

POLITECHNIKA WROCŁAWSKA  
WYDZIAŁ INFORMATYKI I TELEKOMUNIKACJI

# METODY ANALIZY I EKSPLORACJI DANYCH

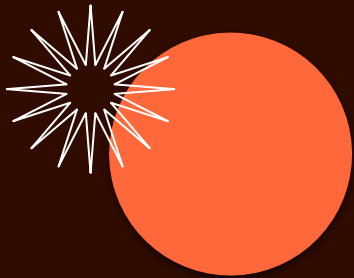
Wykład 10 - Odkrywanie wzorców: reguły  
asocjacyjne.

DR INŻ. AGATA MIGALSKA

---



Wykład  
10

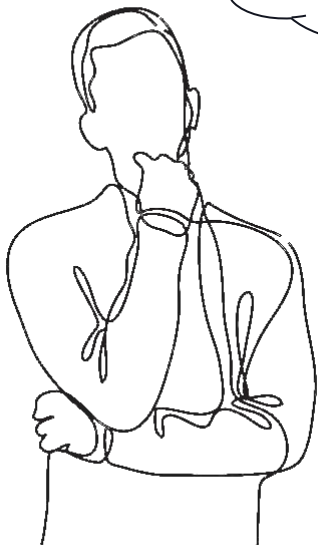


**CEL I MOTYWACJA**

---

# ODKRYWANIE WZORCÓW (PATTERN DISCOVERY)

Które artykuły są często  
kupowane razem przez  
klientów?



ANALITYK RYNKU



KLIENT 1



KLIENT 2



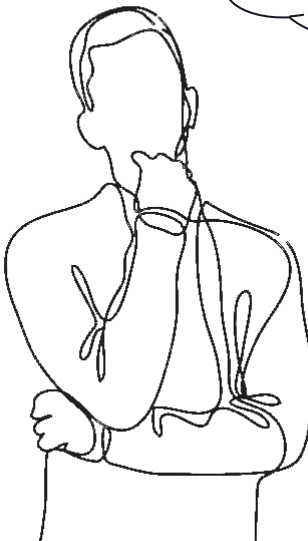
KLIENT N



KLIENT 3

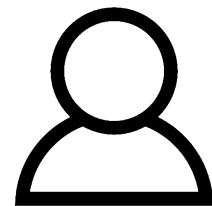


# ODKRYWANIE WZORCÓW (PATTERN DISCOVERY)




Jaka sekwencja leków  
stosowana jest w  
leczeniu tej choroby?

<(lek\_3),  
(lek\_9)>



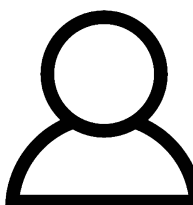
PACJENT 1

<(lek\_3),  
(lek\_4, lek\_7),  
(lek\_9)>




PACJENT 2

<(lek\_9)>



PACJENT N

<(lek\_3, lek\_5, lek\_7)>



PACJENT 3

ANALITYK



# CZYM JEST ODKRYWANIE WZORCÓW?

Biorąc pod uwagę ogromne dane dotyczące transakcji zakupowych, wykrywanie wzorców może pomóc odpowiedzieć na następujące pytania:

- Jakie grupy artykułów są często kupowane razem?
- Jeśli ktoś kupuje pieluchy w nocy, jakie jest prawdopodobieństwo, że kupi też piwo?
- Jeśli klient kupi iPhone'a 5 lub iPhone'a 7, jakie inne produkty elektroniczne klient najprawdopodobniej kupi w ciągu najbliższych 3 miesięcy?

**Wzorzec:** zestaw elementów, podsekwencji lub podstruktur, które często występują razem (lub są silnie skorelowane) w zbiorze danych.

---



# ZASTOSOWANIA

- Przewidywanie danych transakcji zakupowych
  - W przypadku klienta, który kupuje produkty A i B, jakie jest prawdopodobieństwo, że klient kupi produkt C?
  - Przewidywanie strumieni kliknięć na stronie internetowej:
  - Która strona internetowa zostanie kliknięta jako następna?
  - Błędy oprogramowania do wydobywania: gdzie jest prawdopodobny błąd w tym programie błąd w tym programie?
  - Identyfikacja obiektów lub podstruktur w obrazach, filmach i mediach społecznościowych
  - Znajdowanie wysokiej jakości fraz, jednostek i atrybutów w obszernym tekście
  - Znalezienie powtarzających się sekwencji DNA i białek w genomach
  - Znajdowanie „ukrytych” społeczności w ogromnej sieci społecznościowej
-

---

# ODKRYWANIE WZORCÓW

jest zadaniem

a) uczenia nadzorowanego

czy

b) uczenia nienadzorowanego?

---

---

**01**  
ODKRYWANIE  
ASOCJACJI

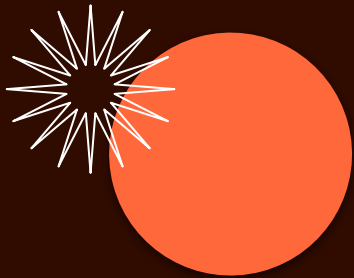
**02**  
ALGORYTMY  
ODKRYWANIA  
ASOCJACJI  
BINARNYCH

