МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет "Львівська політехніка"

Інтелектуальний аналіз даних за допомогою програмного пакета WEKA та MS Excel.

Асоціативний аналіз. Методи побудови асоціативних правил.

МЕТОДИЧНІ ВКАЗІВКИ
до лабораторної роботи № 6

з курсу **"Системи інтелектуального аналізу та візуалізації даних"**

для студентів за освітньою програмою Комп'ютерні науки (Проектування і програмування інтелектуальних систем та пристроїв)

Затверджено на засіданні кафедри
"Системи автоматизованого проектування"

Протокол N 1 від 28.08.2023р.

ЛЬВІВ 2023

1. МЕТА РОБОТИ

Метою лабораторної роботи є ознайомлення студентів з методами пошуку асоціативних правил за допомогою алгоритмів Apriori та FPGrowth в середовищах Weka та Excel.

Студенти мають набути навичок роботи з цими алгоритмами, виконавши певні тренувальні завдання в середовищі Weka, а потім застосувати ці навички для виконання індивідуальних завдань на власних наборах даних.

У програмі Excel студенти повинні застосувати алгоритм Apriori до свого власного набору даних про покупки для знаходження двійкових частих наборів.

2. КОРОТКІ ТЕОРЕТИЧНІ ВІДОМОСТІ

2.1. Алгоритми пошуку асоціативних правил, які розглядаються у лабораторній роботі

У лабораторній роботі розглядаються два методи пошуку асоціативних правил:

- алгоритм Apriori,
- алгоритм FPGrowth.

2.2. Сутність асоціації.

Кореляція вимірює силу лінійного зв'язку між двома числовими змінними. Сила виражається коефіцієнтом кореляції, який повинен бути в діапазоні від -1 до 1 включно. Дано дві змінні X і Y, якщо вони позитивно корельовані, то X і Y рухаються в одному напрямку. Наприклад, X – денна температура, а Y – продаж морозива. Чим вище X, тим більше Y; або чим менше X, тим менше Y. Якщо X і Y негативно корельовані, тоді вони рухаються в протилежних напрямках. Наприклад, пробіг автомобіля та вага автомобіля. Коли коефіцієнт кореляції близький до нуля, кореляції між X і Y немає. Обчислення коефіцієнта кореляції дуже просте в Excel за допомогою функції CORREL. Це показано на рис. 1.

	A	B	C	D	E
1	1	11		20	10
2	2	12		19	9
3	3	13		18	8
4	4	14		17	7
5	5	15		16	6
6	6	16		15	5
7	7	17		14	4
8	8	18		13	3
9	9	19		12	2
10	10	20		11	1
11					
12			=CORREL(A1:B10, D1:E10)		

Рис.1. Обчислення кореляції в Excel

Кореляція — особливий вид асоціації. У той час як кореляційне дослідження вимагає, щоб дві змінні були числовими та вимірювали лише лінійний зв'язок, асоціативний аналіз не має такого обмеження. Аналіз асоціацій вимірює силу спільного співпадіння між двома чи більше змінними.

Також, на відміну від методів побудови моделей класифікації, методи пошуку асоціативних правил не потребують вибору атрибуту класу, усі атрибути вважаються атрибутами ознак, класом є комбінація значень окремих атрибутів ознак. *Методи побудови асоціативних правил призначені для пошуку комбінацій значень атрибутів, на базі яких за комбінацією значень атрибутів першої множини («умова») можна спрогнозувати значення атрибутів другої множини («наслідок»).*

Пошук асоціативних правил часто виконують супермаркети під час аналізу споживчого кошику для визначення продуктів, які покупці часто купують разом, після чого знайдені комбінації товарів розміщують поруч, щоб збільшити ймовірність їх купівлі.

2.3. Алгоритм Apriori пошуку асоціативних правил у Weka Explorer

Першим розглянемо алгоритм Apriori – класичний метод пошуку закономірностей у значеннях атрибутів у вигляді асоціативних правил. Метод Apriori перебирає усі можливі комбінації значень заданої кількості атрибутів, наприклад, *комбінація$_1$={атрибут А значення $А_1$, атрибут Б значення $Б_1$}*, та вибирає ті комбінації, які перевищують мінімальне значення критерію підтримки (support criterion, задається у параметрах методу).

Використання алгоритму.

- Початкова інформація: транзакційна база даних D і визначений користувачем числовий мінімальний поріг підтримки min_sup
- Алгоритм використовує знання з попередньої фази ітерації для створення частих наборів елементів
- Це відображено в латинському походженні назви алгоритму Apriori, що означає «від того, що було раніше».

Створення частих наборів.

- Визначимо Ck як набір елементів-кандидатів розміром k, а Lk-1 як набір частих елементів розміру k
- Основні кроки ітерації:

1. Знайти частий набір L_{k-1}
2. Крок об'єднання: C_k генерується об'єднанням L_{k-1} із собою (декартів добуток L_k x L_{k-1})
3. Крок скорочення (апріорна властивість): будь-який набір елементів розміру (k − 1), який не є частим, не може бути підмножиною частого набору елементів розміру k, тому його слід видалити
4. Отримано частий набір L_k.

Алгоритм використовує пошук у ширину та хеш-деревоподібну структуру для ефективного створення наборів елементів-кандидатів. Потім підраховується частота появи для кожного набору елементів-кандидатів. Ті набори елементів-кандидатів, які мають вищу частоту, ніж мінімальний поріг підтримки, вважаються частими наборами елементів.

Псевдокод алгоритму Апріорі:

L_1= {frequent items}; **for**(k= 2; L_{k-1}!=∅; k++) **do begin**
C_k= candidates generated from L_{k-1} (that is: Cartesian product L_{k-1} x L_{k-1} and eliminating any k-1 size itemset that is not frequent); **for each** transaction t in database **do** increment the count of all candidates in C_k that are contained in t L_k = candidates in C_k with *min_sup*
end **return** $_k L_k$;

Приклад 1.

1. Відкрийте **Excel** та підготуйте датасет і збережіть як **apriori.csv**

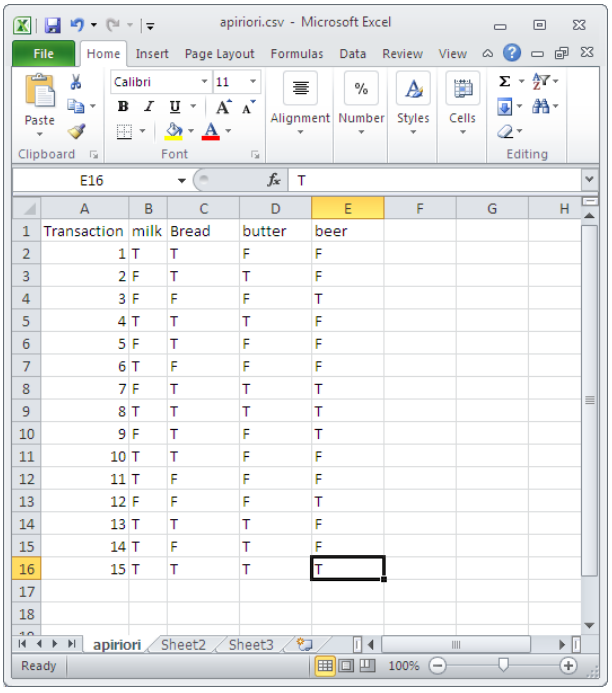

Рис. 2. Підготовка датасету для експерименту

2. Відкрийте **Weka** в режимі **Explorer**.

