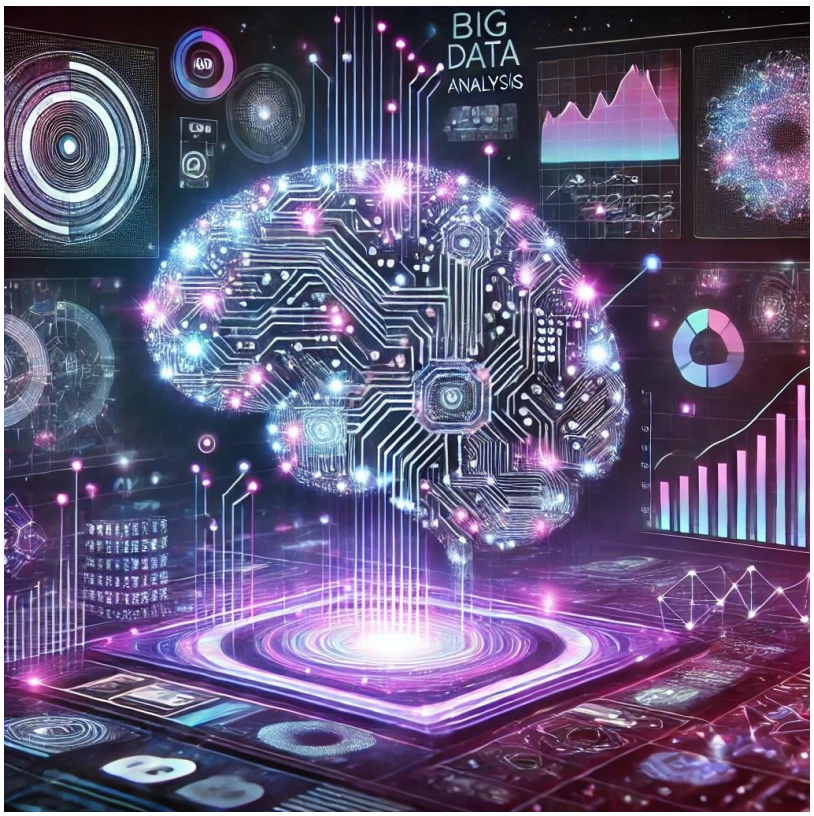
Міністерство освіти і науки України
Центральноукраїнський національний технічний університет
Кафедра кібербезпеки та програмного забезпечення

МЕТОДИЧНІ РЕКОМЕНДАЦІЇ ДО ВИКОНАННЯ ЛАБОРАТОРНИХ РОБІТ

з навчальної дисципліни

“ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ ДАНИХ”

для студентів денної та заочної форм навчання
за спеціальністю 122 «Комп’ютерні науки»



Кропивницький
2024

Міністерство освіти і науки України
Центральноукраїнський національний технічний університет
Кафедра кібербезпеки та програмного забезпечення

МЕТОДИЧНІ РЕКОМЕНДАЦІЇ ДО ВИКОНАННЯ ЛАБОРАТОРНИХ РОБІТ

з навчальної дисципліни

“ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ ДАНИХ”

для студентів денної та заочної форм навчання
за спеціальністю 122 «Комп’ютерні науки»

ЗАТВЕРДЖЕНО
на засіданні кафедри
кібербезпеки та програмного
забезпечення,
Протокол № 3 від 18.09.2024

Кропивницький
2024

Методичні рекомендації до виконання лабораторних робіт з навчальної дисципліни “Інтелектуальний аналіз даних” [для студ. денної та заочної форми навч. за спеціальністю 122 “Комп’ютерні науки”] / Уклад. І. А. Лисенко – Кропивницький: ЦНТУ, 2024.– 50 с.

Укладач Лисенко І.А., канд. техн. наук

Рецензенти: Смірнов О. А., д-р техн. наук, професор;
Якименко Н. М., канд. фіз.-мат. наук, доцент.

Методичні рекомендації висвітлюють організаційні та практичні аспекти виконання лабораторних робіт з навчальної дисципліни “інтелектуальний аналіз даних” для студентів денної та заочної форм навчання за спеціальністю 122 “Комп’ютерні науки”, а також рекомендації щодо ходу виконання робіт, підготовки та представлення отриманих результатів.

ЗМІСТ

Вступ…… …...6

Лабораторна робота № 1. Попередній статистичний аналіз…… ….11

Лабораторна робота № 2. Ознайомлення з інтерфейсом пакетів RapidMiner та KNIME……. ….16

Лабораторна робота № 3. Дискримінантний аналіз…….. ….23

Лабораторна робота № 4. Пошук асоціативних правил…… ….29

Лабораторна робота № 5. Аналіз зв'язків…… ….42

Список рекомендованої літератури…… ….49

Вступ

Під інформацією (лат. Informatio – роз'яснення, виклад) розуміють «повідомлення» у будь-якій формі; відомості, які є об'єктом зберігання, переробки та передавання. Іншими поясненнями цього поняття є такі: Абстрактне значення виразів, висловлень, графічних зображень тощо (М. Брой). Кількісна міра усунення невизначеності (ентропія), міра організації системи (К. Шеннон). Будь-які раніше невідомі відомості (знання) про подію (сутність або процес), які є об'єктами операцій, для яких існує змістова інтерпретація. «Під інформацією розуміють документовані або публічно оголошені відомості про події та явища, що відбуваються у суспільстві, державі та навколишньому природному середовищі». «Інформація – це відомості, що передаються усно, письмово чи якимось іншим способом, за допомогою умовних сигналів з використанням технічних засобів також, та сам процес передавання чи здобування цих відомостей». У кібернетиці: «Інформація – це будь-яка сукупність сигналів, впливів або відомостей, які деяка система сприймає від оточуючого середовища (вхідна інформація), видає в оточуюче середовище (вихідна інформація) або зберігає в собі (внутрішня інформація)» (А.О. Дородніцин). Під даними розуміють представлення деяких фактів у формалізованому вигляді, придатному для зберігання, обробки та передавання, тобто дані – це зареєстровані сигнали будь-якої природи, використані для передавання змісту повідомлень. Це факти, тексти, графіки, картинки, звуки, аналогові та цифрові відео-матеріали, представлені у формі, придатній для зберігання, передавання та обробки. Підсумовуючи, слід зауважити, що інформація є продуктом взаємодії даних та адекватних методів інтерпретації. Інтерпретування, в свою чергу, – це перехід від подання (зовнішньої форми) до значення (абстрактного змісту) інформації. Близьким за значенням, але більш глибоким за сутністю є розуміння – встановлення зв'язку між інформацією, поданою у деякій зовнішній формі, та реальним світом.

На відміну від даних інформація має деякий контекст. Інформація – це спосіб та система обробки даних.

Отже, під знаннями розуміють певним чином структуровану сукупність інформації про світ, властивості об'єктів, закономірності процесів і явищ, а також правила їх використання для прийняття рішень. Тобто знання представляють собою 1) сукупність понять, фактів, закономірностей та евристичних правил, організовану у деяку структуру за допомогою підходящих відношень; 2) продукт інформаційної діяльності, реалізований як система суджень або тверджень стосовно об'єктів, процесів або явищ.

Дані перетворюються в інформацію за допомогою категоризації, калькуляції, контекстуалізації, коректування та конденсації.

Інформація перетворюється в знання за допомогою порівняння, наслідку, зв'язку, судження та усвідомлення, що включає збір інформації, аналіз, синтез, обмін та використання.

Основними властивостями інформації називають:

• Повноту – достатність для прийняття рішення;

• Достовірність – відповідність реальному стану справ, яка знижується наявністю інформаційного шуму, недосконалими методами передавання та інтерпретації, інформаційними втратами та спотвореннями при зберіганні та обробці, тощо;

• Цінність та корисність – для конкретної категорії користувачів, що оцінюється за тим, наприклад, наскільки інформація підвищує імовірність досягнення поставлених цілей, або за матеріальним ефектом від використання інформації;

• Адекватність = повнота+достовірність;

• Актуальність – відповідність теперішньому моменту часу;

• Об'єктивність – незалежність від джерела та приймача;

• Доступність – складається з доступності даних та доступності методів;

• Надлишковість – фактично надлишковість даних – допускає зменшення обсягу повідомлень при збереженні їхнього абстрактного значення.

Знання можна поділити на такі типи:

1) причини, цілі (бачення), що дають відповідь на питання «чому?», а також дають підстави для структурування проблем та прагнення до досягнення успіху;

2) предмет знання (факти, концепції, теорії, конструкції), що дають відповідь на питання «що?»;

3) алгоритми (процедури, методи, ноу-хау, технології, уміння виконати на практиці), що дають відповідь на питання «як зробити?»;

4) альтернативи (варіанти, нюанси), що дають відповідь на питання «хто?», «де?», «коли?», «в яких умовах?».

Для знань важливими властивостями є:

• Структурованість;

• Зручність для доступу та засвоєння;

• Лаконічність;

• Несуперечливість;

• Наявність процедур обробки знань.

Явні знання (кодифіковані та формалізовані) виражаються у словах, цифрах, знаках, формулах, схемах, образах і т.д. Їх легко передавати та поширювати, вони належать усьому людству та впливають на продуктивну діяльність.

Люди у процесі мислення та практичної діяльності в основному оперують неявними знаннями, що знаходяться у їхній свідомості.

У теорії інформації – науці, заснованій Клодом Шенноном у 1948 році, що вивчає кількісні закономірності, пов'язані з отриманням, передаванням, обробкою та зберіганням інформації, – однією з основних задач є стиснення даних, тобто пошук найбільш ефективних способів кодування інформації (фактично, даних). Тому необхідно вміти кількісно вимірювати обсяг даних, що зберігаються або передаються.

Для цього введено поняття ентропії, як універсальної міри різноманітності системи та невизначеності наших знань про її можливі стани (К. Шеннон, У. Уівер).

Ентропія – це міра кількості інформації, що виробляється джерелом, пропускається каналом або потрапляє до користувача.

Інтелектуальним аналізом даних (Data Mining) називають процес визначення нових, коректних та потенційно корисних знань на основі великих масивів даних. «Інтелектуальний аналіз даних» деякі дослідники вважають синонімом ще одного популярного терміна – виявлення знань у даних – "Knowledge Discovery in Databases" (KDD), на думку інших – інтелектуальний аналіз даних є лише важливим кроком у процесі виявлення знань.

Виявлені в результаті інтелектуального аналізу знання називають патерном (зразком). Тобто задача інтелектуального аналізу полягає в ефективному виявленні осмислених патернів з наявного масиву даних великого розміру.

Отримані знання мають бути цікавими.

Ознаками цікавих знань є

− несподіваність – отримані знання мають дивувати (бути нетривіальними) та нести нову інформацію;

− застосовність – нові знання мають бути придатними до застосування для досягнення поставлених цілей, або до формулювання на їхній основі нових корисних цілей.

Аналіз даних є природним результатом еволюції інформаційних технологій. Розвиток баз даних та технологій керування (менеджменту) потребує також розвитку технологій збору та зберігання даних, керування та аналізу даних, оскільки серйозною проблемою інформаційного суспільства залишається

«суперечність між збільшенням обсягів інформації та зменшенням темпів зростання обсягів істинних знань».

Етапами інтелектуального аналізу є

1. Вивчення предметної області.

2. Збір даних.

3. Попередня обробка даних.

а) очищення даних – виключення «шумів» та суперечностей;

б) інтеграція даних – об'єднання даних з різних джерел в одному сховищі;

в) перетворення даних до підходящої форми (агрегація, стиснення, скорочення розмірності, дискретизація атрибутів, тощо).

4. Аналіз даних з метою виявлення патернів.

5. Інтерпретація знайдених патернів (візуалізація, відбір корисних патернів відповідно до функції корисності).

6. Використання нових знань.

Приклади сфер застосування інтелектуального аналізу великих даних:

1. електронні бібліотеки;

2. архіви зображень;

3. бази даних геномних досліджень;

4. медичні зображення;

5. фінансові дані;

6. бази даних підприємств;

7. телекомунікаційні системи;

8. всесвітня павутина;

9. біометричні дані людей…

Основними напрямками (аспектами) методології аналізу даних є:

1. добування різних видів нових знань;

2. добування знань у багатовимірному просторі (аналіз у просторі куба даних може перевищувати потужність та гнучкість інтелектуального аналізу, особливо з урахуванням міждисциплінарного характеру досліджень);

3. розширення можливостей пошуку у мережевому середовищі;

4. обробка невизначеностей та шумів або неповних даних;

5. оцінювання отриманих в результаті аналізу шаблонів.

Відповідно, основними типами задач аналізу даних є задачі опису та задачі прогнозування, тобто:

• Класифікація – процес знаходження моделей чи функцій, які описують та розрізняють класи для прогнозування класу довільно заданого об'єкта з відомими атрибутами на основі навчаючої вибірки;

• Кластеризація – виявлення ознак, за якими можна буде здійснювати класифікацію, шляхом групування “схожих” між собою об’єктів, генерування міток класів на основі відстаней між об’єктами;

• Регресія – встановлення залежностей неперервних результуючих змінних від вихідних;

• Асоціація – пошук закономірностей між декількома подіями, що відбуваються одночасно;

• Послідовність – пошук часових закономірностей між транзакціями;

• Прогнозування – оцінювання пропущених або майбутніх значень цільових чисельних показників;

• Виявлення відхилень або викидів;

• Оцінювання – передбачення неперервних значень ознаки;

• Аналіз зв’язків – пошук залежностей у наборі даних;

• Візуалізація – створення графічного образу аналізованих даних;

• Підведення підсумків – опис конкретних груп об’єктів з аналізованого набору;

• Еволюційний аналіз – опис та моделювання регулярностей та трендів для об’єктів, чия поведінка змінюється у часі.

До глибокого аналізу даних відносять групи задач, розв’язування яких передбачає

− відтворення «портрета» об’єкта у середовищі, тобто виведення моделі, яка «прозоро» інтерпретується і пояснює функціонування об’єкта;

− виявлення структури в даних, наприклад, ідентифікація системи зв’язків та впливів між характеристиками об’єкта у середовищі;

− знаходження закономірностей поведінки системи (об’єкта) – регулярності, періодичності, інваріантів, пошук аномалій [1].

Основними методами аналізу даних є:

• Множинний регресійний аналіз;

• Дерева рішень;

• Дискримінантний аналіз;

• Кластерний аналіз;

• Факторний аналіз;

• Виведення правил асоціації;

• Нейронні мережі;

• Аналіз часових рядів;

• Генетичні алгоритми;

• Візуалізація даних;

• Нечітка логіка.

Лабораторна робота №1. Попередній статистичний аналіз

Візуалізація даних становить важливу складову аналізу, оскільки 80% інформації люди сприймають саме зором.

У статистиці візуалізація первинних даних здійснюється за допомогою гістограми, полігона частот, блочної діаграми, квартильних графіків, діаграми розсіювання тощо.

Нехай маємо деякий набір статистичних даних – результати статистичного дослідження, – причому для кожного об'єкта визначено два атрибути X та Y (рис. 1.1):

X	***Y***	***X***	***Y***	***X***	***Y***	***X***	***Y***
73	291	57	219	61	241	68	264
69	270	71	281	62	243	62	240
72	279	66	262	63	245	70	277
72	282	76	302	71	282	70	279
65	254	70	275	65	252	65	253
67	264	68	267	70	276	70	275
56	216	74	290	70	276	63	248
70	276	68	266	63	246	63	243
63	248	69	270	73	284	67	264
64	253	71	283	68	271	68	267
70	276	60	237	59	227	55	213
67	262	56	222	64	256	56	218
60	234	71	281	79	309	58	223
63	243	68	269	77	300	70	278
80	313	66	257	78	310	59	236

Рис. 1.1

Побудуємо *варіаційний ряд* розподілу однієї з ознак, наприклад, X, підрахувавши частоту появи кожного значення X для діапазону значень від min(X) до max(X) (рис. 1.2):

Значення	Частота	Значення	Частота	Значення	Частота	Значення	Частота	Значення	Частота
55	1	60	2	65	3	70	9	75	0
56	3	61	1	66	2	71	4	76	1
57	1	62	2	67	3	72	2	77	1
58	1	63	6	68	6	73	2	78	1
59	2	64	2	69	2	74	1	79	1
								80	1

Рис. 1.2.

Для побудови *інтервального ряду* слід визначити необхідну кількість та ширину класових інтервалів. У даному випадку візьмемо 6 класів шириною 5, тобто першим буде клас 51-55, а останнім – 76-80 (рис. 1.3):

Діапазон значень	51-55	56-60	61-65	66-70	71-75	76-80
Частота X	1	9	14	22	9	5

Рис. 1.3.

На рис. 1.4 представлено *гістограму* та *полігон* для варіаційного ряду, а на рис. 1.5 – для інтервального ряду.

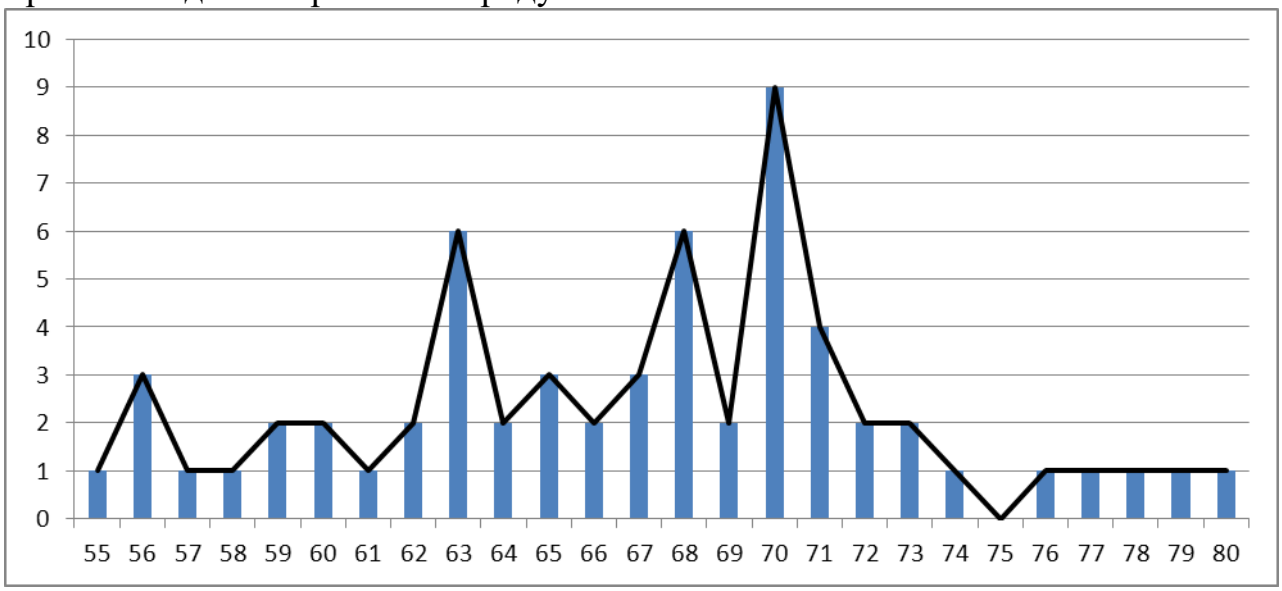


Рис. 1.4

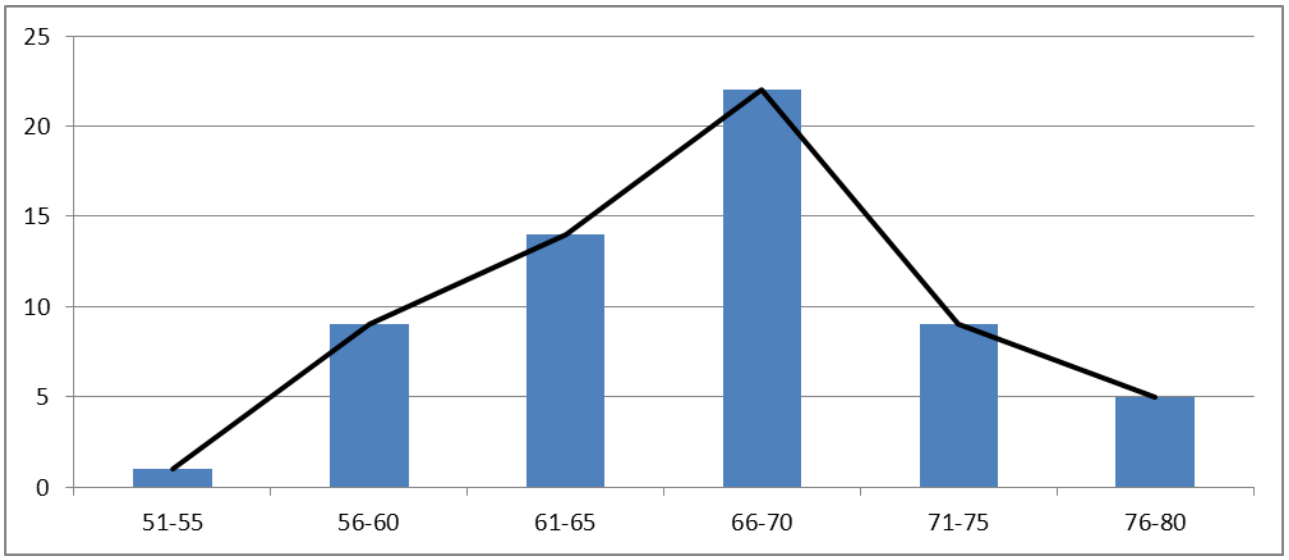


Рис. 1.5

Блочна діаграма (boxplot), розроблена у 70-ті роки Джоном Тьюки, – це графік, що компактно зображує одновимірний розподіл імовірностей. За блочною діаграмою зручно оцінювати медіани, квартилі, дисперсію та асиметрію розподілу, а також виявляти викиди. Для побудови блочної діаграми для змінної необхідно визначити 5 чисел: мінімальне та максимальне значення, перший, другий (медіану або середнє) та третій квартиль і стандартне квадратичне відхилення. В електронних таблицях для цього виконують функції МІН (діапазон), МАКС(діапазон), КВАРТИЛЬ (діапазон), МЕДІАНА (діапазон), КВАРТИЛЬ (діапазон), СТАНДОТКЛОН (діапазон).