**03**  
JAK  
INTERESUJĄCY  
JEST TEN  
WZORZEC?

**04**  
BARDZIEJ  
ZAAWANSOWANE  
TEMATY

---





01

# ODKRYWANIE ASOCJACJI

---

# ODKRYWANIE ASOCJACJI





# ZBIORY CZĘSTE (FREQUENT ITEMSETS)

- **Zbiór (itemset):** zbiór jednego lub więcej elementów
- **k-zbiór (k-itemset):**  $X = \{x_1, \dots, x_k\}$
- **(bezwzględne) wsparcie X:** Częstość lub liczba wystąpień zestawu przedmiotów X
- **(względne) wsparcie, s:** Ułamek transakcji zawierających X (tj. prawdopodobieństwo, że transakcja zawiera X)
- Zbiór X jest **częsty**, jeśli wsparcie X jest nie mniejsze niż próg minsup

T_id	Towary
1	piwo, orzeszki, pieluszki
2	piwo, kawa, pieluszki
3	piwo, pieluszki, jajka
4	orzeszki, jajka, mleko
5	orzeszki, kawa, pieluszki, jajka, mleko

Niech minsup = 50%

**Częste zestawy 1-elementowe:**

Piwo: 3 (60%), Orzeszki: 3 (60%), Pieluszki: 4 (80%), Jajka: 3 (60%)

**Częste zestawy 2-elementowe:**

{Piwo, pieluszki}: 3 (60%)

---

# OD ZBIORÓW CZĘSTYCH DO REGUŁ ASOCJACYJNYCH

- Reguły asocjacyjne:  $X \rightarrow Y(s, c)$
- Wsparcie,  $s$ : Prawdopodobieństwo, że transakcja zawiera  $X \cup Y$ 
  - Uwaga: Notacja!  $X \cup Y$  oznacza, że transakcja zawiera oba elementy,  $X$  i  $Y$
- Ufność,  $c$ : Prawdopodobieństwo warunkowe, że transakcja zawierająca  $X$  zawiera również  $Y$ .

$$c = \frac{s(X \cup Y)}{s(X)}$$

- **Odkrywanie reguł asocjacyjnych:** znajdowanie wszystkich **silnych** reguł asocjacyjnych, tzn. spełniających warunek minimalnego wsparcia i ufności

Niech minsup = 50%

**Częste zestawy 1-elementowe:**

Piwo: 3 (60%), Orzeszki: 3 (60%), Pieluszki: 4 (80%), Jajka: 3 (60%)

**Częste zestawy 2-elementowe:**

{Piwo, pieluszki}: 3 (60%)



Reguły asocjacyjne: Niech minconf=50%

Piwo->Pieluchy (60%, 100%)

Pielucha->Piwo (60%, 75%)



# ODKRYWANIE ZBIORÓW CZĘSTYCH

- Algorytm naiwny
  - Wygeneruj dla danego zbioru elementów  $L$  i bazy danych  $D$  wszystkie możliwe binarne reguły asocjacyjne
  - Oblicz dla każdej reguły wsparcie i ufność
  - Odrzuć te reguły, które nie spełniają warunków minimalnego wsparcia i minimalnej ufności.
- Liczba wszystkich możliwych podzbiorów zbioru elementów  $L$  wynosi  $2^{|L|} - 1$





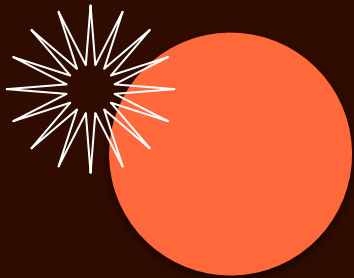
# WŁASNOŚĆ ANTYMONOTONICZNOŚCI

- Jeżeli {piwo, pieluszki, orzeszki} jest zbiorem częstym, to {piwo, pieluszki} też jest zbiorem częstym.
- Każda transakcja zawierająca {piwo, pieluszki, orzeszki} zawiera również {piwo, pieluszki}.
- **Obserwacja:** Każdy podzbiór zbioru częstego musi być zbiorem częstym.
- Wydajna metodologia eksploracji:
  - jeżeli jakkolwiek podzbiór zbioru  $S$  jest nieczęsty, odetnij gałąź zawierającą  $S$ .

## Własność antymonotoniczności

Niech będzie dany zbiór elementów  $L$  oraz jego zbiór potęgowy  $J = 2^L$ . Mówimy, że miara jest antymonotoniczna na zbiorze  $J$ , jeżeli  $\forall X, Y \in J : (X \subseteq Y) \rightarrow f(Y) \leq f(X)$ .

---



02

# ALGORYTMY ODKRYWANIA ASOCJACJI BINARNYCH

---



# ALGORYTM APRIORI

1. Niech  $k=1$
  2. Powtarzaj
    - a. Stwórz zbiór kandydatów  $C_k$  składający się z  $k$ -elementowych zbiorów częstych
    - b. Oblicz wsparcie dla każdego zbioru w  $C_k$
    - c. Filtrowanie zbiorów kandydujących:  $L_k$  - zbiór tych zbiorów  $c$  z  $C_k$ , dla których  $s(c) \geqslant \text{minsup}$
    - d.  $k := k + 1$
    - e. Jeżeli  $L_{k-1} = \emptyset$ , to przerwij pętlę.
  3. Zwróć  $\bigcup_k L_k$
-



# APRIORI - PRZYKŁAD

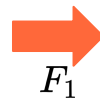
Transakcje

T_id	Towary
1	A, C, D
2	B, C, E
3	A, B, C, E
4	B, E

minsup=2



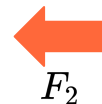
Zbiór k-elem	Wsparcie
{A}	2
{B}	3
{C}	3
{D}	1
{E}	3



