УКРАЇНСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ЗАЛІЗНИЧНОГО ТРАНСПОРТУ

ФАКУЛЬТЕТ ІНФОРМАЦІЙНО-КЕРУЮЧИХ СИСТЕМ
ТА ТЕХНОЛОГІЙ

Кафедра інформаційних технологій

МЕТОДИЧНІ ВКАЗІВКИ

до практичних занять

з дисципліни

«*ІНФОРМАЦІЙНІ СИСТЕМИ І ТЕХНОЛОГІЇ*»

Харків — 2024

Методичні вказівки розглянуто і рекомендовано до друку на засіданні кафедри інформаційних технологій 03 червня 2024 р., протокол № 10.

Методичні вказівки призначено для здобувачів вищої освіти першого (бакалаврського) рівня спеціальності 275.02 «Транспортні технології (на залізничному транспорті)» усіх форм навчання, що вивчають дисципліну «Інформаційні системи і технології».

Укладач

старш. викл. О. І. Іванюк

Рецензент

доц. Н. А. Корольова

ЗМІСТ

Вступ 4

Практична робота 1

Планування проєкту. Побудова діаграми Ганта 5

Практична робота 2

Планування проєкту. Створення Канбан-проєкту 9

Практична робота 3

Проєктування інформаційної системи. Створення діаграми ERD 18

Практична робота 4

Проєктування інформаційної системи. Використання методології IDEF0 25

Практична робота 5

Аналіз даних. Використання методів регресії 32

Практична робота 6

Аналіз даних. Використання методів класифікації 42

Практична робота 7

Аналіз даних. Використання методів кластеризації 50

Список літератури 60

ВСТУП

У сучасному світі інформаційні системи і технології відіграють ключову роль у розвитку будь-якої галузі, включаючи транспортні технології. Від ефективного планування та управління проєктами до аналізу та обробки даних — всі ці процеси неможливо уявити без використання сучасних інформаційних технологій. Ця дисципліна покликана надати здобувачам фундаментальні знання та практичні навички, необхідні для успішного впровадження та використання інформаційних систем у професійній діяльності.

Методичні вказівки містять практичні завдання, які спрямовані на розвиток умінь самостійного планування проєктів, проєктування інформаційних систем, а також аналізу даних з використанням різних методів. Вони містять короткі теоретичні відомості, інструкції, приклади виконання завдань та запитання для самоконтролю, що допоможуть здобувачам краще зрозуміти матеріал та підготуватися до виконання практичних робіт.

Практична робота 1

ПЛАНУВАННЯ ПРОЄКТУ. ПОБУДОВА ДІАГРАМИ ГАНТА

1.1 Мета роботи

Отримати навички планування проєкту, використовуючи діаграми Ганта.

1.2 Теоретичні відомості

Діаграма Ганта є важливим інструментом управління проєктами, що дає змогу візуально представити етапи проєкту, з урахуванням їхньої тривалості та послідовності виконання. Вісь *x* відображає час, а вісь *y* — перелік етапів проєкту. Прямокутниками на діаграмі позначають інтервали часу, впродовж яких мають виконуватись відповідні етапи проєкту. Довжина прямокутника відповідає тривалості етапу.

До діаграми Ганта можна застосувати метод критичного шляху, що дає змогу визначити найважливіші завдання, затримка у виконанні яких призводить до збільшення загального терміну реалізації проєкту.

Діаграма Ганта використовується для наочного подання різних сценаріїв виконання проєкту, корегування робочих процесів і розподілу ресурсів. Динамічна природа діаграми Ганта дає можливість моделювати різні результати проєкту на основі різних сценаріїв. Ця адаптивність має вирішальне значення для оптимізації робочих процесів і перерозподілу ресурсів, необхідних для вирішення будь-яких непередбачуваних проблем або вузьких місць.

1.3 Порядок виконання роботи

1 Для вихідних даних за варіантом розрахувати дати початку та завершення кожного з етапів проєкту, враховуючи залежності між етапами. У якості дати початку проєкту обрати 01 жовтня поточного року.

2 Побудувати діаграму Ганта. Для побудови можна використовувати будь-які інструменти: табличні процесори (Google Таблиці, MS Excel тощо), таск-менеджери, системи управління проєктами.

3 Визначити загальну тривалість та дату завершення проєкту.

4 Визначити критичний шлях проєкту.

5 Визначити етапи проєкту, що мають резерви часу.

1.4 Вихідні дані

Варіанти вихідних даних до завдання доступні за посиланням [12].

1.5 Приклад виконання роботи

Вихідні дані для побудови діаграми Ганта наведено у таблиці 1.1.

Таблиця 1.1 — Вихідні дані

Номер етапу	**Назва етапу**	**Попередні етапи**	**Тривалість етапу, *днів***
1	Етап 1	–	2
2	Етап 2	1	3
3	Етап 3	1	5
4	Етап 4	2	4
5	Етап 5	3, 4	8
6	Етап 6	4	5

Дата початку виконання проєкту: 1 березня 2024 року.

Розрахунок дат початку та завершення виконання кожного з етапів проєкту здійснимо з урахуванням залежностей між етапами: етап починається після завершення всіх етапів, що є попередніми.

У таблиці 1.2 наведено розраховані дати початку та завершення виконання кожного з етапів проєкту.

Таблиця 1.2 — Розраховані дати початку та завершення виконання кожного з етапів проєкту

Номер етапу	**Назва етапу**	**Попередні етапи**	**Тривалість етапу, *днів***	**Дата початку етапу**	**Дата завершення етапу**
1	Етап 1	–	2	01.03.2024	02.03.2024
2	Етап 2	1	3	03.03.2024	05.03.2024
3	Етап 3	1	5	03.03.2024	07.03.2024
4	Етап 4	2	4	06.03.2024	09.03.2024
5	Етап 5	3, 4	8	10.03.2024	17.03.2024
6	Етап 6	4	5	10.03.2024	14.03.2024

Побудову діаграми Ганта виконаємо в Google Таблицях з використанням інструменту «Умовне форматування».

На рисунок 1.1 наведено побудовану діаграму Ганта.

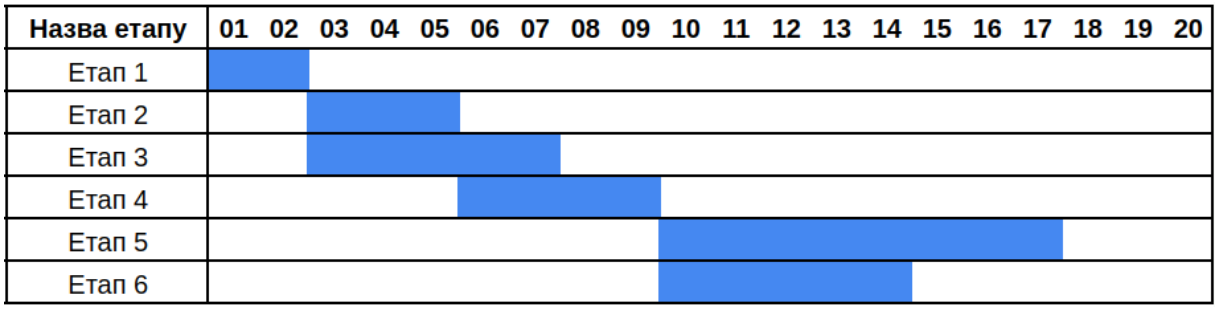


Рисунок 1.1 — Побудована діаграма Ганта

Загальна тривалість проєкту: 17 днів.

Дата завершення виконання проєкту: 17 березня 2024 року.

Критичний шлях: етап 1 — етап 2 — етап 4 — етап 5.

Етапи, що мають резерви часу: етап 3 (2 дні), етап 6 (3 дні).

Висновки. У роботі побудовано діаграму Ганта. Для цього розраховано дати початку та завершення виконання кожного з етапів проєкту. Аналіз отриманої діаграми Ганта встановив загальну тривалість проєкту, дату завершення виконання проєкту, критичний шлях, етапи, що мають резерви часу.

Запитання для самоконтролю

1 Що таке діаграма Ганта?

2 Що відображає кожна з осей діаграми Ганта?

3 Як діаграма Ганта враховує залежності між етапами проєкту?

4 Як діаграма Ганта відображає тривалість кожного з етапів проєкту?

5 Що таке критичний шлях проєкту?

6 Як визначити етапи, що мають резерви часу?

Практична робота 2

ПЛАНУВАННЯ ПРОЄКТУ. СТВОРЕННЯ КАНБАН-ПРОЄКТУ

2.1 Мета роботи

Отримати навички створення Канбан-проєкту і керування епіками та завданнями у Jira.

2.2 Теоретичні відомості

Канбан — це одна з найпоширеніших методологій управління роботою в проєктах. Канбан-дошка — це візуальний інструмент, який дає змогу контролювати робочий процес, відображаючи завдання та їхні стани.

Кожна колонка на Канбан-дошці відповідає окремому етапу роботи в проєкті. Наприклад, «To Do» для завдань, що ще не розпочаті, «In Progress» для тих, що в процесі виконання, і «Done» для завдань, які вже завершено.

Канбан-дошка візуалізує, коли завдання були додані до проєкту та коли вони мають бути завершені. Це надає команді чітку картину про терміни виконання завдань.

Канбан також допомагає контролювати обсяг роботи, який команда може прийняти. Кожна колонка може мати ліміт на кількість завдань, які можуть бути в ній одночасно.

Канбан допомагає виявити проблеми в робочому процесі та ресурсах, що можуть бути оптимізовані для покращення продуктивності та якості роботи.

Канбан є ефективним інструментом для візуалізації та оптимізації робочих процесів, і його використання може покращити управління проєктами та продуктивність команди.

2.3 Порядок виконання роботи

1 Дослідити та коротко описати предметну галузь.

2 Налаштувати проєкт Канбан у Jira [4]:

а) створити та увійти у свій обліковий запис Jira;

б) створити новий проєкт Канбан типу Team-managed;

в) обрати назву для проєкту.

3 Налаштувати Канбан-дошку: налаштувати стовпці, які подають етапи робочого процесу (workflow) для вибраної предметної галузі (наприклад, To Do, In Progress, Review, Done) (рисунок 2.1).

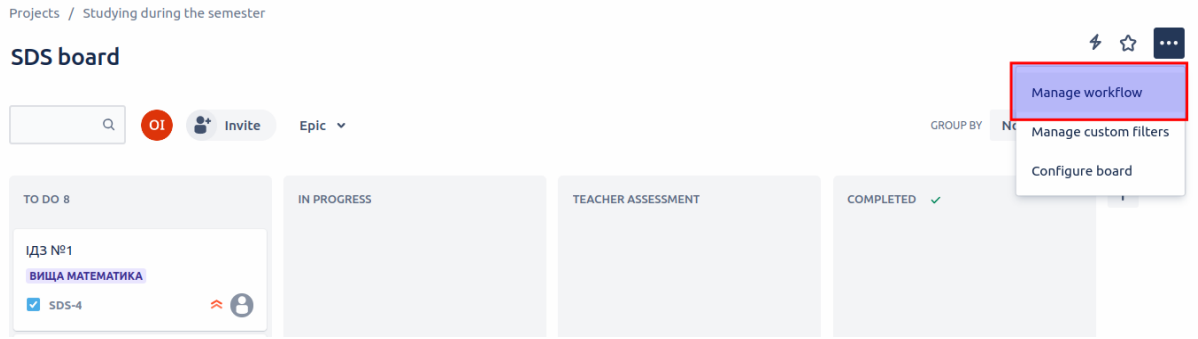


Рисунок 2.1 — Пункт меню налаштування робочого процесу проєкту

4 Створити епіки (epic) та завдання (task):

а) створити щонайменше 5 епіків, які представляють укрупнені етапи проєкту в межах обраної предметної галузі;

б) створити щонайменше 15 завдань, що відносяться до епіків, розбиваючи роботу на менші одиниці;

в) додати до завдань додаткове поле — пріоритет (Priority) (рисунок 2.2) та встановити значення цього поля для створених завдань;

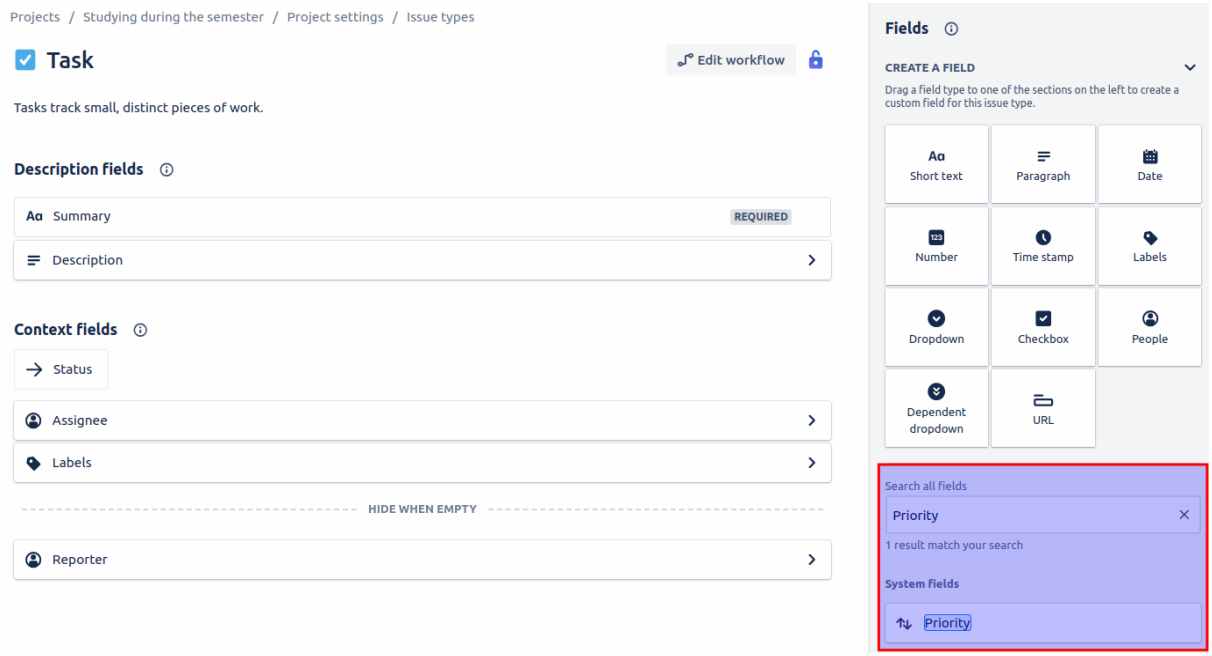


Рисунок 2.2 — Додавання поля пріоритет

г) встановити часові обмеження або терміни виконання відповідно до вимог проєкту та експортувати графічне подання хронології (Timeline) проєкту (рисунок 2.3).