Довжину «вусиків» визначають декількома способами:

1) як мінімальне та максимальне значення по вибірці (у такому разі викиди відсутні);

2) довжину першого з «вусиків» діаграми визначають як різницю першого квартиля та півтори міжквартильних відстані, а другого – як суму третього

квартиля та півтори міжквартильних відстані:

$$X_1=Q_1-k(Q_3-Q_1),\ X_2=Q_3+k(Q_3-Q_1),$$

де X_1 – нижня границя «вусика», X_2 – верхня границя «вусика», Q_1 – перший квартить, Q_3 – третій квартиль, k – коефіцієнт (зазвичай обирають k=1,5).

3) як середнє арифметичне (або медіану) по вибірці плюс/мінус одне стандартне квадратичне відхилення:

$$X_1 = \bar{X} - \sigma,\ X_2 = \bar{X} + \sigma \quad (\text{або } X_1 = Q_2 - \sigma,\ X_2 = Q_2 + \sigma).$$

Щоб побудувати таку блочну діаграму в Excel можна скористатись інструментом «Блокова діаграма» (рис. 1.6.).

Дані для графіка слід подати у такому порядку:

Перший квартиль (Q1)	Максимальне	Мінімальне	Третій квартиль (Q3)	Середнє
63	80	55	70	66,82

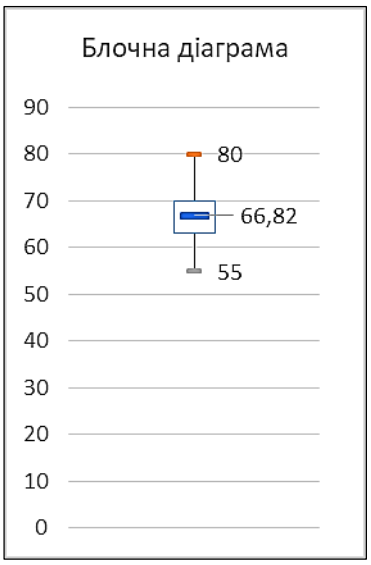


Рис. 1.6

Значення відповідних параметрів розраховано для змінної X (див. рис. 1.1). Рядки з даними слід декілька разів продублювати: для біржової діаграми кількість рядків має перевищувати кількість параметрів побудови, – а після побудови зайві діаграми можна не показувати. Стовбець із значеннями середнього додають окремо до вже побудованої діаграми та у переліку рядів даних переміщують на передостаннє місце, щоб не зменшити висоту блока, оскільки положення його верхньої планки визначається останнім параметром. Залишається для ряду «Середнє» встановити маркер та додати підписи даних.

Для побудови *квантиль-квантильного графіка* (QQ-plot) необхідно поставити у відповідність квантилям нормального розподілу квантилі емпіричного розподілу.

Для цього:

1) у першому стовпці розташуємо у порядку зростання (неспадання) усі значення масиву X – від 55 до 80, – усього 60 значень;

2) у другому стовпці запишемо проміжки розбиття одиничного інтервала, наприклад, на 20 рівних частин (назвемо цю змінну k);

3) у третьому стовпці обчислимо значення k-того квантиля для розподілу X, тобто визначимо, яке значення досліджуваної змінної поділяє усю вибірку у співвідношенні k/(1-k) за формулою ПЕРСЕНТИЛЬ(масив X; k);

4) в останньому стовпці слід обчислити відповідні квантилі нормального розподілу (формула НОРМСТОБР(k) або NORM.S.INV(k));

5) за даними двох останніх стовпців (рис. 1.7) будуємо точкову діаграму (діаграму розсіювання) (див. рис. 1.8).

X	K	PERCENTILE.EXC($A$2:$A$61;K)	NORM.S.INV(k)
55	0,05	56	-1,644853627
56	0,1	58,1	-1,281551566
56	0,15	60	-1,036433389
56	0,2	62	-0,841621234
57	0,25	63	-0,67448975
58	0,3	63	-0,524400513
59	0,35	64,35	-0,385320466
59	0,4	65,4	-0,253347103
60	0,45	67	-0,125661347
60	0,5	68	0
61	0,55	68	0,125661347
62	0,6	69	0,253347103
62	0,65	70	0,385320466
63	0,7	70	0,524400513
63	0,75	70	0,67448975
63	0,8	71	0,841621234
63	0,85	72	1,036433389
63	0,9	73,9	1,281551566
63	0,95	77,95	1,644853627
79			
80			

Рис. 1.7

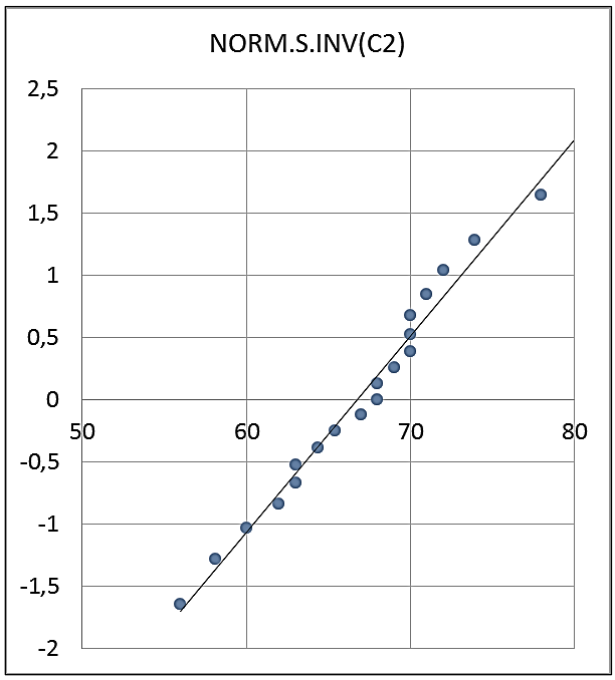


Рис. 1.8

QQ-plot використовують для первинного візуального порівняння емпіричного розподілу з нормальним: якщо точки побудованого графіка добре вкладаються на пряму, то можна припустити, що емпіричний розподіл близький до нормального.

Діаграму розсіювання (Scatter plot) також застосовують для візуалізації статистичних зв'язків між змінними. Значення змінних X та Y задають координати точок на площині. Отже для набору даних (рис. 1.1) матимемо таку діаграму (рис. 1.9):

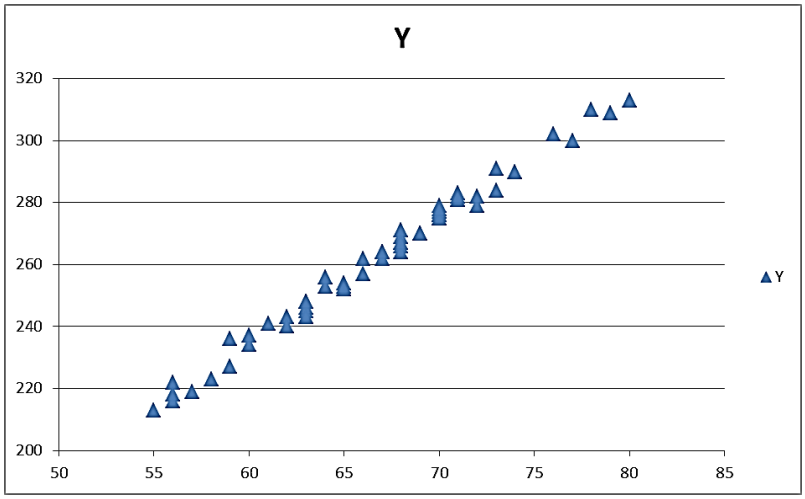


Рис. 1.9

Іншими видами статистичних графіків є кругова та пелюсткова діаграми тощо.

Завдання

1. Ознайомтеся з цікавими для Вас даними на сайті https://www.google.com/publicdata/. Розгляньте різні способи візуалізації цих даних. Додайте до звіту ваші міркування (інтерпретацію) стосовно розглянутих даних.
2. Знайдіть цікаві для Вас статистичні дані. Виконайте статистичний аналіз цих даних (див. пункт 3).
3. Виконайте процедури описової статистики у пакеті MS Excel:

   а) обчисліть середнє, дисперсію, стандартне квадратичне відхилення, мінімальне та максимальне значення, моду та медіану за допомогою вбудованих функцій;
   б) застосуйте процедури описової статистики з Пакета Аналізу;
   в) побудуйте варіаційний ряд;
   г) побудуйте гістограму та полігон розподілу частот;
   д) побудуйте блочну діаграму.

Лабораторна робота №2. Ознайомлення з інтерфейсом пакетів RapidMiner та KNIME

2.1 Аналіз даних з пакетом RapidMiner

Пакет RapidMiner, раніше відомий як YALE (*у перекладі з англ.:* "*Ще одне навчальне середовище"*), почали розробляти у 2001 році Ральф Клінкенберг, Інго Мірсва та Симон Фішер у відділі штучного інтелекту Дортмундського технічного університету. У 2006 році Ральф Клінкенберг та Інго Мірсва заснували компанію Rapid-I. Відповідно, у 2007 році назву програмного продукту було змінено з YALE на RapidMiner. У 2013 році компанію також було перейменовано на RapidMiner.

Зараз RapidMiner (RM) – це програмна платформа, яка забезпечує інтегроване середовище для підготовки даних, машинного навчання, глибокого навчання, виведення тексту та прогнозної аналітики (рис. 2.1). Вона використовується для бізнесу і комерційних застосувань, а також для досліджень, освіти, підготовки кадрів, швидкого створення прототипів та розробки додатків, а також підтримує всі етапи процесу машинного навчання, включаючи підготовку даних, візуалізацію результатів, перевірку моделей та оптимізацію моделей. RapidMiner Studio Free Edition, яка обмежується одним логічним процесором та 10 000 рядків даних, доступна під ліцензією AGPL. Детальну інформацію з установки продукту розміщено на сайті компанії: https://rapidminer.com/products/studio/

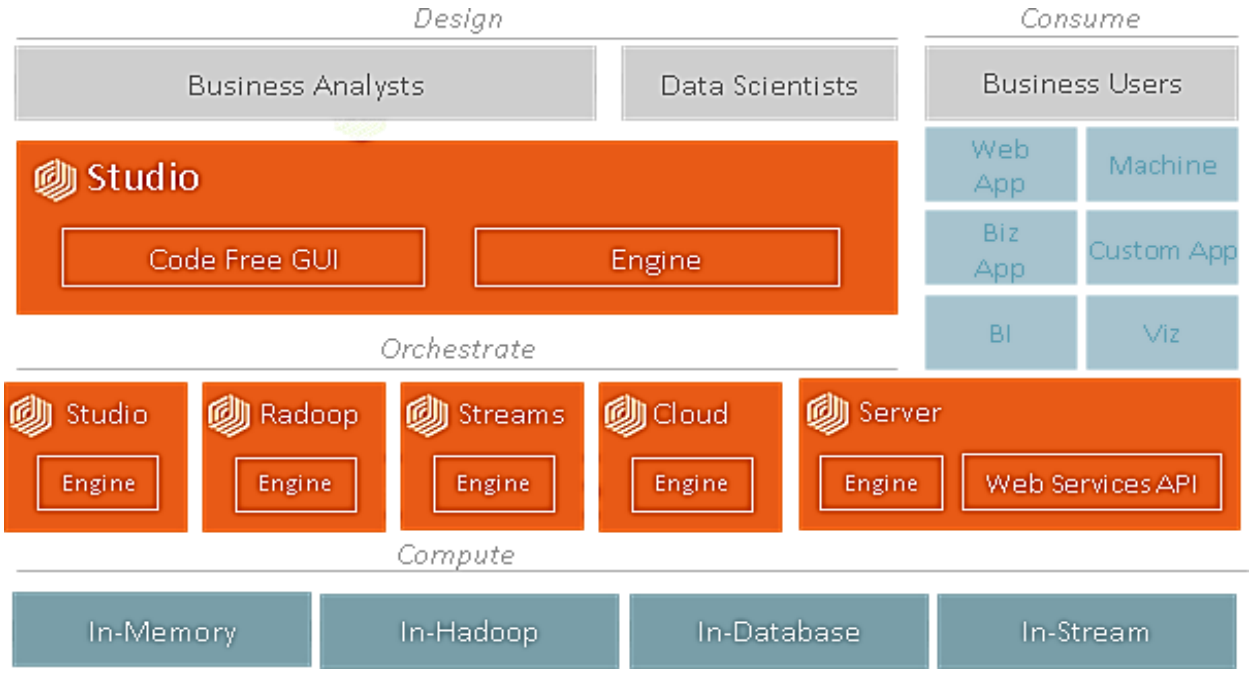


Рис. 2.1 Архітектура пакета RapidMiner

Основні поняття:

Основними об'єктами в RM є процес, оператор та репозиторій.

Процес – це сукупність операторів, з'єднаних між собою у заданому порядку для виконання потрібної задачі аналізу або обробки даних (рис. 2.2).

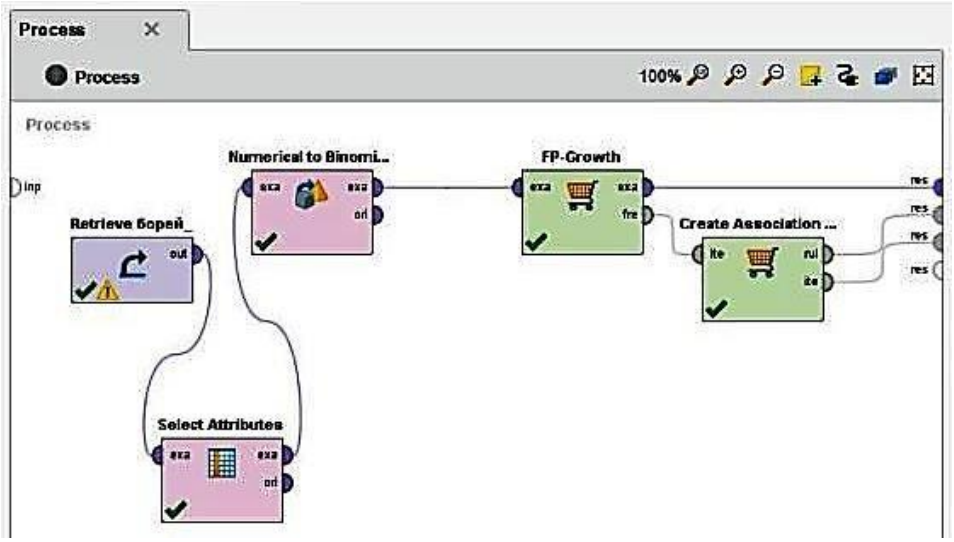


Рис. 2.2

Оператор – логічна одиниця процесу. Оператор виконує дії над даними, у нього є вхід-вихід (так звані «порти»). На вхід дані приходять з попереднього оператора або первинного джерела даних, а на вихід передаються результати виконання. Таким чином можна створювати ланцюжки опрацювання даних, в тому числі і розпаралелювані.

В інтерфейсі програми операторам відповідає вкладка ***Operators***, де вони згруповані за функціональністю. Для використання оператора його необхідно перенести в робочу область процесу.

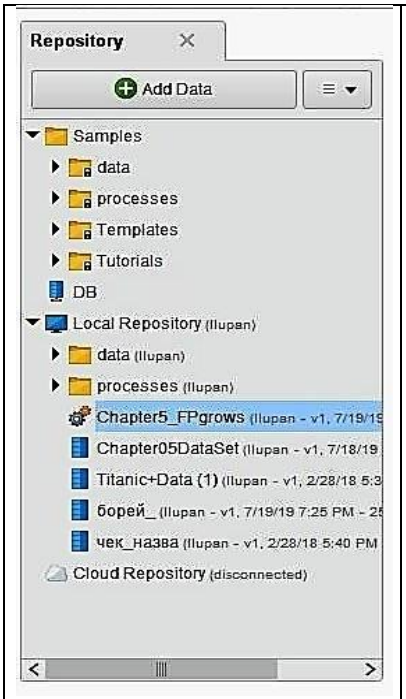


Рис. 2.3

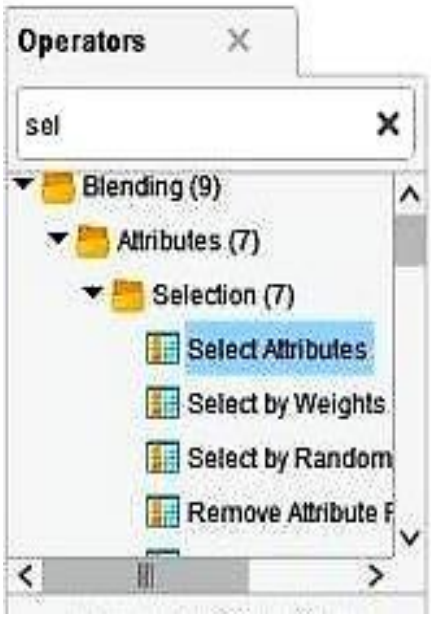


Рис. 2.4

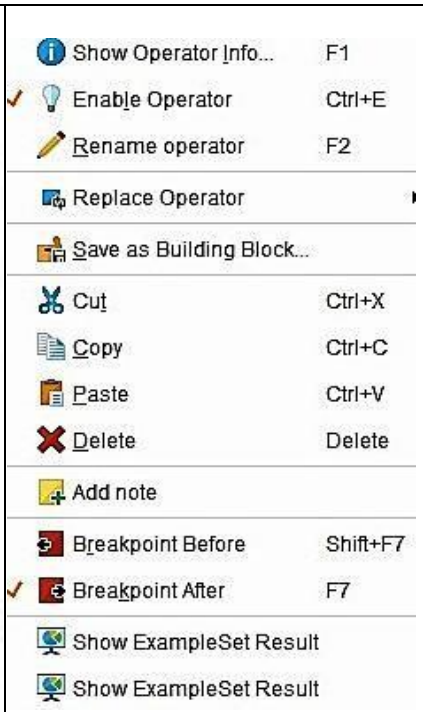


Рис. 2.5

За початковими літерами назви оператора його швидко можна знайти (рис. 2.4).

Репозиторій – місце для зберігання процесів RM та даних, – може бути локальним, а також віддаленим (RapidMiner Server), для якого можливе виконання процесів на боці сервера і т.п. (рис. 2.3).

У вкладці **Repositories** в RM можна побачити набір прикладів (*Samples*), поточні з'єднання з базами даних, визначені через Tools → Manage Database Connections (*DB*) та місце для зберігання власних процесів на комп'ютері (*Local Repository*).

Процес може бути перенесений у вікно **Process** як один оператор (перетягуванням піктограми) або розгорнутий у вигляді послідовності операторів (подвійним «кліком»).

На рис. 2.6 бачимо виділений оператор **Execute**, який виконуватиме процес агломеративного кластерного аналізу (04_AgglomerativeHierarchicalClustering), а вище – той самий процес кластерного аналізу у розгорнутому вигляді як послідовність операторів Retrieve (отримати дані) та **AgglomerativeClustering.**

Для виділеного у вікні **Process** процеса або оператора можна за допомогою розташованого справа вікна **Parameters** переглянути та налаштувати параметри.

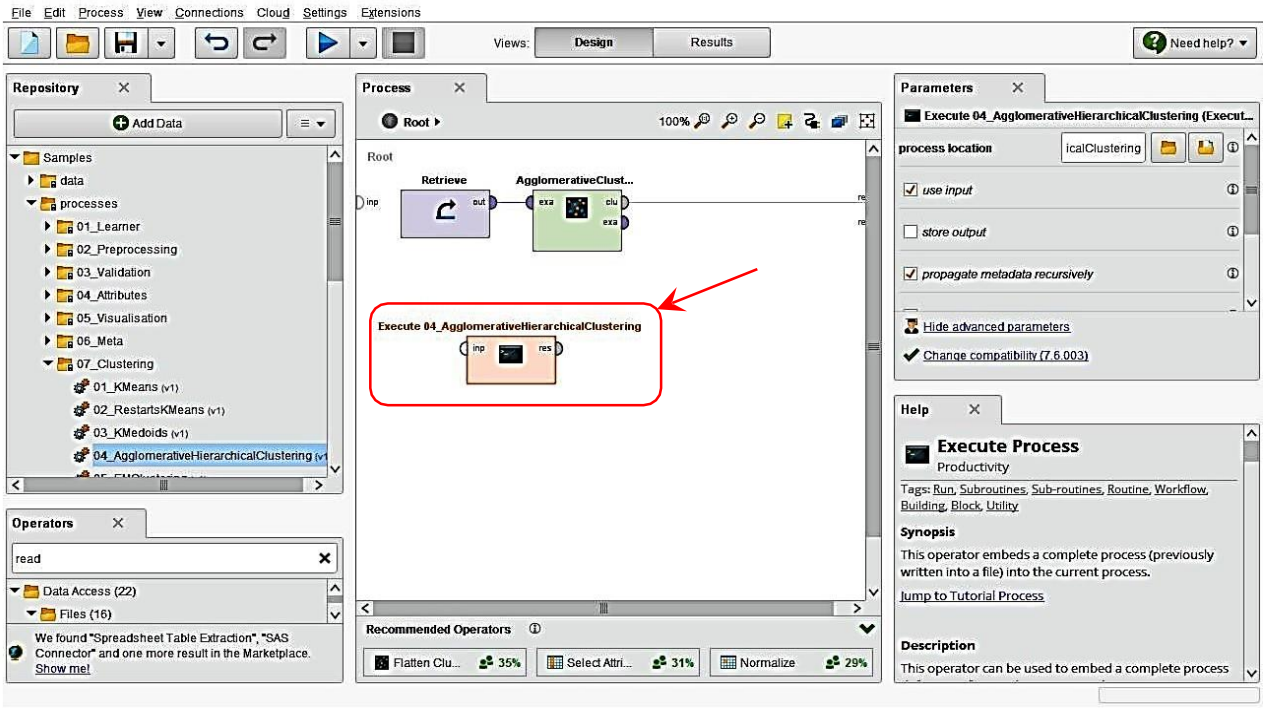


Рис. 2.6

Запуск процеса на виконання здійснюється кнопкою  або клавішею F11. У режимі **Design** здійснюється налаштування процеса в цілому та окремих операторів. Для перегляду результатів слід перейти у режим **Result**.