Zbiór k-elem	Wsparcie
{A}	2
{B}	3
{C}	3
{E}	3



Zbiór k-elem	Wsparcie
{A,B}	1
{A,C}	2
{A,E}	1
{B,C}	2
{B,E}	3
{C,E}	2



Zbiór k-elem	Wsparcie
{A,C}	2
{B,C}	2
{B,E}	3
{C,E}	2



Zbiór k-elem	Wsparcie
{B,C,E}	2



$F_3 = C_3$

# APRIORI - PRZYKŁAD

Transakcje

T_id	Towary
1	A, C, D
2	B, C, E
3	A, B, C, E
4	B, E

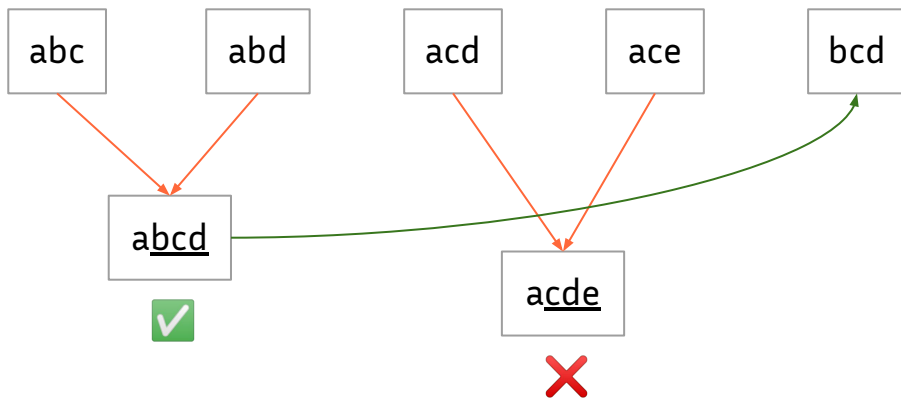
minsup=2



Zbiór k-elem	Wsparcie
{A}	2
{B}	3
{C}	3
{E}	3
{A,C}	2
{B,C}	2
{B,E}	3
{C,E}	2
{B,C,E}	2

# APRIORI: TWORZENIE KANDYDATÓW

1. Tworzenie kandydatów  $C_k$ 
  - a. Łączenie - iloczyn kartezjański zbiorów  $L_{k-1}$
  - b. Odcięcie - usunięcie tych zbiorów, które mają choć jeden podzbiór nienależący do  $L_{k-1}$





# ECLAT: ALGORYTM EKSPLOKACJI PIONOWEJ

- ECLAT (Equivalence Class Transformation) - algorytm oparty na przeszukiwaniu w głąb oraz przecięciu zbiorów transakcji
- Algorytm rekurencyjnie generuje k-elementowe zbiory kandydujące, łącząc (k-1)-elementowe zbiory częste (tak jak Apriori).
- Do obliczenia wartości wsparcia k-elementowego zbioru kandydującego  $X$ , wygenerowanego przez połączenie dwóch zbiorów (k-1)-elementowych  $X_1$  i  $X_2$ , wykorzystuje listy identyfikatorów transakcji zbiorów  $X_1$  i  $X_2$  (inaczej niż Apriori).
- Własności listy transakcji:
  - $t(X) = t(Y)$ :  $X$  i  $Y$  zawsze występują razem, np.  $t(B) = t(E)$
  - $t(X) \subset t(Y)$ : transakcje zawierające  $X$  zawsze zawierają  $Y$ , np.  $t(A) \subset t(C)$
- Algorytm zakłada sortowanie leksykograficzne elementów w zbiorach częstych.

Transakcje w orientacji poziomej

T_id	Towary
1	A, C, D
2	B, C, E
3	A, B, C, E
4	B, E

Transakcje w orientacji pionowej

Towar	Transakcje
A	1, 3
B	2, 3, 4
C	1, 2, 3
D	1
E	2, 3, 4

# ECLAT - PRZYKŁAD

Transakcje

Towar	Transakcje
A	1, 3
B	2, 3, 4
C	1, 2, 3
D	1
E	2, 3, 4

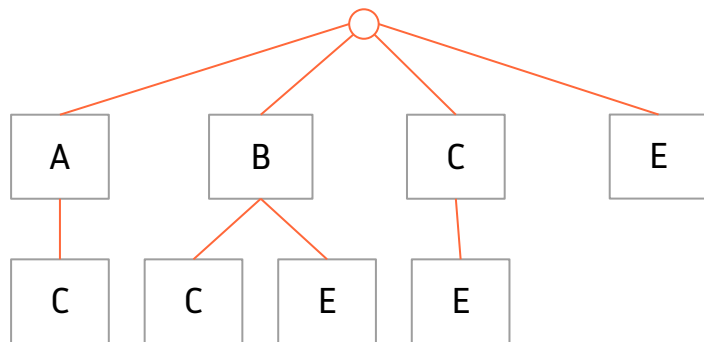
minsup=2



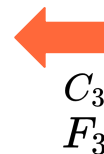
Zbiór k-elem	Transakcje	Wsparcie
{A}	1, 3	2
{B}	2, 3, 4	3
{C}	1, 2, 3	3
{D}	1	1
{E}	2, 3, 4	3



Zbiór k-elem	Transakcje	Wsparcie
{A, B}	3	1
{A, C}	1, 3	2
{A, E}	3	1
{B, C}	2, 3	2
{B, E}	2, 3, 4	3
{C, E}	2, 3	2