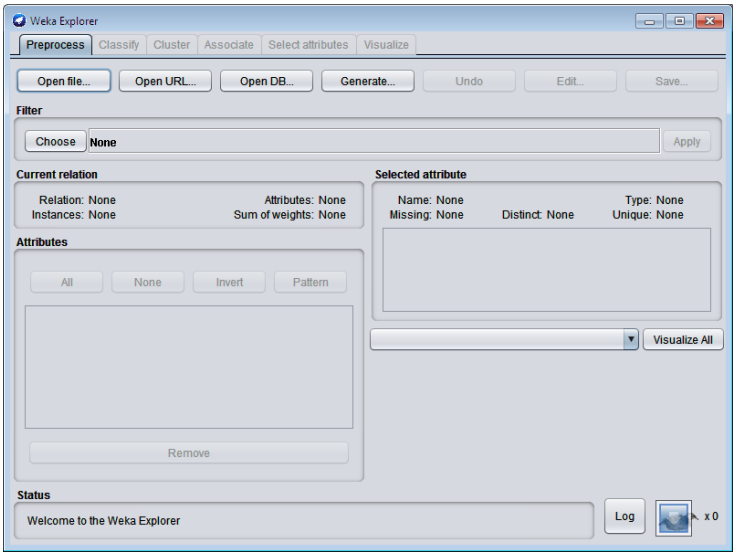

Рис. 3 Weka у режимі Explorer

3. Натисніть кнопку «Відкрити файл..» на вкладці «Preprocess» та виберіть apriori.csv.

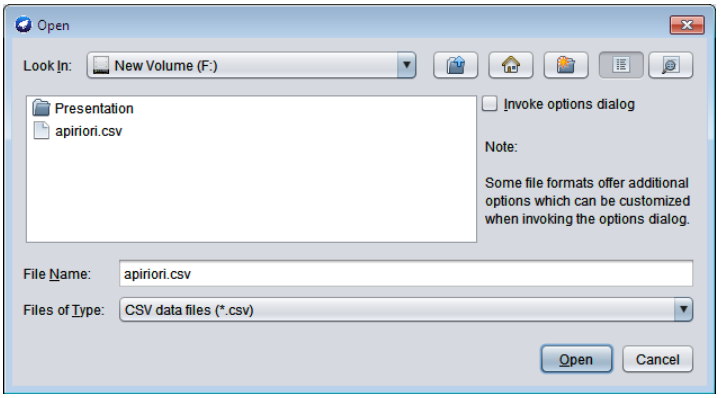

Рис. 4. Завантаження набору даних

4. Видаліть поле **Transaction** і збережіть файл під іменем aprioritest.arff

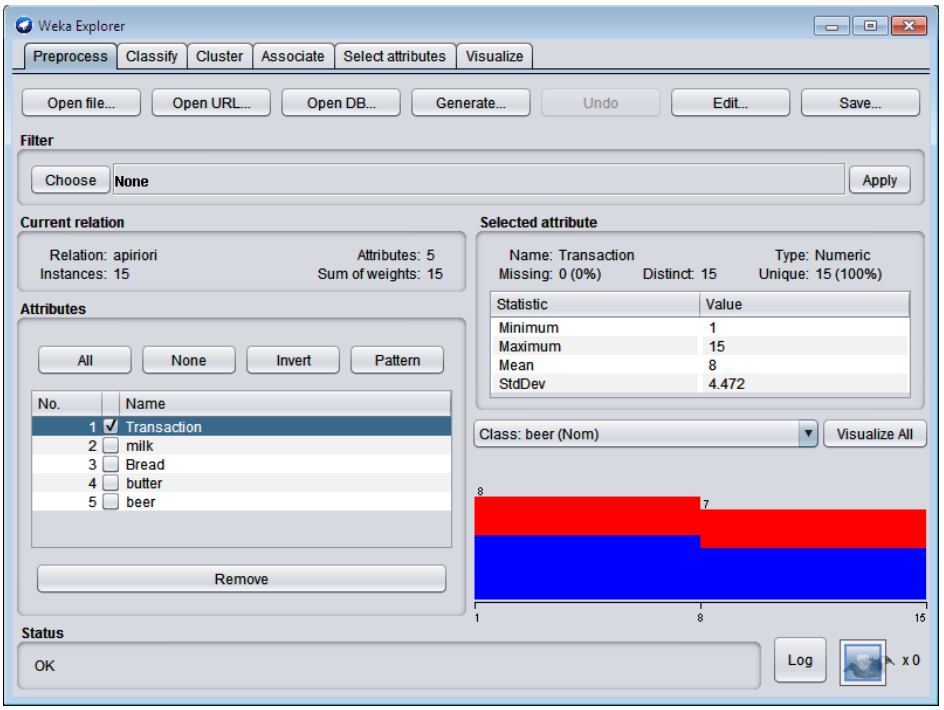

Рис. 5. Видалення атрибуту Transaction

5. Відкрийте закладку **Associate**, виберіть **Apriori** і клацніть на кнопку **Start**.

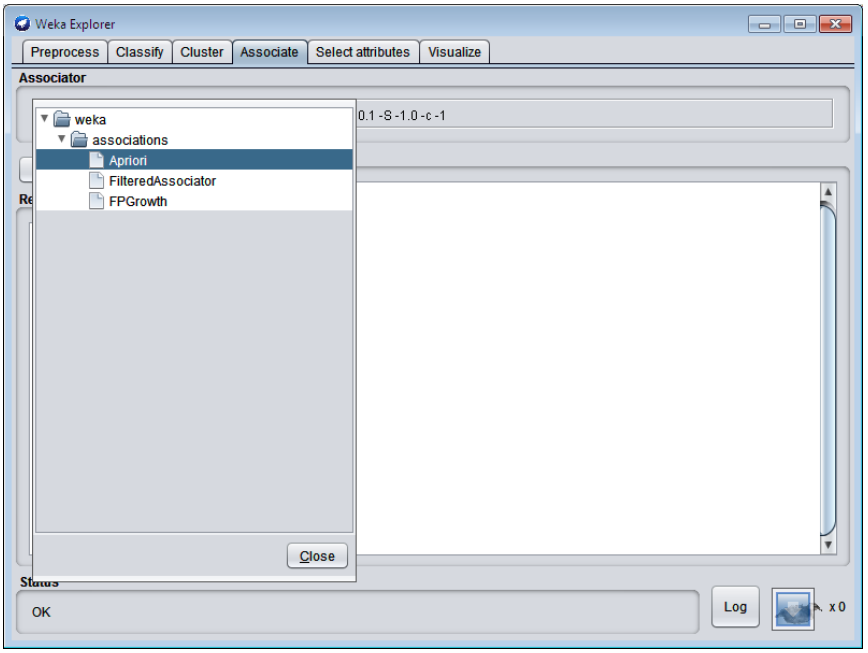

Рис. 6. Вибір алгоритму Apriori

Результат: Наведені нижче знімки екрану показують асоціативні правила, які були згенеровані під час застосування алгоритму Апріорі до заданого набору даних.