Початок роботи. Перегляд вхідних даних.

Першою дією в аналізі є перегляд статистичних даних та здійснення процедур описової статистики. Дані можна завантажити до репозиторію, застосувавши команду **Add data** у вікні **Repository** та вказавши у відповідному діалоговому вікні місце знаходження даних (відповідну папку). В результаті дані будуть переміщені до відповідного репозиторію і зберігатимуться у спеціальному форматі. При цьому їх можна буде редагувати, зокрема, вилучати стовпці, змінювати типи та ролі атрибутів.

При перетягуванні піктограми, що відповідає набору даних, з вікна репозиторію до вікна процесу автоматично створюється оператор **Retrieve**.

В RM також можна використовувати дані у різних форматах без переміщення до репозиторія. У такому разі використовують оператор **Read**. Якщо форма представлення цих даних не відповідає цільовому оператору обробки даних, то доведеться застосовувати додатково оператори попередньої обробки.

Далі слід з'єднати кружечок ***out*** оператора **Retrieve Data** (або іншого, за допомогою якого отримано доступ до даних) з кружечком ***res*** на правому боці вікна **Process**. На цьому дизайн процесу завершено, процес можна запустити на виконання (**Run**), а результати переглянути у режимі **Results**.

На рис. 2.7 для даних з репозиторію за допомогою контекстного меню відкрито **Data Editor**, у якому для будь-якого із стовпців дозволено виконати вилучення, змінення типу та заповнення випадковими даними чи константами. Між тим для того, щоб вилучити стовпець із даних, отриманих у форматі SPSS, довелося застосовувати оператор **Remove Attribute Range** з параметром, що задає діапазон стовпців, які слід вилучити.

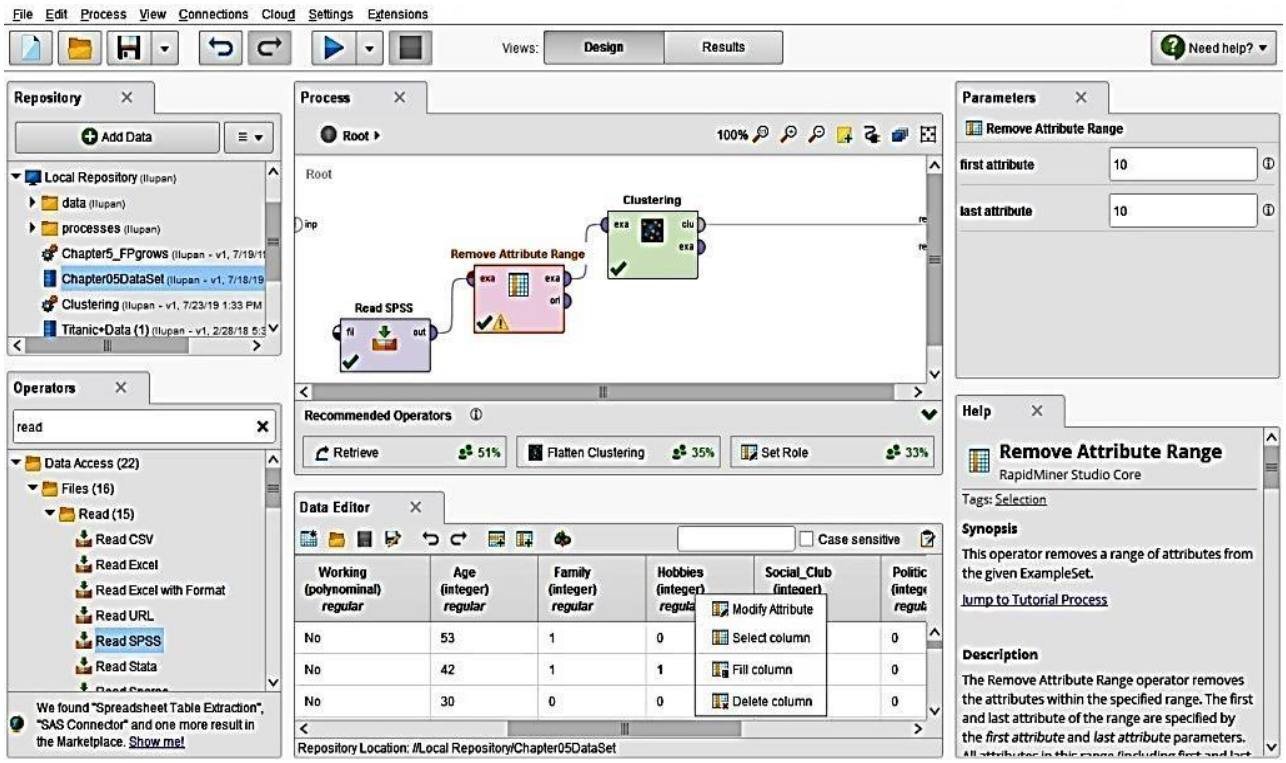
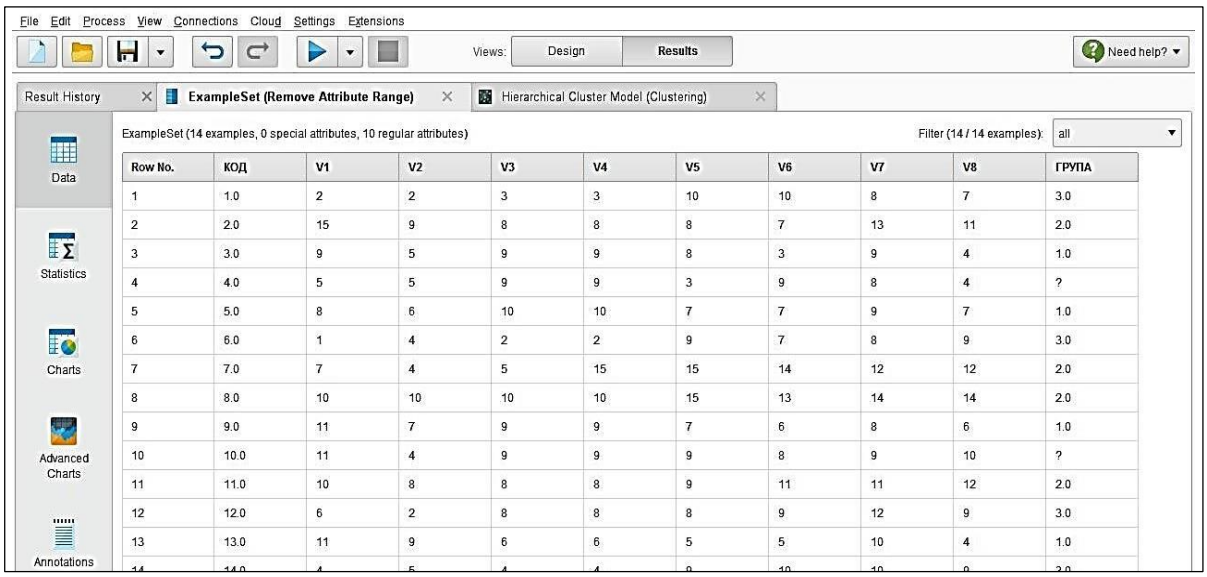


Рис. 2.7

Результатами для процесу перегляду даних є Data – власне таблиця даних (рис. 2.8); Statistics – короткі статистичні дані по кожній змінній – тип даних, мінімальне та максимальне значення, середнє та стандартне квадратичне відхилення для числових змінних, гістограму розподілу (рис. 2.9). У закладці Charts – надається можливість побудувати необхідні графіки (рис. 2.10).



Row No.	КОД	V1	V2	V3	V4	V5	V6	V7	V8	ГРУПА
1	1.0	2	2	3	3	10	10	8	7	3.0
2	2.0	15	9	8	8	8	7	13	11	2.0
3	3.0	9	5	9	9	8	3	9	4	1.0
4	4.0	5	5	9	9	3	9	8	4	?
5	5.0	8	6	10	10	7	7	9	7	1.0
6	6.0	1	4	2	2	9	7	8	9	3.0
7	7.0	7	4	5	15	15	14	12	12	2.0
8	8.0	10	10	10	10	15	13	14	14	2.0
9	9.0	11	7	9	9	7	6	8	6	1.0
10	10.0	11	4	9	9	9	8	9	10	?
11	11.0	10	8	8	8	9	11	11	12	2.0
12	12.0	6	2	8	8	8	9	12	9	3.0
13	13.0	11	9	6	6	5	5	10	4	1.0

Рис. 2.8

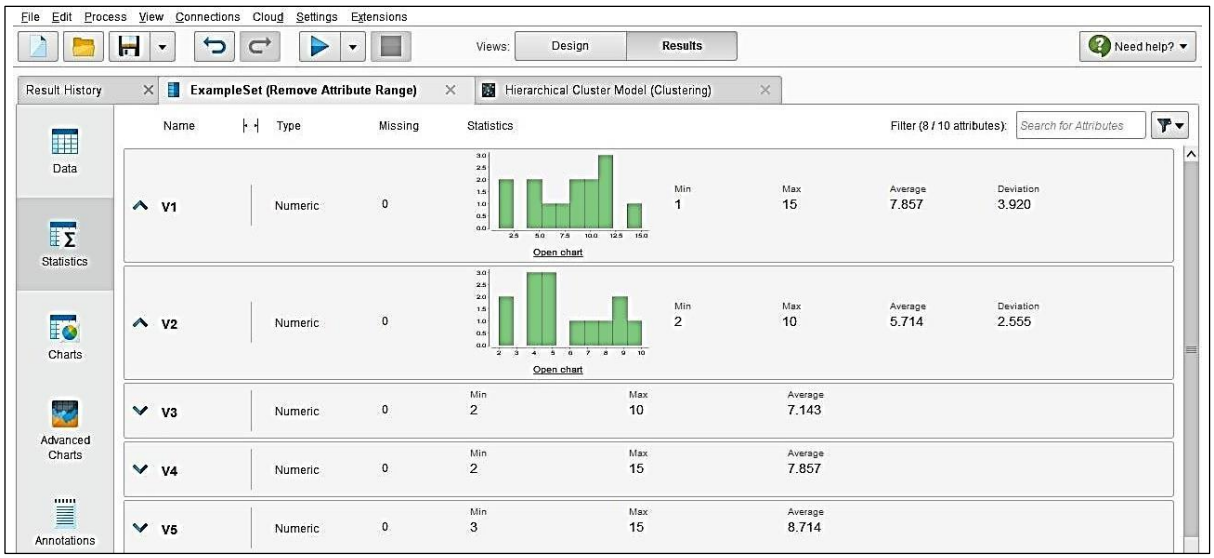


Рис. 2.9

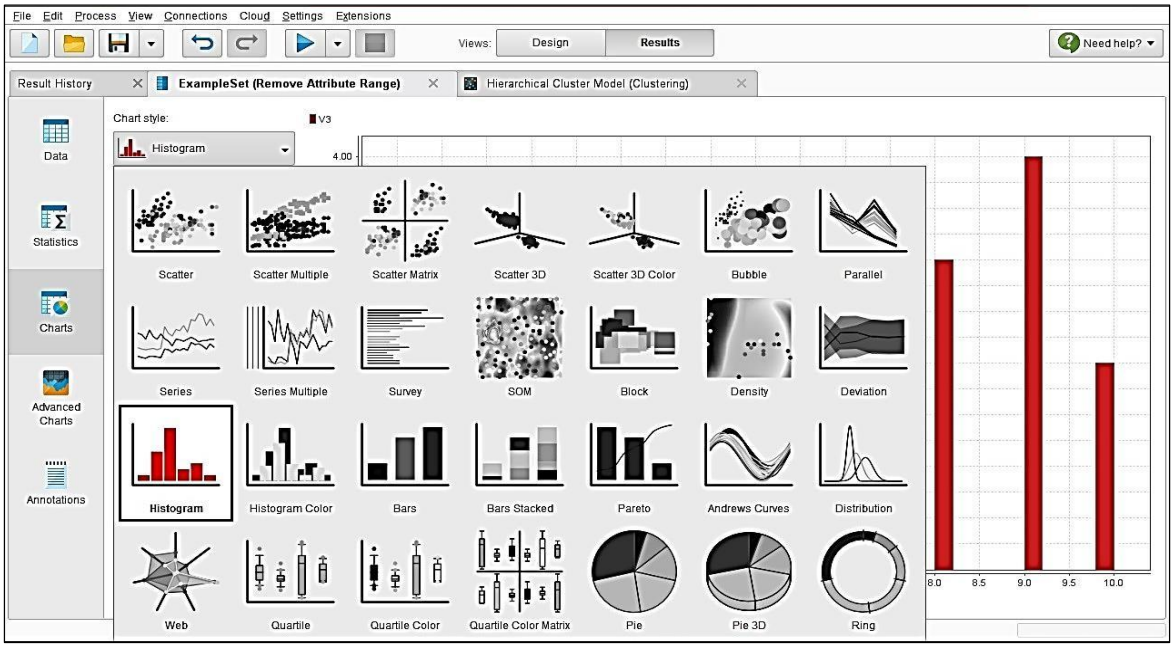


Рис. 2.10

Таким чином можна переглянути набори даних, що надаються разом з

пакетом RapidMiner у навчальних прикладах.

Щоб переглянути проміжні результати виконання окремих операторів складного процесу, після його запуску можна за допомогою контекстного меню до вибраного оператора застосувати команду **Show ExamplesSet Result** (рис. 2.5). Причому в одному випадку отримаємо вигляд даних до, а в іншому – після виконання оператора.

Також є можливість призупинити виконання процеса для перегляду проміжних результатів, встановивши контрольні точки до або після виконання вибраного оператора (**Breackpoint**).

2.2. Основи роботи з пакетом KNIME

KNIME – вільно поширювана аналітична платформа з широким набором засобів інтелектуального аналізу даних (доступу до даних різноманітних форматів, трансформації даних, аналітичних функцій, засобів візуалізації та підготовки звітів).

Процес аналізу, тобто перетворення вхідних даних у результати, в KNIME представлено потоком робіт (workflow), який графічно зображено в основному вікні програми (рис. 2.11):

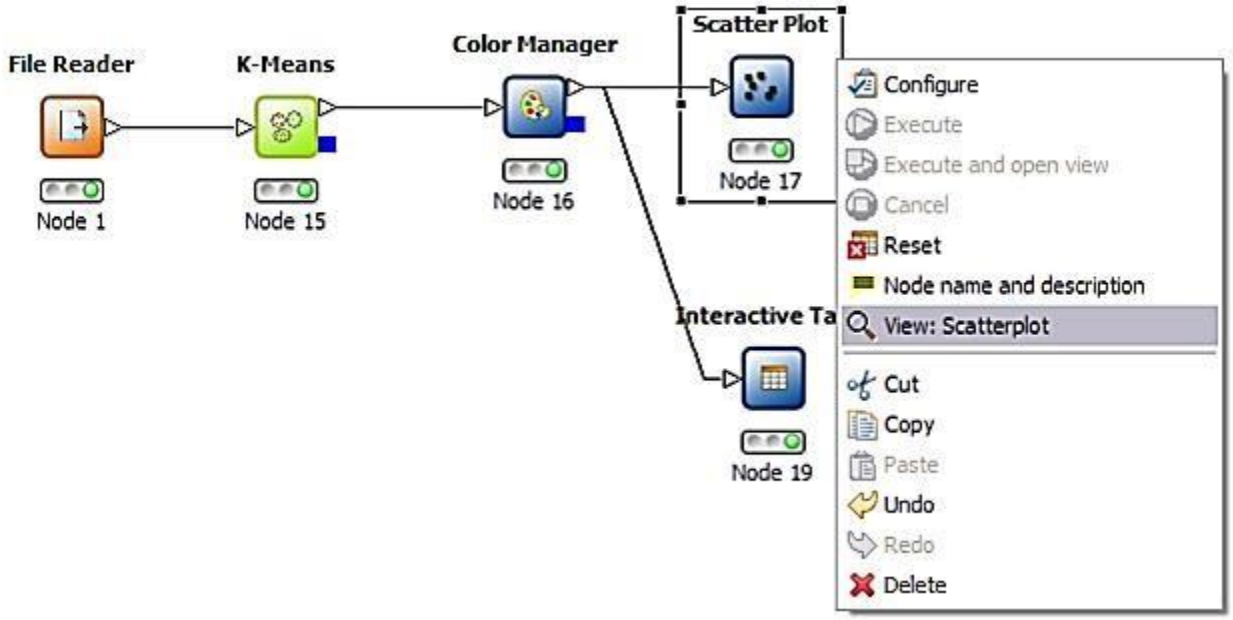


Рис. 2.11. Простий потік робіт

Зображення складається з вузлів (nodes), які інкапсулюють операції над даними, та стрілок, що з'єднують «порти» вузлів у напрямку обміну даними між операціями. Кожен вузол має бути налаштований (відконфігурований – Configure) відповідним чином. Вузол також можна знищити, скопіювати, переіменувати, використовуючи контекстне меню, або переглянути результати його виконання.

Стан виконання тої чи іншої операції або її готовність до виконання (node status) візуалізується індикатором (рис. 1.12):

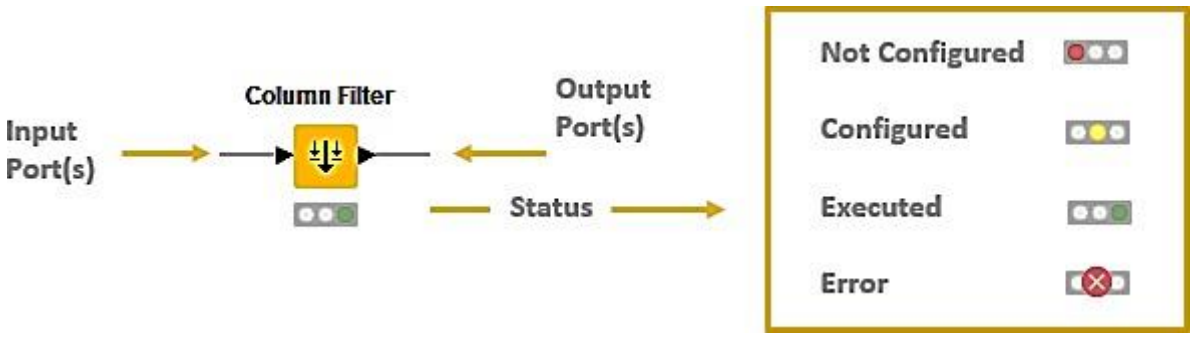


Рис. 2.12

Рекомендації до початку роботи можна отримати на сайті розробника: https://www.knime.com/getting-started.

Завдання

1. Розгляньте декілька наборів даних з прикладів, представлених у репозиторії пакета RapidMiner або KNIME.
2. Для обраних наборів (див. лаб. роб. №1) створіть процеси для перегляду та отримайте результати (описову статистику та графіки).
3. З'ясуйте, у які формати можна експортувати результати та з яких форматів дані можна отримати (імпортувати). Проаналізуйте та зробіть висновки.

Лабораторна робота №3. Дискримінантний аналіз

Виконання дискримінантного аналізу у пакеті KNIME

Виконаємо дискримінантний аналіз для тих даних, які були використані у кластерному аналізі. У прикладі нам довелося взяти файл з більшою кількістю даних, оскільки у пакеті KNIME є обмеження: розмір найменшої групи об'єктів має бути більшим за кількість дискримінантних змінних, тобто $n_i \geq p$. З цією ж метою були змінені налаштування ієрархічного кластерного аналізу: замість евклідової відстані (*Euclidean*) використано манхеттенську (*Manhattan*) та встановлено 3 кластери для виведення (у вихідній таблиці дані буде розподілено на три кластери-групи). Використані дані наведено у таблиця 3.1.

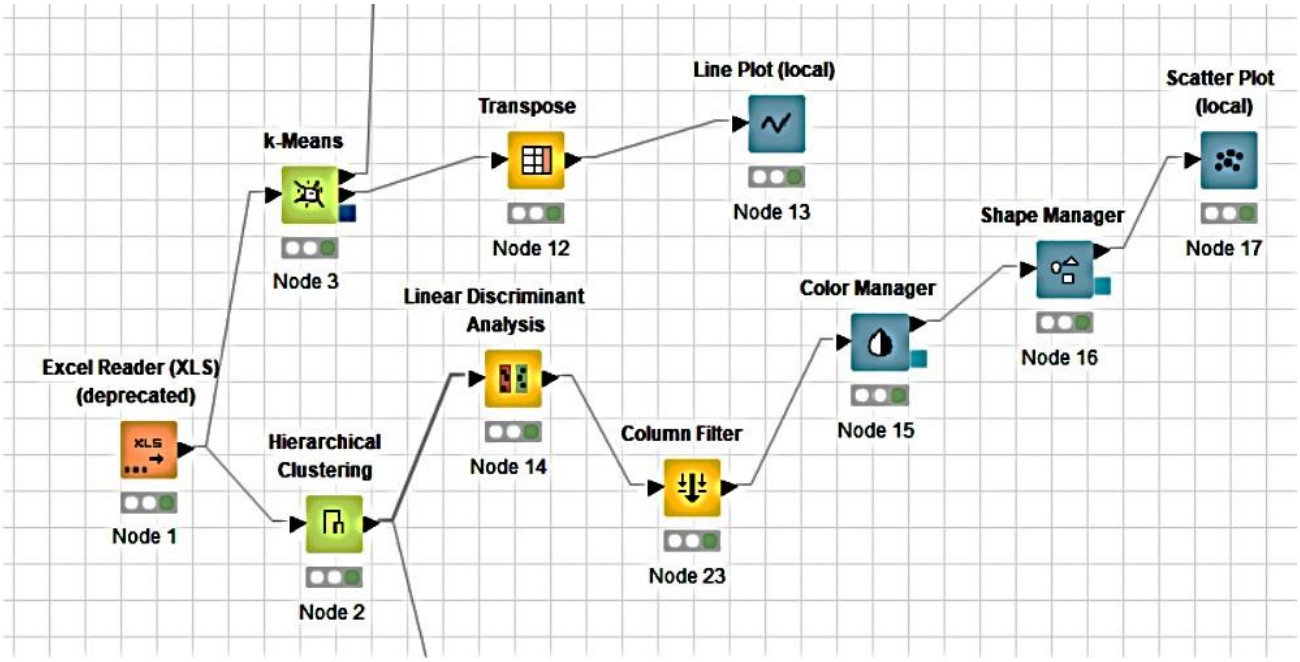


Рис. 3.1

Для виконання дискримінантного аналізу доповнимо створений раніше проект новими вузлами, як показано на рис. 3.1 та встановимо налаштування: вкажемо цільову розмірність – 2 (кількість результуючих канонічних дискримінантних функцій має бути на одиницю меншою за кількість визначених груп об'єктів), та оберемо групуючу змінну (номінативну; у даному випадку вона одна – це приналежність до кластеру).