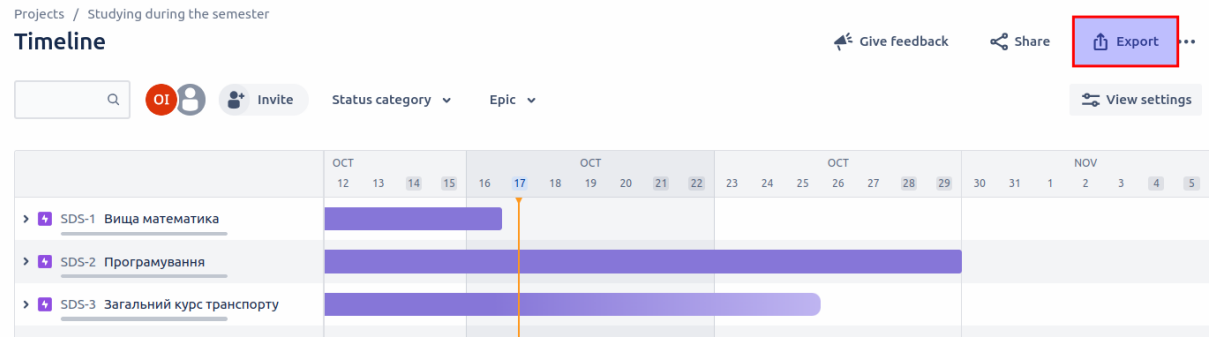


Рисунок 2.3 — Експорт хронології проєкту

5 Провести моделювання використання Канбан-дошки командою:

а) за можливості, запросити одногрупників до співпраці над проєктом (рисунок 2.4);

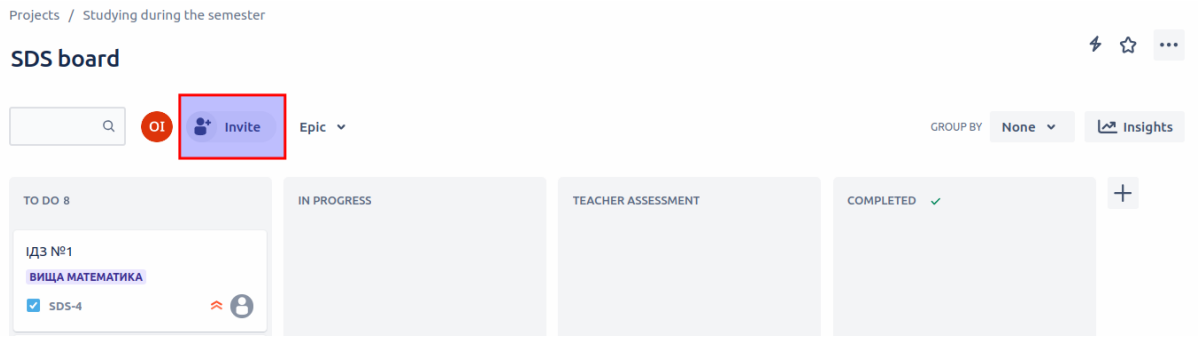


Рисунок 2.4 — Кнопка запрошення інших користувачів до проєкту

б) імітувати роботу над проєктом, переміщуючи завдання по стовпцях Канбан-дошки.

6 Створити кумулятивну схему (Cumulative flow diagram). Для цього необхідно увімкнути пункт Звіти (Reports) у Project settings > Features (рисунок 2.5).

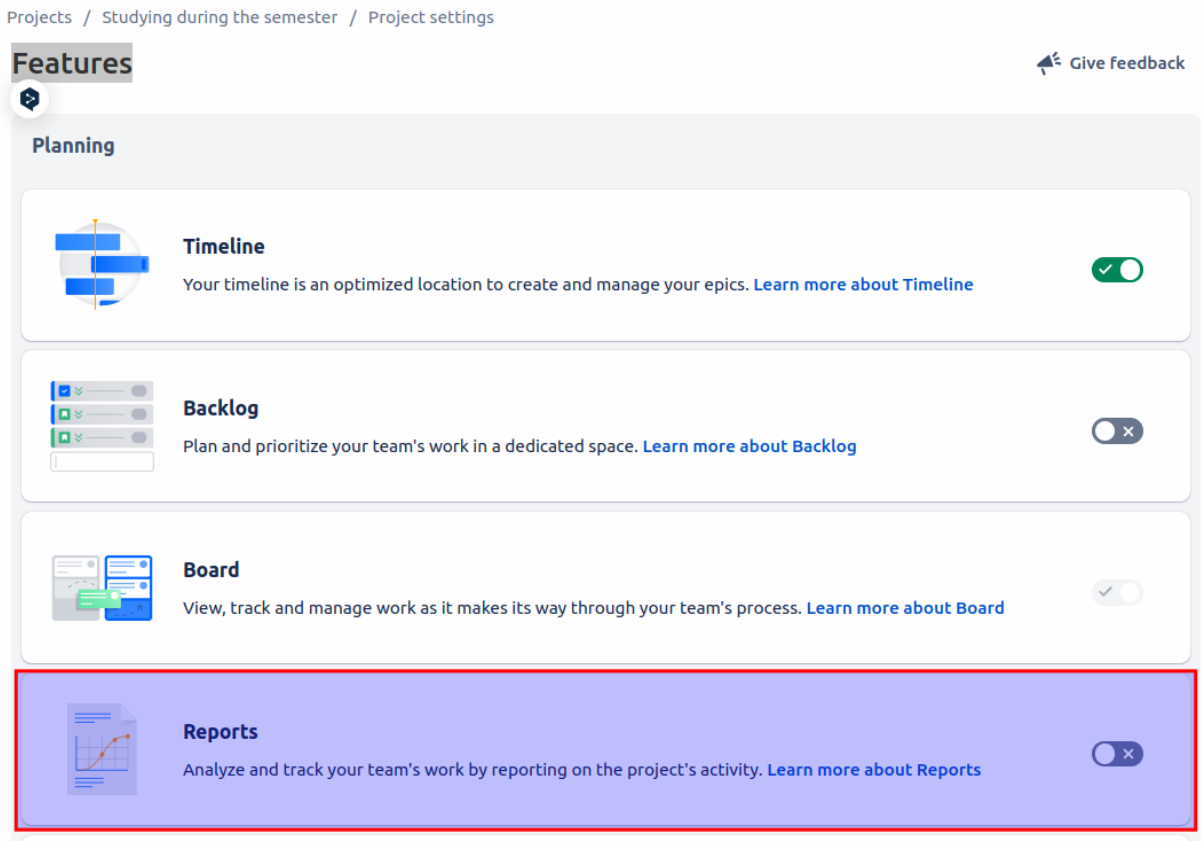


Рисунок 2.5 — Увімкнення звітів

2.4 Вихідні дані

Варіанти вихідних даних до завдання доступні за посиланням [13].

2.5 Приклад виконання роботи

В якості предметної галузі оберемо процес виконання здобувачем робіт за різними дисциплінами впродовж навчання протягом семестру. Епіками проєкту є дисципліни, завданнями — практичні, лабораторні, розрахунково-графічні роботи тощо.

Відповідно до обраної предметної галузі, у системі Jira створимо проєкт Канбан типу Team-managed під назвою «Studying during the semester» (рисунок 2.6).

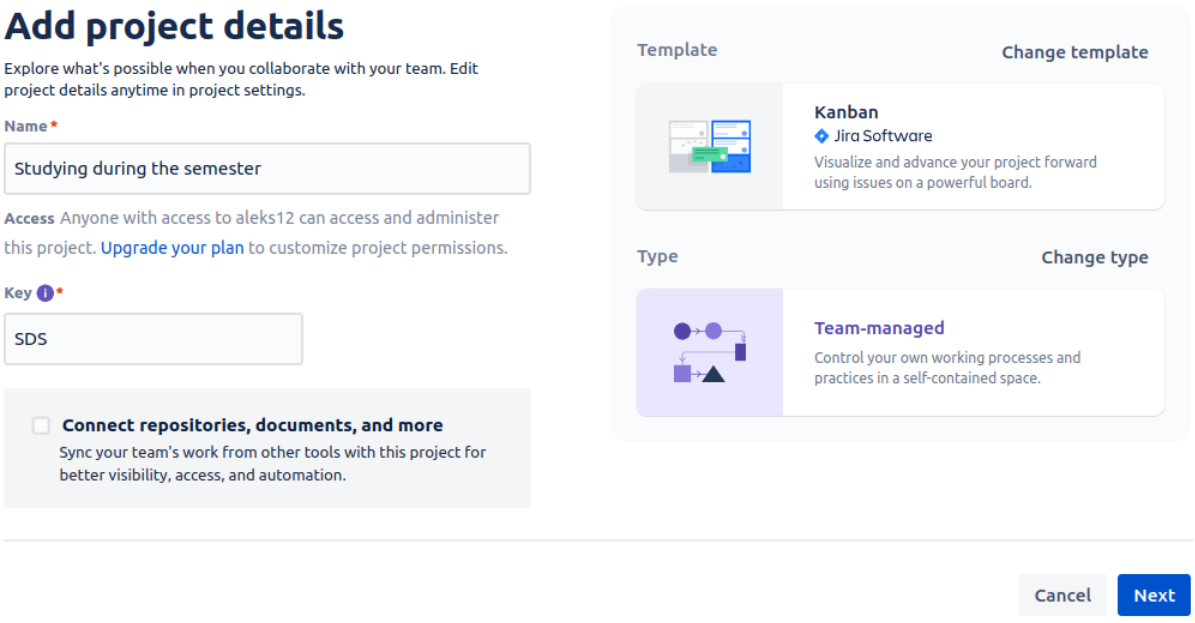


Рисунок 2.6 — Створення проєкту Канбан у системі Jira

До Канбан-дошки, додано такі статуси:

– To Do — завдання, яке ще не розпочато;

– In Progress — завдання в процесі виконання;

– Teacher Assessment — завдання, що перевіряється викладачем;

– Completed — завершене завдання.

На рисунку 2.7 показано адаптований робочий процес із доданими статусами і можливими переходами завдань між ними. До прикладу, завдання із колонки зі статусом Teacher Assessment може бути переміщеним лише в колонку зі статусом In Progress (коли викладач вважає, що робота потребує доопрацювання) або в колонку зі статусом Completed (коли викладач приймає роботу), а перехід у колонку зі статусом To Do заборонено.

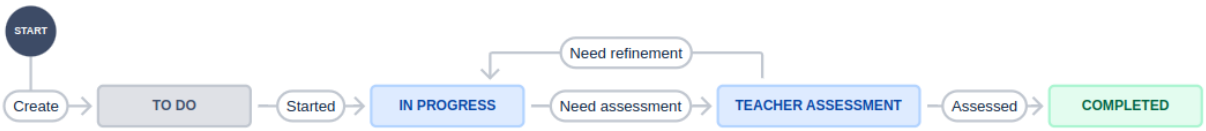


Рисунок 2.7 — Робочий процес, що адаптовано до предметної галузі

До проєкту додамо 3 епіки, що відповідають навчальним дисциплінам: Вища математика, Програмування та Загальний курс транспорту. До епіків додано 8 завдань, що відповідають студентським активностям в межах цих дисциплін (рисунок 2.8).

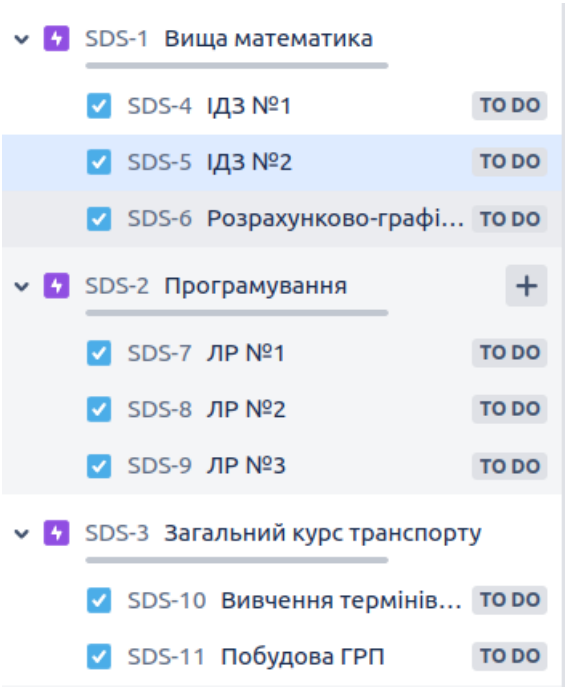


Рисунок 2.8 — Епіки та завдання

Для кожного з завдань встановимо поточний пріоритет (рисунок 2.9).

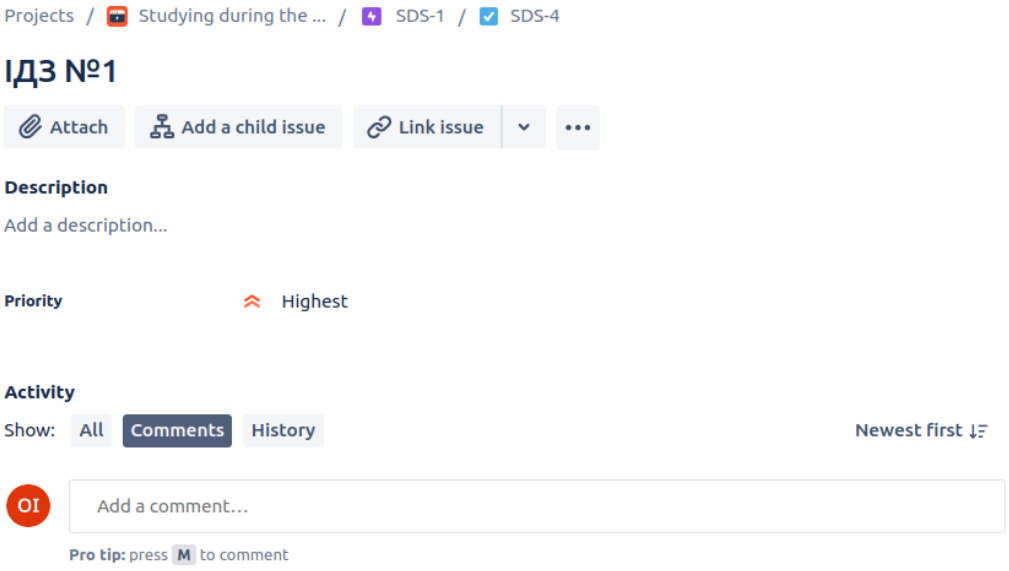


Рисунок 2.9 — Приклад завдання ІДЗ №1 із заданим значенням поля пріоритет

Для епіків встановимо терміни виконання (рисунок 2.10).

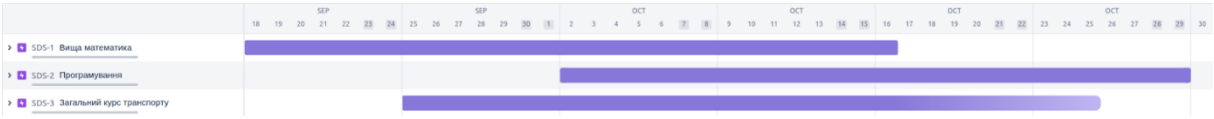


Рисунок 2.10 — Хронологія проєкту

Проведемо імітацію процесу роботи над проєктом. Результати наведено на рисунках 2.11–2.12.