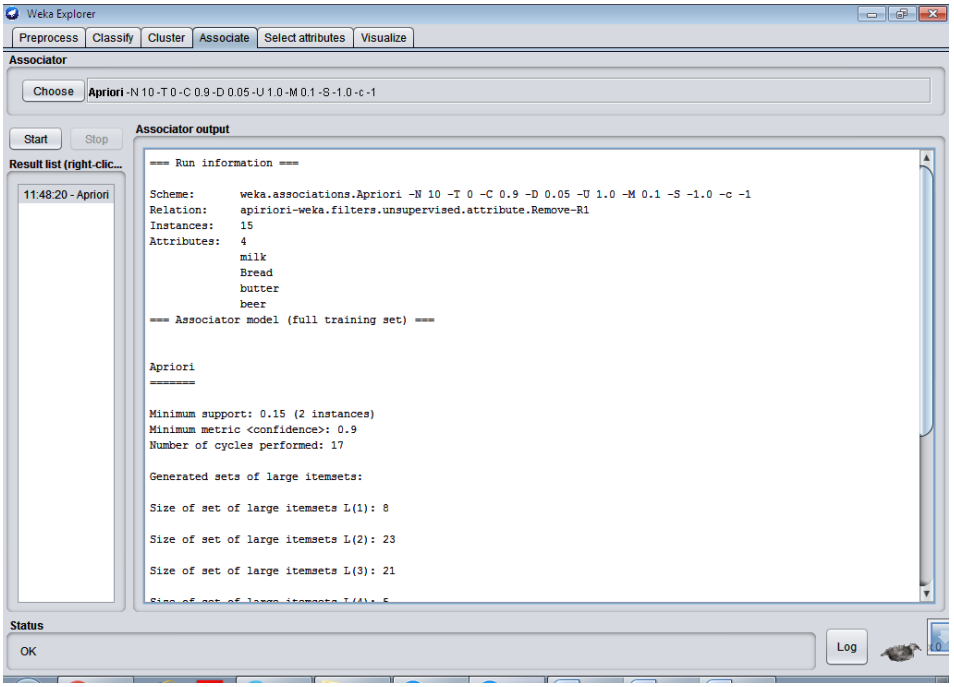

Рис. 7. Результат застосування алгоритму Апріорі

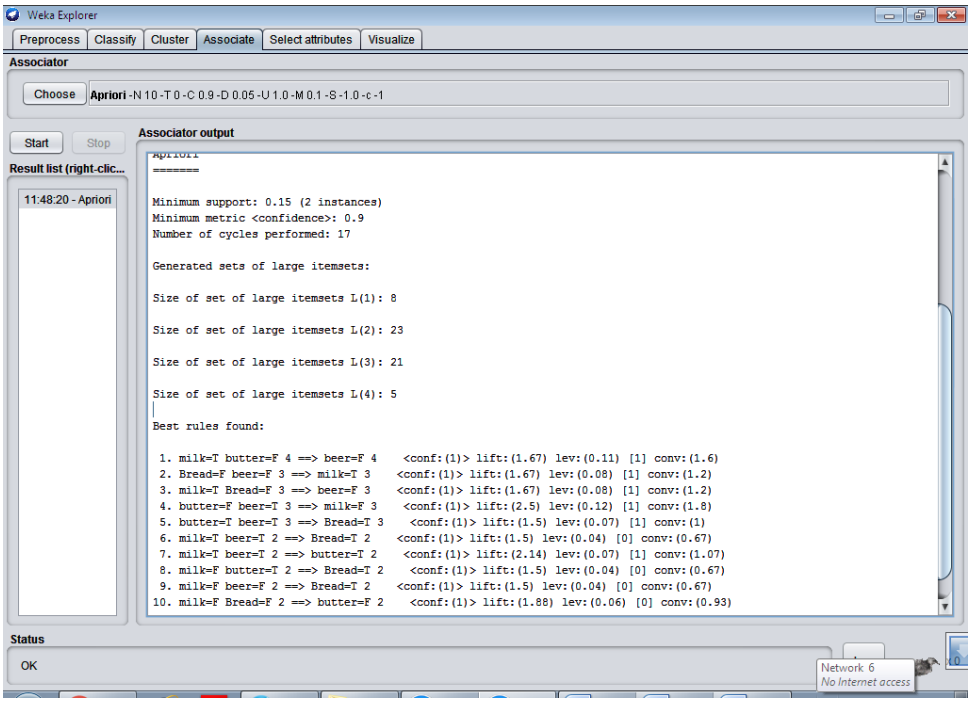

Рис. 8. Результат застосування алгоритму Апріорі

Приклад 2.

Для ілюстрації ідеї пошуку асоціативних правил виконаємо аналіз тестового файлу contact-lenses.arff (у папці data з каталогу Weka) з записами рекомендацій контактних лінз за фізіологічними характеристиками пацієнтів. Файл contact-lenses.arff містить 24 екземпляри даних, тому можна послідовно шукати закономірності, які містять від 1 до 24 атрибутів. Для кожної комбінації задається мінімальне значення критерію підтримки, нехай буде 20% від загальної вибірки з 24-х екземплярів, тоді мінімальна допустима кількість екземплярів, у яких буде знайдено конкретну закономірність має перевищувати чи дорівнювати $24 \times 0.1 = 4.8 \geq 5$ екземплярів.

Передбачається, що необхідні поля даних були дискретизовані.

Виконаємо у пакеті Weka пошук асоціативних правил за допомогою методу Apriori з заданими параметрами (див. Рисунок 1). Для відображення закономірностей вкажіть значення параметру outputItemSets=True.

Метод Apriori починає пошук з закономірностей, які містять лише одне значення атрибуту. Далі, в процесі пошуку, виходячи з того факту, що менші комбінації атрибутів зустрічаються частіше, кількість атрибутів у шуканих закономірностях поступово збільшується. Пошук зупиняється за умови відсутності знайдених екземплярів для поточної кількості атрибутів.

Результат пошуку для різних кількостей атрибутів зображено на Лістингу 1.

Лістинг 1. Закономірності для різних кількостей атрибутів (Apriori)

Minimum support: 0.2 (5 instances) Minimum metric <confidence>: 0.9 Number of cycles performed: 16 Generated sets of large itemsets: Size of set of large itemsets L(1): 11 Large Itemsets L(1): age=young 8 age=pre-presbyopic 8 age=presbyopic 8 spectacle-prescrip=myope 12 spectacle-prescrip=hypermetrope 12 astigmatism=no 12 astigmatism=yes 12 tear-prod-rate=reduced 12 tear-prod-rate=normal 12 contact-lenses=soft 5 contact-lenses=none 15 Size of set of large itemsets L(2): 21 Large Itemsets L(2): age=pre-presbyopic contact-lenses=none 5 age=presbyopic contact-lenses=none 6 spectacle-prescrip=myope astigmatism=no 6 spectacle-prescrip=myope astigmatism=yes 6 spectacle-prescrip=myope tear-prod-rate=reduced 6 spectacle-prescrip=myope tear-prod-rate=normal 6 spectacle-prescrip=myope contact-lenses=none 7 spectacle-prescrip=hypermetrope astigmatism=no 6 spectacle-prescrip=hypermetrope astigmatism=yes 6 spectacle-prescrip=hypermetrope tear-prod-rate=reduced 6 spectacle-prescrip=hypermetrope tear-prod-rate=normal 6 spectacle-prescrip=hypermetrope contact-lenses=none 8 astigmatism=no tear-prod-rate=reduced 6 astigmatism=no tear-prod-rate=normal 6 astigmatism=no contact-lenses=soft 5 astigmatism=no contact-lenses=none 7 astigmatism=yes tear-prod-rate=reduced 6 astigmatism=yes tear-prod-rate=normal 6 astigmatism=yes contact-lenses=none 8 tear-prod-rate=reduced contact-lenses=none 12 tear-prod-rate=normal contact-lenses=soft 5 Size of set of large itemsets L(3): 6 Large Itemsets L(3): spectacle-prescrip=myope tear-prod-rate=reduced contact-lenses=none 6	знайдено 21 закономірностей правило знайдено для 5 екземплярів з 24-х можливих

spectacle-prescrip=hypermetrope astigmatism=yes contact-lenses=none 5 spectacle-prescrip=hypermetrope tear-prod-rate=reduced contactlenses= none 6 astigmatism=no tear-prod-rate=reduced contact-lenses=none 6 astigmatism=no tear-prod-rate=normal contact-lenses=soft 5 astigmatism=yes tear-prod-rate=reduced contact-lenses=none 6	

Асоціативне правило складається з двох множин *X* («умова») і *Y* («наслідок») у вигляді конструкції IF-THEN: $X \rightarrow Y$, тобто, якщо знайдені значення атрибутів множини *X*, тоді з ним ймовірно будуть знайдені значення атрибутів множини *Y*.