В результаті виконання кластерного аналізу отримаємо таку дендрограму (рис. 3.2):

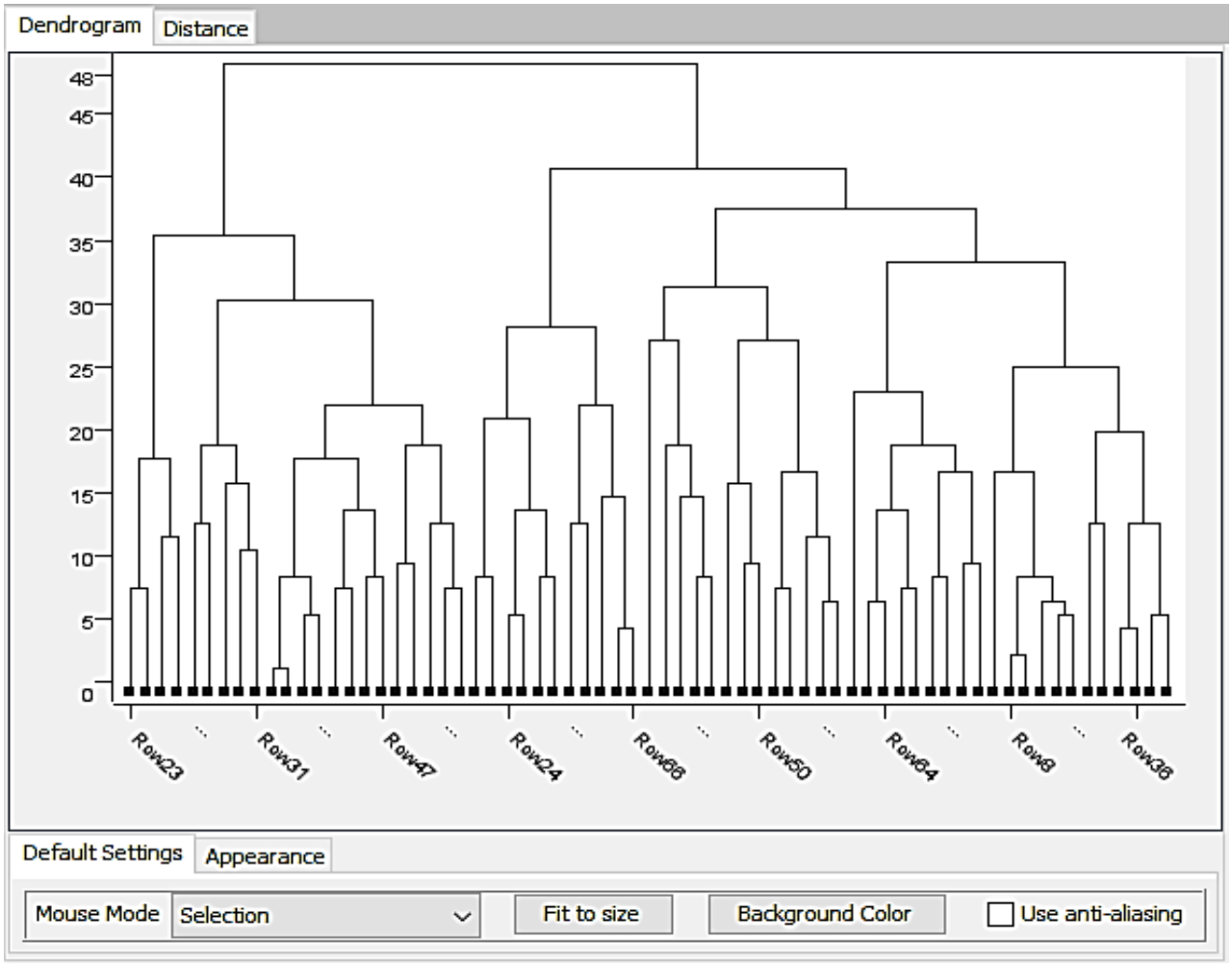


Рис. 3.2

Допоміжні вузли Color Manager та Shape Manager потрібні для налаштування графічного подання об’єктів різних груп на діаграмі розсіювання, а вузол Column Filter – для того, щоб відсіяти зайві змінні (слід залишити тільки отримані канонічні дискримінантні функції). Сама діаграма буде мати такий вигляд як показано на рис. 3.3.

Як бачимо, об’єкти, що належать різним кластерам досить непогано розрізняються, хоча границі кластерів не надто чіткі і мають перетини.

У пакеті SPSS для тих самих даних діаграма розсіювання матиме такий самий вигляд (рис. 3.4).

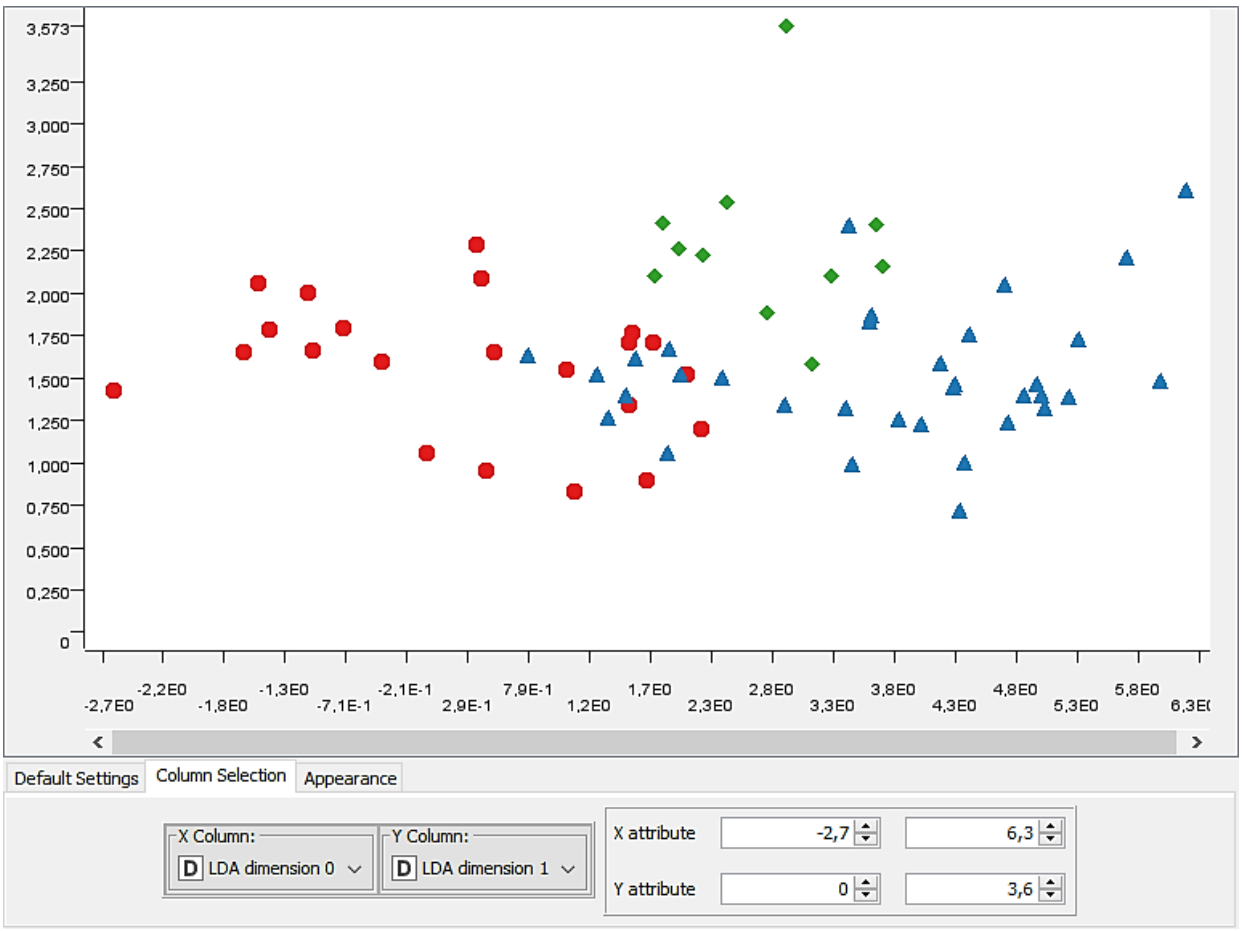

3,573
3,250
3,000
2,750
2,500
2,250
2,000
1,750
1,500
1,250
1,000
0,750
0,500
0,250
0
-2,7E0
-2,2E0
-1,8E0
-1,3E0
-7,1E-1
-2,1E-1
2,9E-1
7,9E-1
1,2E0
1,7E0
2,3E0
2,8E0
3,3E0
3,8E0
4,3E0
4,8E0
5,3E0
5,8E0
6,3E(
Default Settings
Column Selection
Appearance
X Column:
D LDA dimension 0
Y Column:
D LDA dimension 1
X attribute
-2,7
6,3
Y attribute
0
3,6

Рис. 3.3

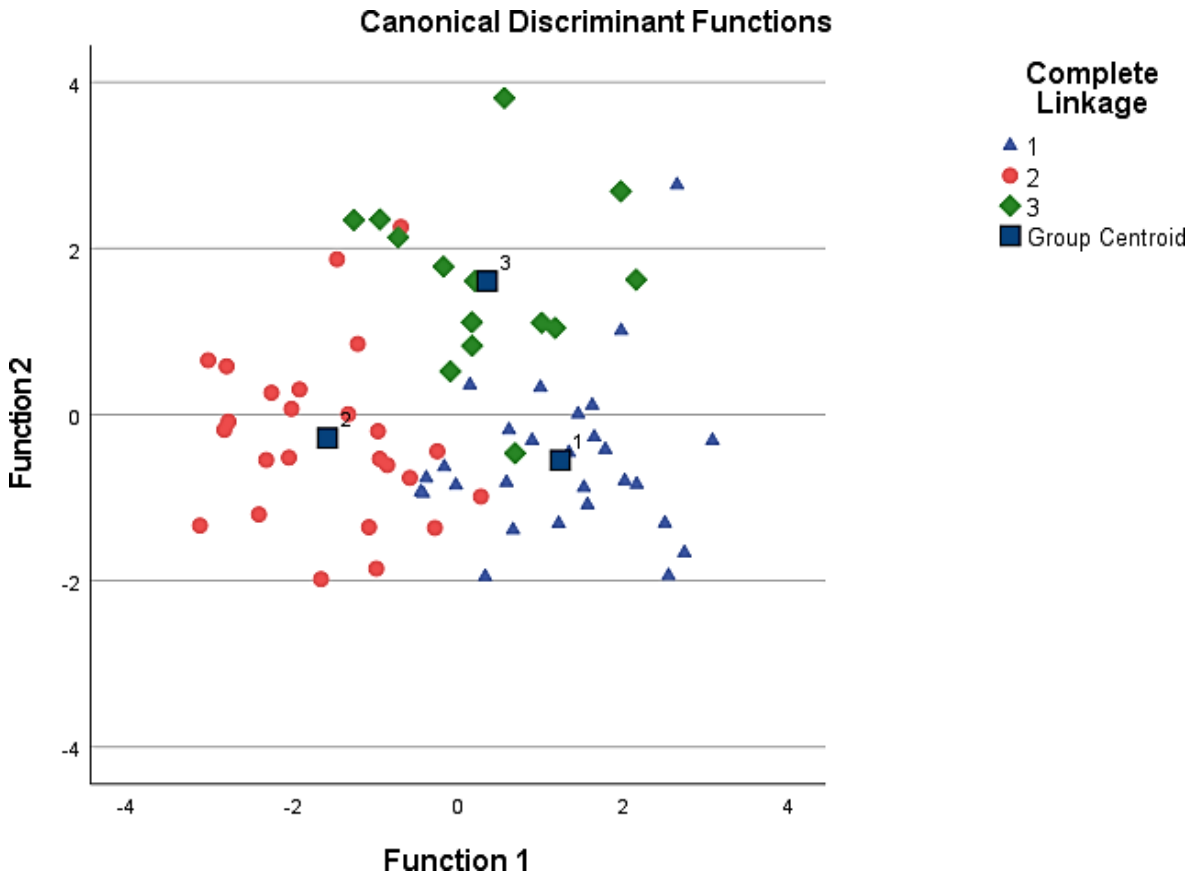

Canonical Discriminant Functions
Complete Linkage
1
2
3
Group Centroid
Function 2
Function 1
4
2
0
-2
-4
-4
-2
0
2
4

Рис. 3.4

Однак, крім діаграми SPSS дає таблиці коефіцієнтів канонічних дискримінантних функцій, структурну матрицю, яка показує кореляцію кожної з вхідних змінних з утвореними канонічними, координати центроїдів:

Standardized Canonical Discriminant Function Coefficients

Function	1	2
лідерство	,230	-,046
впевненість	-,164	,795
вимогливість	-,731	-1,304
скептицизм	,466	,962
поступливість	,742	-,412
довірливість	,104	,859
добросердя	,298	,035
чуйність	,264	,118

Structure Matrix

	Function 1	2
поступливість	,777*	-,086
добросердя	,503*	,106
вимогливість	-,368*	-,083
скептицизм	-,365*	-,048
чуйність	,286*	-,087
довірливість	,407	,562*
впевненість	-,260	,548*
лідерство	,061	,208*

Pooled within-groups correlations between discriminating variables and standardized canonical discriminant functions
Variables ordered by absolute size of correlation within function.
*. Largest absolute correlation between each variable and any discriminant function

Functions at Group Centroids

Complete Linkage	Function 1	2
1	1,236	-,551
2	-1,581	-,283
3	,351	1,606

Unstandardized canonical discriminant functions evaluated at group means

Цікавим результатом дискримінантного аналізу в SPSS є таблиця результатів класифікації, у якій підраховано відсоток елементів первинних груп, які залишаться в них згідно із обчисленими канонічними функціями. По ній видно, що тільки 86,6% об'єктів залишилося у кластерах, визначених кластерним аналізом.

Classification Results[a]

		Complete Linkage	Predicted Group Membership 1	2	3	Total
Original	Count	1	23	3	2	28
		2	1	22	2	25
		3	1	0	13	14
	%	1	82,1	10,7	7,1	100,0
		2	4,0	88,0	8,0	100,0
		3	7,1	,0	92,9	100,0

a. **86,6%** of original grouped cases correctly classified.

Дані, використані у прикладі

Таблиця 3.1

Код	лідерство	впевне-ність	вимогли-вість	скепти-цизм	поступли-вість	довірли-вість	добро-сердя	чуйність
10001	9	5	10	10	7	7	7	8
10003	8	5	10	10	12	8	12	11
10004	10	6	7	7	4	5	8	9
10005	10	9	6	6	4	7	9	10
10006	2	7	8	8	8	7	9	12
10007	12	10	4	4	11	10	10	11
10009	5	8	11	11	4	9	9	12
10010	10	4	5	5	10	5	12	13
10011	9	5	9	9	6	6	10	10
10013	12	5	8	8	4	5	9	10
10015	6	5	4	4	8	5	7	10
10016	7	5	4	4	7	7	10	11
10017	8	9	11	11	3	4	9	3
10018	9	7	5	5	6	7	8	11
10020	7	3	6	6	8	7	13	9
10022	9	8	6	6	10	8	11	11
10023	8	4	6	6	2	7	5	6
10024	5	4	9	9	8	7	7	7
10025	6	7	7	7	12	12	10	8
10027	4	10	5	5	12	4	7	6
10028	9	6	8	8	6	6	7	5
10030	10	7	7	7	11	7	10	7
10032	13	7	8	8	6	8	12	13
10033	13	12	12	12	6	4	6	5
10034	10	7	6	6	7	4	9	13
10035	7	10	6	6	7	4	9	13
10037	9	7	8	8	7	8	11	3
10038	10	9	8	8	5	9	7	9
10039	5	4	9	9	6	5	9	5
10040	11	5	10	10	6	7	8	11
10041	4	6	7	7	4	3	6	4
10042	7	9	10	10	5	6	9	10
10043	7	6	8	8	9	6	7	8
10046	5	5	6	6	5	2	5	7
10047	6	5	7	7	6	6	8	11
10051	3	3	8	8	13	8	10	11
10052	7	3	6	6	9	8	12	11
10053	9	4	5	5	5	6	10	14
10054	12	10	12	12	3	5	6	6
10055	9	4	7	7	9	10	11	8
10056	11	10	6	8	7	10	12	0
10057	8	8	10	10	5	11	9	5
10059	5	10	9	9	5	3	7	8

10060	8	6	8	8	7	9	12	14
10061	7	2	7	7	10	10	7	10
10062	10	7	8	8	9	7	11	11
10065	11	4	8	8	3	4	8	11
10067	5	6	8	8	5	6	5	12
10068	9	5	9	9	6	6	8	9
10069	6	8	12	12	3	2	6	8
10073	8	6	6	6	2	3	8	7
10074	11	8	13	13	2	5	9	5
10075	10	4	10	10	8	6	8	9
10078	4	6	9	9	2	7	6	6
10079	8	4	7	7	8	6	12	9
10082	6	5	5	5	5	4	9	10
10083	5	4	7	7	8	6	12	6
10084	9	4	4	4	8	7	10	8
10085	10	6	7	7	4	4	9	9
10086	9	4	8	8	11	10	15	10
10087	11	7	6	6	3	9	11	7
10088	6	5	5	5	10	6	12	11
10089	5	6	6	6	8	11	9	4
10090	1	9	10	10	4	2	9	10
10091	8	5	6	6	10	8	8	11
10093	8	7	6	6	10	13	10	10
10094	10	9	9	9	7	9	8	9

Завдання:

Завдання виконати в одному з пакетів на вибір: KNIME, SPSS, RapidMiner або іншому.

1. Ознайомитися з прикладами застосування дискримінантного аналізу у тексті або в інших джерелах.
2. Використати дані, для яких було виконано кластерний аналіз, наприклад, дані, наведені у додатку, або у рейтинговій таблиці університетів України.
3. Виконати дискримінантний аналіз. Побудувати за результатами аналізу діаграму розсіювання елементів кластерів у просторі визначених канонічних функцій.
4. Проінтерпретувати отримані результати.
5. Проаналізувати результати дискримінантного аналізу. Зробити висновки про якість прогнозування значень групуючої змінної від незалежних: яка із незалежних змінних має найбільший вплив на класифікацію? Чи всі обрані змінні мають вплив на групуючу змінну? Наскільки точний отриманий прогноз? Чи можна в подальшому класифікувати об’єкти лише за допомогою обраних змінних? Спробувати проінтерпретувати отримані канонічні функції.

Лабораторна робота № 4. Пошук асоціативних правил

В пакеті RapidMiner пошук асоціативних правил здійснюють за допомогою операторів **FP-Growth** – пошуку частих елементів та наборів елементів у базі даних транзакцій, – та **Create Association Rules**, який генерує набір асоціативних правил із сформованої оператором **FP-Growt** множини частих наборів.

Проблема полягає у тому, що дані для аналізу зазвичай формуються як результат виконання запиту до бази даних і мають такий вигляд:

ID_транзакції	ID_товару
100	хліб
100	молоко
200	печиво
200	…

Рис. 4.1

А оператор **FP-Growt** в RM потребує такого вигляду вхідних даних:

ID_транзакції	хліб	молоко	печиво	…
100	true	true	false	…
200	false	false	true	…
…	…	…	…	…

Рис. 4.2

Отже дані, передані для аналізу, незалежно від формату – текстового чи у вигляді електронної таблиці, – потребують попередньої підготовки. У шаблоні *Market basket analysis* пакета RM пропонується послідовність операторів, показана на рис. 4.3, яка при запуску на виконання формує правила за тестовим набором даних. Однак підготовчі дії для іншого набору даних (з іншими атрибутами) можуть відрізнятися.