Zbiór k-elem	Transakcje	Wsparcie
{B, C, E}	2, 3	2





# FP-GROWTH: DRZEWO WZORCÓW

- FP-Growth: Frequent Pattern Growth
    - Znajduje 1-elementowe zbiory częste i dzieli bazę danych wg tych zbiorów na rozdzielne partycje.
    - Rekurencyjnie rozszerza zbiory częste poprzez znajdowanie ich pod-partycji.
    - Wykorzystując wydajną strukturę FP-drzewa.
  - Algorytm:
    - Rekurencyjnie tworzy i eksploruje (warunkowe (tj. wewnątrz partycji)) FP-drzewa dopóki przetwarzane FP-drzewo nie jest puste lub dopóki zawiera więcej niż jedną krawędź
  - Algorytm zakłada sortowanie leksykograficzne elementów w zbiorach częstych.
  - Algorytm eksploracji poziomej.
  - Algorytm utrzymuje pomocniczą tablicę nagłówkową, która dla każdego elementu posiada listę wskaźników do tego elementu w FP-drzewie.
-



# FP-GROWTH: PRZYKŁAD

T_id	Towary	1-elementowe zbiory częste po usunięciu zbiorów nieczęstych, posortowane wg malejącego wsparcia
1	{f, a, c, d, g, i, m, p}	{f, c, a, m, p}
2	{a, b, c, f, l, m, o}	{f, c, a, b, m}
3	{b, f, h, j, o, w}	{f, b}
4	{b, c, k, s, p}	{c, b, p}
5	{a, f, c, e, l, p, m, n}	{f, c, a, m, p}

1. Przeskanuj bazę danych raz w celu znalezienia 1-elementowych zbiorów częstych.  
f:4, a:3, c:4, b:3, m:3, p:3
2. Posortuj zbiory częste wg malejącego wsparcia  
F-list = f-c-a-b-m-p
3. Przeskanuj skompresowaną bazę danych i skonstruuj FP-drzewo.

Niech minsup=3.

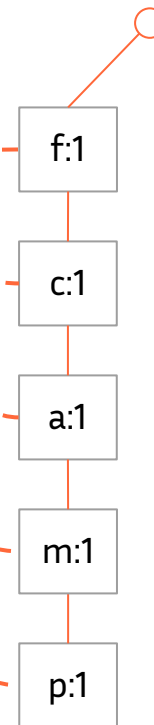
---

# FP-GROWTH: PRZYKŁAD

T_id	1-elementowe zbiory częste po usunięciu zbiorów nieczęstych, posortowane wg malejącego wsparcia
1	{f, c, a, m, p}
2	{f, c, a, b, m}
3	{f, b}
4	{c, b, p}
5	{f, c, a, m, p}

Niech minsup=3.

Element	Wsparcie	Wskaźnik
f	4	
c	4	
a	3	
b	3	
m	3	
p	3	



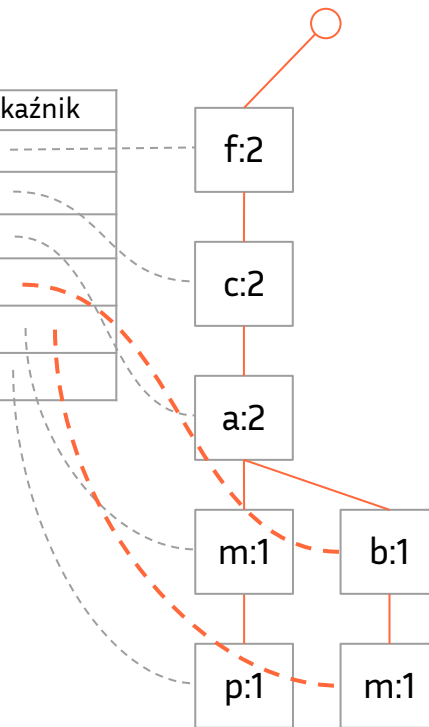


# FP-GROWTH: PRZYKŁAD

T_id	1-elementowe zbiory częste po usunięciu zbiorów nieczęstych, posortowane wg malejącego wsparcia
1	{f, c, a, m, p}
2	{f, c, a, b, m}
3	{f, b}
4	{c, b, p}
5	{f, c, a, m, p}

Niech minsup=3.

Element	Wsparcie	Wskaźnik
f	4	
c	4	
a	3	
b	3	
m	3	
p	3	



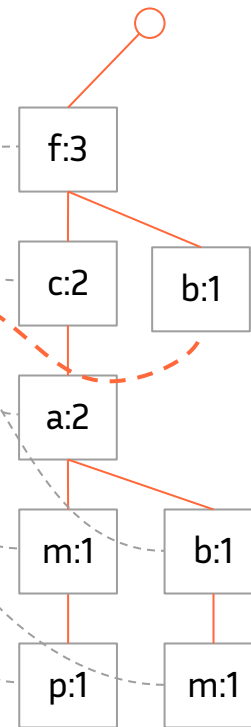


# FP-GROWTH: PRZYKŁAD

T_id	1-elementowe zbiory częste po usunięciu zbiorów nieczęstych, posortowane wg malejącego wsparcia
1	{f, c, a, m, p}
2	{f, c, a, b, m}
3	{f, b}
4	{c, b, p}
5	{f, c, a, m, p}

Niech minsup=3.

Element	Wsparcie	Wskaźnik
f	4	
c	4	
a	3	
b	3	
m	3	
p	3	

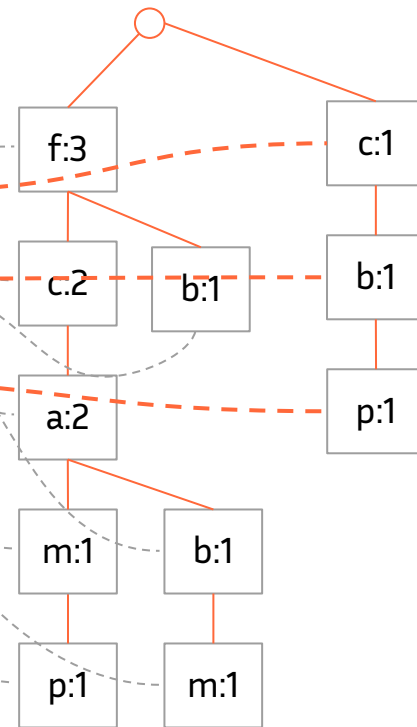


# FP-GROWTH: PRZYKŁAD

T_id	1-elementowe zbiory częste po usunięciu zbiorów nieczęstych, posortowane wg malejącego wsparcia
1	{f, c, a, m, p}
2	{f, c, a, b, m}
3	{f, b}
4	{c, b, p}
5	{f, c, a, m, p}

Niech minsup=3.

