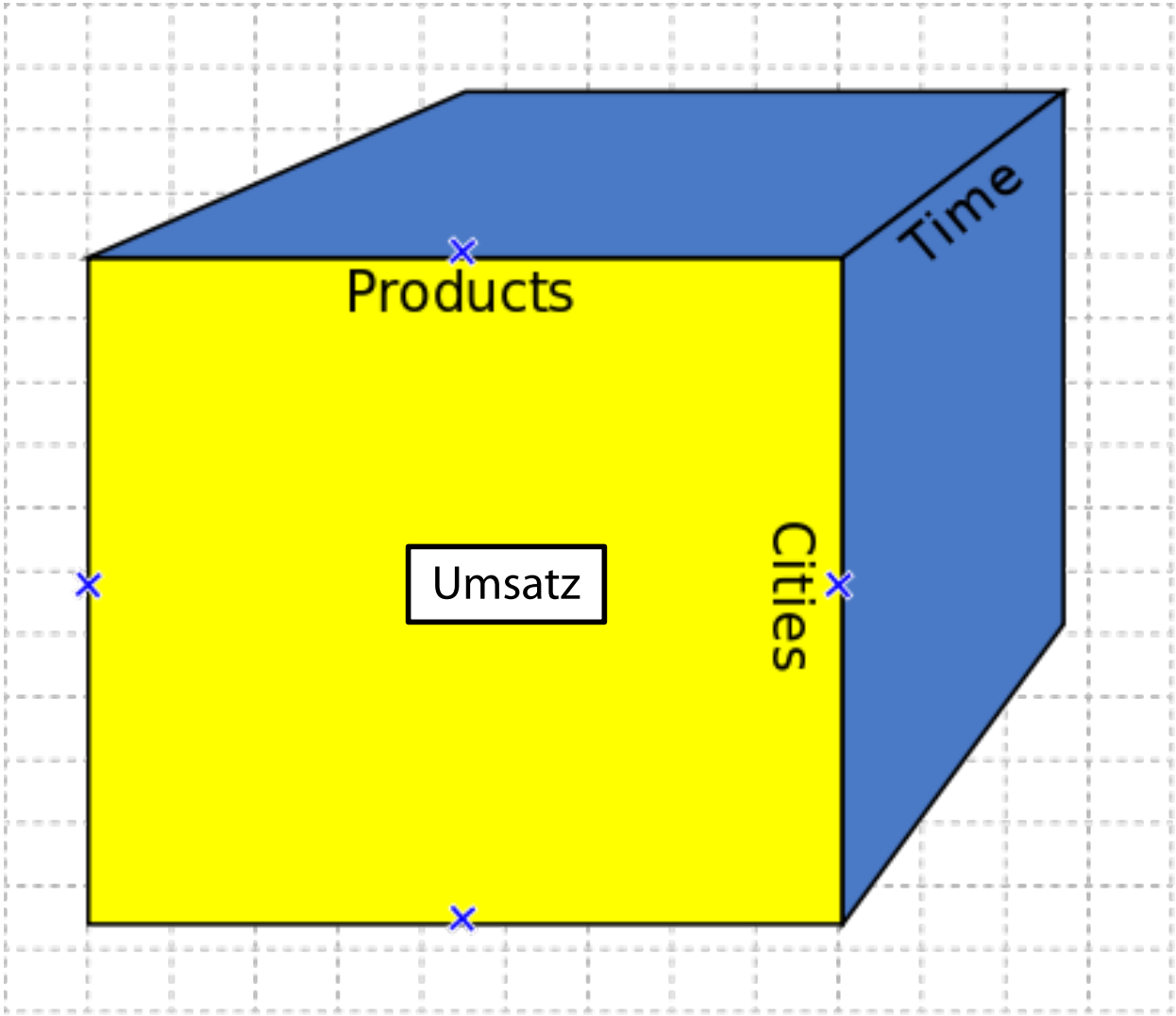
- Alternative zur Berechnung von r_{cs} ?
  - Suche nach ähnlichen Artikeln
    - Artikel-Artikel-kollaborative-Filterung
    - Sonst gleiches Vorgehen
  - Suche Assoziationsregeln mit s als Vorbedingung
  - Suche häufige Artikelmengen mit s als Element