Критерій підтримки (support criterion) – частка кількості екземплярів, яка припадає на кожне асоціативне правило (від 0.0 до 1.0). Для 24 екземплярів підтримка у 20% (minSupport=0.2) дорівнює 5, тобто кожне асоціативне правило має виконуватися щонайменше для 5 екземплярів з усього набору даних.

Критерій достовірності (confidence criterion) – відношення кількості екземплярів множини (*X U Y) до кількості екземплярів множини X у знайденому правилі. Відношення* характеризує рівень зв'язку атрибутів множин *X* та *Y*.

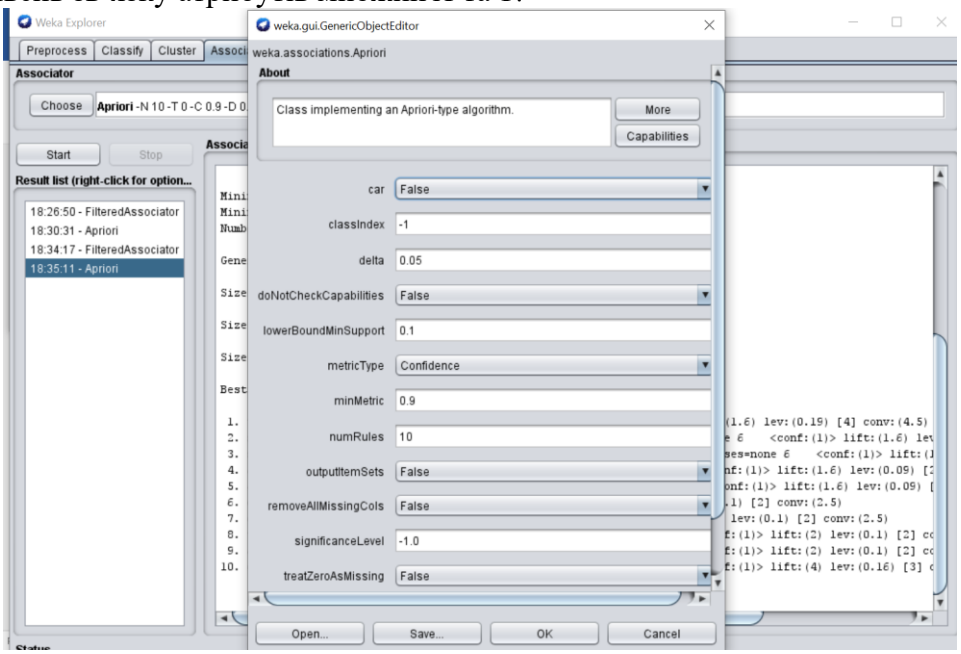

Рис 9. Параметри методу Apriori у Weka

Розглянемо правило № 5 зображене на Лістингу 2.

Лістинг 2. Приклад знайденого асоціативного правила

5. astigmatism=yes tear-prod-rate=reduced 6 ==> contact-lenses=none 6
<conf:(1)> lift:(1.6) lev:(0.09) [2] conv:(2.25)

Значення атрибутів множини X («умова») (astigmatism=yes tear-prodrate=reduced) були знайдені у 6 екземплярів так само, як і значення атрибутів множини Y («наслідок») (contact-lenses=none), відповідно достовірність асоціативного правила дорівнює conf = 6/6 = 1.0, тобто виконується умова minMetric=0.9. Розробіть програму та побудуйте перелік асоціативних правил за допомогою метода, зазначеного у варіанті.

2.4. Алгоритм FP-growth пошуку асоціативних правил у Weka Explorer

Apriori: використовує підхід генерування та тестування – генерує набори кандидатів і тестує чи вони часті.

- Створення наборів кандидатів є дорогим (як у просторі, так і в часі)

- Підрахунок підтримки коштує дорого
- Перевірка підмножини (обчислювально дорога)
- Неодноразові сканування бази даних (I/O)

FP-Growth: дозволяє знаходити набори частих елементів без створення набору кандидатів.

Двоетапний підхід:

- Крок 1: Побудова компактної структуру даних під назвою FP-дерево.

Створюється за допомогою 2 проходів набором даних.

- Крок 2: Витягує часті набори елементів безпосередньо з FP-дерева

FP-Tree створюється за допомогою 2 проходів над набором даних:

Прохід 1:

- Сканування даних та знаходження підтримки для кожного елемента.
- Відкинути нечасті елементи.
- Відсортувати часті елементи у порядку зменшення на основі їх підтримки.

Прохід 2:

Вузли відповідають елементам і мають лічильник

1. FP-Growth зчитує 1 транзакцію за раз і відображає її на шляху

- Використовується фіксований порядок, тому шляхи можуть накладатися, коли транзакції спільно використовують елементи. У цьому випадку лічильники збільшуються.

2. Вказівники підтримуються між вузлами, що містять той самий елемент, створюючи однозв'язані списки (пунктирні лінії)

- Чим більше шляхів перекриваються, тим вище стиснення. FP-дерево може все поміститися у пам'ять.

3. Часті набори елементів, витягуються з FP-Tree.

Приклад 3:

1. Завантажте файл **aprioritest.arff**, який був сформований у Прикладі 1.

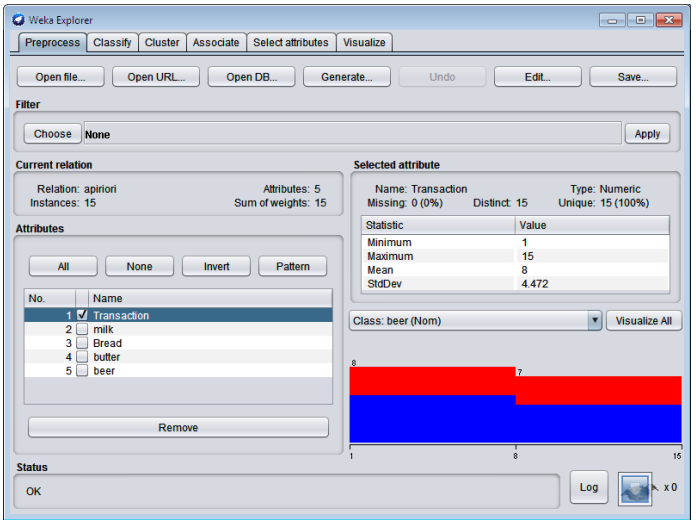

Рис. 10. Завантаження датасету

2. Відкрийте закладку **Associate**, виберіть **FPGrowth** і клацніть на кнопку **FPGrowth**.