Element	Wsparcie	Wskaźnik
f	4	
c	4	
a	3	
b	3	
m	3	
p	3	

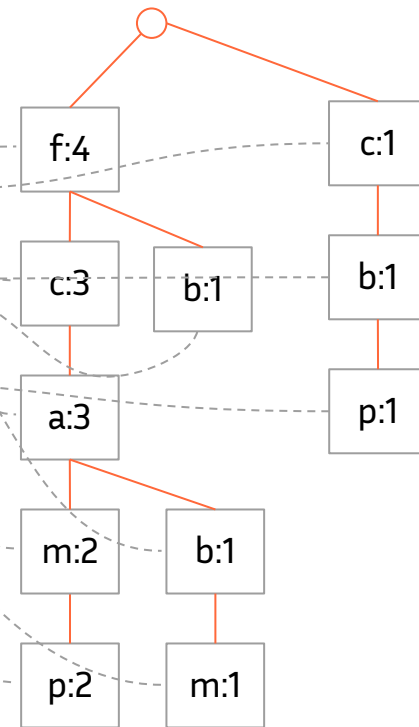


# FP-GROWTH: PRZYKŁAD

T_id	1-elementowe zbiory częste po usunięciu zbiorów nieczęstych, posortowane wg malejącego wsparcia
1	{f, c, a, m, p}
2	{f, c, a, b, m}
3	{f, b}
4	{c, b, p}
5	{f, c, a, m, p}

Niech minsup=3.

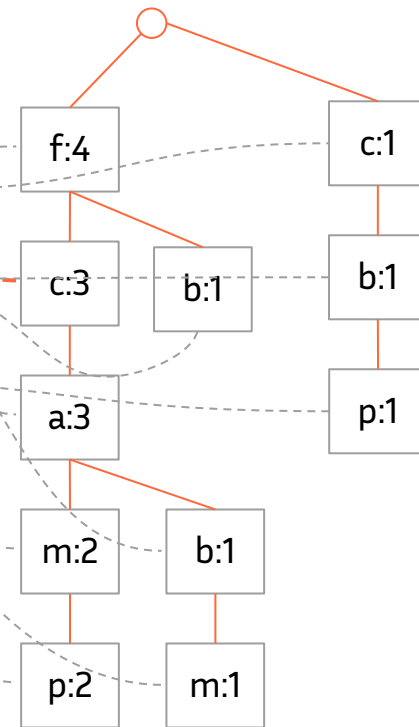
Element	Wsparcie	Wskaźnik
f	4	
c	4	
a	3	
b	3	
m	3	
p	3	



# FP-GROWTH: WARUNKOWA BAZA WZORCA

Wzorzec	Warunkowa baza wzorca
c	f: 3
a	
b	
m	
p	

Element	Wsparcie	Wskaźnik
f	4	
c	4	
a	3	
b	3	
m	3	
p	3	

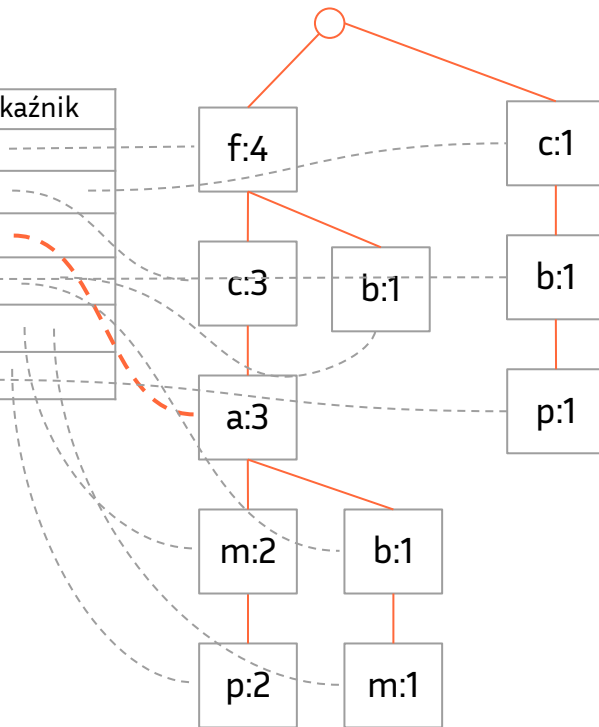


Ścieżka prefiksowa wzorca  $\alpha$  - pojedyncza ścieżka, której końcowym wierzchołkiem jest  $\alpha$ .