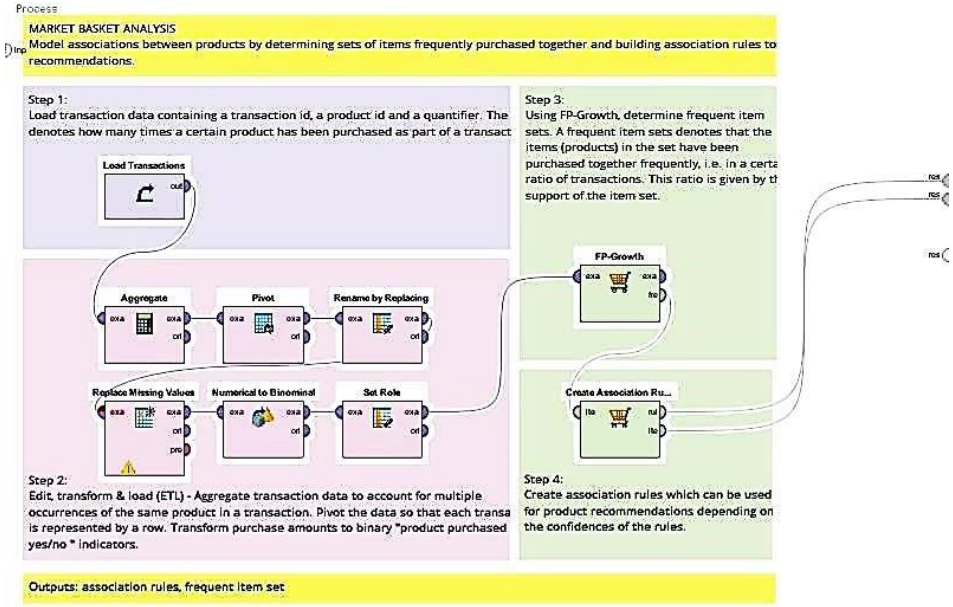


Рис. 4.3

Взявши за основу шаблон, виконаємо пошук асоціативних правил для даних, отриманих з деякої бази даних, як збережений у текстовому форматі результат виконання запиту, що містить фіскальний номер товарного чека (ідентифікатор транзакції) та назву товару (рис. 4.4). Результат запиту можна

зберегти і у форматі електронної таблиці.

Файл з даними можна завантажити до локального репозиторія (**Add data**) і потім перетягнути в область виконання проекта (**Retrieve**), або відкрити з Excel оператором **Read Excel**. В результаті дані однаково будуть перетворені до вигляду як на рис. 4.5:

фіс_ном назва
601 яйце
601 банан
580 ківі
580 грейпфрут рожевий
580 банан
51403 яйце
51403 ківі
51403 банан
51398 грейпфрут рожевий
51398 банан
478 ківі
478 банан
387 ківі
387 Кардамон

Рис. 4.4

Row No.	фіс_ном (integer) regular	назва (polynomial) regular
	601	яйце
	601	банан
	580	ківі
	580	грейпфрут рожевий
	580	банан
	51403	яйце
	51403	ківі
	51403	банан
	51398	грейпфрут рожевий
	51398	банан
	478	ківі
	478	банан
	387	ківі
	387	Кардамон
	387	банан

Рис. 4.5

Якщо слідувати шаблону, то послідовність операторів для наведеного набору даних буде такою, як на рис. 4.6:

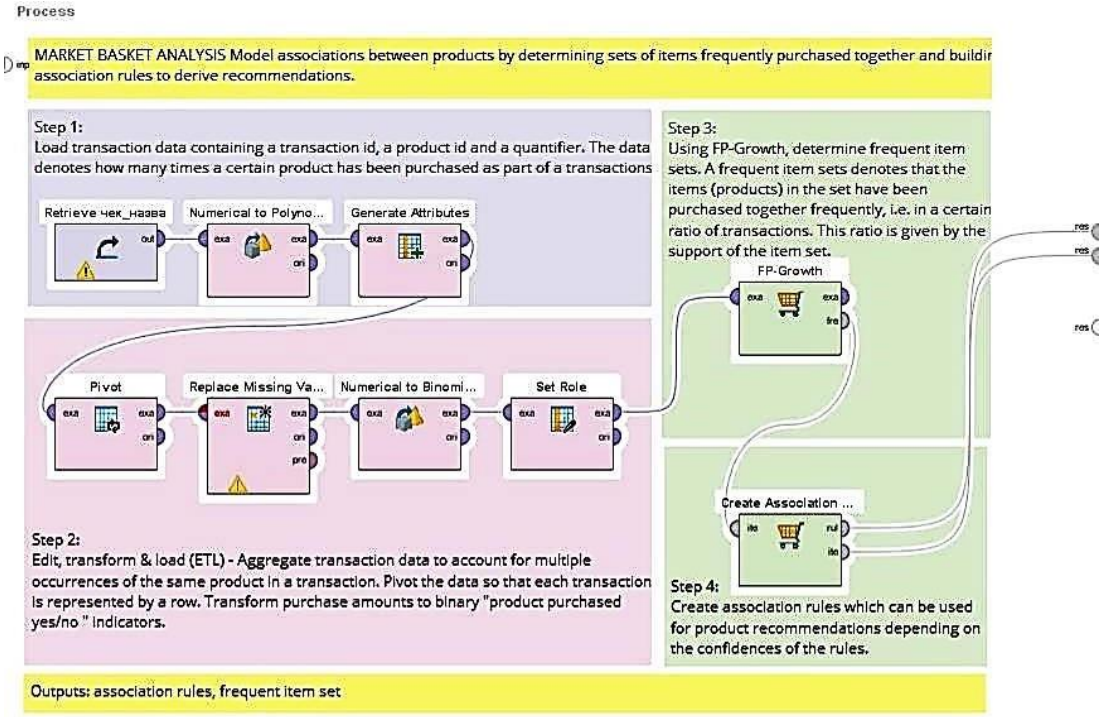


Рис. 4.6

Тут

1) **Retrieve** – завантаження первинних даних з репозиторію;

2) **Numerical to Polynomial** – перетворення атрибута фіс_ном з числового типу до номінативного;

3) **Generate Attribute** – добавлення атрибута item із значенням 1 для кожного рядка вхідних даних, заповнивши відповідні значення function description у параметрах оператора (це потрібно для створення зведення у наступному операторі – у шаблоні перед цим ще здійнюють агрегацію даних для випадку, якщо товар в одному чеку буде вказаний більше одного разу);

4) **Pivot** – оператор перетворення таблиці до вигляду, де кожній транзакції відповідає рядок, а атрибутами є назви продуктів:

Row No.	фіс_ном	item_банан	item_яйце	…
1	601	1	1	?
2	580	1	?	?
…	…	…	…	…

Рис. 4.7

Про входження продукту до транзакції свідчить значення 1, яке утворюється додаванням відповідних значень атрибуту item (згенерованого попереднім оператором), оскільки фактично формується зведена таблиця, в якій атрибут, за яким буде здійснюватися групування (у даному випадку необхідно вказати це «фіс_ном»), та атрибут для індексування (у даному випадку це «назва»). Атрибут для об'єднання та узагальнюючу операцію можна вказати явно, але у нашому випадку, якщо налаштувати параметри оператора як на рис. 4.8, його буде визначено автоматично і об'єднання здійснюватиметься за тим атрибутом, що залишився, тобто «item».

У разі відсутності товару в транзакції і неможливості виконати об'єднання значень, відповідна комірка таблиці не заповнюється (на екрані виводиться знак «?»). – Це пропущене значення, яке слід заповнити, для чого і призначено наступний оператор;

5) **Replace Missing Value** – за замовченням усі пропущені значення заповнюються нулем;

6) Numerical to Binomial – нулі та одиниці будуть замінені відповідно значеннями false та true для перетворення таблиці до вигляду, представленого на рис. 4.2.

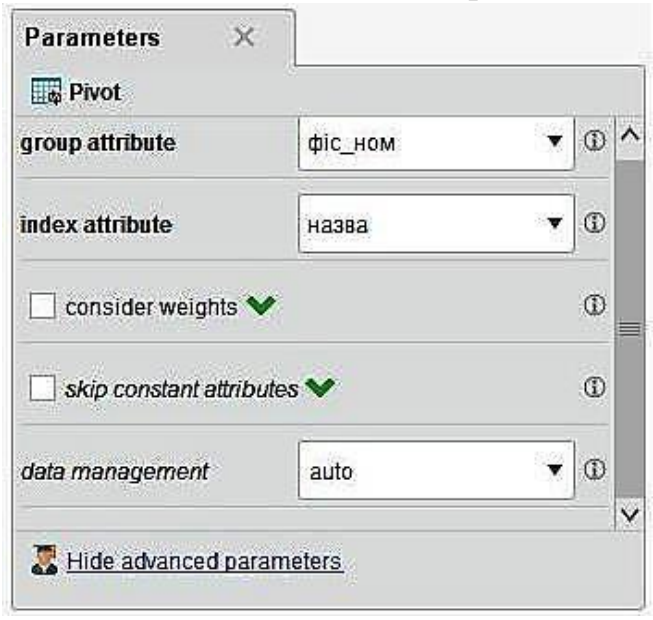
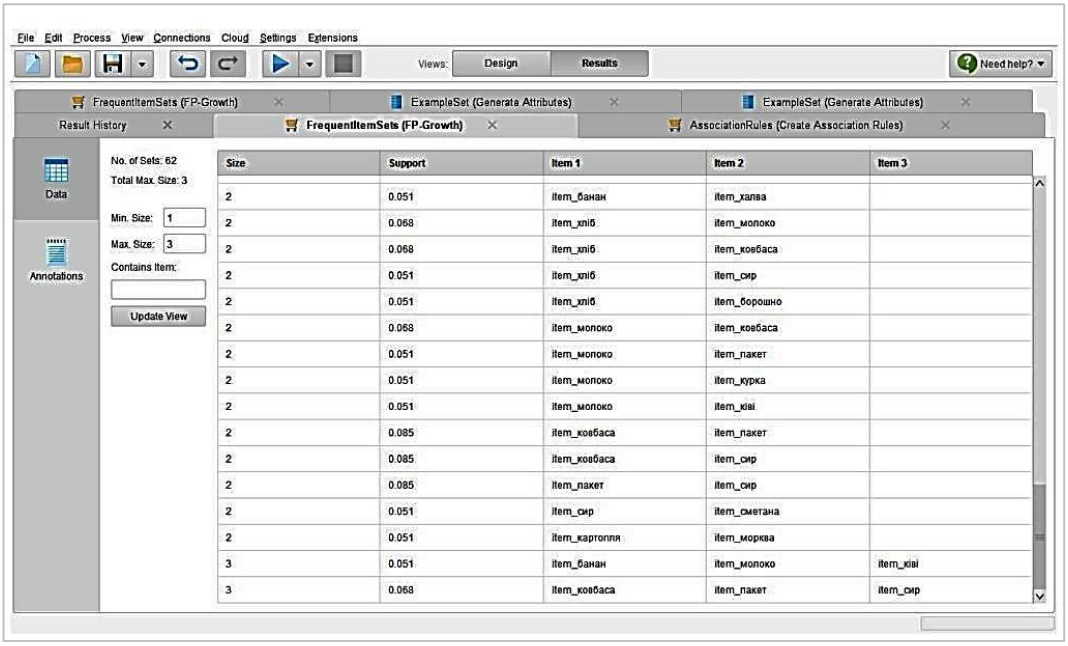


Рис. 4.8

Попередня підготовка завершена. Далі один за одним виконуватимуться оператори **FP-Growth** та **Create Association Rules**.

Для формування множини частих наборів товарів у параметрах оператора **FP-Growth** необхідно встановити порогове значення мінімальної підтримки. За замовченням встановлено значення 0,95, якому у нашому прикладі не відповідає жоден елемент, тому встановлено значення 0,09. В результаті отрималося 62 набори з одного, двох та трьох елементів (рис. 4.9):



Size	Support	Item 1	Item 2	Item 3
2	0.051	item_банан	item_халва	
2	0.068	item_хліб	item_молоко	
2	0.068	item_хліб	item_ковбаса	
2	0.051	item_хліб	item_сир	
2	0.051	item_хліб	item_борошно	
2	0.068	item_молоко	item_ковбаса	
2	0.051	item_молоко	item_пакет	
2	0.051	item_молоко	item_курка	
2	0.051	item_молоко	item_ківі	
2	0.085	item_ковбаса	item_пакет	
2	0.085	item_ковбаса	item_сир	
2	0.085	item_пакет	item_сир	
2	0.051	item_сир	item_сметана	
2	0.051	item_картопля	item_морква	
3	0.051	item_банан	item_молоко	item_ківі
3	0.068	item_ковбаса	item_пакет	item_сир

Рис. 4.9

Далі потрібно встановити порогове значення для одного з параметрів оператора Create Association Rules. Наприклад, у наведеному випадку встановлено значення достовірності=0,2, але його можна регулювати після виконання процеса, зменшуючи відповідно кількість правил, побудованих на сформованій множині частих шаблонів (Frequent Patterns). Так, за вказаними параметрами буде створено 57 правил (рис. 4.10):

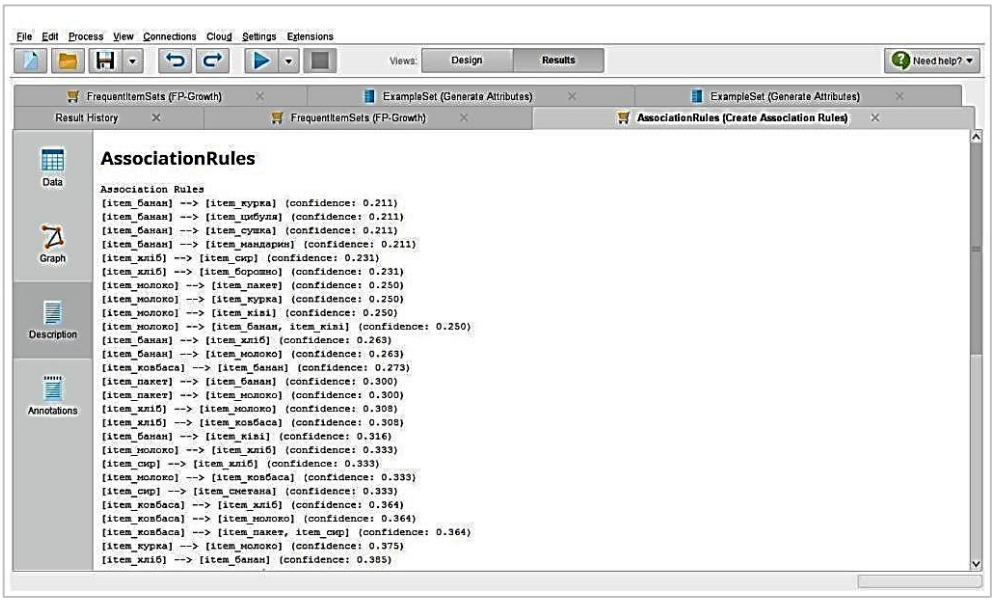


Рис. 4.10

Але, для аналізу, регулюючи обраний параметр повзунком, можна залишити лише найбільш значущі правила, які в результатах процесу будуть представлені у вигляді таблиці (рис. 4.11) або у вигляді графа (рис. 4.12). Для кожного правила наведено антецедент (тут *Premise*) та наслідок (тут *Conclusion*), підтримку та достовірність, а у таблиці, крім того ще й впевненість, ліфт, а також gain, laplace та ps:

- gain (виграш, підсилення) – коефіцієнт підсилення, який обчислюється за допомогою тета-параметра коефіцієнта підсилення;
- laplace – обчислюється за допомогою параметра k laplace;
- ps – ще один критерій для вибору правил.

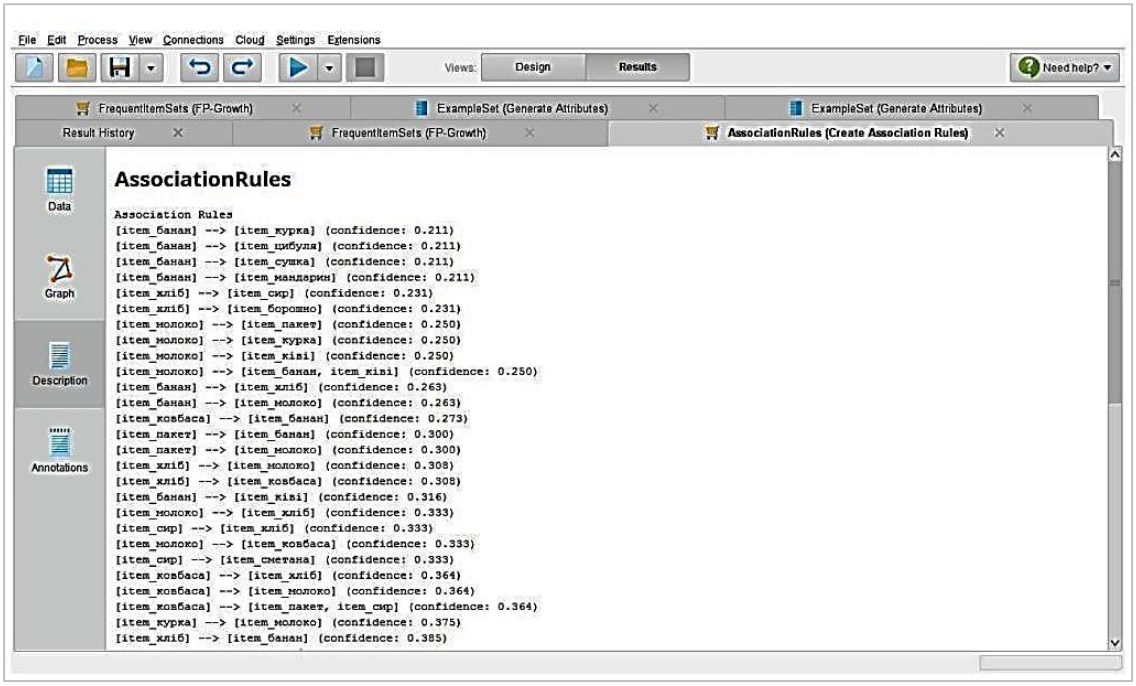


Рис. 4.11

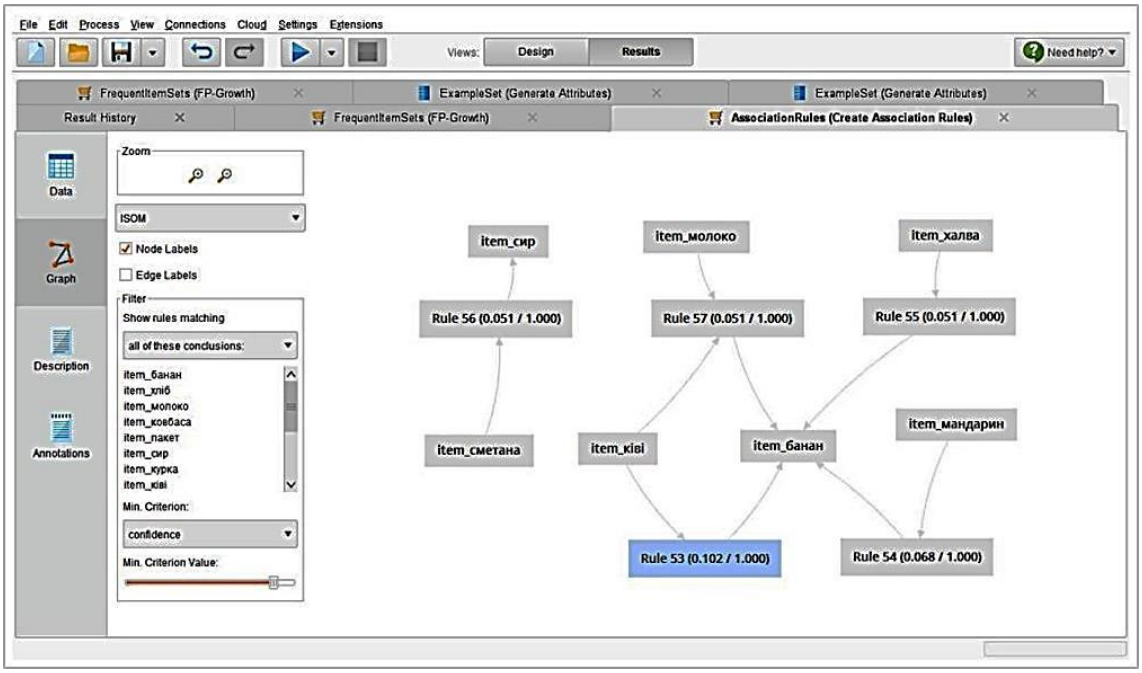


Рис. 4.12

Слід зауважити, що справді цікавих правил у наведеному наборі не знайдено – усі вони доволі банальні, але і проаналізована база транзакцій містить лише близько трьохсот записів.

1.2. Пошук асоціативних правил з пакетом KNIME

Ознайомитися з особливостями застосування процедур пошуку асоціативних правил в пакеті KNIME можна на блозі розробника (рис. 4.13) та проаналізувавши приклади з Explorer'а.