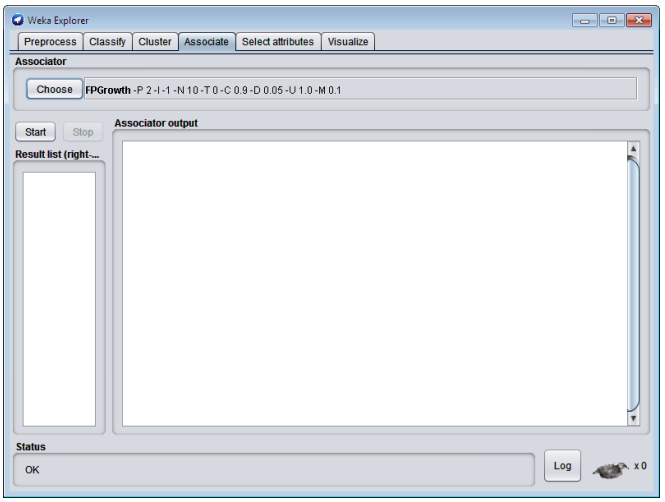

Рис. 11. Вибір алгоритму FP-Growth

Результат: FP-Growth знайшов 2 правила.

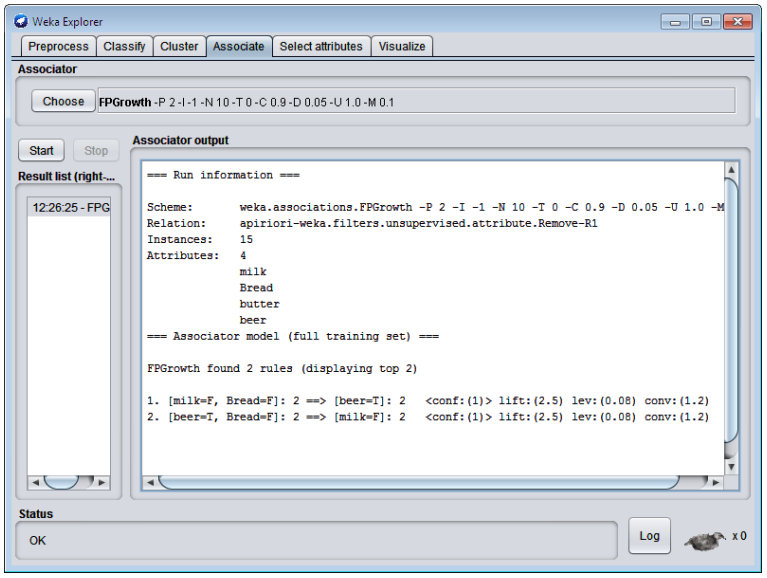

Рис. 12 Результати алгоритму FP-Growth

2.5. Параметри налаштування алгоритмів у Weka

Розглянемо параметри налаштування використовуваних алгоритмів пошуку асоціативних правил в WEKA (табл. 1).

Таблиця 1. Параметри налаштування алгоритмів

Метод	**Параметр**
Apriori	*car* – пошук класових (зі значенням цільового атрибута в правій частині) або звичайних асоціативних правил. *classIndex* – індекс цільового атрибута. Якщо встановлено значення -1, буде обраний останній атрибут. *delta* – ітеративно зменшувати значення порогу підтримки на дане значення. Зменшення буде відбуватися доти, поки не буде досягнуто мінімальне значення підтримки чи не буде згенеровано задану кількість правил.

	lowerBoundMinSupport – нижня межа порогу підтримки. *metricType* – встановлює тип метрики, за якою будуть ранжуватися правила (Confidence, Lift, Leverage, Conviction). *minMetric* – мінімальне граничне значення для обраної метрики. *numRules* – кількість правил, які необхідно знайти. *outputItemSets* – чи виводити часті набори. *removeAllMissingCols* – прибирати чи колонки (атрибути) в яких всі значення відсутні. *significanceLevel* – рівень значущості (тільки для достовірності). *upperBoundMinSupport* – верхня межа мінімальної підтримки. Ітеративне зменшення підтримки починається з цього значення.
FPGrowth	*delta* – ітеративно зменшувати значення порогу підтримки на дане значення. Зменшення буде відбуватися до тих пір, поки не буде досягнуто мінімальне значення підтримки чи не буде згенеровано задану кількість правил. *findAllRulesForSupportLevel* – знайти всі правила, які задовольняють нижній межі мінімального значення підтримки та мінімальному значенню метрики. Включення цього режиму скасує виконання ітеративного зменшення підтримки для знаходження заданого кількості правил. *lowerBoundMinSupport* - нижня межа порогу підтримки як частка кількості примірників. *maxNumberOfItems* – максимальна кількість примірників у частому наборі; значення -1 означає без обмежень. *metricType* – встановлює тип метрики, за якою будуть ранжуватися правила. *minMetric* – мінімальне граничне значення для метрики. *numRulesToFind* – кількість правил, які необхідно знайти. *positiveIndex* – встановлює індекс бінарного атрибуту, який буде розглядатися як позитивний. *rulesMustContain* – виводити правила, які містять задані об'єкти (список об'єктів, розділених комою). *transactionsMustContain* – для роботи алгоритму використовувати транзакції (примірники), які містять задані об'єкти. *upperBoundMinSupport* – верхня межа мінімальної підтримки. Ітеративне зменшення підтримки починається з цього значення. *useORForMustContainList* – - використовувати логічний зв'язку «або» замість «і» для списків обов'язкових елементів у транзакціях і правилах.

2.6. Асоціативний аналіз в Excel

Нехай у нас є файл, який містить 3538 записів для 1000 чеків (транзакцій).

Дані відсортовано за номерами чеків. У стовпці A наведено номери чеків, у стовпці C показано товари в чеку, а в стовпці B перераховано куплену кількість кожного товару. Наприклад, рядки 2–5 показують, що позиції 7, 15, 49 і 44 є у чеку 1. Ми проігноруємо кількість покупок у цьому дослідженні.

	A	B	C	D
1	Receipt	Quantity	ItemOnReceipt	
2	1	3	7	
3	1	4	15	
4	1	2	49	
5	1	5	44	
6	2	1	1	
7	2	2	19	
8	3	1	1	
9	3	1	19	
10	4	1	18	
11	4	1	35	
12	4	5	3	

Рис. 13. Дані транзакцій по чеках супермаркету

Для аналізу асоціативних правил дані необхідно реорганізувати у вигляді таблиці на рис. 14. Застосувавши дві функції =MAX(C2:C3539) і =MIN(C2:C3539), ми знаємо, що існує 50 різних елементів, пронумерованих від 0 до 49. Щоб відрізнити числові значення 0 і 1 від номерів елементів 0 і 1, ми поставимо літеру «I» перед кожним номером товару. Упорядкуйте дані у своюму файлі:

1. Введіть «Item» у клітинку D1 і «Receipt#» у клітинку E1.

2. Введіть формулу ="I"&C2 у клітинку D2. Буква I знаходиться всередині пари подвійних лапок.

3. Автозаповніть клітинки D2 до клітинки D3539.

4. Оскільки існує 1000 чеків, пронумерованих від 1 до 1000, введіть 1 у клітинку E2 і 2 у клітинку E3. Виберіть клітинки E2 і E3 та заповніть автоматично клітинку до клітинки E1001.

5. Введіть «I0» у клітинку F1 та горизонтально автозаповніть клітинку F1 до клітинки BC1. Комірки F1:BC1 містять номери елементів I0, I1, …, I49. Частина нашого аркуша виглядає так, як на рисунку 14.