Online Analytical Processing (OLAP)

- Hypothesengestützte Analysemethode
- Daten aus den operationalen Datenbeständen eines Unternehmens oder aus einem Data-Warehouse (Datenlager)
- Ziel: Durch multidimensionale Betrachtung dieser Daten ein entscheidungsunterstützendes Analyseergebnis in Bezug auf die Hypothese zu gewinnen
- zugrundeliegende Struktur ist ein OLAP-Würfel (englisch cube), der aus der operationalen Datenbank erstellt wird

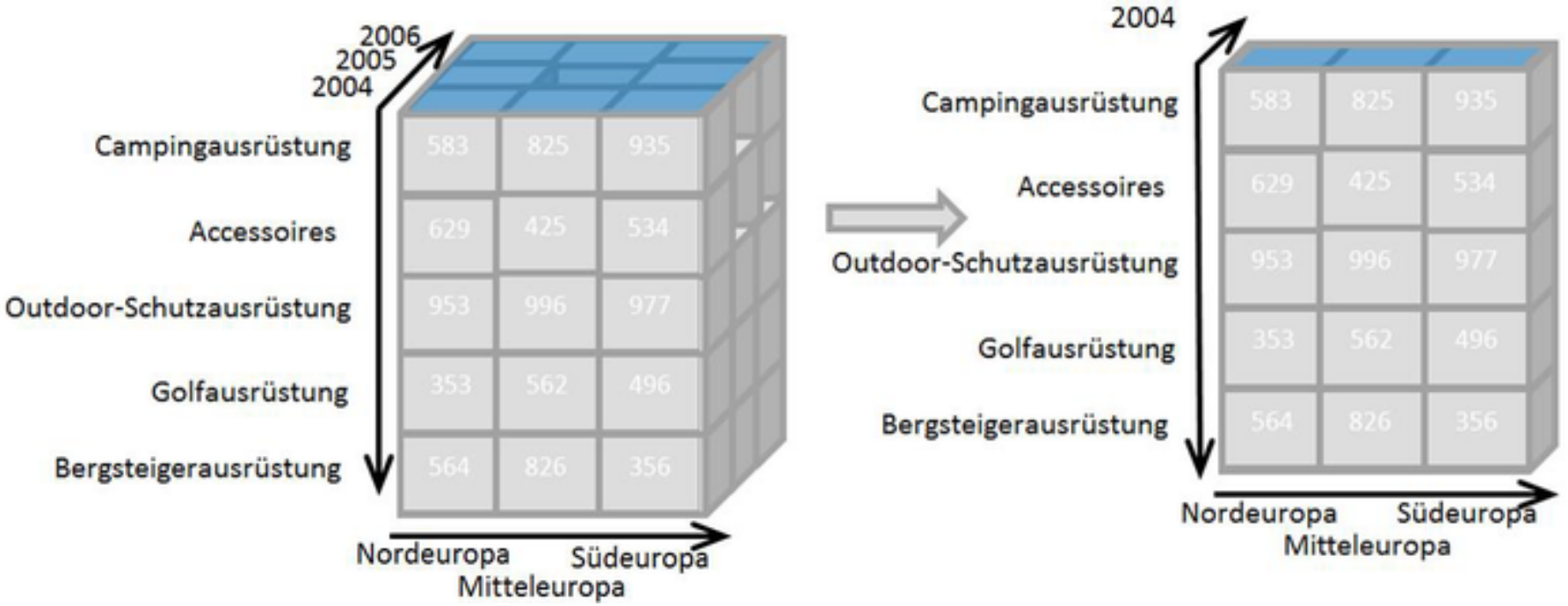
Wikipedia

OLAP-Würfel