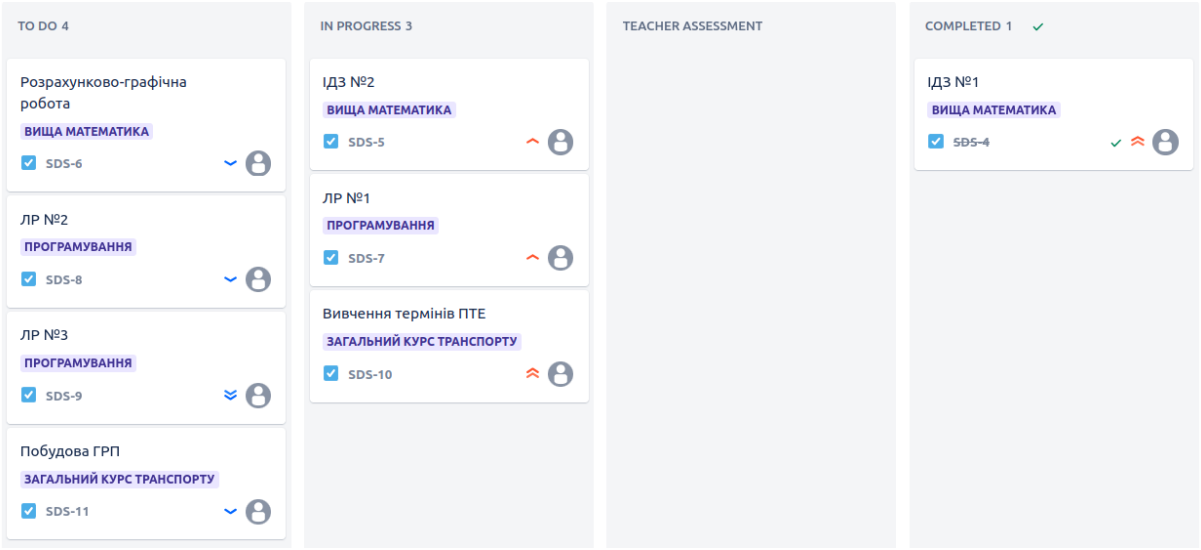


Рисунок 2.11 — Стан Канбан-дошки в процесі моделювання 1

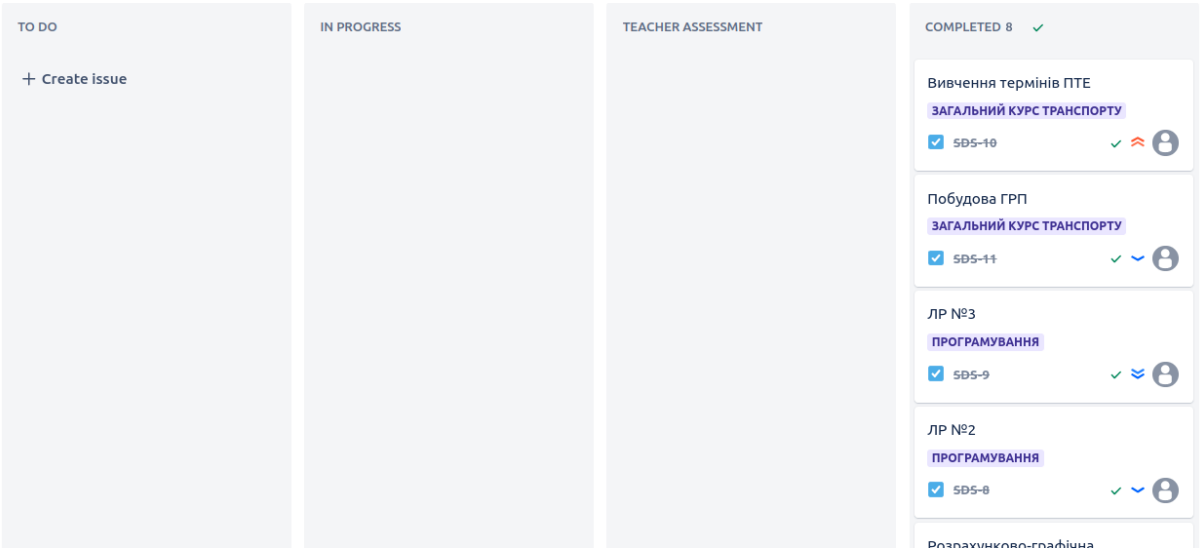


Рисунок 2.12 — Стан Канбан-дошки в процесі моделювання 2

Кумулятивну схему виконання завдань проєкту наведено на рисунку 2.13.

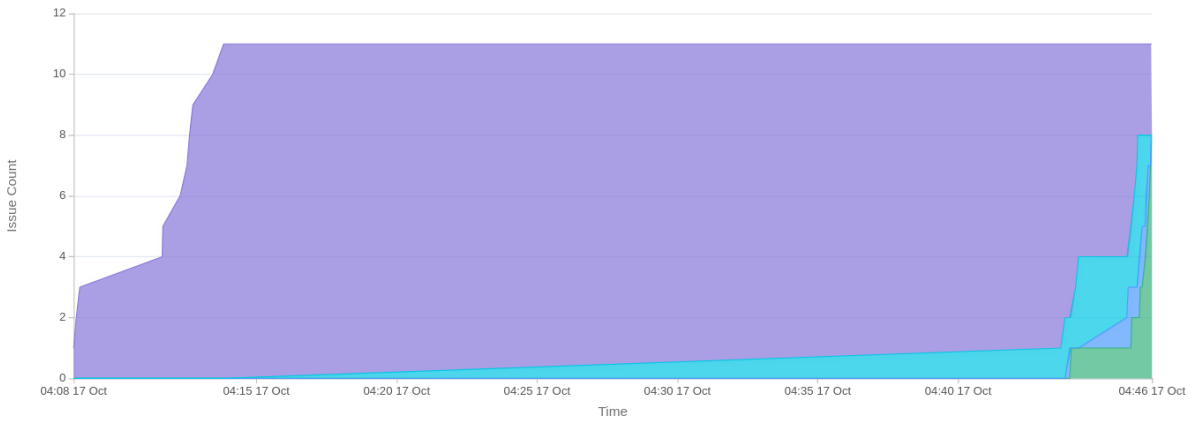


Рисунок 2.13 — Кумулятивна схема виконання завдань проєкту

Висновки. У роботі створено проєкт у системі Jira. Предметна галузь проєкту — процес виконання здобувачем робіт за різними дисциплінами впродовж навчання протягом семестру. Налаштовано робочий процес проєкту, додано необхідні статуси та обмежено переходи між ними. Створено епіки, що відповідають дисциплінам семестру, та завдання, що позначають студентські активності в межах відповідних дисциплін. Для епіків встановлено часові межі, а для завдань — пріоритети, що позначають важливість. Проведено імітацію виконання проєкту — пересування завдань відповідно до термінів, пріоритетів і ролей. Наведено кумулятивну схему виконання завдань проєкту.

Запитання для самоконтролю

1 Що таке методологія Канбан?

2 Як організована Канбан-дошка?

3 Що є горизонтальною віссю Канбан-дошки?

4 Як подаються завдання на Канбан-дошці?

5 Чим відрізняються епіки та завдання?

6 Які переваги надає організація проєкту відповідно до методології Канбан?

Практична робота 3

ПРОЄКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ. СТВОРЕННЯ ДІАГРАМИ ERD

3.1 Мета роботи

Отримати навички проєктування інформаційної системи, навчитись створювати діаграми сутність-зв’язок.

3.2 Теоретичні відомості

Діаграма сутність-зв’язок (Entity-Relationship Diagram, ERD) — це графічне подання сутностей, їхніх атрибутів і зв’язків між ними у межах певної системи. Вона використовується для моделювання структури бази даних на концептуальному рівні та допомагає зрозуміти, як дані організовані та пов’язані між собою.

Основні елементи ERD:

1 Сутність — це об’єкт або поняття, яке може бути ідентифіковане в системі. Сутності можуть бути фізичними об’єктами (наприклад, студент, викладач) або абстрактними поняттями (наприклад, курс, оцінка). Сутності зображуються прямокутниками на діаграмі;

2 Атрибут — це характеристика або властивість сутності. Атрибути надають детальну інформацію про сутності. Атрибути зазвичай вказуються всередині прямокутника сутності у вигляді списку. Наприклад, для сутності «Студент» атрибутами можуть бути: ім’я, прізвище, дата народження;

3 Зв’язок — це асоціація між двома або більше сутностями. Зв’язки описують, як сутності взаємодіють між собою. Зв’язки зображуються лініями, що з’єднують сутності, часто з позначеннями типу зв’язку.

Наприклад, зв’язок «записаний на» може з’єднувати сутність «Студент» із сутністю «Курс». Зв’язки бувають різних типів (таблиця 3.1).

Таблиця 3.1 — Типи зв’язків

Назва зв’язку	**Позначення**	**Опис**	**Приклад**
Нуль або один (0..1)		Зв’язок, при якому екземпляр сутності може бути пов’язаний з одним або жодним екземпляром іншої сутності	Співробітник може мати або не мати закріплене місце для паркування
Один і тільки один (1..1)		Зв’язок, при якому екземпляр сутності має бути пов’язаний рівно з одним екземпляром іншої сутності	Кожен студент має рівно один студентський квиток
Один або багато (1..*)		Зв’язок, при якому екземпляр сутності має бути пов’язаний щонайменше з одним, але можливо з багатьма екземплярами іншої сутності	Кожен курс повинен мати щонайменше одного зареєстрованого студента, але може мати багато студентів
Нуль або багато (0..*)		Зв’язок, при якому екземпляр сутності може бути пов’язаний з нульовим або багатьма екземплярами іншої сутності	Викладач може консультувати нуль або багато студентів

Крім того, можуть використовуватись зв’язки Один (1) та Багато (*) (рисунок 3.1).

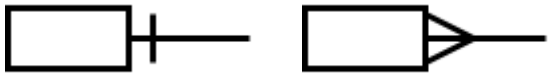


Рисунок 3.1 — Зв’язки типів Один та Багато

У кожної сутності можуть бути ключі, які допомагають унікально ідентифікувати записи та встановлювати зв’язки між сутностями:

– первинний ключ (Primary Key, PK) — це унікальний ідентифікатор кожного екземпляра сутності. Наприклад, id студента є первинним ключем для сутності «Студент»;

– зовнішній ключ (Foreign Key, FK) — це атрибут, що вказує на первинний ключ іншої сутності. Він використовується для встановлення зв’язків між сутностями. Наприклад, id групи у сутності «Студент» може бути зовнішнім ключем, що посилається на первинний ключ у сутності «Група».

3.3 Порядок виконання роботи

1 Дослідити та коротко описати предметну галузь.

2 Для предметної галузі визначити основні сутності та їхні атрибути. Має бути щонайменше 8 сутностей.

3 Визначити зв’язки між сутностями.

4 Створити ERD, використовуючи відповідні інструменти (наприклад, draw.io [2], Lucidchart, MS Visio).

3.4 Вихідні дані

Варіанти вихідних даних до завдання доступні за посиланням [14].

3.5 Приклад виконання роботи

У якості прикладу розглянемо предметну галузь «Університетська інформаційна система (ІС)». Мета цієї системи полягає в управлінні інформацією про студентів, дисципліни, викладачів, групи, розклад занять та оцінки. Для створення ERD визначимо основні сутності, їхні атрибути та зв’язки між ними (таблиця 3.2).

Таблиця 3.2 — Сутності та їхні атрибути університетської ІС

Сутність	Атрибути
Студент (Student)	Ідентифікатор студента (student_id, PK), ім’я (first_name), прізвище (last_name), дата народження (birth_date), ідентифікатор групи (group_id, FK)
Викладач (Teacher)	Ідентифікатор викладача (teacher_id, PK), ім’я (first_name), прізвище (last_name), дата народження (birth_date), посада (job_title), ідентифікатор кафедри (department_id, FK)
Курс (Course)	Ідентифікатор курсу (course_id, PK), назва (title), опис (description), ідентифікатор викладача (teacher_id, FK)
Група (Group)	Ідентифікатор групи (group_id, PK), номер (number), ідентифікатор факультету (faculty_id, FK)
Розклад (Schedule)	Ідентифікатор розкладу (schedule_id, PK), ідентифікатор курсу (course_id, FK), ідентифікатор викладача (teacher_id, FK), ідентифікатор групи (group_id, FK), дата та час (datetime)
Оцінка (Mark)	Ідентифікатор оцінки (mark_id, PK), ідентифікатор студента (student_id, FK), ідентифікатор курсу (course_id, FK), дата (date), значення (value)
Кафедра (Department)	Ідентифікатор кафедри (department_id, PK), назва (title), ідентифікатор факультету (faculty_id, FK)
Факультет (Faculty)	Ідентифікатор факультету (faculty_id, PK), назва (title)

Визначимо зв’язки між сутностями:

– Студент належить до Групи. Один студент належить до однієї групи (1..1), але одна група може включати багато студентів (1..*).

– Викладач належить до Кафедри. Один викладач належить до однієї кафедри (1..1), але одна кафедра може включати багато викладачів (1..*).

– Курс викладається Викладачем. Один курс викладається одним викладачем (1..1), але один викладач може викладати багато курсів (0..*).

– Група належить до Факультету. Одна група належить до одного факультету (1..1), але один факультет може включати багато груп (1..*).

– Кафедра належить до Факультету. Одна кафедра належить до одного факультету (1..1), але один факультет може включати багато кафедр (1..*).

– Розклад стосується Курсів, Викладачів та Груп. Один елемент розкладу належить до одного курсу (1..1), одного викладача (1..1) і однієї групи (1..1), але один курс може мати багато розкладів (0..*), один викладач може мати багато розкладів (0..*), і одна група може мати багато розкладів (*).

– Оцінка стосується Студентів та Курсів. Одна оцінка належить до одного студента (1..1) та одного курсу (1..1), але один студент може мати багато оцінок (0..*), і один курс може мати багато оцінок (0..*).

На основі опису сутностей та зв’язків побудуємо діаграму сутність-зв’язок для університетської ІС (рисунок 3.2).

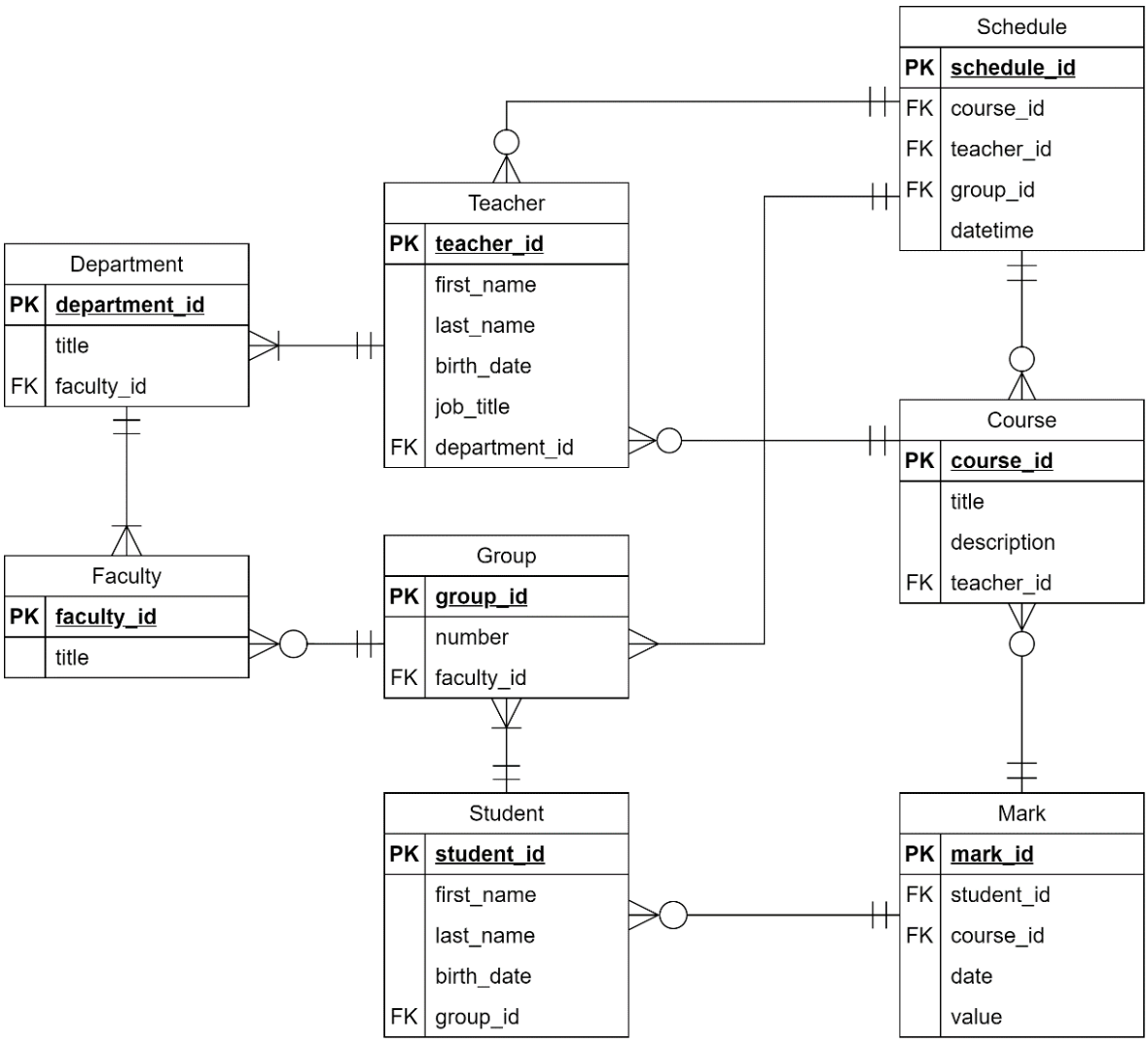


Рисунок 3.2 — Діаграма сутність-зв’язок для університетської ІС

Висновки. У роботі створено діаграму сутність-зв’язок для університетської інформаційної системи. Мета системи полягає в управлінні інформацією про студентів, дисципліни, викладачів, групи, розклад занять та оцінки. Визначено основні сутності: студент, викладач, курс, група, розклад, оцінка, кафедра, факультет, а також їхні атрибути. Побудовано зв’язки між сутностями, що відображають реальні відносини в системі. Отже, ERD для університетської ІС забезпечує ефективне управління та організацію академічної інформації.