	A	B	C	D	E	F	G	H	I	J
1	Receipt	Quantity	ItemOnReceipt	Item	Receipt#	I0	I1	I2	I3	I4
2	1	3	7	I7	1					
3	1	4	15	I15	2					
4	1	2	49	I49	3					
5	1	5	44	I44	4					
6	2	1	1	I1	5					
7	2	2	19	I19	6					
8	3	1	1	I1	7					
9	3	1	19	I19	8					
10	4	1	18	I18	9					
11	4	1	35	I35	10					
12	4	5	3	I3	11					
13	4	5	15	I15	12					
14	4	1	44	I44	13					
15	4	1	4	I4	14					
16	5	4	4	I4	15					

Рис. 14. Організуйте дані

6. Товар, який відображається на чеку, повинен мати числове значення 1 у відповідній клітинці. Наприклад, клітинка G3 повинна мати значення 1, оскільки елемент I1 є у чеку №2. Подібним чином, оскільки I7 знаходиться у чеку №1, клітинка M2 також має мати значення 1. Клітинки F2, G2 і H2 повинні мати значення 0, оскільки елементи I0, I1 і I3 відсутні у чеку №1. Нам потрібна формула, щоб призначити 1 під елемнт, якщо він є у квитанції; інакше призначте 0. Введіть у клітинку F2 наступну формулу: =COUNTIFS($A$2:$A$3539,$E2,$D$2:$D$3539,F$1)

У цій формулі «$A$2:$A$3539,$E2» повертає рядки 2–5, оскільки $E2 відповідає 1. Однак жоден із чотирьох рядків не має значення у стовпці D, що дорівнює F$1 (I0). Таким чином, ця формула повертає 0 у клітинці F2.

7. Автоматично заповніть формулу від F2 до клітинки BC2, а потім автоматично заповніть разом клітинки F1001:BC1001. Частина наших даних виглядатиме так, як показано на рис. 15.

	C	D	E	F	G	H	I	J	K	L	M	N	O
1	ItemOnReceipt	Items	Receipt#	I0	I1	I2	I3	I4	I5	I6	I7	I8	I9
2	7	I7	1	0	0	0	0	0	0	0	1	0	0
3	15	I15	2	0	1	0	0	0	0	0	0	0	0
4	49	I49	3	0	1	0	0	0	0	0	0	0	0
5	44	I44	4	0	0	0	1	1	0	0	0	0	0
6	1	I1	5	0	0	1	0	1	0	0	1	0	1
7	19	I19	6	0	0	0	0	0	0	0	0	0	0
8	1	I1	7	0	0	0	0	1	0	0	0	0	0
9	19	I19	8	0	0	0	0	0	0	0	0	0	0
10	18	I18	9	0	0	1	0	0	0	0	0	0	0
11	35	I35	10	0	0	0	1	0	0	0	0	0	0
12	3	I3	11	0	0	0	0	0	0	0	0	0	0
13	15	I15	12	0	0	0	0	0	0	0	0	0	0
14	44	I44	13	0	0	0	0	0	0	0	0	0	0
15	4	I4	14	0	1	0	0	0	0	0	0	0	0
16	4	I4	15	0	0	0	0	0	1	0	0	0	0
17	9	I9	16	0	0	0	0	0	0	0	0	0	0
18	23	I23	17	0	0	0	0	1	0	1	0	0	1
19	2	I2	18	0	1	0	0	0	0	0	0	0	0
20	7	I7	19	0	0	0	0	0	0	0	1	0	0

Рис. 15. Створіть таблицю з даними для пошуку асоціативних правил

У цьому прикладі ми перевіримо всі можливі набори елементів розміром 2; отже, нам потрібно налаштувати таблицю так, щоб і рядки, і стовпці таблиці були позначені кожним елементом. Пізніше нам також потрібно обчислити значення впевненості (confidence) та підтримки (support) для кожного набору елементів. Для цього нам потрібно обчислити (1) кількість разів, коли два товари одночасно зустрічаються в одному чеку, і (2) кількість чеків, у яких з'являється товар.

8. У клітинку BG1 введіть текст «I0», а потім автоматично заповніть горизонтально клітинку DD1.

9. Введіть текст «Occurrences» у BF2, і введіть формулу =COUNTIFS(F2:F1001,1) у комірку BG2 та заповніть автозаповнення від комірки BG2 до комірки DD2. Клітинки BG2:DD2 посилаються на «кількість чеків, у яких відображається товар». Наприклад, позиція I0 з'являється у 84 чеках (показано на рисунку 16).

10. Введіть 1 у клітинку BE3 і 2 у клітинку BE4. Виберіть клітинки BE3 і BE4 та автозаповніть клітинки до клітинки BE52. Ця дія заповнить числа 1, 2, 3, …, 50 у клітинках BE3, BE4, BE5, …, BE52. Ці номери використовуватимуться для посилання на стовпець у таблиці даних F2:BC1001, оскільки у F2:BC1001 є 50 стовпців, один стовпець для одного елемента.

11. Введіть «I0» у комірку BF3 та заповніть автозаповнення від комірки BF3 до комірки BF52. Порівняйте ваш аркуш із рис. 16. Якщо наш робочий аркуш виглядає інакше, ніж на рис. 16, нам потрібно (1) перевірити наші формули та (2) переконатися, що в назвах елементів немає порожніх місць. Наприклад, якщо в комірці F1 або комірці BG1 ім'я елемента введено як «I0», обчислення I0 буде неправильним.

	BC	BD	BE	BF	BG	BH	BI	BJ	BK	BL
1	I49				I0	I1	I2	I3	I4	I5
2	1			Occurrences	84	85	72	78	91	103
3	0		1	I0						
4	0		2	I1						
5	0		3	I2						
6	0		4	I3						
7	0		5	I4						
8	0		6	I5						
9	0		7	I6						
10	0		8	I7						
11	0		9	I8						
12	0		10	I9						
13	0		11	I10						
14	1		12	I11						
15	0		13	I12						

Рис. 16. Організуйте таблицю асоціацій

12. Давайте обчислимо, скільки разів два елементи одночасно зустрічаються в одному чеку. Введіть таку формулу в клітинку BG3: =COUNTIFS(INDEX($F$2:$BC$1001,0,$BE3), 1;F$2:F$1001,1)

BE3 = 1, тому функція INDEX($F$2:$BC$1001,0,$BE3) отримує перший стовпець із таблиці F2:BC1001, який є стовпцем для елемента I0, тобто F2:F1001. Таким чином, попередня формула стає =COUNTIFS(F2:F1001,1,F$2:F$1001,1) у клітинці BG3. Ця формула підраховує, скільки рядків одночасно відповідають двом умовам: клітинки всередині F2:F1001, які мають значення 1, і клітинки всередині F$2:F$1001, які також мають значення 1.

У цій конкретній формулі трапляється, що дві умови однакові, оскільки клітинка BG3 призначена для набору елементів (I0, I0). Це може ввести в оману.

13. Автозаповнення від комірки BG3 до комірки DD3, а потім разом автозаповнення до BG52:DD52. Давайте подивимося на формулу в клітинці BH3. Це =COUNTIFS(INDEX($F$2:$BC$1001,0,$BE3), 1,G$2:G$1001,1) INDEX($F$2:$BC$1001,0,$BE3) все одно дасть нам F2:F1001 для елемента I0.

Однак G$2:G$1001 призначено для елемента I1. Таким чином, ця формула тепер підраховує, скільки разів елемент I0 і елемент I1 одночасно з'являються в одному чеку.

