Kapitel DM:V

V. Association Analysis

- Assoziationsanalyse
- Frequent Itemset Mining
- Regel-Mining

Assoziationsanalyse

Motivation/Überblick

Warenkorbanalyse:

Transaktion	Produkte
1	{ Milch, Butter}
2	{ Milch, Kaffee, Kuchen}
3	{ Milch, Kakao, Kuchen}
4	{ Kaffee, Zucker, Tee }
5	{ Milch, Kaffee, Zucker }
6	{ Tee, Zucker }

- Welche Produkte werden zusammen gekauft (Itemsets mit $P(A \cap B) > \sigma$)?
- Welches Produkt bedingt den Kauf von weiteren Produkten ($P(A \mid B) > \gamma$)?

Assoziationsanalyse

Spezifikation des Problems der Assoziationsanalyse

- $I = \{i_1 \dots i_d\}$ sei eine Menge von d binären Merkmalen (Items)
- $X \subseteq \{0, 1\}^d$ sei ein mit I korrespondierender binärer Instanz/Merkmalsraum, $X \subseteq \mathcal{P}(I)$, auch als Transaktionen bezeichnet

Assoziationsanalyse

Spezifikation des Problems der Assoziationsanalyse

- $I = \{i_1 \dots i_d\}$ sei eine Menge von d binären Merkmalen (Items)
- $X \subseteq \{0,1\}^d$ sei ein mit I korrespondierender binärer Instanz/Merkmalsraum, $X \subseteq \mathcal{P}(I)$, auch als Transaktionen bezeichnet

X				I			
Transaktion/Instanz	Milch	Butter	Kaffee	Kakao	Kuchen	Zucker	Tee
x_1	1	1	0	0	0	0	0
x_2	1	0	1	0	1	0	0
x_3	1	0	0	1	1	0	0
x_4	0	0	1	0	0	1	1
x_5	1	0	1	0	0	1	0
x_6	0	0	0	0	0	1	1

Assoziationsanalyse

k-Itemset, Instanzmenge, Support

Definition 1 (k-Itemset)

Ein k-Itemset $I^k \subseteq I$ bezeichnet alle Teilmengen einer Menge an Items I mit Mächtigkeit k, d.h., $\{I^k \mid I^k \subseteq I \wedge \mid I^k \mid = k\}$

Definition 2 (Instanzmenge X_{I^k} eines k-Itemset)

Die Mengen von Instanzen X_{I^k} eines k-Itemset k-Itemset sei definiert als die Menge der Instanzen $x_i \in X$, welche den k-Itemset I^k enthalten, d.h., $\{x_i \in X \mid I^k \subseteq x_i\}$

Definition 3 (Support)

Der Support σ_{I^k} bezeichnet die relative Häufigkeit des Auftretens eines k-Itemset I^k im Merkmalsraum X, d.h., $\sigma_{I^k} = p(I^k) = \frac{X_{I^k}}{|X|}$

Assoziationsanalyse

k-Itemset, Support

X	I						
Transaktion/Instanz	Milch	Butter	Kaffee	Kakao	Kuchen	Zucker	Tee
x_1	1	1	0	0	0	0	0
x_2	1	0	1	0	1	0	0
x_3	1	0	0	1	1	0	0
x_4	0	0	1	0	0	1	1
x_5	1	0	1	0	0	1	0
x_6	0	0	0	0	0	1	1

Beispiele:

- Beispiele für 2-Itemsets:
  $I_1 = \{\text{Milch}, \text{Butter}\}$, $I_2 = \{\text{Butter}, \text{Zucker}\}$, $I_3 = \{\text{Milch}, \text{Kaffee}\}$
- Beispiele für Instanzmengen:
  $X_{I_1} = \{x_1\}$, $X_{I_2} = \{\}$, $X_{I_3} = \{x_2, x_5\}$
- Beispiele für Support:
  $\sigma_{I_1} = \frac{1}{6}$, $\sigma_{I_2} = 0$, $\sigma_{I_1} = \frac{2}{6}$

Kapitel DM:V

V. Association Analysis

- Assoziationsanalyse
- Frequent Itemset Mining
- Regel-Mining

Frequent Itemset Mining

Häufige k-Itemsets

Definition 4 (Menge häufiger k-Itemsets)