# FP-GROWTH: WARUNKOWA BAZA WZORCA

Wzorzec	Warunkowa baza wzorca
c	f: 3
a	fc: 3
b	
m	
p	

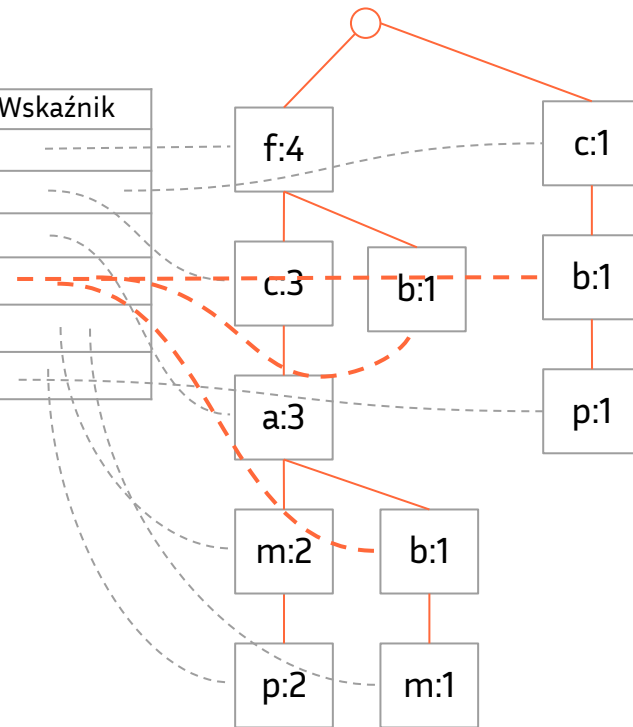
Element	Wsparcie	Wskaźnik
f	4	
c	4	
a	3	
b	3	
m	3	
p	3	



# FP-GROWTH: WARUNKOWA BAZA WZORCA

Wzorzec	Warunkowa baza wzorca
c	f: 3
a	fc: 3
b	fca: 1, f: 1, c: 1
m	
p	

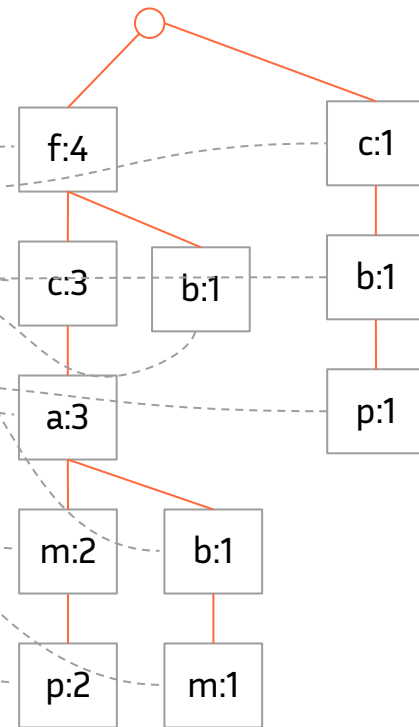
Element	Wsparcie	Wskaźnik
f	4	
c	4	
a	3	
b	3	
m	3	
p	3	



# FP-GROWTH: WARUNKOWA BAZA WZORCA

Wzorzec	Warunkowa baza wzorca
c	f: 3
a	fc: 3
b	fca: 1, f: 1, c: 1
m	fca: 2, fcab: 1
p	fcam: 2, cb: 1

Element	Wsparcie	Wskaźnik
f	4	
c	4	
a	3	
b	3	
m	3	
p	3	

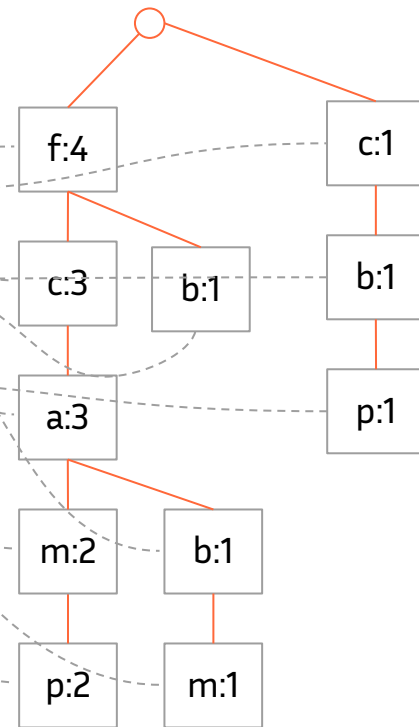




# FP-GROWTH: WARUNKOWA BAZA WZORCA

Wzorzec	Warunkowa baza wzorca
c	f: 3
a	fc: 3
b	fca: 1, f: 1, c: 1
m	fca: 2, fcab: 1
p	fcam: 2, cb: 1

Element	Wsparcie	Wskaźnik
f	4	
c	4	
a	3	
b	3	
m	3	
p	3	





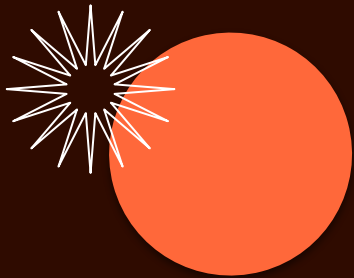
# PODSUMOWANIE

- Zbiór X jest **częsty**, jeśli wsparcie X jest nie mniejsze niż próg *minsup*
  - **Odkrywanie reguł asocjacyjnych** - znajdowanie wszystkich **silnych** reguł asocjacyjnych, tzn. spełniających warunek minimalnego wsparcia *minsup* i ufności *minconf*
  - Obserwacja: Każdy podzbiór zbioru częstego musi być zbiorem częstym.
  - Skalowalne metody odkrywania asocjacji w danych:
    - Eksploracja pozioma:
      - Apriori
      - FP-Growth
    - Eksploracja pionowa:
      - Eclat
-