Іншим прикладом є формула в клітинці BJ4. Це =COUNTIFS(INDEX($F$2:$BC$1001,0,$BE4), 1,I$2:I$1001,1) Оскільки BE4 = 2, ця формула фактично така ж, як =COUNTIFS(G$2: G$1001,1,I$2:I$1001,1) Очевидно, клітинка BJ4 підраховує, скільки разів елемент I1 і елемент I3 одночасно з'являються в одній квитанції.

А тепер давайте порівняємо ваш аркуш із рисунком 17.

	BE	BF	BG	BH	BI	BJ	BK	BL	BM	BN
1			I0	I1	I2	I3	I4	I5	I6	I7
2		Occurrences	84	85	72	78	91	103	34	93
3	1	I0	84	3	40	2	2	4	5	2
4	2	I1	3	85	7	8	1	6	2	4
5	3	I2	40	7	72	2	5	1	2	3
6	4	I3	2	8	2	78	6	4	1	2
7	5	I4	2	1	5	6	91	6	4	6
8	6	I5	4	6	1	4	6	103	1	7
9	7	I6	5	2	2	1	4	1	34	0
10	8	I7	2	4	3	2	6	7	0	93
11	9	I8	0	7	3	2	3	4	2	2
12	10	I9	2	2	4	5	49	6	5	8
13	11	I10	4	3	4	4	8	5	3	3
14	12	I11	2	5	2	4	4	1	0	33
15	13	I12	4	3	3	1	9	5	2	0

Рис. 17. Обчислені співпадіння двох елементів

14. Нам потрібно побудувати іншу таблицю, щоб показати, які набори елементів є дійсними на основі нашого мінімального значення підтримки та мінімального значення достовірності.

• Введіть текст «Support» у клітинку DF2, число 0,03 у клітинку DF3, текст «Confidence» у клітинку DF4 та число 0,5 у клітинку DF5.

• Введіть «I0» у клітинку DH1 та заповніть клітинку FE1 по горизонталі.

• Введіть «I0» у клітинку DG3 та автоматично заповніть вертикально клітинку DG52.

Порівняйте вашу таблицю з рисунком 18.

	DD	DE	DF	DG	DH	DI	DJ	DK	DL	DM
1	I49				I0	I1	I2	I3	I4	I5
2	59		Support							
3	2		0.03	I0						
4	2		Confidence	I1						
5	2		0.5	I2						
6	2			I3						
7	2			I4						
8	3			I5						
9	3			I6						
10	24			I7						
11	1			I8						
12	2			I9						
13	1			I10						
14	1			I11						
15	5			I12						

Рис. 18. Формування таблиці асоціативного аналізу

15. Введіть наступну формулу в комірку DH3, автозаповнення від DH3 до FE3, а потім разом автозаповнення від комірок DH3:FE3 до комірок DH52:FE52:

=IF($DG3=DH$1,"",IF(AND(BG$2>0,BG3/1000>=$DF$3, BG3/BG$2>=$DF$5),TEXT(BG3/1000,"0,000") & ", " &TEXT(BG3/BG$2,"0,000"),""))

Для цієї формули нам потрібно розуміти наступне:

a. Зауважте, клітинка DH3 посилається на набір елементів (DH1, DG3), який є (I0, I0). Так само клітинка DI7 посилається на набір елементів (DI1, DG7), який є набором елементів (I1, I4).

b. Якщо DG3 і DH1 стосуються одного і того ж елемента (вони є в цьому випадку), немає необхідності обчислювати значення підтримки або достовірності.

в. BG$2>0, щоб уникнути можливої помилки ділення на нуль.

d. BG3/1000>=$DF$3 — переконатися, що значення підтримки набору елементів (DH1, DG3) не менше мінімального значення підтримки.

Зверніть увагу, що жорстко закодована кількість чеків 1000. Кращим підходом є використання посилання на клітинку, тобто збереження кількості чеків у клітинці.

d. BG3/BG$2>=$DF$5, щоб переконатися, що значення достовірності набору елементів (DH1, DG3) відповідає мінімальним вимогам достовірності.

f. Функція AND гарантує, що якщо будь-яка вимога не виконується, нічого не відображається в клітинці DH3.

g. Для форматування результатів використовується функція TEXT.

16. Порівняйте ваш результат із рисунком 19. Налаштуйте мінімальне значення підтримки та значення достовірності, щоб переглянути зміни знайдених наборів елементів.

	DF	DG	DH	DI	DJ	DK	DL
1			I0	I1	I2	I3	I4
2	Support						
3	0.03	I0			0.040, 0.55		
4	Confidence	I1					
5	0.5	I2					
6		I3					
7		I4					
8		I5					
9		I6					
10		I7					
11		I8					
12		I9					0.049, 0.53
13		I10					

Рис. 19. Асоціативний аналіз частих наборів розмірності 2

Використання Excel для проведення аналізу асоціацій для наборів елементів розміром 3 або більше стає непрактичним, тому зупинимося у нашому розгляді на 2 елементних групах.

3. ЛАБОРАТОРНЕ ТРЕНУВАННЯ

3.1 Використовувані датасети:

vote.arff - Цей набір даних містить інформацію про те, як кожен із конгресменів Палати представників США голосував за 16 ключових законів. Один екземпляр представляє історію голосування одного конгресмена та його партійну приналежність. (Для класифікації класова використовувалася партійна приналежність.)

weather.nominal.arff - Це дуже маленький набір даних лише з номінальними атрибутами.

supermarket.arff - Цей набір даних описує купівельні звички покупців супермаркету. Більшість атрибутів позначають одну конкретну групу предметів. Значення є `t', якщо клієнт купив товар поза асортиментом товарів, а в іншому випадку відсутній. Є один екземпляр на клієнта. Набір даних не містить атрибут класу, оскільки це не потрібно для вивчення асоціаціативних правил.

3.2 Панель Associate

Панель Associate виглядає дуже схожою на панель класифікації. Це в основному те саме, лише відсутні вікно параметрів тесту та поле вибору класу. Обидва не стосуються асоціаціативних правил. Асоціативні правила звичайно не надають одному атрибуту спеціальної позиції, як це робить класифікація з атрибутом класу. Панель тестування не потрібна, оскільки вивчення асоціативних правил здебільшого розглядається як завдання дослідницького аналізу даних. Це означає, що немає сильного акценту на точному оцінюванні.

3.3 Результати застосування алгоритму Apriori

Завантажте набір даних `vote.arff' і перейдіть до панелі Associate. Виберіть `Apriori' як асоціатор. Після натискання кнопки «Пуск» Apriori починає будувати свою модель і записує результати в поле виводу. Перша частина виводу (`Інформація про виконання', англ. Run information) описує параметр, який було встановлено, і використаний набір даних.

```
=== Run information ===

Scheme:       weka.associations.Apriori -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1
Relation:     vote
Instances:    435
Attributes:   17
              handicapped-infants
              water-project-cost-sharing
              adoption-of-the-budget-resolution
              physician-fee-freeze
              el-salvador-aid
              religious-groups-in-schools
              anti-satellite-test-ban
              aid-to-nicaraguan-contras
              mx-missile
              immigration
              synfuels-corporation-cutback
              education-spending
              superfund-right-to-sue
              crime
              duty-free-exports
              export-administration-act-south-africa
              Class
```

Наступна частина результату — це інформація про те, як були згенеровані правила. Це має виглядати наступним чином:

```
Apriori
=======

Minimum support: 0.45 (196 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 11

Generated sets of large itemsets:

Size of set of large itemsets L(1): 20

Size of set of large itemsets L(2): 17

Size of set of large itemsets L(3): 6

Size of set of large itemsets L(4): 1
```

Створені правила перераховані в кінці результату. Здебільшого показано не всі правила. За замовчуванням показано 10 найцінніших за рівнем достовірності. Кожне правило складається з деяких значень атрибутів ліворуч від стрілки, знака стрілки та правого списку значень атрибутів. Праворуч від знака стрілки розташовані прогнозовані значення атрибутів.