Eine aus k-Itemsets bestehende Menge $L_k = \{I_1^k, \ldots, I_n^k\}$ wird als *Menge häufiger k-Itemsets* bezeichnet, wenn der Support aller enthaltenen k-Itemsets einen definiert Minimum Support σ_{min} überschreitet, d.h., $L_k = \{I^k \mid \sigma^{I^k} \geq \sigma_{min}\}$

Frequent Itemset Mining

Häufige k-Itemsets

Definition 4 (Menge häufiger k-Itemsets)

Eine aus k-Itemsets bestehende Menge $L_k = \{I_1^k, \ldots, I_n^k\}$ wird als *Menge häufiger k-Itemsets* bezeichnet, wenn der Support aller enthaltenen k-Itemsets einen definiert Minimum Support σ_{min} überschreitet, d.h., $L_k = \{I^k \mid \sigma^{I^k} \geq \sigma_{min}\}$

Zielsetzung Frequent Itemset Mining:

- Finde die häufigsten k-Itemsets für beliebiges k

Frequent Itemset Mining

Häufige k-Itemsets

Definition 4 (Menge häufiger k-Itemsets)

Eine aus k-Itemsets bestehende Menge $L_k = \{I_1^k, \ldots, I_n^k\}$ wird als *Menge häufiger k-Itemsets* bezeichnet, wenn der Support aller enthaltenen k-Itemsets einen definiert Minimum Support σ_{min} überschreitet, d.h., $L_k = \{I^k \mid \sigma^{I^k} \geq \sigma_{min}\}$

Zielsetzung Frequent Itemset Mining:

- Finde die häufigsten k-Itemsets für beliebiges k

Wie viele häufige k-Itemsets existieren?

- Beispiel: Gegeben L^{100}. Wie viele L^k mit $k < 100$ existieren?
- $\binom{100}{1} + \binom{100}{2} + \ldots + \binom{100}{99} = 2^{100} - 1 = 1.27 \times 10^{30}$

Frequent Itemset Mining

Abgeschlossene häufige Itemsets, Maximal häufige Itemsets

Definition 5 (Abgeschlossene häufige Itemsets (Closed Frequent Itemsets) $\mathcal{C}$)

Ein häufiger k-Itemset $A \subseteq I$ wird genau dann als abgeschlossen bezeichnet, wenn kein Itemset $B \subseteq I$ existiert, so dass B eine Übermenge von A ist, $A \subset B$, mit gleichem Support $\sigma^A = \sigma^B$

$\mathcal{C}$ bezeichnet die Menge aller abgeschlossenen Itemsets, d.h.,
$\mathcal{C} = \{A \subseteq I \mid \sigma^A \geq \sigma_{min} \wedge \nexists_{B \subseteq I}$ mit $A \subset B \wedge \sigma^A = \sigma^B\}$.

Frequent Itemset Mining

Abgeschlossene häufige Itemsets, Maximal häufige Itemsets

Definition 5 (Abgeschlossene häufige Itemsets (Closed Frequent Itemsets) $\mathcal{C}$)

Ein häufiger k-Itemset $A \subseteq I$ wird genau dann als abgeschlossen bezeichnet, wenn kein Itemset $B \subseteq I$ existiert, so dass B eine Übermenge von A ist, $A \subset B$, mit gleichem Support $\sigma^A = \sigma^B$

$\mathcal{C}$ bezeichnet die Menge aller abgeschlossenen Itemsets, d.h.,
$\mathcal{C} = \{A \subseteq I \mid \sigma^A \geq \sigma_{min} \wedge \nexists_{B \subseteq I}$ mit $A \subset B \wedge \sigma^A = \sigma^B\}$.

Definition 6 (Maximal häufige Itemsets (Maximum Frequent Itemsets) $\mathcal{M}$)

Ein häufiger k-Itemset $A \subseteq I$ wird genau dann als „Maximal häufiger Itemset“ bezeichnet, wenn kein Itemset $B \subseteq I$ existiert, so dass B eine Übermenge von A ist und B häufig ist, d.h., $\mathcal{C} = \{A \subseteq I \mid \sigma^A \geq \sigma_{min} \wedge \nexists_{B \subseteq I}$ mit $A \subset B \wedge \sigma^B > \sigma_{min}\}$.