Запитання для самоконтролю

1 Що таке діаграма сутність-зв’язок (ERD) і для чого вона використовується?

2 Які основні елементи складають ERD?

3 Що таке сутність в контексті ERD і як вона зображується на діаграмі?

4 Як визначаються атрибути сутності і де вони розміщуються на ERD?

5 Що таке зв’язок в ERD і які типи зв’язків ви знаєте?

6 Як встановити зв’язок між сутностями у діаграмі сутність-зв’язок?

7 Яка роль первинного ключа в ERD і як його позначають?

8 Що таке зовнішній ключ і як він використовується для встановлення зв’язків між сутностями?

9 Які кількісні обмеження можуть бути у зв’язках між сутностями і як вони позначаються на ERD?

10 Які інструменти можна використовувати для створення діаграм сутність-зв’язок і які переваги кожного з них?

Практична робота 4

ПРОЄКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ. ВИКОРИСТАННЯ МЕТОДОЛОГІЇ IDEF0

4.1 Мета роботи

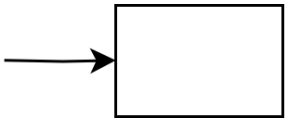
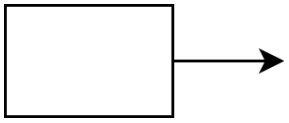
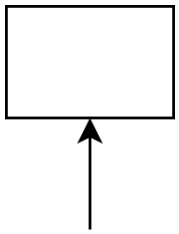
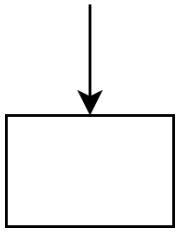
Отримати навички проєктування інформаційної системи, використовуючи методологію моделювання IDEF0.

4.2 Теоретичні відомості

IDEF0 (Integration DEFinition for Function Modeling) — це методологія моделювання, яка використовується для аналізу і розроблення систем та їхніх процесів. Вона надає графічне представлення функціональної моделі, що відображає функції та взаємодії між ними. IDEF0 діаграми є ефективним інструментом для документування, аналізу, реінжинірингу та проєктування бізнес-процесів. У контексті інформаційних систем і технологій IDEF0 допомагає моделювати бізнес-процеси, відображати потоки інформації, взаємодії між різними системами та компонентами, а також визначати ресурси, необхідні для виконання функцій. Вона забезпечує систематичний підхід до розроблення та вдосконалення інформаційних систем, сприяє зрозумілості та узгодженості процесів.

Основні елементи IDEF0 наведені у таблиці 4.1.

Таблиця 4.1 — Основні елементи IDEF0

Елемент	**Опис**	**Графічне зображення**
Функція	Основні дії або процеси, які виконуються в системі. Вони зображуються прямокутниками	
Входи	Матеріали, інформація або продукти, які необхідні для виконання функції. Зображуються стрілками, що спрямовані на функцію зліва	
Виходи	Результати або продукти, які отримуються в результаті виконання функції. Зображуються стрілками, що виходять з функції справа	
Механізми	Ресурси, які використовуються для виконання функції (люди, обладнання). Зображуються стрілками, що спрямовані на функцію знизу	
Управління	Правила, політики, процедури та обмеження, які визначають, як виконується функція. Зображуються стрілками, що спрямовані на функцію зверху	

Контекстна діаграма — це найвищий рівень абстракції в IDEF0 моделі, який представляє загальну картину системи або процесу. Вона показує основну функцію системи та її взаємодію з зовнішнім середовищем через входи, виходи, механізми та управління. Контекстна діаграма допомагає зрозуміти взаємозв'язки між основними бізнес-процесами та інформаційними потоками, а також визначити основні функції та ресурси, необхідні для їхнього виконання. Контекстна діаграма зображає основну функцію (A-0), яка описує головний процес або діяльність.

Декомпозиція — це процес розбиття основної функції на підфункції для більш детального аналізу. Вона дає змогу більш детально моделювати процеси і зрозуміти взаємодію між підфункціями. Кожна підфункція може

бути представлена на окремій діаграмі з рівнем деталізації, необхідним для аналізу. Декомпозиція функцій дає змогу виявити підпроцеси, що складають загальні бізнес-процеси, оптимізувати їх, визначити місця інтеграції різних інформаційних систем та технологій, а також забезпечити ефективне управління ресурсами та потоками інформації.

4.3 Порядок виконання роботи

1 Дослідити та коротко описати предметну галузь.

2 Визначити основну функцію для предметної галузі.

3 Визначити входи, виходи, механізми та управління для основної функції.

4 Створити контекстну діаграму (A-0) для основної функції, використовуючи відповідні інструменти (наприклад, draw.io [2], Lucidchart, MS Visio).

5 Виконати декомпозицію основної функції на підфункції. Має бути 3-6 підфункцій.

6 Визначити входи, виходи, механізми та управління для кожної підфункції.

7 Створити деталізовану діаграму для підфункцій.

4.4 Вихідні дані

Варіанти вихідних даних до завдання доступні за посиланням [15].

4.5 Приклад виконання роботи

У якості прикладу розглянемо предметну галузь «Система управління навчальним процесом студентів». Мета цієї системи полягає в управлінні інформацією про студентів, курси, викладачів, відвідуваність, оцінки та

звіти про академічну успішність. Для створення діаграми IDEF0 визначимо основну функцію, підфункції, входи, виходи, механізми та управління.

Основна функція — управляти навчальним процесом студентів. Входи, виходи, механізми та управління для основної функції наведено в таблиці 4.2.

Таблиця 4.2 — Входи, виходи, механізми та управління основної функції

Елемент	Опис
Входи	Заявки на реєстрацію, навчальний план, дані про відвідуваність, результати модульного контролю
Виходи	Оцінки, звіти про успішність, аналітичні дані
Механізми	Система реєстрації курсів, система управління розкладом, викладачі, система обліку відвідування, система звітування, система аналітики
Управління	Правила реєстрації, академічні календарі, політика відвідуваності, критерії оцінювання, стандарти звітності, методи аналізу

Контекстна діаграма наведена на рисунку 4.1.

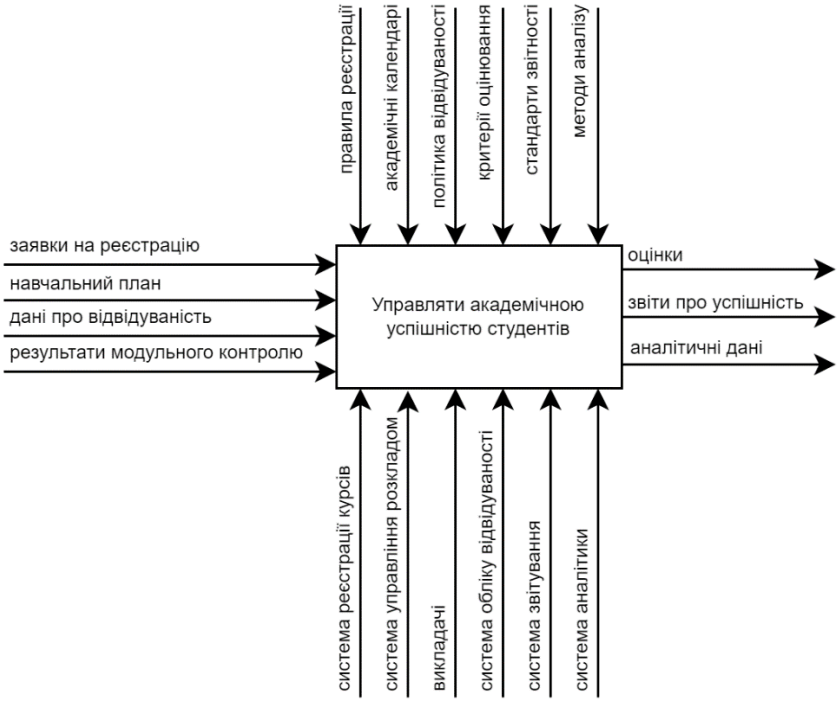


Рисунок 4.1 — Контекстна діаграма (A-0)

Виконаємо декомпозицію основної функції на підфункції та визначимо їхні входи, виходи, механізми та управління (таблиця 4.3).

Таблиця 4.3 — Входи, виходи, механізми та управління підфункції

Підфункція	**Входи**	**Виходи**	**Механізми**	**Управління**
A1: Зареєструвати студентів на курси	Заявки на реєстрацію	Зареєстровані студенти	Система реєстрації курсів	Правила реєстрації
A2: Створити розклад занять	Зареєстровані студенти, навчальний план	Розклад занять	Система управління розкладом	Академічні календарі
A3: Відстежити відвідування	Розклад занять, дані про відвідуваність	Звіти про відвідуваність	Викладачі, система обліку відвідуваності	Політика відвідуваності
A4: Оцінити студентів	Зареєстровані студенти, результати контрольних робіт, іспитів	Оцінки	Викладачі	Критерії оцінювання
A5: Сформувати звіти про успішність	Оцінки, звіти про відвідуваність	Звіти про успішність	Система звітування	Стандарти звітності
A6: Провести аналітичний аналіз успішності	Звіти про успішність	Аналітичні дані	Система аналітики	Методи аналізу

Діаграма для підфункцій наведена на рисунку 4.2.

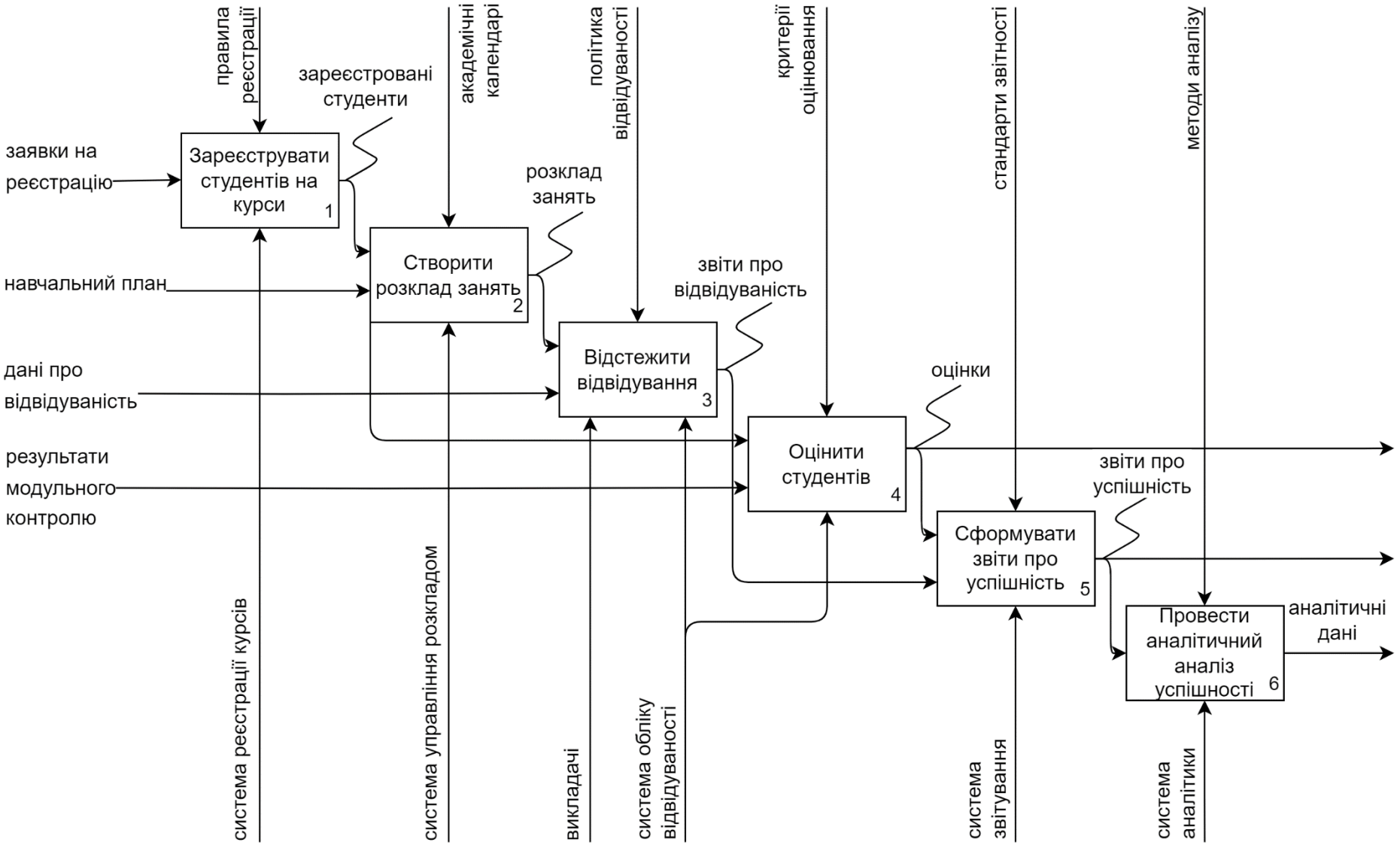


Рисунок 4.2 — Деталізована діаграма для підфункції

Висновки. У роботі створено діаграму IDEF0 для системи управління академічною успішністю студентів. Визначено основну функцію «Управління академічною успішністю студентів», а також входи, виходи, механізми та управління для неї. Виконано декомпозицію основної функції на підфункції: реєстрація студентів на курси, ведення розкладу занять, ведення обліку відвідуваності, оцінювання студентів, формування звітів про успішність, аналітичний аналіз успішності. Створено детальні діаграми для кожної підфункції.