Якщо взяти правило 1 як приклад, це правило означає, що якщо значення для `adoption-of-the-budget-resolution' дорівнює `y', а значення для 'physician-fee-freeze' 'n', то значення атрибута `class ' слід прогнозувати як `democrat' (Зверніть увагу, що НЕ в усіх правилах передбачене значення є значенням атрибута класу.) Це передбачення має певну підтримку та значення довіри.

Число перед знаком стрілки означає кількість випадків, до яких застосовується правило. Число після стрілки означає кількість випадків, передбачених правильно. Число в дужках після `conf:' означає достовірність правила.

3.4 Датасет для голосування

Завантажте набір даних `vote.arff'.

Завдання A1: запустіть Apriori, використовуючи налаштування параметрів за замовчуванням.

Достовірність правила 10 становить 0:96. Як було обчислено це значення впевненості?

Запишіть пропорцію у вигляді ділення.

Завдання A2: Скільки випадків підтримує правило 8?

Завдання A3: Що означає «правило застосовується до певної кількості випадків»?

Поясніть на прикладі правила номер 7. (Підказка: ви можете перевірити числа на панелі попередньої обробки.)

Завдання A4: Що означає «кількість випадків, передбачених правильно»? Поясніть на прикладі правила номер 9.

Завдання A5: Вивчіть опис параметрів для Apriori, натиснувши кнопку `More' у вікні, яке дозволяє вам змінити параметри для `Apriori'. Спробуйте змінити кількість правил, указаних у вихідних даних. Як ви думаєте, чи може кількість згенерованих правил перевищувати 100. Якщо так, то чому?

Завдання A6: Що означає «найкращі правила»? Який критерій використовується для визначення найкращих правил?

Завдання A7: Яке правило визначає, наскільки ймовірно, що якщо конгресмен не голосував за допомогу Сальвадору, він також голосував за допомогу нікарагуанським контрас?

Завдання A8: 10 найкращих правил містять правила, які містять «Class=democrat» у правій частині. Чи говорить це щось про виборчі звички конгресменів-демократів?

3.5 Датасет для погоди

Завантажте набір даних `weather.nominal.arff'.

Завдання B1: Розгляньте правило:

temperature=hot ==> humidity=normal.

Чим підтверджується це правило? Скільки випадків застосовується до цього правила і яке значення достовірності? (Щоб відповісти на це запитання, відкрийте вікно «Перегляд» на панелі попередньої обробки.)

Завдання B2: Розгляньте правило:

temperature=hot humidity=high ==> windy=TRUE.

Чим підтверджується це правило? Скільки випадків застосовується до цього правила і яке значення достовірності? Далі запишіть номери екземплярів, які підтримують правило, і кількість екземплярів, які застосовуються до цього правила.

Завдання B3: Чи може правило мати перевірки двох (чи більше) атрибутів праворуч, як у прикладі нижче:

outlook=sunny temperature=cool ==> humidity=normal play=yes

3.6 Датасет для супермакету

Завантажте набір даних `supermarket.arff'.

Використовуйте Apriori для створення правил і використовуйте їх, щоб сказати щось про купівельні звички клієнтів супермаркету. Згенеруйте близько 30 правил.

Також може бути цікаво створити правила з одним конкретним атрибутом у правій частині. Їх можна згенерувати, встановивши для першого параметра значення `true', а для другого параметра — значення індексу атрибута (індекси атрибутів для цього параметра починаються з 0, а не з 1), яке ви хочете бачити в правій частині правил.

Завдання C1: Вивчіть кілька згенерованих правил і опишіть одне спостереження, яке, на вашу думку, було зроблено щодо купівельних звичок клієнтів супермаркету. Також запишіть відповідні правила для цього спостереження.

Завдання С2: Опишіть друге спостереження, яке, на вашу думку, було зроблено щодо купівельних звичок клієнтів супермаркету. Також запишіть відповідні правила для цього спостереження.

Завдання С3: Чи пропонують спостереження, зроблені вами в завданнях С1 і С2, якісь напрямки дій для менеджера супермаркету? Якщо так, то якими вони можуть бути?

4. ЛАБОРАТОРНЕ ЗАВДАННЯ

1. Виконайте наступні завдання для власного набору даних у Weka:
    - Запустіть алгоритм пошуку асоціативних правил Apriori.
    - Яке значення для порогу підтримки було використано в побудованій моделі? Яке значення для порогу достовірності було використано?
    - Запишіть 10 найкращих знайдених правил, вкажіть для них значення підтримки та достовірності.
    - Що позначають числа ліворуч і праворуч від стрілки в знайдених асоціативних правилах?
2. Виконайте наступні завдання для власного набору даних у Weka
    - Запустіть алгоритм пошуку асоціативних правил FPGrowth.
    - Порівняйте списки десяти найкращих правил, отриманих двома алгоритмами. Поясніть відмінність в роботі двох алгоритмів.
    - Згенеруйте також правила, у правій частині яких буде знаходитися ваш цільовий атрибут.
3. Виконайте наступні завдання для власного набору даних в Excel
    - Сформуйте власний набір транзакцій покупок, як показано у прикладі 3.
    - Знайдіть двоелементні групи частих наборів.
    - Організуйте ваші обчислення на кількох аркушах (мінімум одниаркуш із вхідними даними і один аркуш для аналізу.)

Бібліотеки наборів даних:

1. UCI Machine Learning Repository: https://archive.ics.uci.edu/
2. Datasets section at Weka: http://www.cs.waikato.ac.nz/ml/weka/

5. КОНТРОЛЬНІ ЗАПИТАННЯ

1. Що таке асоціативні правила і в яких випадках вони використовуються?
2. У чому полягає задача пошуку асоціативних правил? Наведіть практичний приклад?
3. Що таке частий набір?
4. Що таке сильне асоціативне правило?
5. З яких двох кроків складається пошук асоціативних правил?
6. У чому полягає принцип Apriori?
7. Як формуються правила зі знайдених частих наборів?
8. Що таке алгоритм Apriori? Опишіть його основні принципи роботи.
9. Що таке алгоритм FPGrowth? В чому його переваги порівняно з алгоритмом Apriori?
10. Як використовувати алгоритм Apriori в середовищі Weka? Продемонструйте кроки.
11. Як використовувати алгоритм FPGrowth в середовищі Weka? Продемонструйте кроки.
12. Як використовувати алгоритм Apriori в Excel? Продемонструйте кроки.
13. Що таке спорідненість, підтримка, достовірність та підйом в контексті асоціативних правил?
14. Які результати ви отримали при виконанні тренувальних задач в Weka?

15. Розкажіть про ваш власний набір даних, який ви використовували для самостійного завдання. Які асоціативні правила ви були в змозі виявити?
16. Наведіть приклади використання алгоритмів пошуку асоціативних правил в реальних застосуваннях.

6. ЗМІСТ ЗВІТУ

1. Тема і мета роботи.
2. Завдання до роботи.
3. Результати виконання завдань.
4. Відповіді на контрольні запитання.
5. Висновки, що відображують результати виконання роботи та їх критичний аналіз.