Bemerkungen:

- $\mathcal{C}$ beinhaltet die vollständige Information über alle häufigen Itemsest, d.h. es sind alle häufigen Itemsets aus $\mathcal{C}$ ermittelbar. $\mathcal{C}$ stellt somit eine kompakte Repräsentation aller häufigen Itemsets dar.
- $\mathcal{M}$ beinhaltet unvollständige Information über alle häufigen Itemsets
- Der wesentliche Unterschied in der Definition zwischen $\mathcal{C}$ und $\mathcal{M}$ besteht darin, das die Übermengen B in $\mathcal{M}$ auch einen geringeren Support als A haben können.

Frequent Itemset Mining

Abgeschlossene häufige Itemsets, Maximal häufige Itemsets

Beispiel:

- Zwei Transaktionen: $X = \{< a_1, \ldots, a_{50} >, < a_1, \ldots, a_{100} >\}$
- Wir setzen den Minimum Support auf 1: $\sigma_{min} = 1$

$\Rightarrow$ $\mathcal{C} = \{\{a_1, \ldots, a_{50}\} : \sigma = 2; \{a_1, \ldots, a_{100}\}; \sigma = 1\}$

$\Rightarrow$ $\mathcal{M} = \{\{a_1, \ldots, a_{100}\} : \sigma = 1\}$

- Wie hoch ist der Support von $I_1^2 = \{a_2, a_{45}\}$ und $I_2^2 = \{a_2, a_{55}\}$?
- Vollständig aus $\mathcal{C}$ ermittelbar: $\sigma_{I_1^2} = 2$, $\sigma_{I_2^2} = 1$
- Nicht ermittelbar aus $\mathcal{M}$: $\sigma_{I_2^2} = 1$

$\Rightarrow$ $\mathcal{C}$ beschreibt alle 2^{100} häufigen Itemsets vollständig

Frequent Itemset Mining

A-Priori Algorithmus

Problem:

- Brute-Force Berechnung aller häufigen k-Itemsets für große k nicht realisierbar, da 2^k mögliche Mengen.

Beobachtung [Agrawal/Srikant 1994] :

- $k+1$ häufige Itemsets können auf Basis der k häufigen Itemsets ermittelt werden (a-priori Wissen $\rightarrow$ „A-priori Algorithmus“).

Lösung:

- Beginne mit $k=1$ und ermittle iterative L^1, L^2, L^3 und beschneide in jedem Schritt die Menge der nicht gültigen abgeschlossenen häufigen Itemsets.

Frequent Itemset Mining

A-Priori Algorithmus

Satz 7 (A-Priori Eigenschaft eines häufigen Itemset)

Eine nichtleere Untermenge $A \subset I^k$ eines häufigen k-Itemsets I^k ist ebenfalls ein häufiger Itemset.

Beweis

Die Instanzmenge $X_{I^k} = \{x_i \in X \mid I^k \subseteq x_i\}$ eines Itemsets I^k ist, laut Definition, auch eine Instanzmenge jeder nichtleeren Untermenge A von I^k, d.h., $X_{A \subset I^k} = \{x_i \in X \mid A \subset I^k \subseteq x_i\}$. Daher hat A den gleichen Support und somit die gleiche Eigenschaft hinsichtlich der Häufigkeit von I^k.

Bemerkungen:

- Der Umkehrschluss erlaubt es, den Suchraum entsprechend zu beschränken. Mengen, deren Teilmengen keine häufigen Itemsets sind, sind keine häufigen Itemsets.
- Eine Eigenschaft, bei der eine Menge ein Kriterium nicht erfüllt, wenn eine beliebige Untermenge dieses Kriterium ebenfalls nicht erfüllt, nennt man antimonoton.