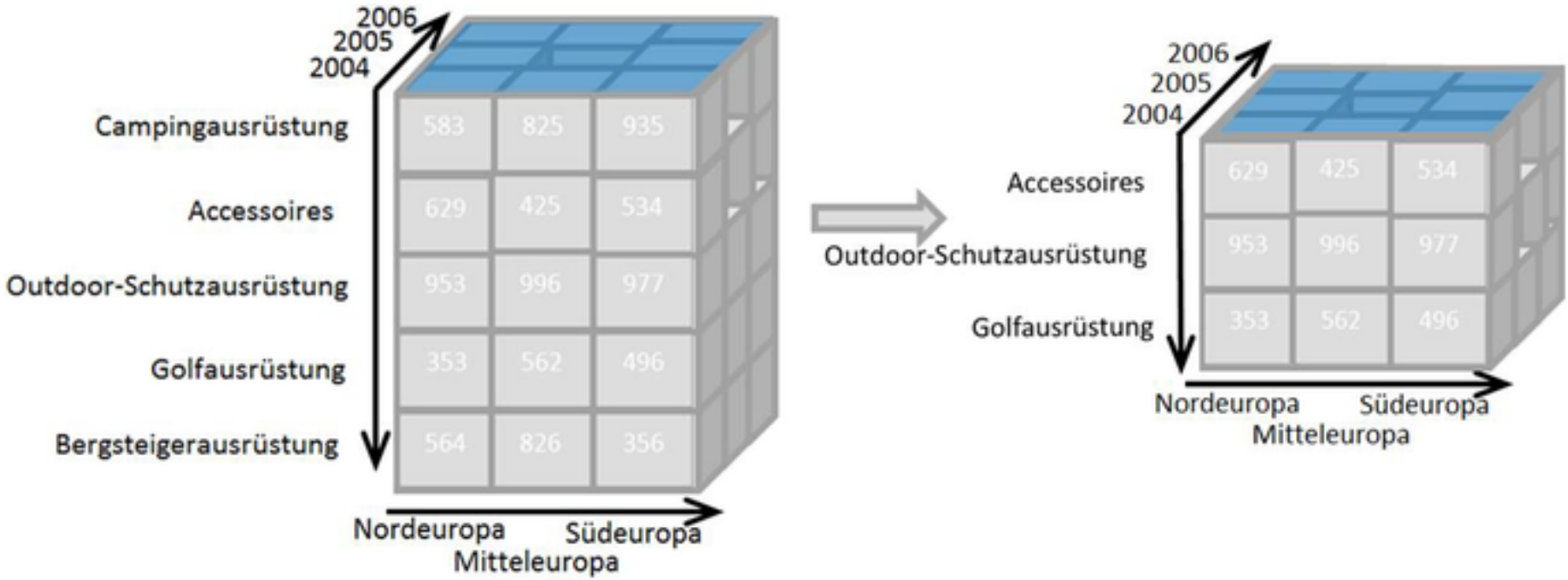
Wikipedia

OLAP Slicing



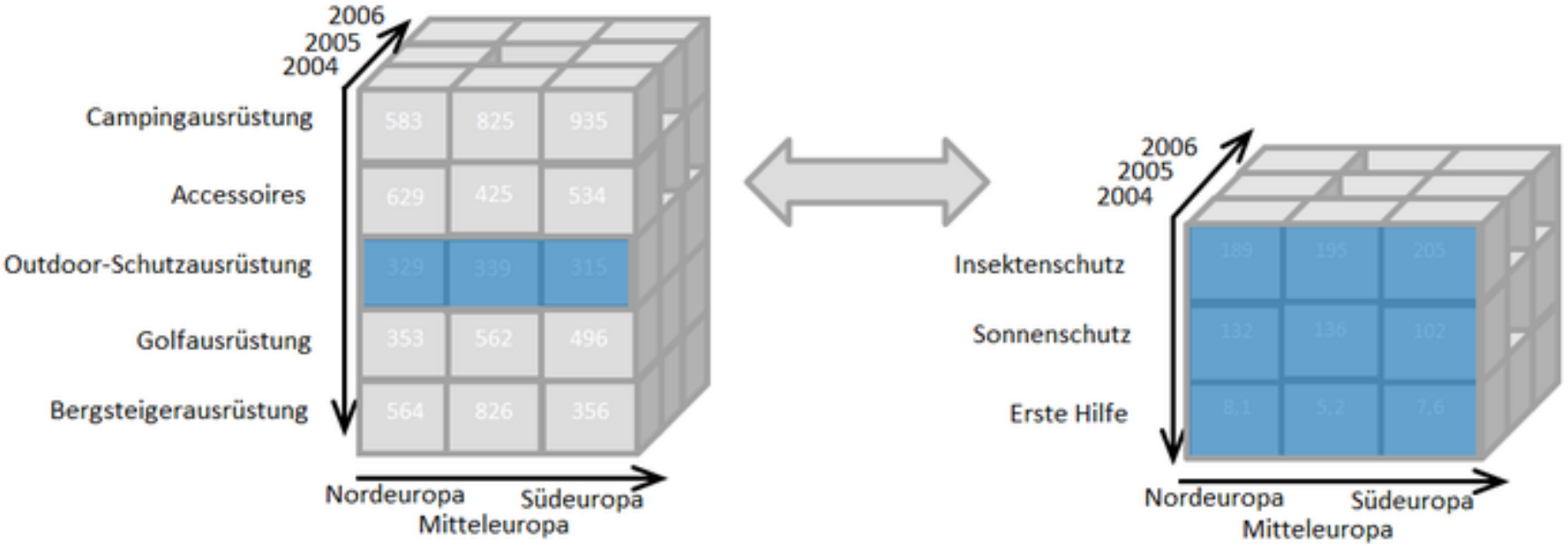
Wikipedia

OLAP Dicing



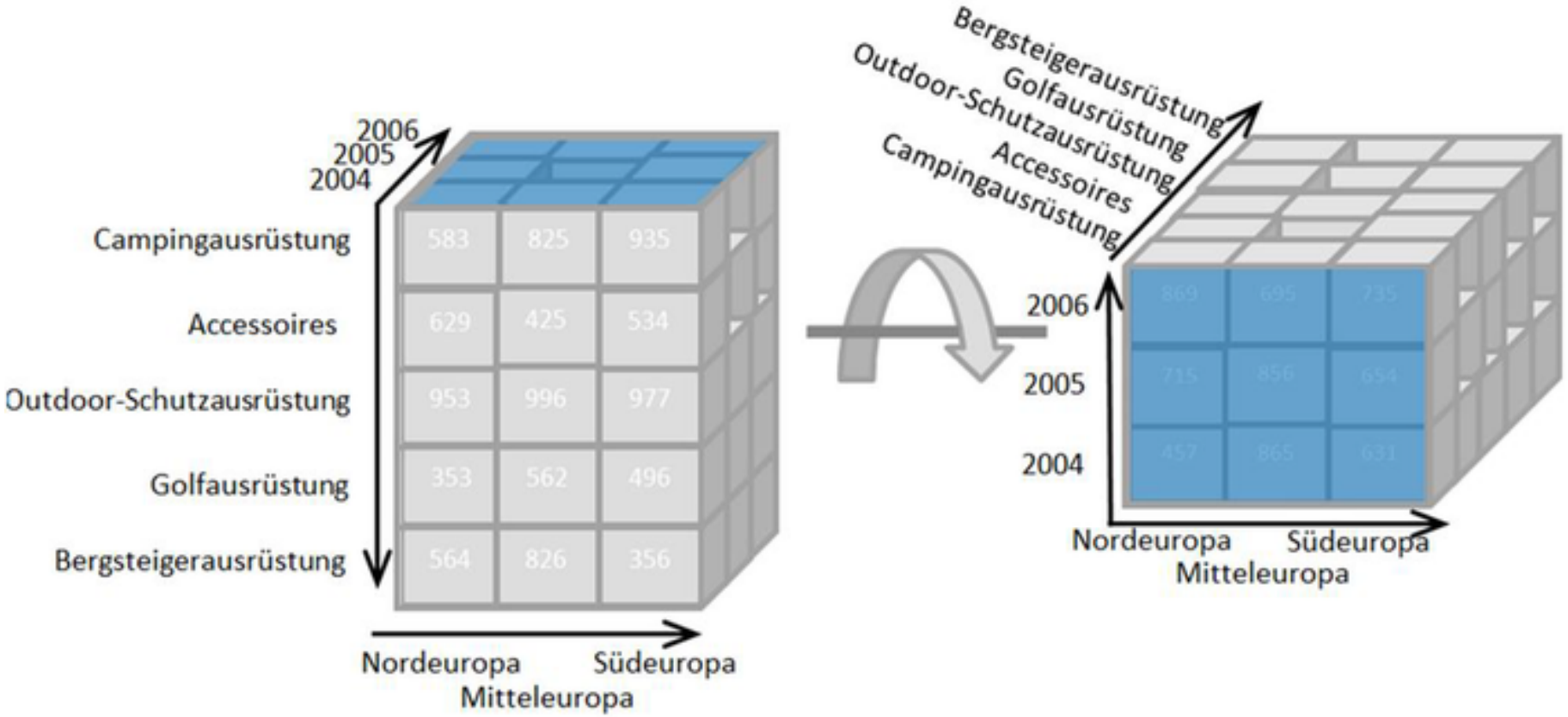
Wikipedia

OLAP Drill-down



Wikipedia

OLAP Pivoting



Wikipedia

Daten und Normalisierung

Auswertung beeinflusst durch verzogene Daten

Parent

ID	Age	Sex	Married	Income
1	40	Male	Yes	400,000
2	59	Female	No	200,000

Child

ID	Parent	Sex	Subsidy
1	1	Male	ES
2	1	Female	NO
3	2	Female	RE

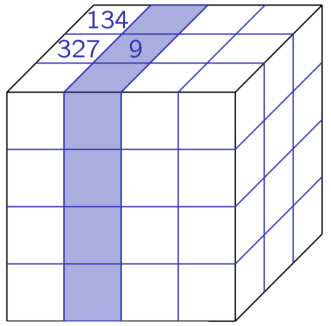
flattening