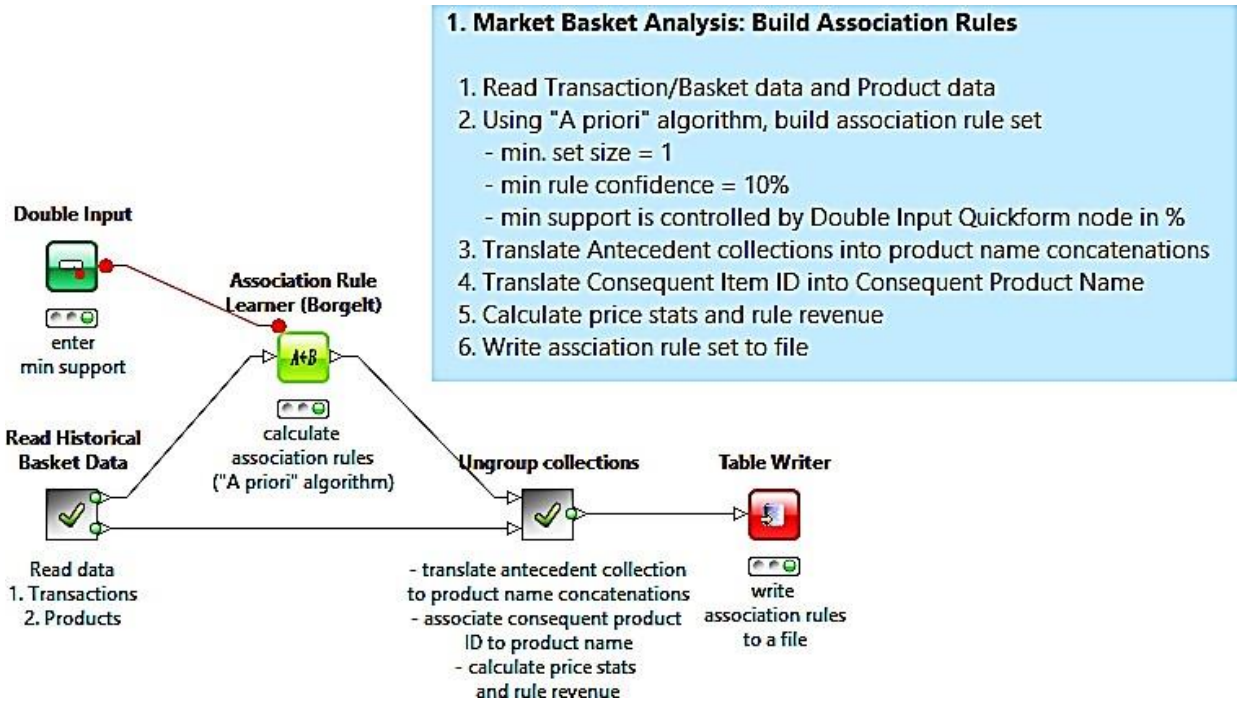


Рис. 4.13

Пропонується використати дані транзакцій. Наприклад, для завдання 4.1. дані матимуть вигляд (рис. 4.14):

	A
1	Транзакції
2	1
3	1,2
4	1,3
5	1,2,4,
6	1,5
7	1,2,3,6
8	1,7
9	1,2,4,8
10	1,3,9,
11	1,2,5,10
12	1,11
13	1,2,3,4,6,12
14	1,13
15	1,2,7,14
16	1,3,5,15
17	1,2,4,8,16,
18	1,17
19	1,2,3,6,9,18

Рис. 4.14

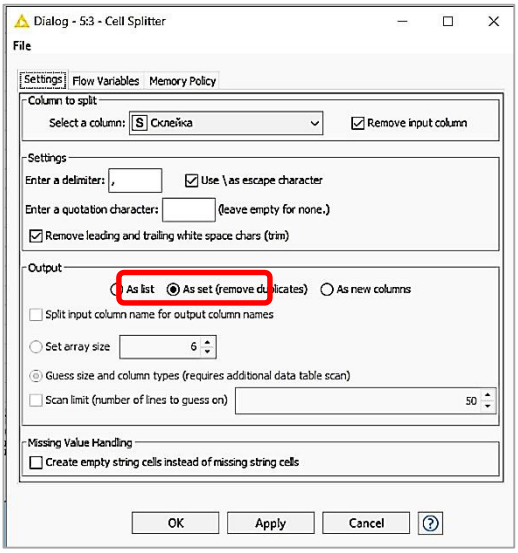


Рис. 4.15

З файла Excel такі дані можна прочитати, використовуючи вузол **Excel Reader**. Дані мають бути записані як рядки чисел, розділених комами. Кожен рядок відповідає транзакції.

Наступний крок – перетворити рядки чисел у множини, застосовуючи **Cell Splitter**. Важливо правильно відконфігурувати цей вузол (рис. 4.15): слід вказати, що послідовності елементів потрібно перетворити *у множину без повторень*.

Після цього можна приступити до побудови асоціативних правил або за допомогою вузла **Association Rule Learner**, який реалізує алгоритм Apriori в традиційній реалізації, або в реалізації **Association Rule Learner (Borgelt)**, яка пропонує кілька покращень продуктивності порівняно з традиційною реалізацією алгоритму. Однак набір правил асоціації вихідних даних залишається незмінним. При конфігуруванні цих вузлів слід вказати граничні значення для підтримки та достовірності для пошуку асоціативних правил (рис. 4.16, рис. 4.17).

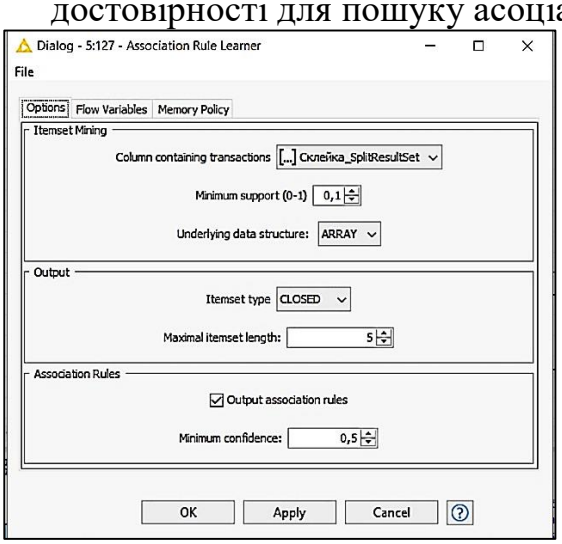


Рис. 4.16

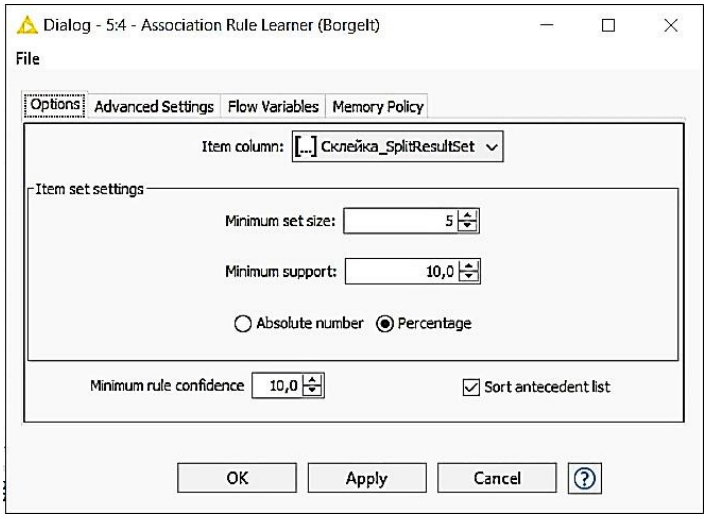


Рис. 4.17

Якщо все налаштовано коректно (рис. 4.19), то вузли виконаються без зауважень у консольному вікні (рис. 4.18).

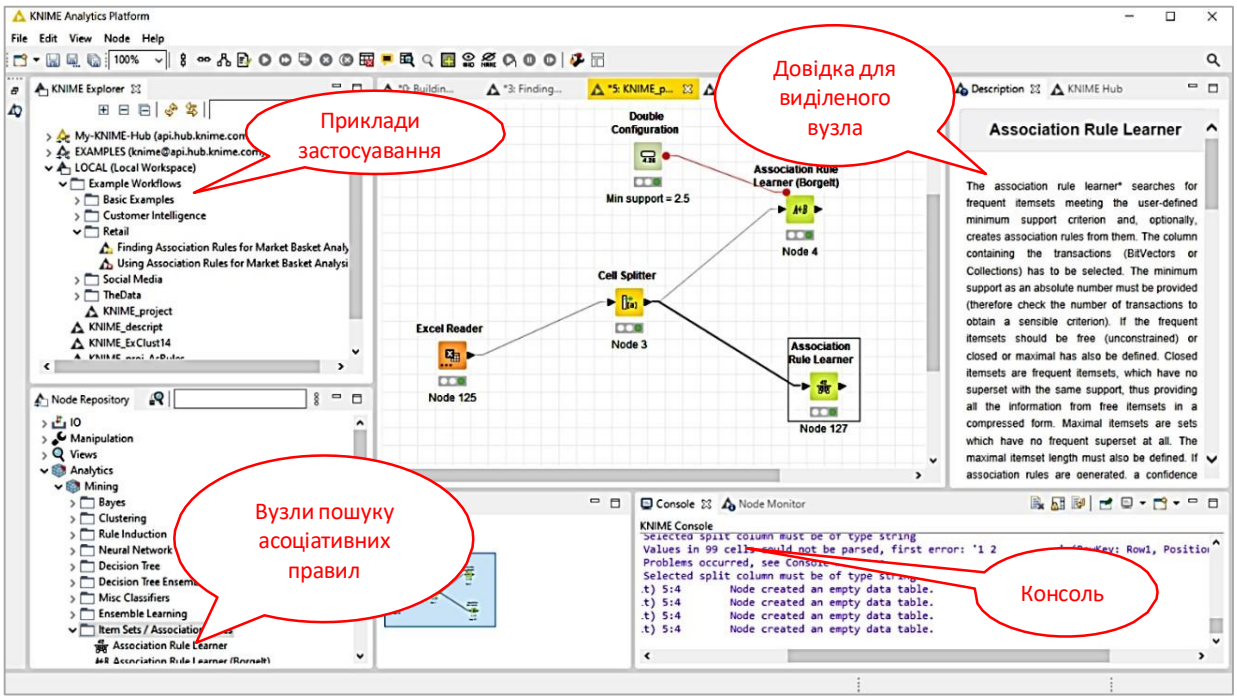


Рис. 4.18

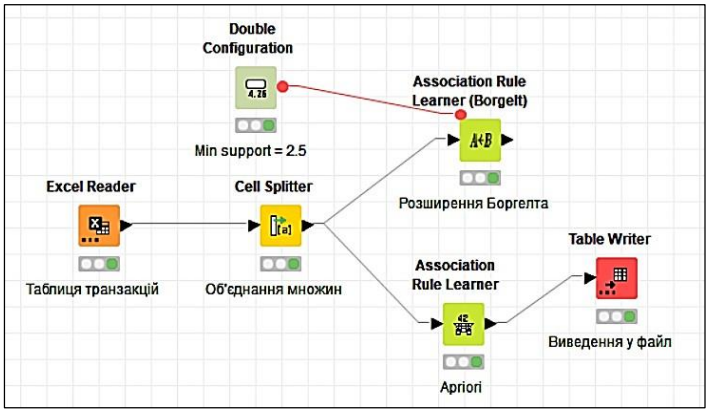


Рис. 4.19

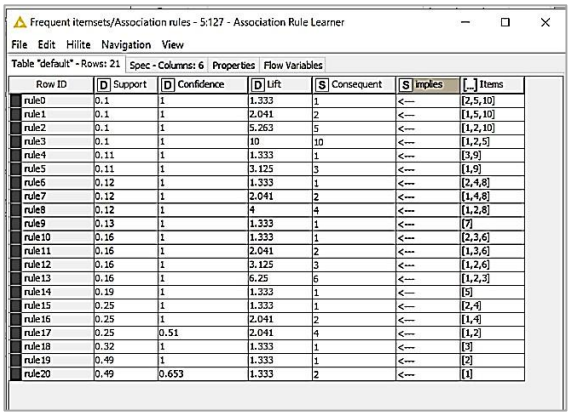


Рис. 4.20

Результатом застосування вузла **Association Rule Learner** буде таблиця (рис. 4.20).

Розглянемо приклад, наведений у п. 4.3 для пакета Rapid Miner: завантажимо файл **чек_Запит**.xlsx формату *.xls або *.xlsx (вузол **Excel Reader**). Потік команд для виконання аналізу асоціативних правил наведено на рис. 4.21:

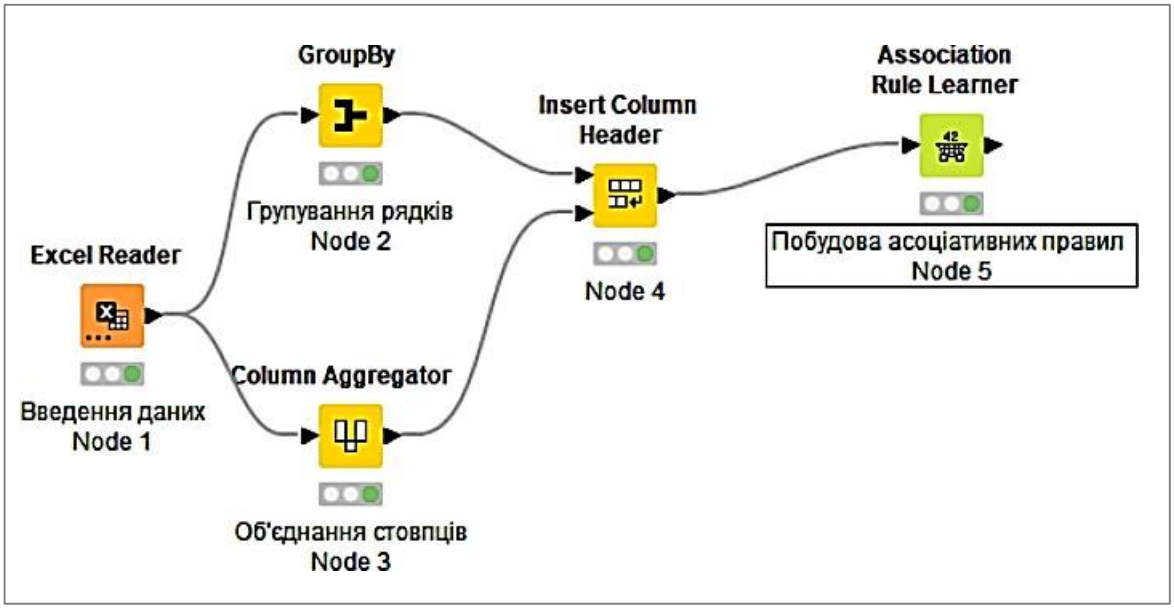


Рис. 4.21

Результатом виконання Вузла 1 (Введення даних з файла Excel) буде таблиця (рис. 4.22).

Для групування рядків (Вузол 2) за номерами чеків застосовують групову операцію «Set» – утворення множини (рис. 4.23).

На рис. 4.24 та рис. 4.25 показано налаштування цього вузла та результати відповідно.

Вузол 3 - Агрегація стовпців (товари) – призначений для об’єднання значень, розміщених у стовпцях. Тут застосовуємо операцію склеювання (Concatenate) (рис. 4.26, рис. 4.27).

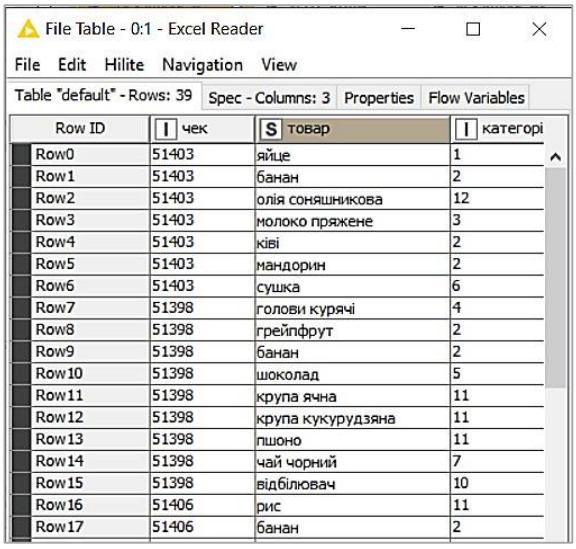


Рис. 4.22

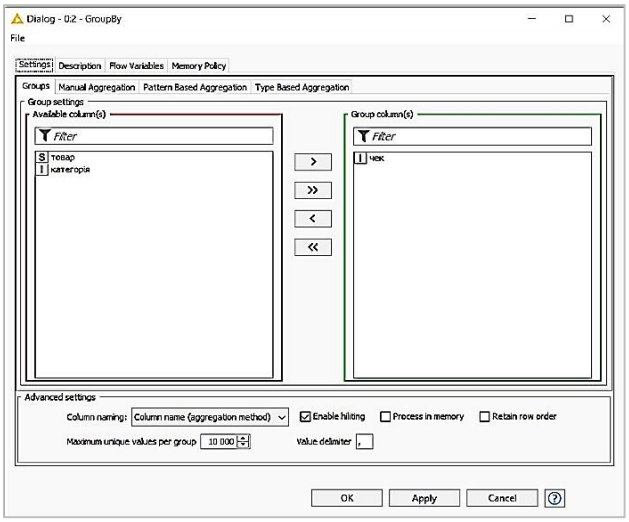


Рис. 4.23

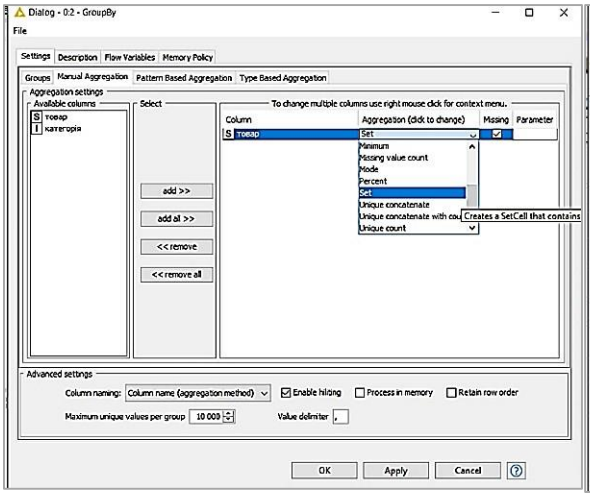


Рис. 4.24

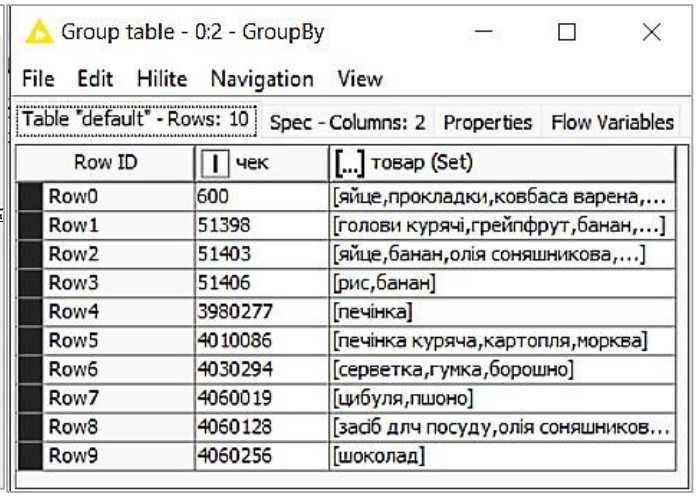


Рис. 4.25

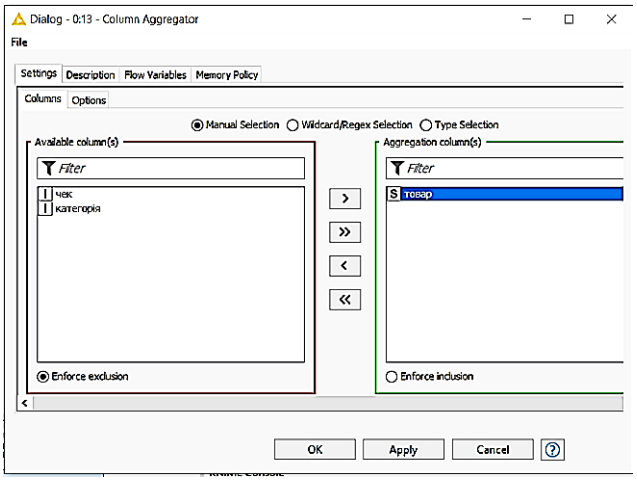


Рис. 4.26

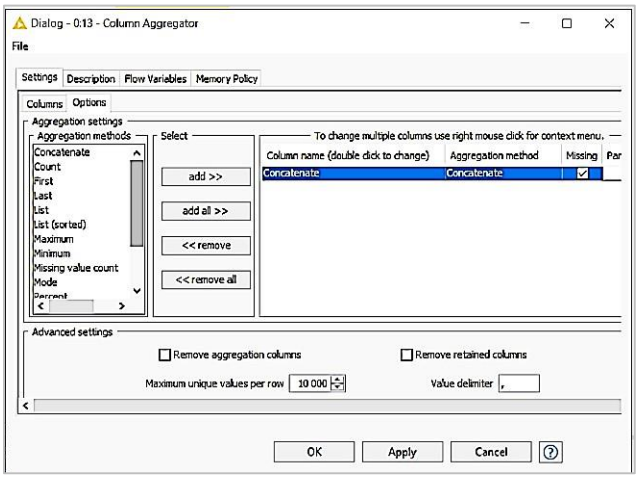


Рис. 4.27

Вузол 4 призначений для того, щоб визначити назви стовпців. У даному прикладі назвами будуть товари з утворених на попередніх кроках об’єднань (рис. 4.28).