Frequent Itemset Mining

A-Priori Algorithmus

Input: $X = \{x_1, \ldots, x_n\}$. Mengen von Instanzen
σ_{min}. Minimum Support

Output: $\mathcal{L}$. Die Menge der häufigen Itemsets

1. L^1 = `find-frequent-1-itemsets(X)` `// Initialisierung`
2. **FOR** $(k = 2; L^{k-1} \neq \emptyset; k++)$
3. $C_k = \{c \mid c \in \{a \in L^{k-1} \cup \{b\} \in I\} \wedge b \notin a \wedge I = \bigcup L^{k-1}\}$
   `// Konkateniere alle Merkmale mit jedem gültigen k-1 Itemset`
4. $C_k = C_k / \{c \in C_k \mid \exists_{I^{k-1}}\ mit\ I^{k-1} \subset c \wedge I^{k-1} \notin L^{k-1}\}$
   `// Entferne a-Priori Eigenschaften verletzende k-Itemsets`
5. **FOR EACH** $(x_i \in X)$ `// Scan über Instanzen`
6. $C_t = \{c \mid c \in C_k \wedge c \subset x_i\}$ `// Finde alle Kandidaten k-Itemsets für` x_i
7. $\forall_{c_j \in C_t} : \sigma^{c_j} = \sigma^{c_j} + \frac{1}{|X|}$ `// Erhöhe Support`
8. $L^k = \{c \in C_k \mid \sigma^c \geq \sigma_{min}\}$
9. **RETURN**$(M = \bigcup_k L^k)$

Bemerkungen:

- Zentraler Schritt ist die Genierung der k-Itemsets aus den (k-1)-Itemsets in Zeile 4.
- Zeile 5 ist ein Optimierungsschritt, der so nicht unbedingt notwendig ist. Zeile 8 würde in Zeile 5 nicht entfernte k-Itemsets entsprechend eliminieren.

Frequent Itemset Mining

Beispiel: A-Priori Algorithmus mit $\sigma_{min} = \frac{2}{9}$

Transaktion	Produkte
1	{ I1, I2, I5 }
2	{ I2, I4 }
3	{ I2, I3 }
4	{ I1, I2, I4 }
5	{ I1, I3 }
6	{ I2, I3 }
7	{ I1, I3 }
8	{ I1, I2, I3, I5 }
9	{ I1, I2, I3 }

Frequent Itemset Mining

Beispiel: A-Priori Algorithmus mit $\sigma_{min} = \frac{2}{9}$

Transaktion	Produkte
1	{ I1, I2, I5 }
2	{ I2, I4 }
3	{ I2, I3 }
4	{ I1, I2, I4 }
5	{ I1, I3 }
6	{ I2, I3 }
7	{ I1, I3 }
8	{ I1, I2, I3, I5 }
9	{ I1, I2, I3 }

Generated C_1	σ	L_1	Generated C_2	σ	L_2	Generated C_3	σ	L_3
{ I1 }	$6/9$	{ I1 }	{ I1, I2 }	$4/9$	{ I1, I2 }	{ I1, I2, I3 }	$2/9$	{ I1, I2, I3 }
{ I2 }	$7/9$	{ I2 }	{ I1, I3 }	$4/9$	{ I1, I3 }	{ I1, I2, I5 }	$2/9$	{ I1, I2, I5 }
{ I3 }	$6/9$	{ I3 }	{ I1, I4 }	$1/9$	{ }			
{ I4 }	$2/9$	{ I4 }	{ I1, I5 }	$2/9$	{ I1, I5 }			
{ I5 }	$2/9$	{ I5 }	{ I2, I3 }	$4/9$	{ I2, I3 }			
			{ I2, I4 }	$2/9$	{ I2, I4 }			
			{ I2, I5 }	$2/9$	{ I2, I5 }			
			{ I3, I4 }	$0/9$	{ }			
			{ I3, I5 }	$1/9$	{ }			
			{ I4, I5 }	$0/9$	{ }			

Frequent Itemset Mining

Effizienzverbesserungen A-Priori Algorithmus

Es existieren verschiedene Varianten zur Beschleunigung des A-Priori Algorithmus:

- Verwendung von Hash-Tabllen zur Reduzierung der Größe von C_k für $k > 1$.
- Eliminierung jener Instanzen $x_i \in X$ für Schritt $k+1$, welche keine häufigen $k-$Itemsets haben.
- Sampling: Ziehe zufällig eine Menge $S \subset X$ und ermittle $\mathcal{M}$ in Bezug auf S mit einem geringeren minimum Support σ_{min} (Tausch Laufzeit gegen Genauigkeit).
- Dynamisches zählen der Itemsets: Erhöhen den Support während dem Scannen der Beispiele X. Sobald der Support eines Itemsets den minimum Support überschreitet, verwende den Itemset in L_k und eliminiere ihn in C_k.

Alternativen zum A-Priori Algorithmus:

- Frequent Pattern Growth Algorithmus (Tiefensuche)
- Eclat (Tiefensuche)