Запитання для самоконтролю

1 Що таке IDEF0 і які її основні цілі?
2 Які основні елементи містить діаграма IDEF0?
3 Як визначаються функції в IDEF0?
4 Які типи зв’язків існують в діаграмах IDEF0?
5 Що таке контекстна діаграма (A-0) і яка її роль у IDEF0?
6 Як входи відображаються в IDEF0 діаграмах?
7 Як виходи відображаються в діаграмах IDEF0?
8 Яка роль механізмів у діаграмах IDEF0?
9 Що таке управління в контексті IDEF0 і як воно відображається?
10 Як виконується декомпозиція функцій в IDEF0?

Практична робота 5

АНАЛІЗ ДАНИХ. ВИКОРИСТАННЯ МЕТОДІВ РЕГРЕСІЇ

5.1 Мета роботи

Отримати навички аналізу даних, використовуючи моделі лінійної регресії та поліноміальної регресії.

5.2 Теоретичні відомості

Регресія — це завдання статистики, аналізу даних та машинного навчання, що полягає у моделюванні залежності між цільовою змінною (y) і вхідними ознаками $(x_1, x_2, ..., x_n)$, коли така залежність є невідомою в аналітичному вигляді (невідома функція $y = \mathrm{f}(x_1, x_2, ..., x_n)$), але при цьому є відомим (експериментально виміряним) набір даних (відомо які значення приймає змінна (y) при конкретних значеннях ознак $(x_1, x_2, ..., x_n)$). У найпростішому випадку ознака може бути одна — (x).

Розглянемо приклад завдання регресії. Дано набір зі 100 вимірювань $(x^{(i)}, y^{(i)})$. Візуалізуємо цей набір даних за допомогою діаграми розсіювання (рисунок 5.1).

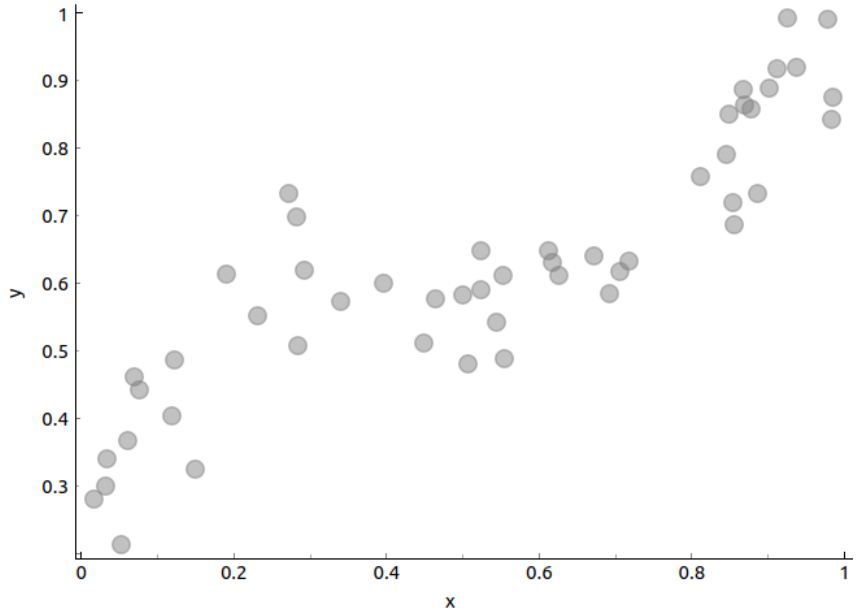


Рисунок 5.1 — Діаграма розсіювання набору даних

Завдання регресії полягає у знаходженні функції $y = \mathrm{f}(x)$, яка найкраще описує цей набір даних. Приклад такої функції наведено на рисунку 5.2.

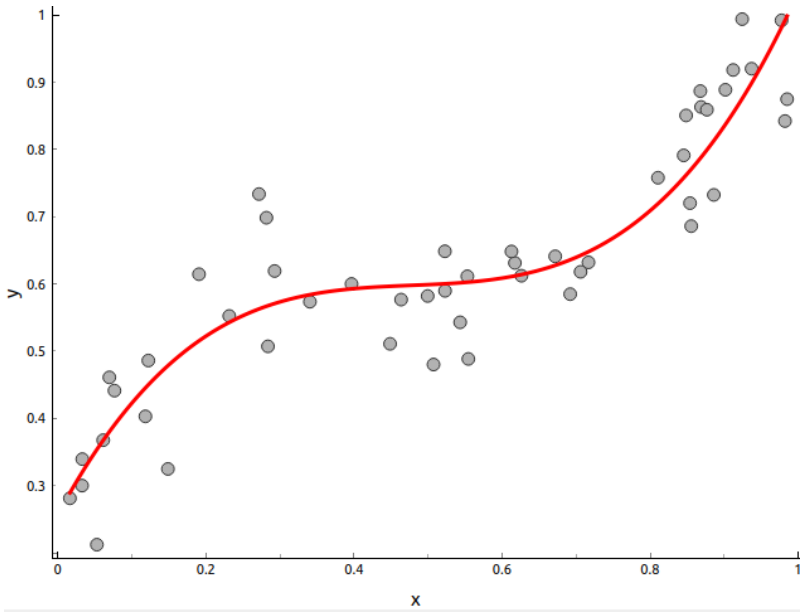


Рисунок 5.2 — Приклад моделі, що описує набір даних

Базовим засобом вирішення завдання регресії є метод лінійної регресії. Цей метод заснований на гіпотезі про те, що залежність між

цільовою змінною (y) і вхідною ознакою (x) є лінійною. Отже, вирішення завдання регресії за допомогою методу лінійної регресії зводиться до знаходження параметрів лінійної функції, що найкраще описує дані:

$$y = \theta_0 + \theta_1 x$$

Існує багато методів знаходження параметрів θ_0, θ_1. Класичним є метод найменших квадратів (МНК).

$$\theta_1 = \frac{\sum_i (x_i - \overline{x})(y_i - \overline{y})}{\sum_i (x_i - \overline{x})^2}$$

$$\theta_0 = \overline{y} - \theta_1 \overline{x}$$

У випадку, коли постає необхідність описати залежність між цільовою зміною і вхідною ознакою за допомогою нелінійної функції, застосовують модифікацію лінійної регресії — поліноміальну регресію. Цей метод полягає у додаванні до набору даних фіктивних ознак, які є результатами застосування нелінійних функцій (часто ступеневих) до відомих ознак. До розширеного набору даних також застосовують МНК.

5.3 Порядок виконання роботи

1 Встановити пакет Orange Data Mining. Пакет є у вільному доступі та може бути завантажений за посиланням [1]. У пакеті необхідно встановити додатковий модуль Educational (Options → Add-ons → Educational → OK). Завантажити у середовище набір навчальних даних (train data) за варіантом.

2 Візуалізувати навчальні дані за допомогою діаграми розсіювання.

3 Висунути гіпотезу про вид залежності в даних (лінійна залежність або поліноміальна певного порядку).

4 Побудувати, візуалізувати і знайти параметри таких регресійних моделей:

а) модель лінійної регресії;

б) модель поліноміальної регресії другого порядку;

в) модель поліноміальної регресії третього порядку.

5 Зробити висновок про те, яка модель є найбільш доцільною.

6 Завантажити у середовище набір тестових даних (test data) за варіантом.

7 За допомогою найбільш доцільної моделі отримати прогнозні значення для набору тестових даних.

5.4 Вихідні дані

Варіанти вихідних даних до завдання доступні за посиланням [16].

5.5 Приклад виконання роботи

Для виконання роботи використаємо пакет Orange Data Mining. Створимо схему з таких віджетів:

– File — віджет для завантаження даних;

– Data Table — віджет для відображення даних;

– Scatter Plot — віджет для побудови діаграми розсіювання;

– Polynomial Regression — віджет для побудови регресійної моделі;

– Predictions — віджет для отримання прогнозів моделі на нових (тестових) даних.

Схему наведено на рисунку 5.3.

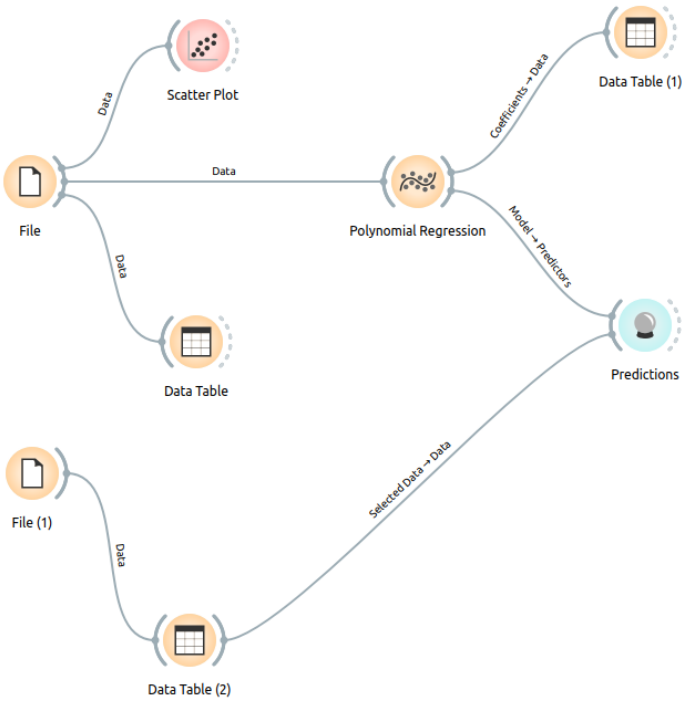
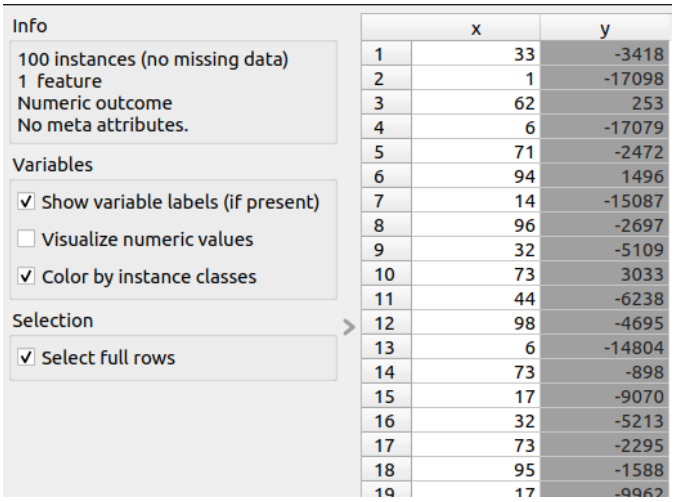


Рисунок 5.3 — Схема у Orange Data Mining

Завантажимо набір навчальних даних, використовуючи віджет File. Ряд *x* позначимо як ознаку (feature), а ряд *y* як цільову змінну (target). Для перегляду набору навчальних даних скористаємося віджетом Data Table (рисунок 5.4).



	x	y
1	33	-3418
2	1	-17098
3	62	253
4	6	-17079
5	71	-2472
6	94	1496
7	14	-15087
8	96	-2697
9	32	-5109
10	73	3033
11	44	-6238
12	98	-4695
13	6	-14804
14	73	-898
15	17	-9070
16	32	-5213
17	73	-2295
18	95	-1588
19	17	-9962

Рисунок 5.4 — Перегляд набору навчальних даних

За допомогою віджета Scatter Plot побудуємо діаграму розсіювання набору навчальних даних (рисунок 5.5).

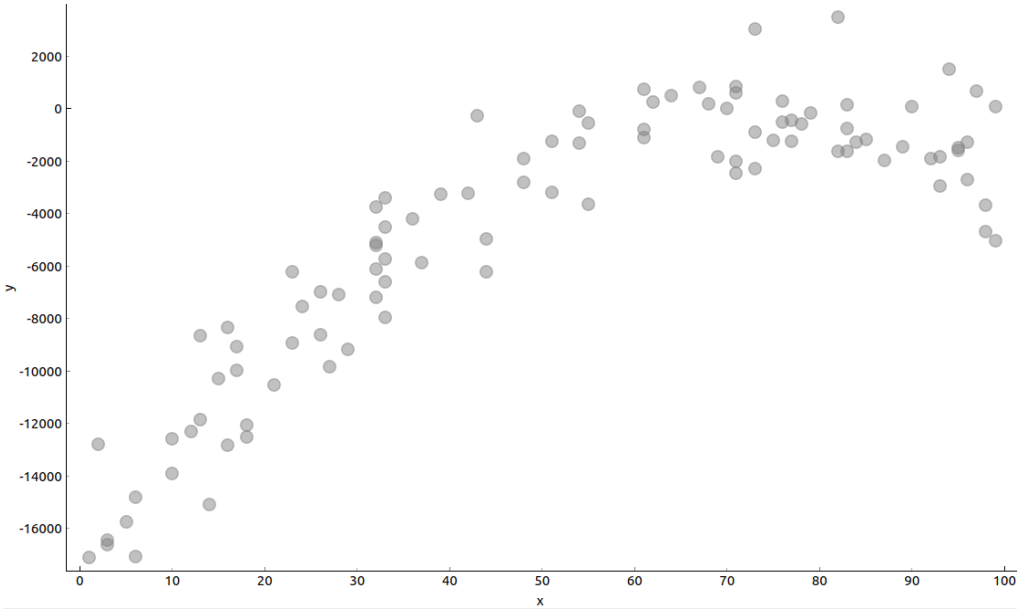


Рисунок 5.5 — Діаграма розсіювання набору навчальних даних

За діаграмою розсіювання можемо висунути гіпотезу про наявність в даних поліноміальної залежності другого порядку.

У віджеті Polynomial Regression встановимо значення ступеня поліному 1, отримавши модель лінійної регресії (рисунок 5.6).

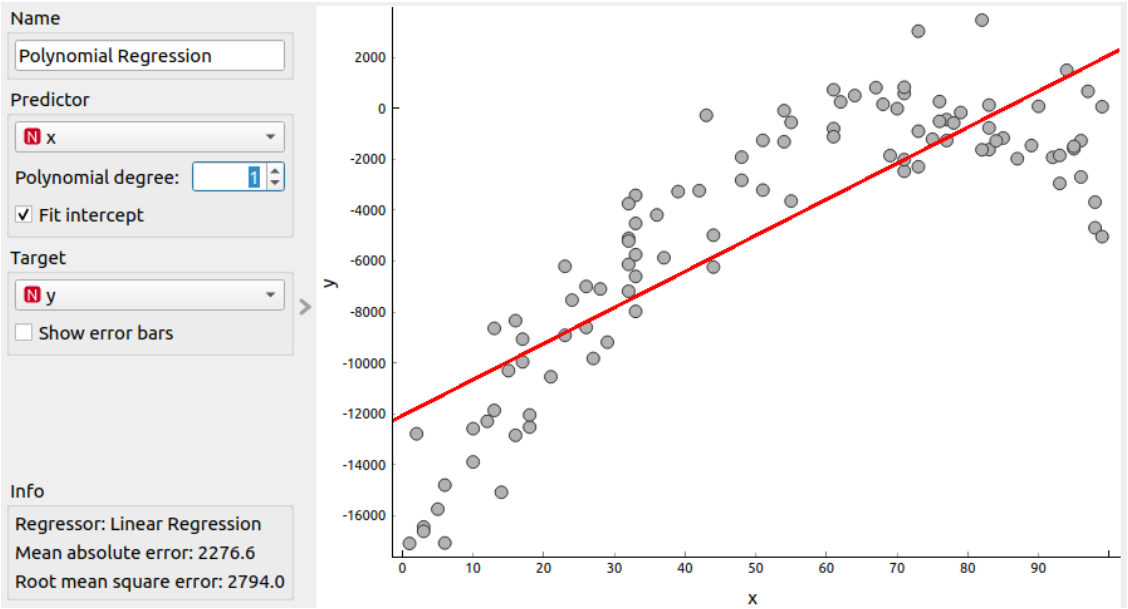


Рисунок 5.6 — Модель лінійної регресії, що описує набір даних

Параметри моделі лінійної регресії встановимо із віджета Data Table (1): $\theta_0 = -12075.7$, $\theta_1 = 141.63$.