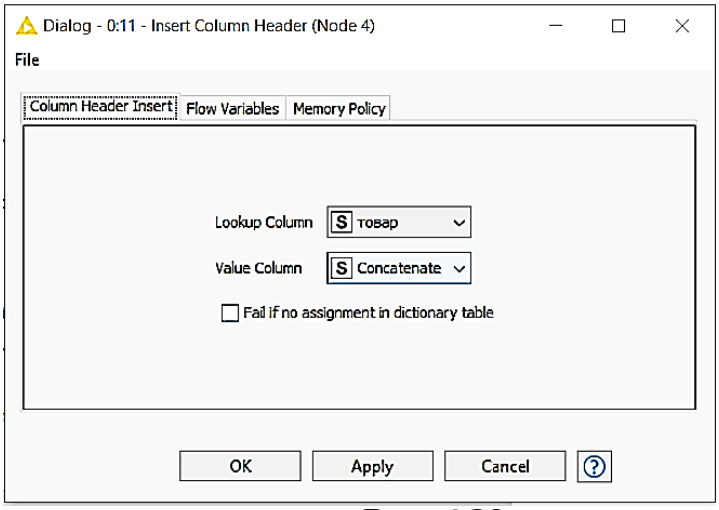


Рис. 4.28

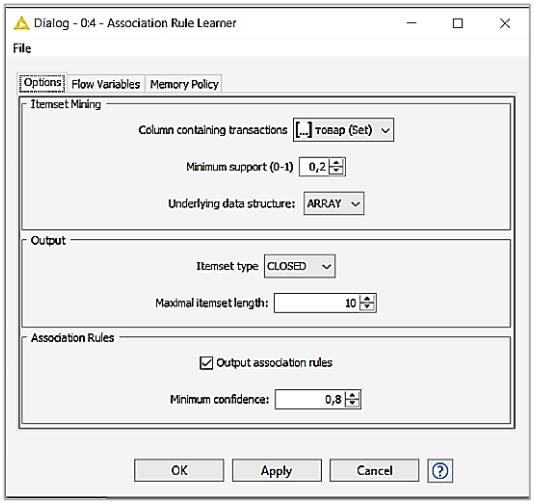
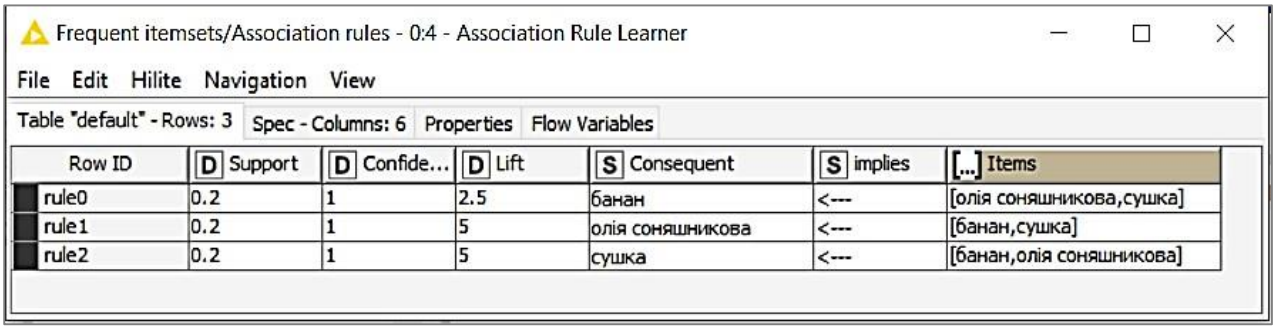


Рис. 4.29

Після виконання усіх підготовчих дій можна приступати до побудови асоціативних правил (Вузол 5): необхідно спочатку налаштувати параметри пошуку – визначити мінімальні значення підтримки та достовірності правил (рис. 4.29).

Для наведених параметрів (мінімальна підтримка=0,2 та мінімальна достовірність=0,8) отримаємо результати як на рис. 4.30. При зменшенні мінімальних значень кількість правил збільшиться. Можливо серед них будуть і цікавіші.



Row ID	Support	Confide...	Lift	Consequent	implies	Items
rule0	0.2	1	2.5	банан	<---	[олія соняшникова,сушка]
rule1	0.2	1	5	олія соняшникова	<---	[банан,сушка]
rule2	0.2	1	5	сушка	<---	[банан,олія соняшникова]

Рис. 4.30

Якщо дані для аналізу подано у форматі *.csv, то потік команд буде таким, як показано на Рис. 4.31:

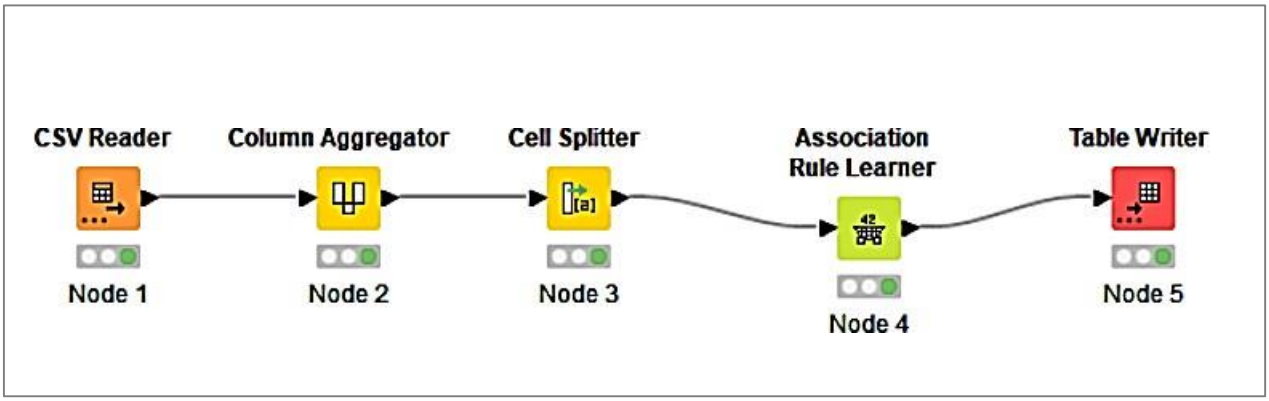


Рис. 4.31

Тут у Вузлі 1 необхідно вказати назву файла, розділовий знак, який використано для відокремлення значень, та наявність або відсутність заголовків стовпців (Рис. 4.32). В результаті вміст файлу (розглянемо для прикладу файл groceries.csv[27]) буде прочитано у таблицю (рис. 4.33):

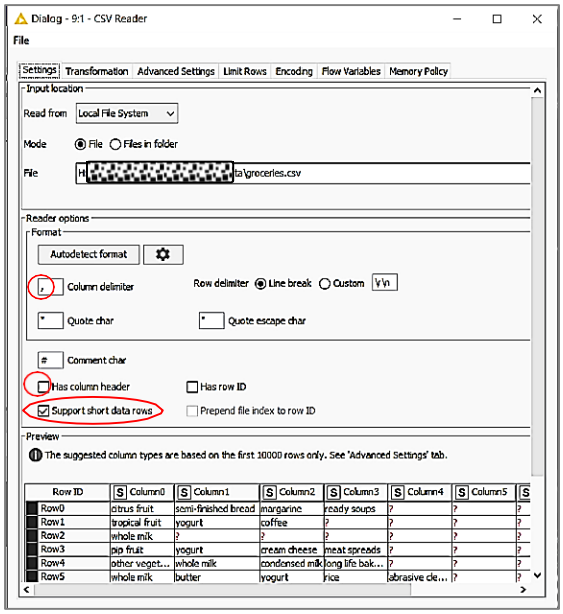


Рис. 4.32

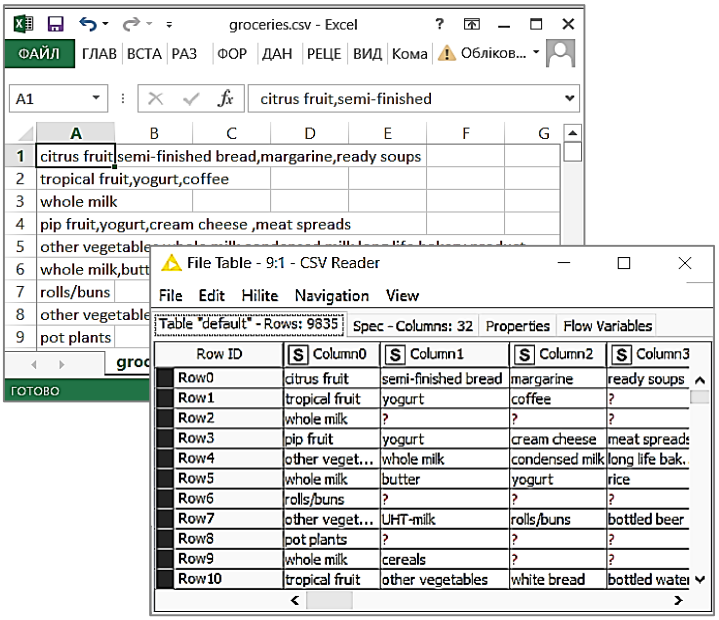


Рис. 4.33

У Вузлі 2 (Column Aggregator) до значень стовпців буде застосовано склеювання («Concatenate»), а у Вузлі 3 (Cell Splitter) зі склеєних значень буде утворено множину (рис. 4.34):

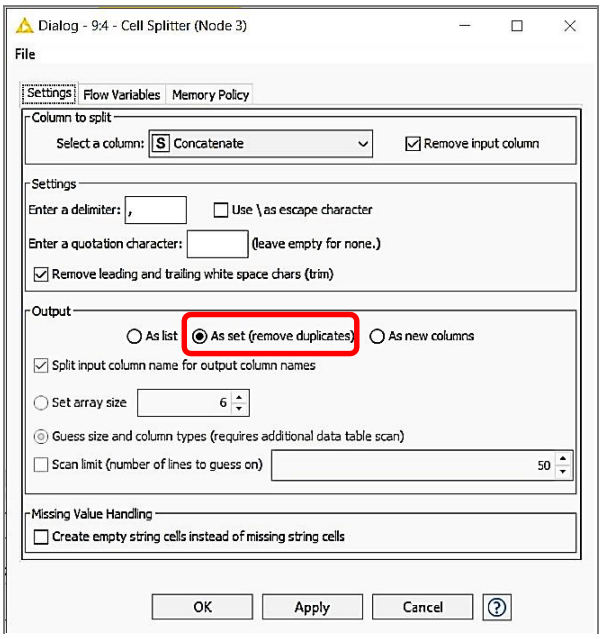


Рис. 4.34

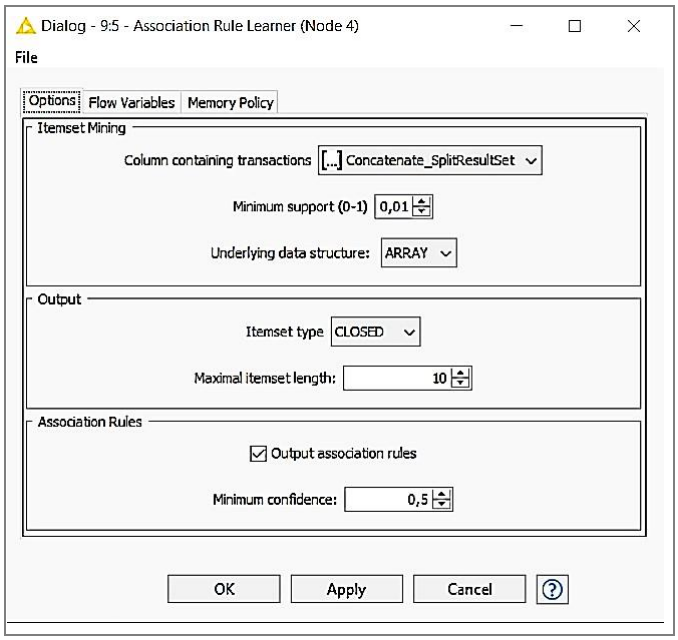
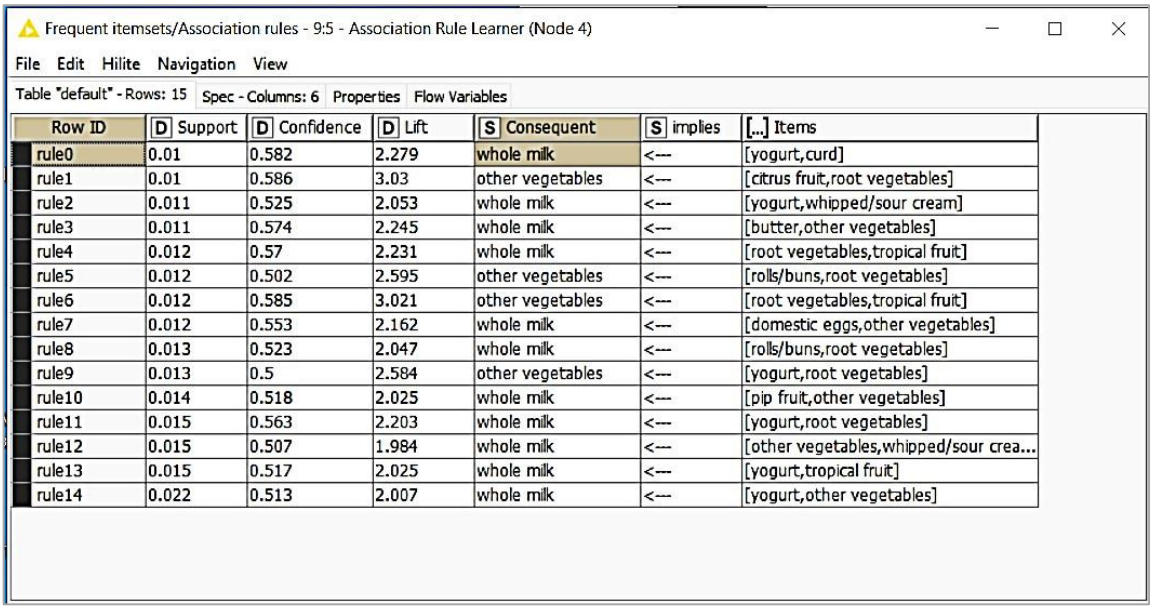


Рис. 4.35

У Вузлі 4 (Association Rules Learner) потрібно налаштувати параметри для пошуку асоціативних правил (рис. 4.35). Для наведених параметрів (*мінімальна підтримка = 0,01* та *мінімальна достовірність = 0,5*) отримаємо 14 правил (рис. 4.36). Всі вони мають значення ліфта більші за одиницю, тобто потенційно цікаві. Остаточне рішення про корисність отриманих правил має приймати експерт.


Frequent itemsets/Association rules - 9:5 - Association Rule Learner (Node 4)

File Edit Hilite Navigation View

Table "default" - Rows: 15 Spec - Columns: 6 Properties Flow Variables

Row ID	D Support	D Confidence	D Lift	S Consequent	S implies	[...] Items
rule0	0.01	0.582	2.279	whole milk	<---	[yogurt,curd]
rule1	0.01	0.586	3.03	other vegetables	<---	[citrus fruit,root vegetables]
rule2	0.011	0.525	2.053	whole milk	<---	[yogurt,whipped/sour cream]
rule3	0.011	0.574	2.245	whole milk	<---	[butter,other vegetables]
rule4	0.012	0.57	2.231	whole milk	<---	[root vegetables,tropical fruit]
rule5	0.012	0.502	2.595	other vegetables	<---	[rolls/buns,root vegetables]
rule6	0.012	0.585	3.021	other vegetables	<---	[root vegetables,tropical fruit]
rule7	0.012	0.553	2.162	whole milk	<---	[domestic eggs,other vegetables]
rule8	0.013	0.523	2.047	whole milk	<---	[rolls/buns,root vegetables]
rule9	0.013	0.5	2.584	other vegetables	<---	[yogurt,root vegetables]
rule10	0.014	0.518	2.025	whole milk	<---	[pip fruit,other vegetables]
rule11	0.015	0.563	2.203	whole milk	<---	[yogurt,root vegetables]
rule12	0.015	0.507	1.984	whole milk	<---	[other vegetables,whipped/sour crea...
rule13	0.015	0.517	2.025	whole milk	<---	[yogurt,tropical fruit]
rule14	0.022	0.513	2.007	whole milk	<---	[yogurt,other vegetables]

Рис. 4.36

Завдання

1. Ознайомтеся з прикладами виконання аналізу асоціативних правил у пакеті KNIME.
2. Підберіть в інтернеті дані для аналізу.
3. Підготуйте дані для аналізу у вигляді таблиць. За необхідності, виконайте дії щодо перетворення даних у своєму наборі даних. Переконайтеся, що всі змінні мають узгоджені для подальших процедур аналізу дані та типи.
4. Виконайте пошук асоціативних правил засобами KNIME або RapidMine.
5. Налаштуйте параметри пошуку правил (підтримку та достовірність) так, щоб визначити найбільш цікаві правила або отримати невелику кількість (10-20) правил для інтерпретації. Розгляньте інші показники сили правил, такі як, наприклад, Lift або Conviction.
6. Візуалізуйте результати пошуку усіма доступними в обраному пакеті способами. З'ясуйте переваги та недоліки різних способів візуалізації правил.
7. Проінтерпретуйте отримані результати, виявлення цікавих правил.
8. Зробіть та задокументуйте свої висновки: Які правила ви знайшли? Які атрибути найбільш сильно пов'язані один з одним. Які правила виявилися несподіваними для Вас? Чому, на вашу думку, таке може бути? Скільки різних значень підтримки та достовірності довелося перевірити, перш ніж були знайдені деякі асоціативні правила? Чи були якісь із знайдених правил достатньо корисними, щоб їх можна було використати для прийняття рішень? Чому так, або чому ні? Які способи представлення асоціативних правил були найбільш переконливими або зрозумілими для вас? Чим сподобався/не сподобався використаний інструментарій, тощо.

9. Оформіть звіт до лабораторної роботи за таким планом:
    1. Опис даних (посилання на джерело даних, перелік та опис використовуваних змінних).
    2. Використовуваний інструмент (вказати пакет та використані у ньому оператори чи процедури).
    3. Встановлені параметри пошуку асоціативних правил:

Підтримка	Достовірність	Кількість правил

    4. Отримані візуальні результати аналізу (скріншоти або інше).
    5. Висновки.

Лабораторна робота №5. АНАЛІЗ ЗВ'ЯЗКІВ.

5.1. Інформаційний пошук

Інформаційний пошук (ІП) (англ. *Information retrieval*) – наука про пошук *неструктурованої* документальної інформації, що включає:

- пошук інформації в документах;
- пошук самих документів;
- добування метаданих з документів;
- пошук тексту, зображень, відео та звуку у локальних реляційних базах даних, у гіпертекстових базах даних таких, як Інтернет та локальні інтранет;
- сортування документів за частотою шуканої фрази. Етапами інформаційного пошуку є:

1) кроулінг (crawling) – збір документів;
2) синтаксичний розбір;
3) аналіз;
4) *індексування**;***
5) пошук.

Якість пошуку визначається такими показниками (рис. 5.1):

1. Точність (*precision*) – відношення кількості добутих релевантних документів до загальної кількості добутих документів = ***RR/Rd,*** тобто *відповідність знайденого шуканому.*
2. Вибірка (*recall*) – відношення кількості добутих релевантних документів до загальної кількості релевантних документів = ***RR/Rt.***

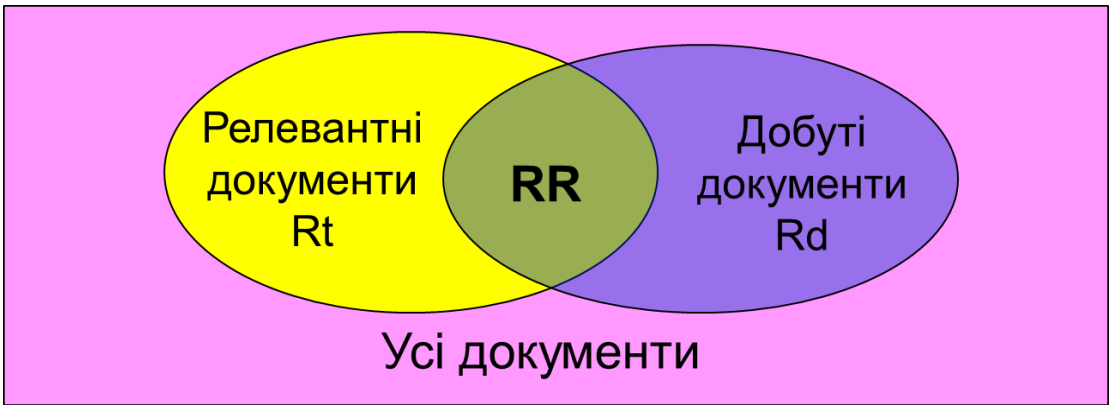


Рис. 5.1

5.2. Обчислення показників PageRank.

Степінь важливості веб-документа оцінюють кількістю підтверджень його зв'язків з іншими сторінками. Відповідно до показника PageRank[31] сторінка є

важливою, якщо до неї прив'язані важливі сторінки (рис. 5.2). Іншими словами, якщо всі дороги ведуть до Рима, то Рим – важливе місто.

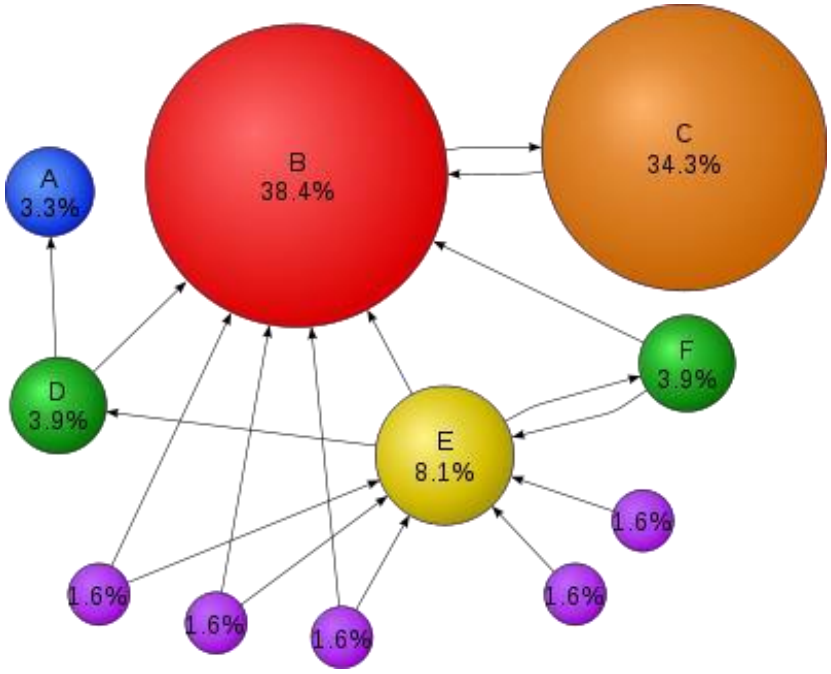


Рис. 5.2

До використання алгоритма PageRank пошукові системи здійснювали кроулінг веба та створювали списки термів, вишукуючи їх на кожній сторінці та створюючи зворотний індекс (*inverted index*). Коли пошуковий список сформований, сторінки з цим термом видобувають із зворотного індекса та ранжують відповідно до кількості входжень терма у сторінці (його релевантності). При такому підході релевантність сторінки можна штучно підвищити багатократним використанням терма, зокрема, завуальовано, наприклад, з використанням прихованого шрифту. Таке явище називають термінологічним спамом або термспамом.