# IMPLEMENTACJE

- FPGrowth w Apache Spark  
<https://spark.apache.org/docs/latest/ml-frequent-pattern-mining.html>
  - Apriori <https://pypi.org/project/apriori-python/>
  - FP-Growth <https://github.com/enaeseth/python-fp-growth>
  - FP-Growth <https://pypi.org/project/pyfpgrowth/>
  - Apriori, ECLAT, FP-Growth <https://github.com/udayRage/PAMI>
-



03

**JAK INTERESUJĄCY  
JEST TEN WZORZEC?**

---

# PRZYPOMNIENIE: REGUŁY ASOCJACYJNE

- Reguły asocjacyjne:  $X \rightarrow Y(s, c)$
- Wsparcie,  $s$ : Prawdopodobieństwo, że transakcja zawiera  $X \cup Y$ 
  - Uwaga: Notacja!  $X \cup Y$  oznacza, że transakcja zawiera oba elementy,  $X$  i  $Y$
- Ufność,  $c$ : Prawdopodobieństwo warunkowe, że transakcja zawierająca  $X$  zawiera również  $Y$ .

$$c = \frac{s(X \cup Y)}{s(X)}$$

- **Odkrywanie reguł asocjacyjnych:** znajdowanie wszystkich **silnych** reguł asocjacyjnych, tzn. spełniających warunek minimalnego wsparcia i ufności

Niech minsup = 50%

**Częste zestawy 1-elementowe:**

Piwo: 3 (60%), Orzeszki: 3 (60%), Pieluszki: 4 (80%), Jajka: 3 (60%)

**Częste zestawy 2-elementowe:**

{Piwo, pieluszki}: 3 (60%)



Reguły asocjacyjne: Niech minconf=50%

Piwo->Pieluchy (60%, 100%)

Pielucha->Piwo (60%, 75%)



# MIARY ATRAKCYJNOŚCI REGUŁ ASOCJACYJNYCH

- Popularne miary:
  - Wsparcie - ocena ogólności reguły asocjacyjnej
  - Ufność - ocena wiarygodności reguły asocjacyjnej
- Miary wsparcia i ufności są niewystarczające do oceny atrakcyjności reguły asocjacyjnej, gdyż:
  - nie uwzględniają korelacji pomiędzy poprzednikiem i następnikiem reguły asocjacyjnej,
  - eliminują możliwość znalezienia interesujących reguł asocjacyjnych o niewielkim wsparciu.

	kawa	nie_kawa	suma
herbata	20	5	25
nie_herbata	70	5	75
suma	90	10	100

$$herbata \rightarrow kawa \left[ s = \frac{20}{100} = 0.2, c = \frac{20}{25} = 0.8 \right]$$

$$nie\_herbata \rightarrow kawa \left[ s = \frac{70}{100} = 0.7, c = \frac{70}{75} = 0.93 \right]$$

---



# LIFT

- Lift - miara zależności / korelacji pomiędzy zdarzeniami

$$lift(A, B) = \frac{P(B|A)}{s(B)} = \frac{c(A \rightarrow B)}{s(B)} = \frac{s(A \cup B)}{s(A)s(B)}$$

- $lift = 1 \Leftrightarrow$  zdarzenia A i B są niezależne
  - $lift < 1 \Leftrightarrow$  zdarzenia A i B są skorelowane negatywnie
  - $lift > 1 \Leftrightarrow$  zdarzenia A i B są skorelowane pozytywnie
-



# INNE MIARY ATRAKCYJNOŚCI

- Nie sprawdza się jeżeli dla danej reguły istnieje wiele transakcji, które nie zawierają elementów reguły
  - np. wiele transakcji nie zawierających ani kawy ani herbaty
- W odpowiedzi, w literaturze zaproponowano wiele innych miar, w tym takich, które są odporne na obecność transakcji nie zawierających m.in.:
  - miara cosinusów

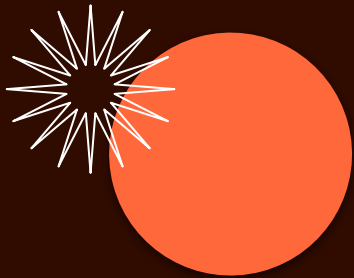
$$\text{Cosine}(A, B) = \frac{s(A \cup B)}{\sqrt{s(A)s(B)}}$$

- miara Jaccarda

$$\text{Jaccard}(A, B) = \frac{s(A \cup B)}{s(A) + s(B) - s(A \cup B)}$$

---





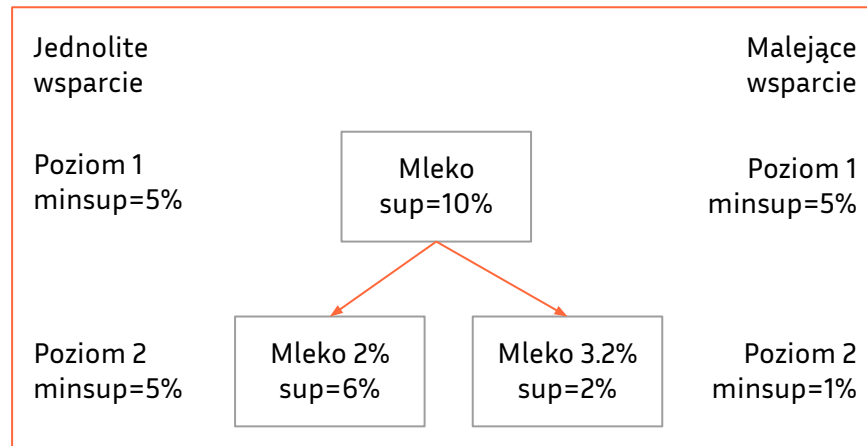
04

**BARDZIEJ  
ZAAWANSOWANE  
TEMATY**

---

# WIELOPOZIOMOWE REGUŁY ASOCJACYJNE

- Obiekty, np. towary, często tworzą hierarchię
  - Mleko → Mleko 2%
  - Chleb → Chleb pszenny
- Wielopoziomowe reguły mają często większą wartość poznawczą niż reguły jednopoziomowe.
- W jaki sposób ustawić próg minsup?
  - jednolity próg dla wszystkich poziomów,
  - próg wsparcia coraz mniejszy wraz z głębokością w hierarchii.
- Jak wydajnie eksplorować dane jeżeli próg wsparcia jest zależny od poziomu w hierarchii?
  - Użyj najmniejszego wsparcia do filtrowania zbiorów.