У віджеті Polynomial Regression встановимо значення ступеня поліному 2, отримавши модель поліноміальної регресії другого порядку (рисунок 5.7).

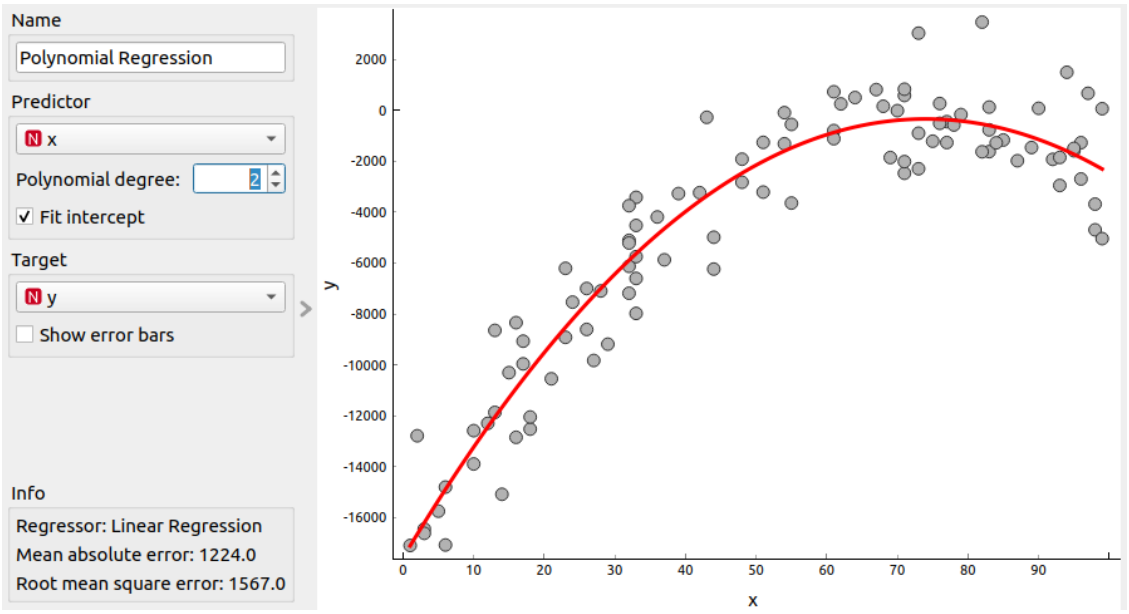


Рисунок 5.7— Модель поліноміальної регресії другого порядку, що описує набір даних

Параметри моделі поліноміальної регресії другого порядку: $\theta_0 = -17576$, $\theta_1 = 465.87$, $\theta_2 = -3.15$.

У віджеті Polynomial Regression встановимо значення ступеня поліному 3, отримавши модель поліноміальної регресії третього порядку (рисунок 5.8).

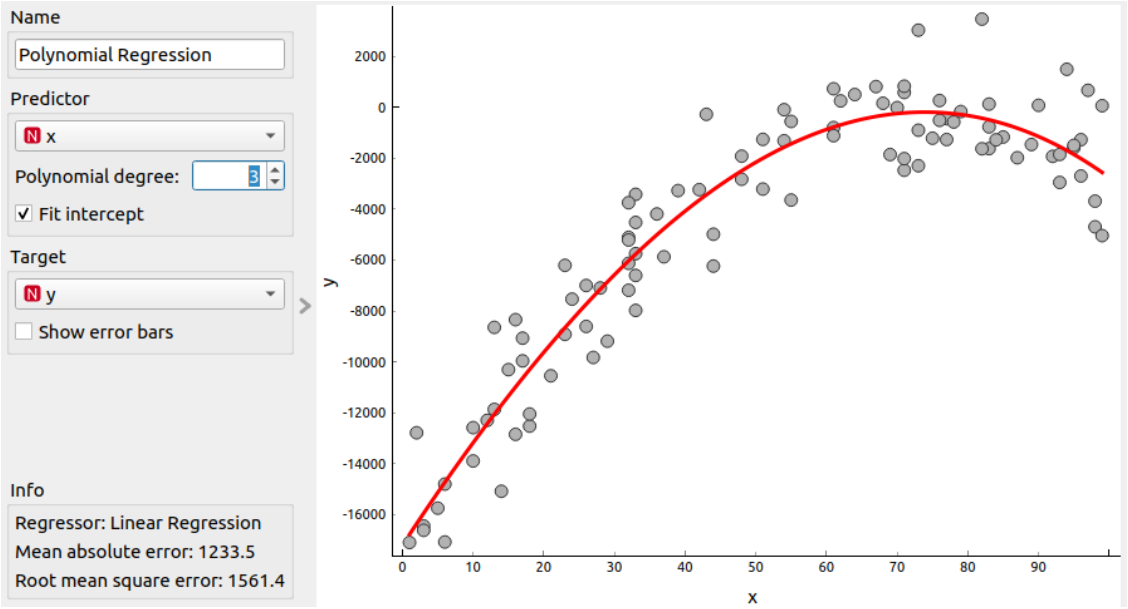


Рисунок 5.8 — Модель поліноміальної регресії третього порядку, що описує набір даних

Параметри моделі поліноміальної регресії третього порядку: $\theta_0 = -17203.3$ $\theta_1 = 424.25$ $\theta_2 = -2.14$ $\theta_3 = -0.007$.

За візуалізаціями можемо зробити висновок про те, що модель лінійної регресії гірше описує дані, ніж поліноміальні моделі, оскільки не враховує нелінійних залежностей. Модель поліноміальної регресії другого порядку описує дані достатньо точно. Застосування моделей вищих порядків недоцільне.

Завантажимо тестові дані, використовуючи віджет File (1). Для перегляду тестових даних скористаємося віджетом Data Table (2) (рисунок 5.9).

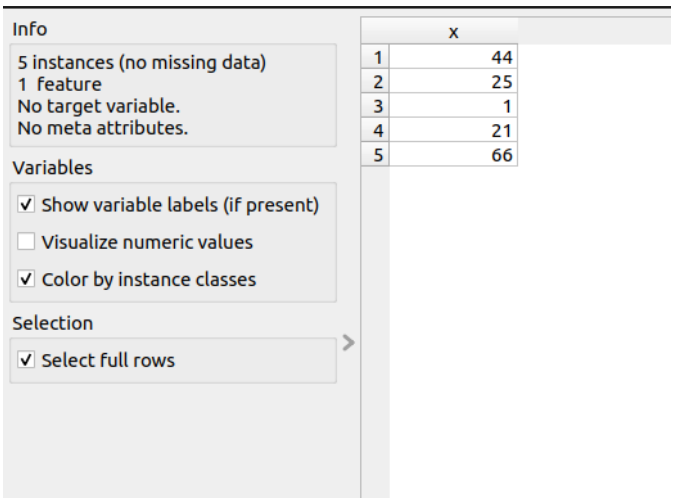


Рисунок 5.9 — Набір тестових даних

Отримаємо прогнози моделі поліноміальної регресії другого порядку на тестових даних, використовуючи віджет Predictions (рисунок 5.10).

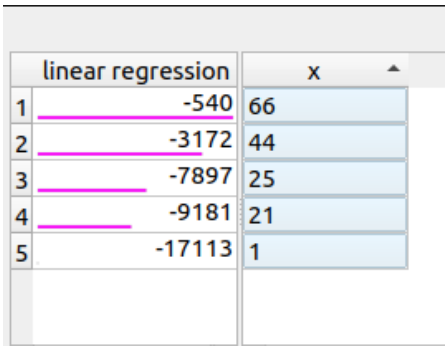


Рисунок 5.10 — Прогнози моделі на тестових даних

Висновки. У роботі виконано аналіз даних, з використанням регресійних моделей. Застосовано моделі лінійної регресії та поліноміальної регресії.

Запитання для самоконтролю

1 Що таке завдання регресії в машинному навчанні?

2 Що таке метод лінійної регресії?

3 Які основні компоненти моделі лінійної регресії?

4 На якій гіпотезі засновано метод лінійної регресії?

5 Як припущення про лінійність впливають на результати моделювання?

6 Чим відрізняється метод лінійної регресії від методу поліноміальної регресії?

7 Для чого потрібно два набори даних: навчальний та тестовий?

8 Як можна пересвідчитись у якості регресійної моделі?

9 Як впливає кількість спостережень на точність моделі регресії?

10 Чому великий обсяг даних може бути критично важливим для моделі?

Практична робота 6

АНАЛІЗ ДАНИХ. ВИКОРИСТАННЯ МЕТОДІВ КЛАСИФІКАЦІЇ

6.1 Мета роботи

Отримати навички аналізу даних, використовуючи метод k найближчих сусідів.

6.2 Теоретичні відомості

Класифікація — це одне з основних завдань статистики, аналізу даних та машинного навчання, що містить у собі моделювання залежності між категоріальною цільовою змінною (y) і набором вхідних ознак $(x_1, x_2, ..., x_n)$. Це моделювання виконується, коли аналітична форма залежності $y = \mathrm{f}(x_1, x_2, ..., x_n)$ є невідомою, але при цьому доступний експериментально виміряний набір даних, де відомо, які категорії приймає змінна (y) у при певних значеннях ознак $(x_1, x_2, ..., x_n)$. Таке завдання може містити одну або декілька вхідних ознак.

Розглянемо ілюстративний приклад класифікації: дано набір зі 100 прикладів $(x^{(i)}, y^{(i)})$. Для наочності можна візуалізувати цей набір даних за допомогою діаграми розсіювання, як показано на рисунку 6.1.

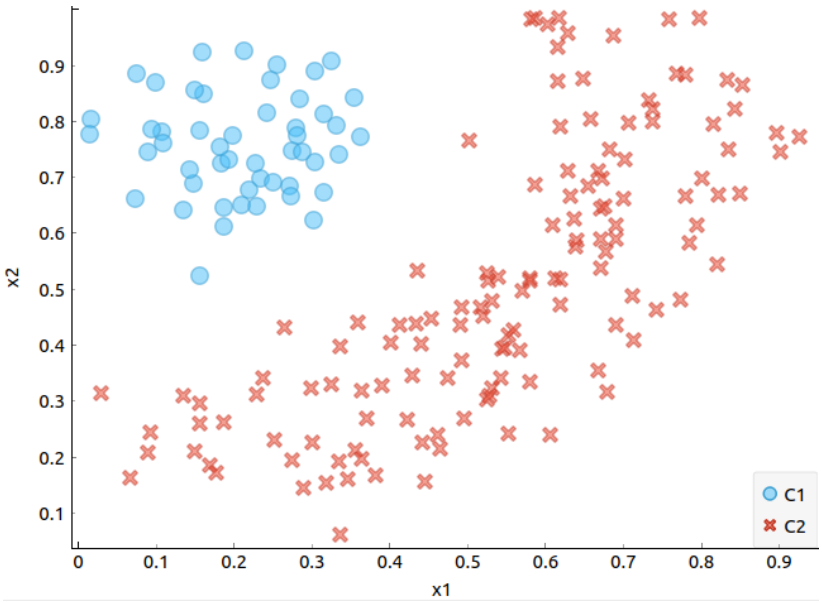


Рисунок 6.1 — Діаграма розсіювання набору даних

Завдання класифікації полягає у визначенні функції $y = f(x_1, x_2)$, яка найточніше описує наявний набір даних і може ефективно класифікувати нові приклади, не включені в навчальний набір (рисунок 6.2).

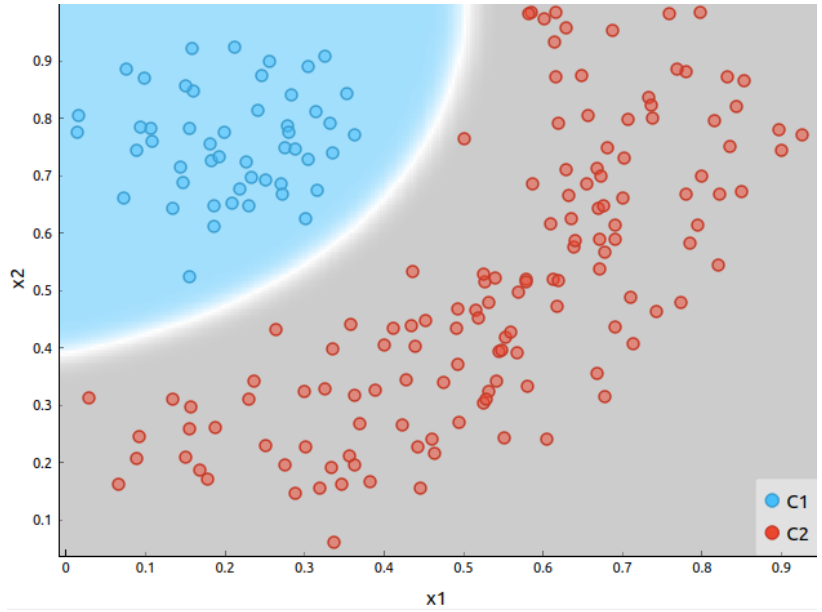


Рисунок 6.2 — Приклад моделі, що класифікує дані

Серед базових методів вирішення завдання класифікації вирізняють метод k найближчих сусідів (kNN): клас нового прикладу визначається як

найбільш поширений клас серед *k* найближчих сусідів у навчальному наборі даних. Цей метод не вимагає будування експліцитної моделі, але є чутливим до вибору параметра *k* та масштабування ознак.

Для оцінки точності та ефективності цих моделей використовується валідаційний набір даних, який зазвичай формується шляхом виділення приблизно 20 % від загального навчального набору. Точність кожної моделі на валідаційному наборі дає змогу не тільки порівняти їхню здатність до класифікації, але й визначити, наскільки добре модель здатна узагальнювати нові дані, не беручи участь у навчальному процесі.

Такий підхід допомагає уникнути перенавчання та забезпечує об’єктивну оцінку ефективності класифікаційних моделей перед їхнім застосуванням у реальних умовах.

6.3 Порядок виконання роботи

1 Встановити пакет Orange Data Mining. Пакет є у вільному доступі та може бути завантажений за посиланням [1].

2 Завантажити у середовище набір навчальних даних (train data) за варіантом.

3 Візуалізувати навчальні дані за допомогою діаграми розсіювання.

4 Створити три моделі методу kNN, встановлюючи для *k* значення 3, 5, та 7 сусідів відповідно.

5 Розділити набір даних на основний набір для навчання (80 % загальної кількості даних) та валідаційний набір (20 % загальної кількості даних).

6 Оцінити точність кожної моделі на валідаційному наборі даних.

7 Порівняти точність моделей та визначити, яка з них є найефективнішою на використаному наборі даних.

8 Побудувати та проаналізувати матриці невідповідностей для кожної моделі.

6.4 Вихідні дані

Варіанти вихідних даних до завдання доступні за посиланням [17].