Для подолання термспаму розробники алгоритму PageRank в компанії Google запропонували наступне:

1. Моделювати поведінку «випадкового серфера», який подорожує вебом, обираючи сторінки випадковим чином. Часто відвідувані сторінки вважають більш «важливими».

2. Про контент сторінки судять не тільки за термами самої сторінки, а і за термами пов'язаних сторінок, які значно важче сфальсифікувати.

Власне PageRank – це функція, яка кожній веб-сторінці ставить у відповідність дійсне число – «важливість» сторінки. Сторінка тим важливіша, чим вищий її PageRank.

Для обчислення PageRank веб слід представити як орієнтований граф. Для графа, зображеного на Рис. 5.3, матриця гіперпосилань M (*transition matrix*) показує, що з вузла A випадковий серфер може з однаковою імовірністю (p=1/3) перейти в кожен з решти вузлів B, C або D[32]. Вузли B та D пов'язані кожен з двома вузлами, тому імовірність переходів p=1/2. З вузла C можна перейти тільки у вузол A з імовірністю p=1. Нехай випадковий серфер у початковий момент

знаходиться у будь-якому з вузлів (веб-сторінок). Тоді вектор V_0 складається з елементів 1/n, де n – загальна кількість веб-сторінок. Наступне значення вектора обчислюють множенням на матрицю гіперпосилань:

$$\boldsymbol{v}_i = \boldsymbol{M} \cdot \boldsymbol{v}_{i-1} \qquad (5.1)$$

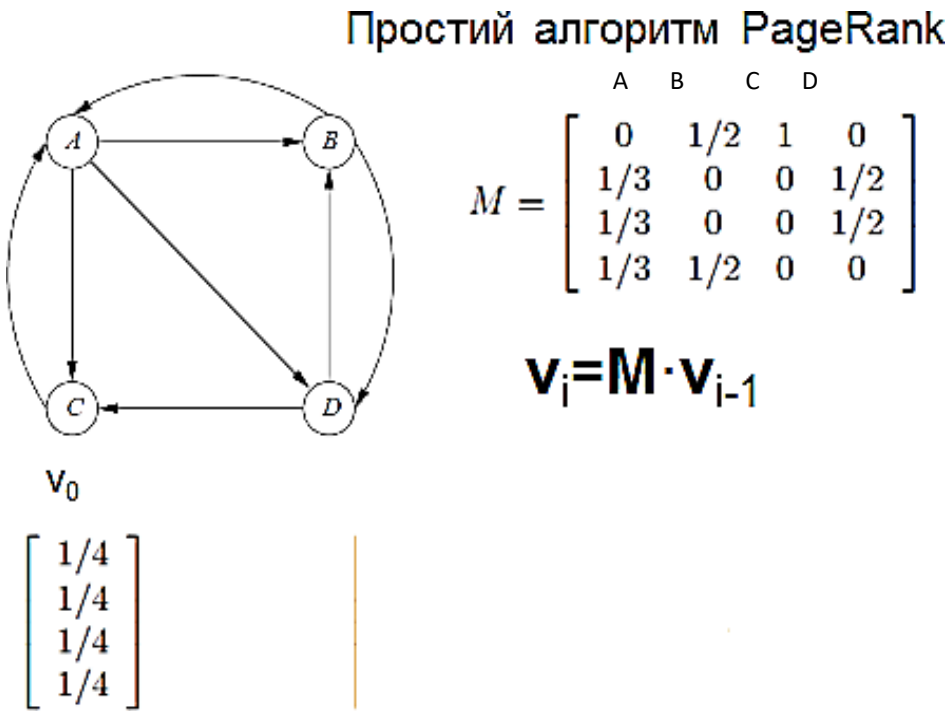


Рис. 5.3

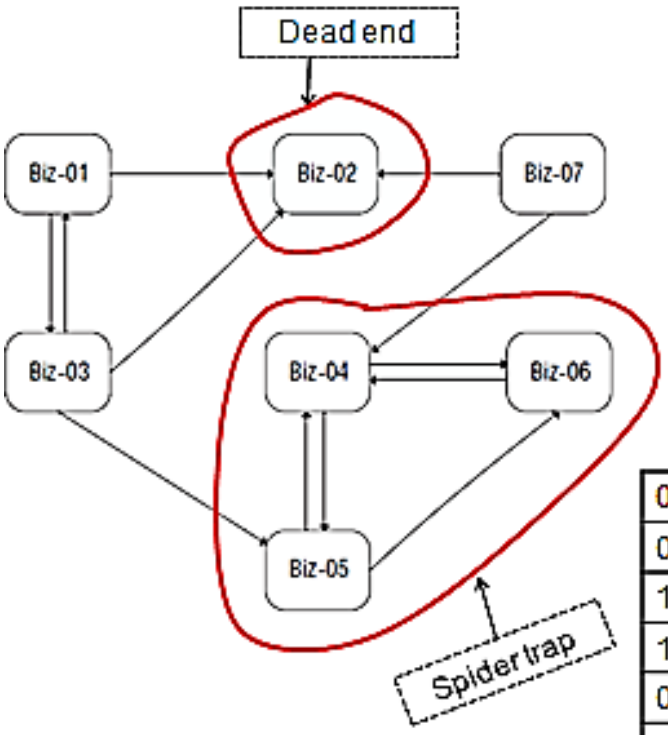
На *k*-тій ітерації отримують розподіл серфера у вебі після *k* кроків (рис. 5.4). В решті решт результат сходиться до значення, яке і показує PageRank кожної сторінки. У даному випадку найбільший PageRank має перша сторінка, тобто сторінка A:

V_0 V_1 V_2 V_3

$$\begin{bmatrix} 1/4 \\ 1/4 \\ 1/4 \\ 1/4 \end{bmatrix}, \begin{bmatrix} 9/24 \\ 5/24 \\ 5/24 \\ 5/24 \end{bmatrix}, \begin{bmatrix} 15/48 \\ 11/48 \\ 11/48 \\ 11/48 \end{bmatrix}, \begin{bmatrix} 11/32 \\ 7/32 \\ 7/32 \\ 7/32 \end{bmatrix}, \ldots, \begin{bmatrix} 3/9 \\ 2/9 \\ 2/9 \\ 2/9 \end{bmatrix}$$

Рис. 5.4

Простий PageRank коректно працює для «коректного» веба. Реальна ситуація складніша: крім строго пов'язаних компонентів (SCC – *strong connected components*) веб може містити компоненти, *з яких можна* потрапити в SCC в один бік, *в які можна* потрапити з SCC в один бік, ізольваних компонентів і т.д. Отже в реальному вебі часто виникають глухі кути або тупики (*dead ends*) та пастки (*spider traps*) (рис. 5.5).



Приклад мережі веб-сторінок

Файл	Посилання на
Biz-01	Biz-02, Biz-03
Biz-02	Немає посилань
Biz-03	Biz-01, Biz-02, Biz-05
Biz-04	Biz-05, Biz-06
Biz-05	Biz-04, Biz-06
Biz-06	Biz-04
Biz-07	Biz-02, Biz-04

Матриця гіперпосилань =

01	02	03	04	05	06	07
0	0	1/3	0	0	0	0
1/2	0	1/3	0	0	0	1/2
1/2	0	0	0	0	0	0
0	0	0	0	1/2	1	1/2
0	0	1/3	1/2	0	0	0
0	0	0	1/2	1/2	0	0
0	0	0	0	0	0	0

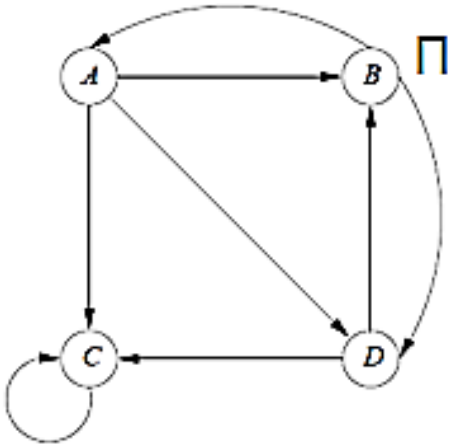
Рис. 5.5

Для обчислення PageRank з урахуванням пасток і тупиків вводять механізм «таксації» – скінчену імовірність для серфера вийти з веба на будь-якому етапі та запустити нового серфера на будь-яку сторінку. Таку імовірність позначають β.

$$\mathbf{v}' = \beta M \mathbf{v} + (1-\beta)\mathbf{e}/n \qquad (5.2)$$

На рис. 5.6 маємо структуру, в якій вершина *C* – пастка. Потрапивши на цю сторінку, серфер не зможе перейти на інші сторінки. За алгоритмом простого PageRank ця сторінка отримає найвищий коефіцієнт, а решта сторінок отримають PageRank=0. Модифікований PageRank дозволяє серферу «телепортацію» з пастки.

Покращити результати дозволяє і так званий тематично-чутливий PageRank: якщо дослідити зміст (контент) сторінок, то можна дати перевагу відвідувати лише тематично пов'язані сторінки, тобто сторінки з високою релевантністю до пошукового запиту. Так само можна обмежити відвідування сторінок, позначених як спам. Такий коефіцієнт ще називають TrustRank (рис. 5.7). Для обчислення TrustRank формують множину S вершин, тематично пов'язаних або вільних від спаму, і в одиничному векторі e_S обнуляють компоненти не пов'язані з елементами множини S.



Подолання пасток (taxation)

$$M = \begin{bmatrix} 0 & 1/2 & 0 & 0 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 0 & 1 & 1/2 \\ 1/3 & 1/2 & 0 & 0 \end{bmatrix}$$

$$\mathbf{v}' = \beta M \mathbf{v} + (1-\beta)\mathrm{e}/n$$

β=0,8
1-β=0,2
n=4

$$\mathbf{v}' = \begin{bmatrix} 0 & 2/5 & 0 & 0 \\ 4/15 & 0 & 0 & 2/5 \\ 4/15 & 0 & 4/5 & 2/5 \\ 4/15 & 2/5 & 0 & 0 \end{bmatrix} \mathbf{v} + \begin{bmatrix} 1/20 \\ 1/20 \\ 1/20 \\ 1/20 \end{bmatrix}$$

$$\begin{bmatrix} 1/4 \\ 1/4 \\ 1/4 \\ 1/4 \end{bmatrix}, \begin{bmatrix} 9/60 \\ 13/60 \\ 25/60 \\ 13/60 \end{bmatrix}, \begin{bmatrix} 41/300 \\ 53/300 \\ 153/300 \\ 53/300 \end{bmatrix}, \begin{bmatrix} 543/4500 \\ 707/4500 \\ 2543/4500 \\ 707/4500 \end{bmatrix}, \ldots, \begin{bmatrix} 15/148 \\ 19/148 \\ 95/148 \\ 19/148 \end{bmatrix}$$

Імовірність залишитися у пастці обмежена 14

Рис. 5.6

Тематично-чутливий PageRank

$$\mathbf{v}' = \beta M \mathbf{v} + (1-\beta)\mathrm{e}/n$$

$$\mathbf{v}' = \beta M \mathbf{v} + (1-\beta)\mathrm{e}_S/|S|$$

set $S = \{B, D\}$

$$\mathbf{v}' = \begin{bmatrix} 0 & 2/5 & 4/5 & 0 \\ 4/15 & 0 & 0 & 2/5 \\ 4/15 & 0 & 0 & 2/5 \\ 4/15 & 2/5 & 0 & 0 \end{bmatrix} \mathbf{v} + \begin{bmatrix} 0 \\ 1/10 \\ 0 \\ 1/10 \end{bmatrix}$$

$$\begin{bmatrix} 0/2 \\ 1/2 \\ 0/2 \\ 1/2 \end{bmatrix}, \begin{bmatrix} 2/10 \\ 3/10 \\ 2/10 \\ 3/10 \end{bmatrix}, \begin{bmatrix} 42/150 \\ 41/150 \\ 26/150 \\ 41/150 \end{bmatrix}, \begin{bmatrix} 62/250 \\ 71/250 \\ 46/250 \\ 71/250 \end{bmatrix}, \ldots, \begin{bmatrix} 54/210 \\ 59/210 \\ 38/210 \\ 59/210 \end{bmatrix}$$

Рис. 5.7

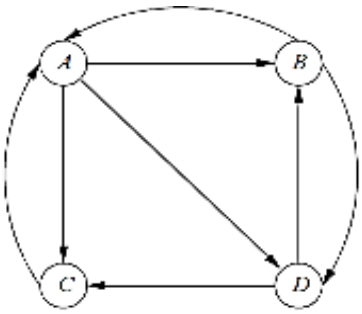
Узагальненим показником важливості сторінки є SpamMass (рис. 5.8):

Обчислення SpamMass

SpamMass=(r-t)/r,

де r – PageRank,

t – TrustRank



Node	PageRank	TrustRank	Spam Mass
A	3/9	54/210	0.229
B	2/9	59/210	-0.264
C	2/9	38/210	0.186
D	2/9	59/210	-0.264

Тут немає спама

Якщо SM≈1, то сторінка імовірно є спамом.
Якщо SM<0, або SM≈0, то сторінка – не спам

21

Рис. 5.8

В оригінальній роботі авторів формула PageRank мала такий вигляд: припустимо, що на сторінку A вказують (тобто цитують її) сторінки $T_1 \ldots T_n$. Встановимо параметр d – коефіцієнт затухання, який може бути визначений в діапазоні від 0 до 1, – рівним 0,85. Позначимо C(A) – кількість посилань, що виходять зі сторінки A. Тоді PageRank сторінки A буде визначений так:

$$PR(A) = (1-d) + d\,(PR(T_1)/C(T_1) + \ldots + PR(T_n)/C(T_n))$$

5.3. HITS

Інший підхід до визначення важливості веб-сторінки запропоновано в алгоритмі HITS. На відміну PageRank HITS ранжує тільки відповіді на конкретний запит. Важливість сторінок визначається або її авторитетом (*authorities*) – сторінка важлива, оскільки забезпечує інформацію по темі, – або її посередництвом (*hub*) – сторінка є хабом, концентратором або посередником, якщо містить посилання на авторитетні сторінки.

Тобто, якщо для PageRank *«сторінка важлива, якщо до неї прив'язані важливі сторінки»*, то для HITS *«сторінка є гарним хабом, коли вона пов'язана з гарним авторитетом, і навпаки»*.

Для обчислення показників авторитетності та хабності спочатку будують матрицю зв'язків L (L_{ij}=1, коли сторінки i та j пов'язані) та транспонують її. Далі обчислюють $\bar{h}$ та $\bar{a}$ (Рис. 5.9). Тут λ та μ – масштабні коефіцієнти.

Граничні значення векторів $\bar{h}$ та $\bar{a}$ і будуть містити показники хабності та авторитетності для сторінок веба. Як видно з рис. 5.9, сторінка A є найбільшим концентратором, оскільки вона пов'язана з найбільшими авторитетами. Відповідно, сторінки B та C – найбільші авторитети.

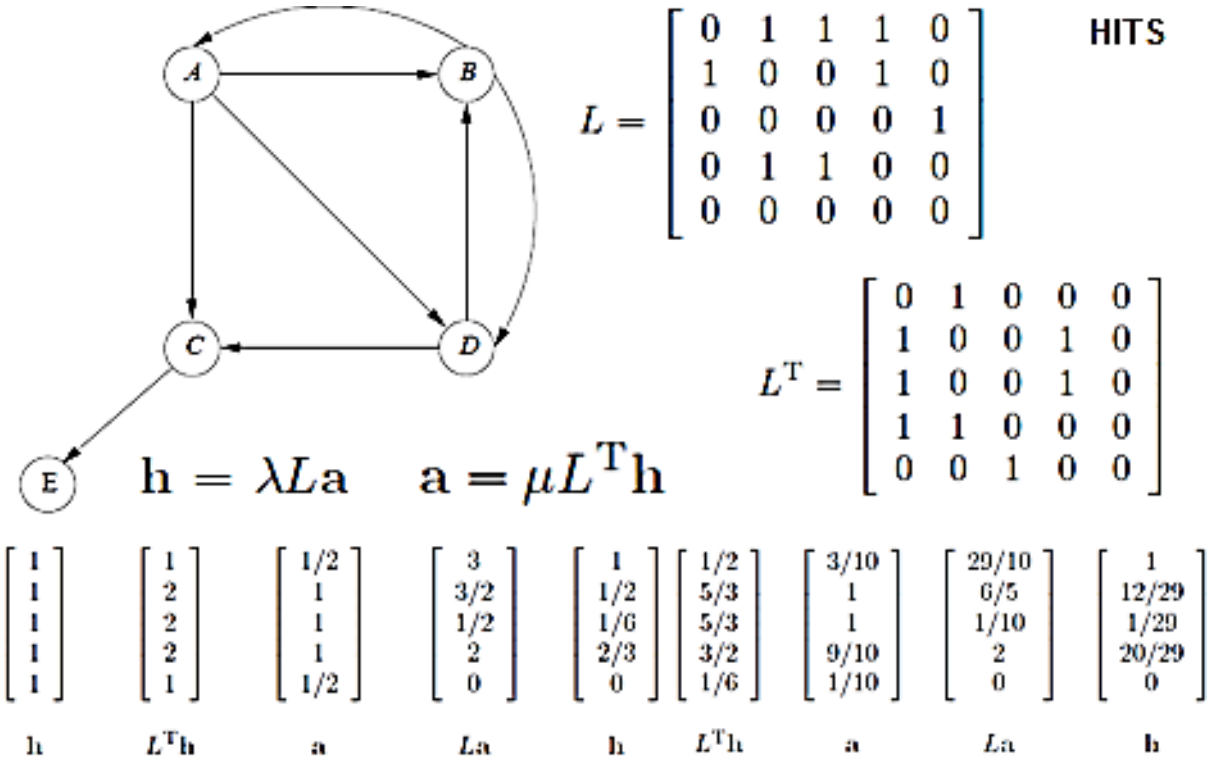


Рис. 5.9. Алгоритм HITS

Завдання:

Виконати або завдання 1 або завдання 2 в залежності від номера в списку журналу академічної групи:

Завдання1. Розробити програму для реалізації одного із алгоритмів аналізу зв'язків (однієї з модифікацій PageRank або HITS)

a. використовуючи масиви;

b. використовуючи динамічні структури;

c. використовуючи засоби розпаралелювання.

Завдання 2. Розробити граф мережі з n вузлів (n=6, 8 або 10) та дослідити її на спам. При використанні симулятора дослідити не менше двох графів (з пасткою/ без пасток або з мертвими вершинами/ без них) та прокоментувати отримані результати.

Список рекомендованої літератури

Основна

1. Болюбаш Н. М. Інтелектуальний аналіз даних : навчальний посібник**.** Миколаїв: ЧДУ ім. Петра Могили, 2023. – 320 с. Системний курс з ІАД: CRISP-DM, класифікація, кластеризація, асоціативні правила, нейромережі, приклади в економіці та бізнесі. https://www.researchgate.net/publication/379994907_Bolubas_N_M_Intelektualnij_analiz_danih
2. George J. Klir, Bo Yuan. Fuzzy sets and fuzzy logic: theory and application. New Jersey. 2018. 763 p.
3. Талах М. В., Дворжак В. В. Інтелектуальний аналіз даних. Частина 1: навчальний посібник. Чернівці : Технодрук, 2022. – 367 с. Archer Розглядаються базові поняття Data Mining, моделі даних, класи задач (класифікація, кластеризація, регресія, пошук асоціацій), приклади реалізації алгоритмів. https://archer.chnu.edu.ua/handle/123456789/6751
4. Іванчук Я. В., Месюра В. І., Яровий А. А., Манжілевський О. Д. Інтелектуальний аналіз даних та машинне навчання. Частина 1. Базові методи та засоби аналізу даних: навч. посібник. Вінниця : ВНТУ, 2021. – 69 с. https://pdf.lib.vntu.edu.ua/books/2022/Ivanchuk_P1_2021_69.pdf
5. Гороховатський В. О., Творошенко І. С. Методи інтелектуального аналізу та оброблення даних : навч. посібник. Харків : ХНУРЕ, 2021. – 92 с. https://openarchive.nure.ua/handle/document/15868

Додаткова

1. Литвин В. В., Пасічник В. В., Нікольський Ю. В. Аналіз даних та знань: навчальний посібник. Львів : Магнолія, 2021. – 276 с.
2. Винничук Р. Інтелектуальний аналіз даних (Data Mining) в харчовій промисловості. Матеріали конференції, 2024. Приклад прикладного застосування Data Mining у харчовій промисловості, містить реальний кейс, структуру задач, інструментарій. https://www.researchgate.net/publication/384673396_INTELEKTUALNIJ_ANALIZ_DANIH_DATA_MINING_V_HARCOVIJ_PROMISLOVOSTI
3. Фісун М. Т., Кравець І. О., Казмірчук П. П., Ніколенко С. Г. Інтелектуальний аналіз даних : практикум. Львів : Новий Світ-2000, 2023. – 160 с.
4. Савеленко О. К., Лисенко І. А., Іванченко О. О. CASE-технології у проектуванні інформаційних систем: навчальний посібник / Мін-во освіти і науки України, Центральноукраїн. нац. техн. ун-т. - Кропивницький: Видавець Лисенко В.Ф., 2018.- 240 с. https://dspace.kntu.kr.ua/handle/123456789/10278

Інформаційні ресурси

1. Weka 3: Data Mining Software in Java. URL: http://www.cs.waikato.ac.nz/ml/weka/

Методичне видання

МЕТОДИЧНІ РЕКОМЕНДАЦІЇ ДО ВИКОНАННЯ ЛАБОРАТОРНИХ РОБІТ З НАВЧАЛЬНОЇ ДИСЦИПЛІНИ **“ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ ДАНИХ”**

для студентів денної та заочної форми навчання
за спеціальністю 122 “Комп’ютерні науки”

Укладач

Лисенко Ірина Анатоліївна