# WIELOWYMIAROWE REGUŁY ASOCJACYJNE

- Jednowymiarowe reguły asocjacyjne
    - wszystkie elementy są w jednym wymiarze, np. w wymiarze “produkt”
      - $\text{kupuje}(X, \text{“mleko”}) \rightarrow \text{kupuje}(X, \text{“chleb”})$
  - Wielowymiarowe reguły asocjacyjne
    - $n \geq 2$  wymiarów
      - $\text{wiek}(X, \text{“18-25”}) \wedge \text{zawód}(X, \text{“student”}) \rightarrow \text{kupuje}(X, \text{“coca-cola”})$
      - $\text{wiek}(X, \text{“18-25”}) \wedge \text{kupuje}(X, \text{“popcorn”}) \rightarrow \text{kupuje}(X, \text{“coca-cola”})$
  - Przekształcenia zmiennych
    - Zmienne kateryczne binaryzujemy
      - $\text{stan\_cywilny} = \{\text{panna}, \text{kawaler}, \text{zamężna}, \text{żonaty}, \dots\} \rightarrow$  dla każdej kategorii tworzymy zmienną binarną  $\text{stan\_cywilny} = \text{panna}, \text{stan\_cywilny} = \text{kawaler}$ , itd.
    - Zmienne ciągłe przekształcamy do kategorii, a następnie binaryzujemy.
      - Przekształcenie do kategorii: binning, klasteryzacja.
-

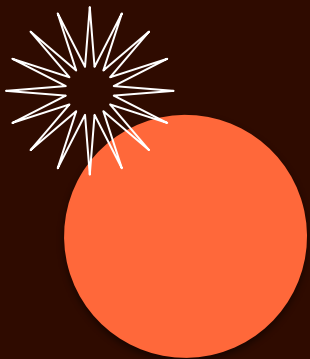


# NEGATYWNE ASOCJACJE

- $\text{parówki} \wedge \text{piwo} \rightarrow \text{musztarda} \wedge \neg \text{czerwone\_wino}$
  - $\text{szalik\_drużyny\_A} \rightarrow \neg \text{szalik\_drużyny\_B}$
  - Definicja 1: Wzorce negatywne zawierają co najmniej jeden zbiór negatywny (w poprzedniku lub w następniku).
  - Definicja 2: Wzorce negatywnie skorelowane.
    - Reguła  $X \rightarrow Y$  o negatywnej korelacji zbiorów X i Y ma de facto charakter “wiedzy negatywnej”, tzn. wystąpienie zbioru X w koszyku klienta zmniejsza prawdopodobieństwo wystąpienia zbioru Y w tym samym koszyku.
  - Problem odkrywania wiedzy negatywnej jest trudniejszy od odkrywania wiedzy pozytywnej.
    - Liczba odkrywanych reguł negatywnych w zbiorze danych jest wielokrotnie większa niż liczba reguł pozytywnych.
    - “Ludzie, którzy kupują chleb w supermarkecie, nie kupują lodówki”.
      - Należy określić podzbiór reguł negatywnych, który będzie interesujący.
-

THANKS!

**DZIĘKUJĘ  
ZA UWAGĘ**





Ewentualnie dodać szczegóły odnośnie tablicy haszującej i redukcja liczby przebiegów przez bazę danych (partycjonowanie) i ogólnie usprawnienia metody





# ZBIORY DOMKNIĘTE I ZBIORY MAKSYMALNE

Wyrażanie wzorców w formie skompresowanej: Wzorce zamknięte

Jak poradzić sobie z takim wyzwaniem?

Rozwiązanie 1: Zamknięte formacje: Formacja (zestaw pozycji)  $X$  jest zamknięta, jeśli  $X$  występuje często i nie istnieje super-wzorec  $Y$  zawierający  $X$ , z takim samym wsparciem jak  $X$

Zamknięty wzór to bezstratna kompresja częstego wzoru

Zmniejszono liczbę wzorów, ale nie traci się informacji pomocniczych

Rozwiązanie 2: Maksymalne wzorce: Wzorec  $X$  jest wzorcem maksymalnym, jeśli  $X$  jest częsty i nie istnieje żaden częsty superwzorec  $Y$  zawierający  $X$ .

Różnica od zamkniętych wzorców?

Nie przejmuj się prawdziwym wsparciem wzorców podrzędnych wzorca maksymalnego

Max-pattern to kompresja stratna

Wiemy tylko, że wzór jest częsty, ale nie znamy już prawdziwego wsparcia

W wielu zastosowaniach eksploracja zamkniętych wzorców jest bardziej pożądana niż eksploracja maksymalnych wzorców.

---



# CLOSET+

- Wydajny algorytm eksploracji zbiorów domkniętych oparty na FP-Growth.
  - Przykład: Jeżeli Y pojawia się zawsze wtedy, gdy X, to Y jest łączone z X.
    - Mając
      - d-proj. db: {acef, acf} -> acfd-proj. db: {e}
    - otrzymujemy
      - acfd: 2
-