6.5 Приклад виконання роботи

Для виконання роботи використаємо пакет Orange Data Mining. Створимо схему з таких віджетів:

– File — віджет для завантаження даних;

– Data Table — віджет для відображення даних;

– Scatter Plot — віджет для побудови діаграми розсіювання;

– kNN — віджет для побудови моделі kNN;

– Test and Score — віджет для отримання показників ефективності моделей класифікації на тестових (валідаційних) даних, порівнює декілька моделей за певним показником;

– Confusion Matrix — віджет, що показує кількості об’єктів, класифікованих моделлю правильно та помилково.

Схему наведено на рисунку 6.3.

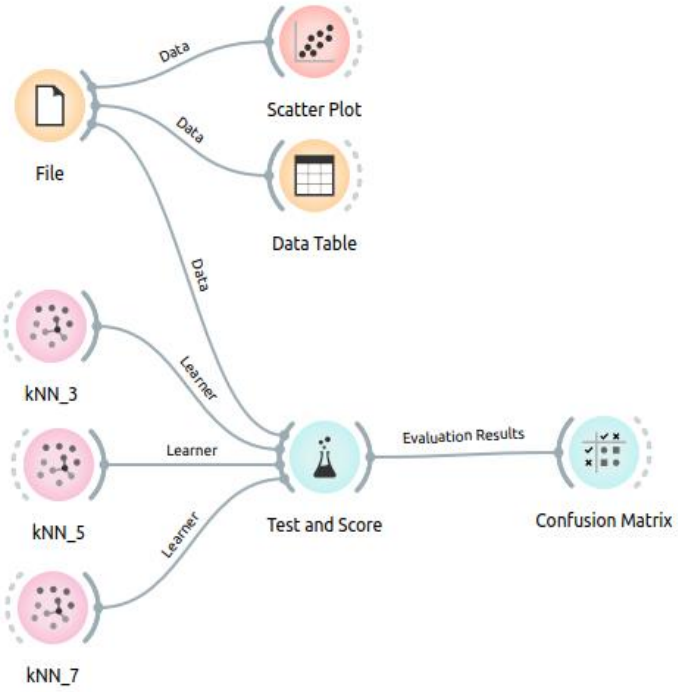


Рисунок 6.3 — Схема в Orange Data Mining

Завантажимо набір навчальних даних, використовуючи віджет File. Для рядів x_1 та x_2 встановимо тип number та позначимо їх як ознаки, для ряду y встановимо тип categorical та позначимо його як цільову змінну (target) (рисунок 6.4).

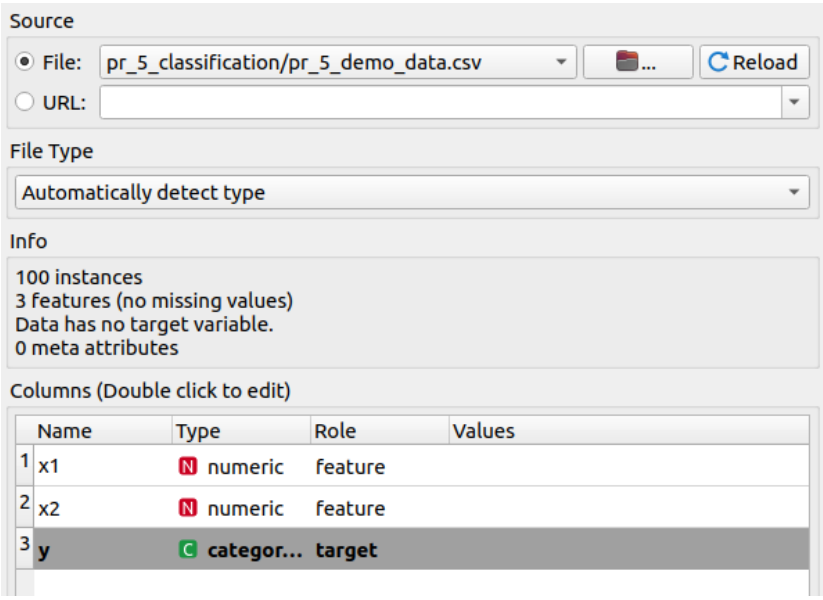
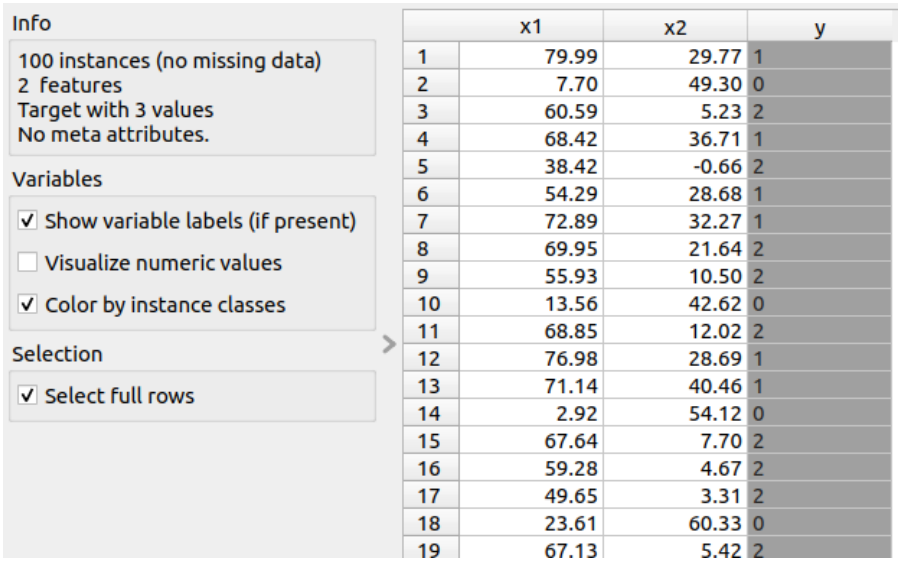


Рисунок 6.4 — Завантаження набору навчальних даних

Для перегляду набору навчальних даних скористаємося віджетом Data Table (рисунок 6.5).



	x1	x2	y
1	79.99	29.77	1
2	7.70	49.30	0
3	60.59	5.23	2
4	68.42	36.71	1
5	38.42	-0.66	2
6	54.29	28.68	1
7	72.89	32.27	1
8	69.95	21.64	2
9	55.93	10.50	2
10	13.56	42.62	0
11	68.85	12.02	2
12	76.98	28.69	1
13	71.14	40.46	1
14	2.92	54.12	0
15	67.64	7.70	2
16	59.28	4.67	2
17	49.65	3.31	2
18	23.61	60.33	0
19	67.13	5.42	2

Рисунок 6.5 — Перегляд набору навчальних даних

За допомогою віджета Scatter Plot побудуємо діаграму розсіювання набору навчальних даних (рисунок 6.6).

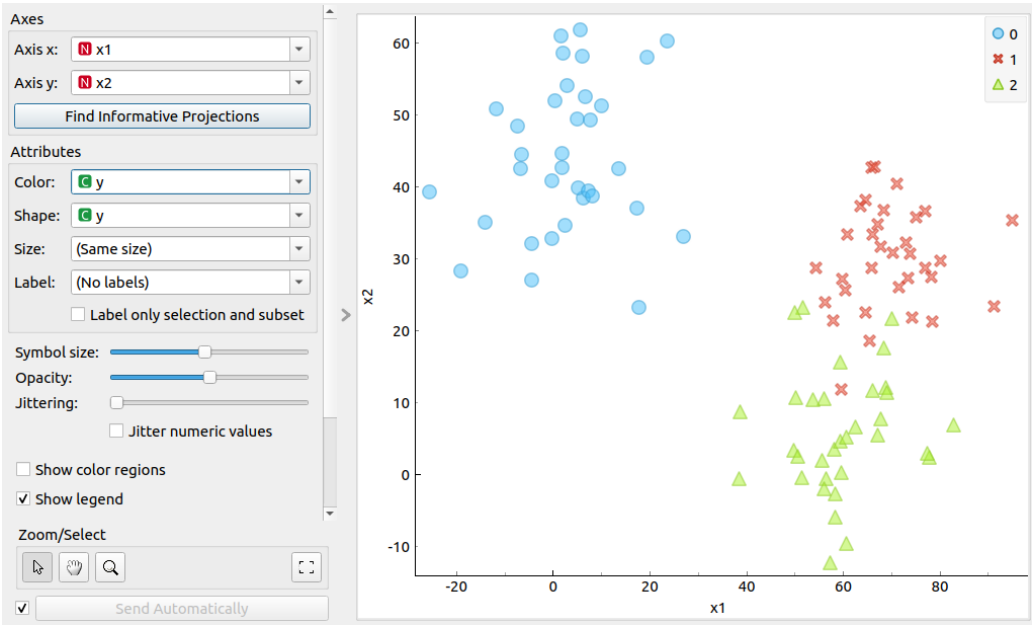


Рисунок 6.6 — Діаграма розсіювання набору навчальних даних

Побудуємо три моделі kNN із значеннями *k* 3, 5, 7 (рисунок 6.7).

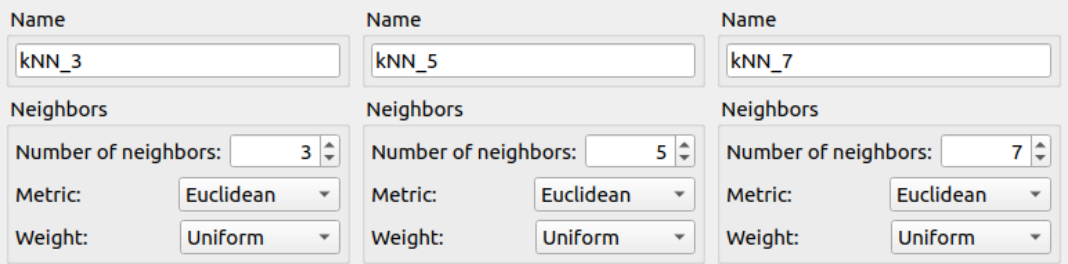


Рисунок 6.7 — Моделі kNN

За допомогою віджета Test and Score розділимо набір даних на навчальні та валідаційні дані в пропорції 80 % / 20 % (рисунок 6.8).

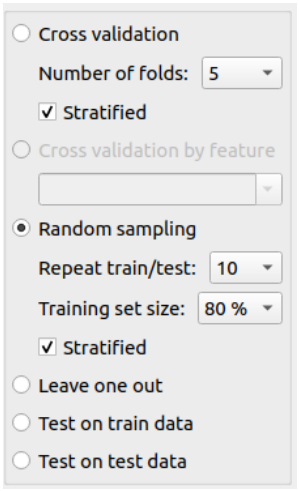
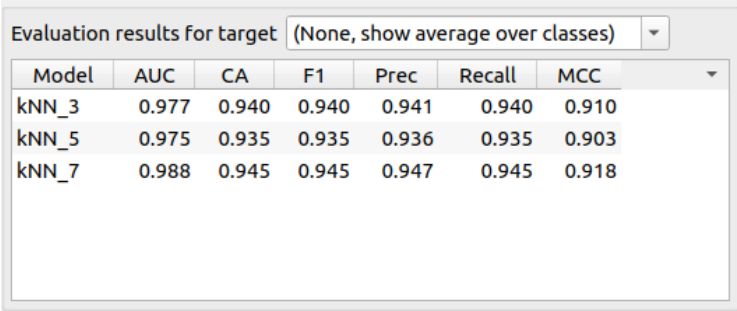


Рисунок 6.8 — Розділення даних на навчальні та валідаційні

За допомогою віджета Test and Score визначимо точність кожної моделі kNN на валідаційному наборі даних (рисунок 6.9).



Model	AUC	CA	F1	Prec	Recall	MCC
kNN_3	0.977	0.940	0.940	0.941	0.940	0.910
kNN_5	0.975	0.935	0.935	0.936	0.935	0.903
kNN_7	0.988	0.945	0.945	0.947	0.945	0.918

Рисунок 6.9 — Значення критеріїв якості для моделей kNN, розраховані на валідаційному наборі даних

Як видно, найвищу класифікаційну точність (CA) на валідаційному наборі має модель kNN із параметром $k = 7$.

Побудуємо матриці невідповідностей для кожної моделі (рисунок 6.10).

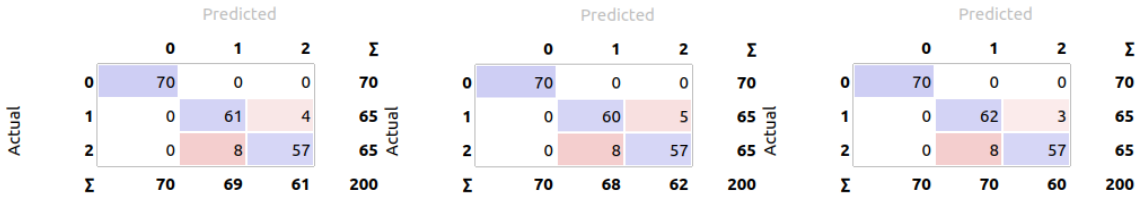


Рисунок 6.10 — Матриці невідповідностей для моделей kNN

Висновки. У роботі проведено аналіз даних за допомогою моделей класифікації, заснованих на методі kNN. Якість моделей оцінено на валідаційному наборі даних, у результаті чого було обрано модель, яка враховує 7 найближчих сусідів.

Запитання для самоконтролю

1 Що таке завдання класифікації в машинному навчанні?

2 Яка різниця між бінарною та багатокласовою класифікацією?

3 Як можна оцінити якість класифікаційної моделі?

4 Як розраховується точність моделі класифікації?

5 Для чого потрібне розділення даних на навчальний та валідаційний набори?

6 Наведіть приклад застосування класифікації.

7 Чим корисна матриця невідповідностей?

8 Що таке метод k найближчих сусідів?

9 На яких принципах базується алгоритм kNN?

10 Як вибір значення k впливає на роботу алгоритму kNN?

Практична робота 7

АНАЛІЗ ДАНИХ. ВИКОРИСТАННЯ МЕТОДІВ КЛАСТЕРИЗАЦІЇ

7.1 Мета роботи

Отримати навички аналізу даних, використовуючи метод k-середніх (k-means) та метод ієрархічної кластеризації.

7.2 Теоретичні відомості

Кластеризація — це завдання аналізу даних та машинного навчання, що полягає у групуванні об’єктів набору даних в однорідні групи (кластери) на основі їхніх ознак. Метою кластеризації є поділ даних так, щоб об’єкти всередині одного кластера були більш схожі між собою, ніж з об’єктами з інших кластерів. Кластеризація відрізняється від класифікації тим, що не вимагає наявності попередньо визначених міток або цільових змінних, що робить її особливо корисною для дослідження структури даних та виявлення прихованих патернів. Приклад вирішення завдання кластеризації наведено на рисунку 7.1.