$$\Pi \text{ Child.ID AS ID, Child.Sex AS Sex, Child.Subsidy AS Subsidy, Parent.Sex AS Psex, Parent.Married AS Pmarried, Parent.Income AS Pincome}$$

$$\text{Child} \bowtie_{\text{Child.Parent} = \text{Parent.ID}} \text{Parent}$$

ChildParent

ID	Sex	Subsidy	PAge	PSex	PMarried	PIncome
1	Male	ES	40	Male	Yes	400,000
2	Female	NO	40	Male	Yes	400,000
3	Female	RE	59	Female	No	200,000



Redundanzen! Bsp. Assoziationsregel {PIncome=400,000} → {PSex=male}?? Support und Konfidenz werden verzerrt!

OLAP hilft hier nicht

Zusammenfassung

- Datenanalyse kann sehr aufwendig sein
- Brute-Force-Ansätze scheitern auf großen Datenmengen
- Genaue Analyse der Problemstellung und Aufteilung in Teilprobleme ermöglichen praxistaugliche Verfahren
- Datenrepräsentation ist entscheidend

Ausblick

- Müssen wir immer alle Daten betrachten um bestimmte Größen zu bestimen (wie z.B. u(c, s))?
- Wenn nein, welche Daten müssen wir betrachten, um bestimmte Ausagen treffen zu können?
- Welche Daten sollten wir wie erfassen?
- Führt uns auf: Statistik