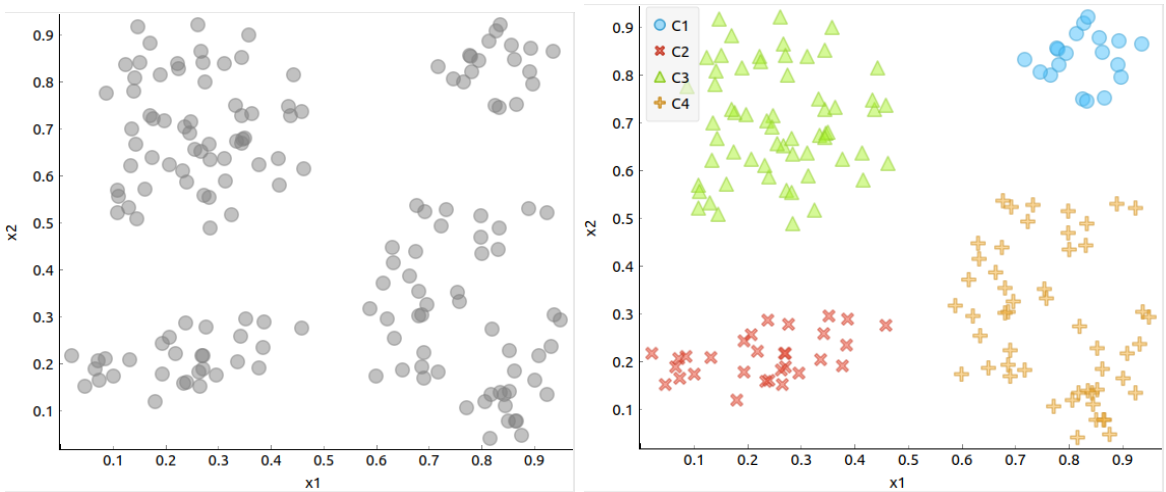


Рисунок 7.1 — Приклад вирішення завдання кластеризації (ліворуч наведено вихідні дані, праворуч — кластеризовані дані)

Існує багато методів кластеризації. Одними з найбільш поширених є метод k-середніх та метод ієрархічної кластеризації.

Метод k-середніх є одним з найбільш відомих методів кластеризації. Він працює шляхом поділу *n* об'єктів на *k* кластерів, де кожен об'єкт належить тому кластеру, центр якого є найближчим. Алгоритм k-середніх містить такі етапи:

1 Випадково обираються *k* початкових центрів кластерів;

2 Об'єкти з набору даних розподіляються на *k* кластерів (кожен об'єкт відноситься до того кластеру, центр якого є найближчим);

3 Виконується перерахунок центрів кластерів шляхом визначення центру тяжіння (центроїду) точок, що до них відносяться;

4 Етапи 2–3 повторюються до тих пір, поки після перерахунку центрів кластерів їхнє положення не припинить змінюватись.

Метод k-середніх є ефективним, але він чутливий до вибору початкових центрів кластерів та значення *k*. Неправильний вибір *k* може призвести до неякісної кластеризації. Для визначення оптимального числа кластерів можуть використовуватися додаткові методи, такі як аналіз графіку ліктя (elbow method) або визначення середнього силуетного коефіцієнту (silhouette coefficient).

Ієрархічна кластеризація є іншим популярним методом кластеризації, який не вимагає попереднього визначення числа кластерів. Ієрархічна кластеризація може бути агломеративною (bottom-up) або розділювальною (top-down).

Алгоритм агломеративної ієрархічної кластеризації:

1 Кожен об'єкт вважається окремим кластером;

2 На кожному кроці два найближчих кластери об'єднуються, поки всі об'єкти не будуть об'єднані в один кластер. Відстань між кластерами може обчислюватися за різними метриками (наприклад, мінімальна відстань

(single linkage), максимальна відстань (complete linkage), середня відстань (average linkage), або відстань за методом Ворда).

Алгоритм розділювальної ієрархічної кластеризації:

1 Всі об’єкти об’єднані в один кластер;

2 На кожному кроці кластер розділяється на два підкластери, поки кожен об’єкт не стане окремим кластером.

Ієрархічна кластеризація дає змогу отримати дендрограму, за якою можна візуально визначити оптимальну кількість кластерів.

Для обчислення відстані між об’єктами або кластерами можуть використовуватися різні метрики відстані:

– евклідова відстань — найчастіше використовується, особливо в методі k-середніх;

– мангеттенська відстань — використовується в деяких випадках, коли важливі різниці по окремих вимірах;

– косинусна подібність — часто використовується для текстових даних та інших даних з великою кількістю нульових значень.

Оцінка якості кластеризації є важливим етапом аналізу даних. Деякі методи оцінки включають:

– визнання силуетного коефіцієнту дає змогу виміряти, наскільки об’єкти схожі на об’єкти свого кластера порівняно з об’єктами інших кластерів. Значення коефіцієнта варіюються від –1 до 1, де вищі значення вказують на кращу кластеризацію;

– порівняння міжкластерної та внутрішньокластерної відстаней, де мінімізація відстаней всередині кластерів та максимізація відстаней між кластерами є бажаною.

Кластеризація знаходить застосування в багатьох галузях, таких як: маркетинг (сегментація клієнтів для створення цільових маркетингових стратегій), біоінформатика (групування генів або білків з подібними функціями), соціальні мережі (виявлення спільнот в соціальних графах),

зменшення розмірності (попередній етап для подальшого аналізу даних або візуалізації).

7.3 Порядок виконання роботи

1 Встановити пакет Orange Data Mining. Пакет є у вільному доступі та може бути завантажений за посиланням [1].

2 Завантажити у середовище набір даних за варіантом.

3 Візуалізувати навчальні дані за допомогою діаграми розсіювання.

4 Виконати кластеризацію даних за допомогою методу k-середніх для різних значень *k* від 2 до 10. Обрати ту кількість кластерів, для якої значення силуетного коефіцієнту є максимальним. Побудувати діаграму розсіювання з результатом кластеризації.

5 Виконати ієрархічну кластеризацію даних та побудувати дендрограму. За дендрограмою встановити найбільш доцільну кількість кластерів. Побудувати діаграму розсіювання з результатом кластеризації.

6 Порівняти результати кластеризації, аналізуючи форми та розміри кластерів, їхню кількість та розташування. Зробити висновок щодо найбільш доцільного методу кластеризації для цього набору даних.

7.4 Вихідні дані

Варіанти вихідних даних до завдання доступні за посиланням [18].

7.5 Приклад виконання роботи

Для виконання роботи використаємо пакет Orange Data Mining. Створимо схему з таких віджетів:

– File — віджет для завантаження даних;

– Data Table — віджет для відображення даних;

– Scatter Plot — віджет для побудови діаграми розсіювання;

– k-Means — віджет для побудови моделі k-середніх;

– Distances — віджет для визначення матриці, що показує відстані між кожною парою об’єктів з набору даних;

– Hierarchical Clustering — віджет для побудови моделі ієрархічної кластеризації.

Схему наведено на рисунку 7.2.

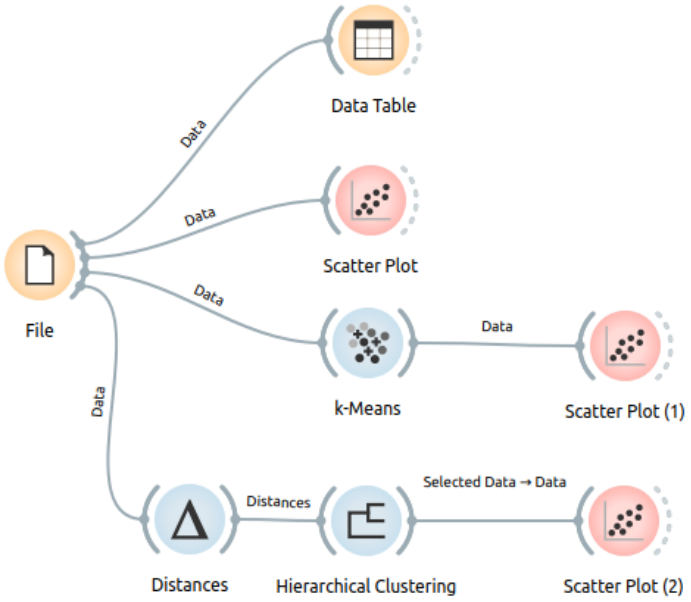


Рисунок 7.2 — Схема у Orange Data Mining

Завантажимо набір навчальних даних, використовуючи віджет File. Для рядів x_1 та x_2 встановимо тип number та позначимо їх як ознаки (рисунок 7.3).

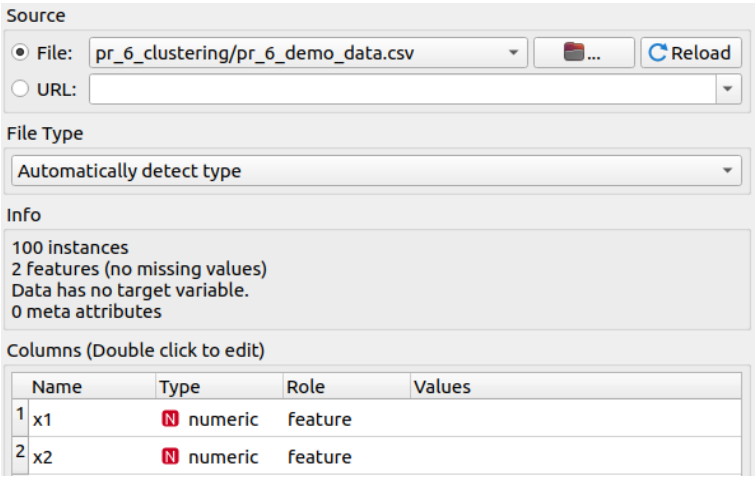


Рисунок 7.3 — Завантаження набору навчальних даних

Для перегляду набору навчальних даних скористаємося віджетом Data Table (рисунок 7.4).

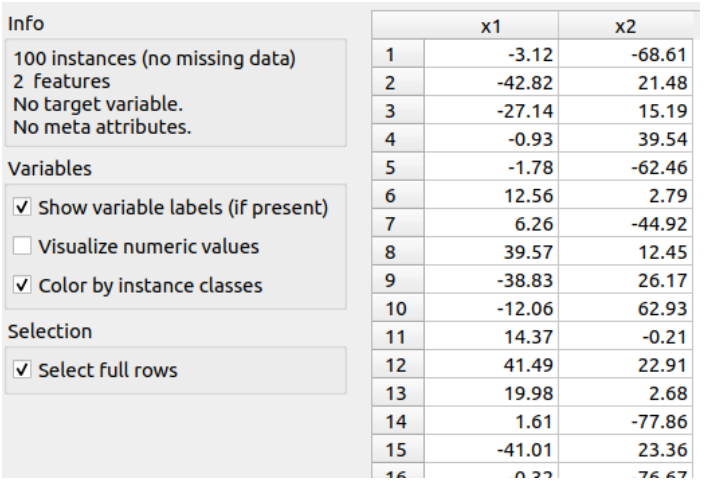


Рисунок 7.4 — Перегляд набору навчальних даних

Використовуючи віджет Scatter Plot, побудуємо діаграму розсіювання набору даних, щоб побачити розподіл даних та їхню структуру (рисунок 7.5).

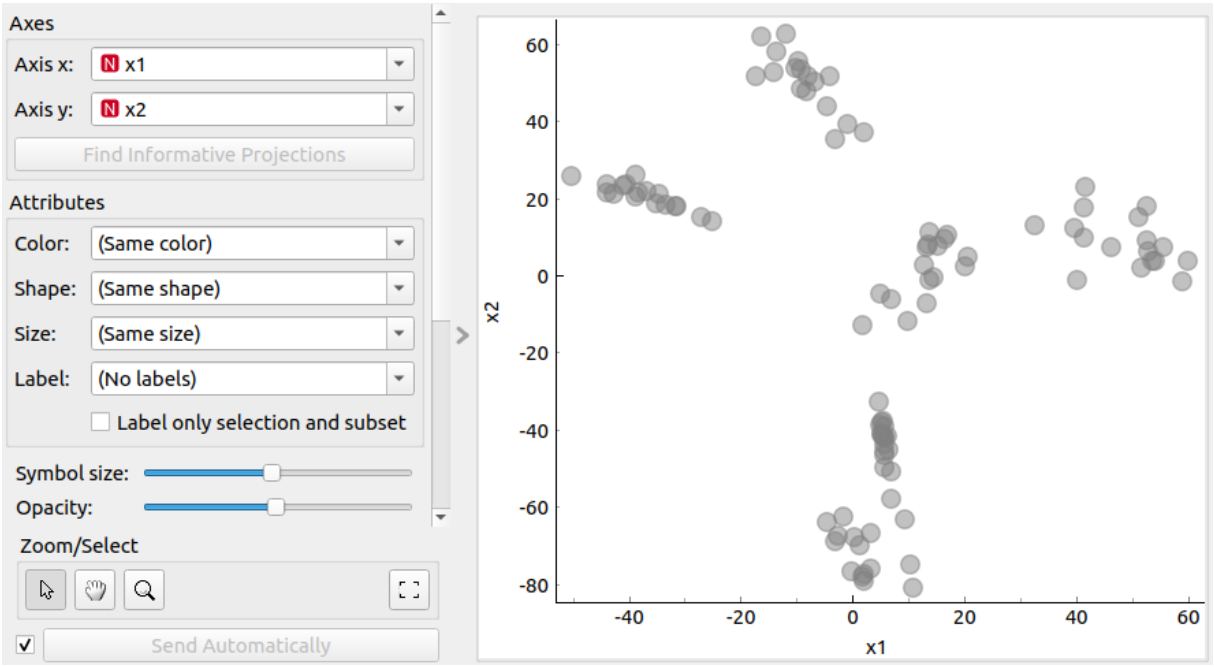


Рисунок 7.5 — Діаграма розсіювання набору даних

За допомогою віджета k-Means створимо 9 моделей із різними кількостями кластерів (від 2 до 10) (рисунок 7.6).

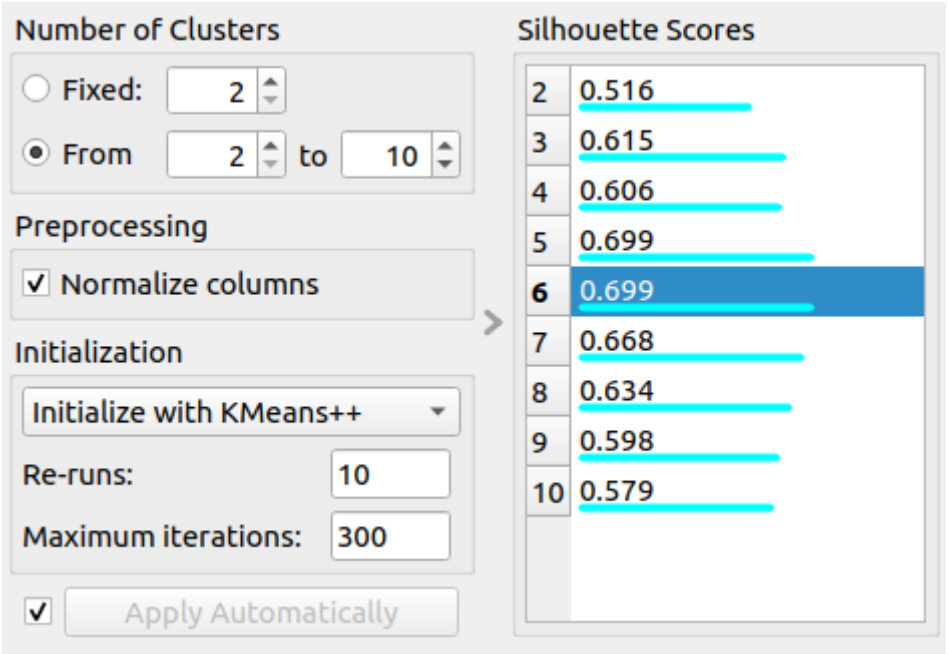


Рисунок 7.6 — Моделі k-середніх із значеннями k від 2 до 10

За значеннями силуетного коефіцієнту можна встановити, що найбільш доцільною кількістю кластерів є 5 або 6. Для візуалізації оберемо значення 6. На рисунку 7.7 наведено діаграму розсіювання з результатом кластеризації, побудовану у віджеті Scatter Plot (1).

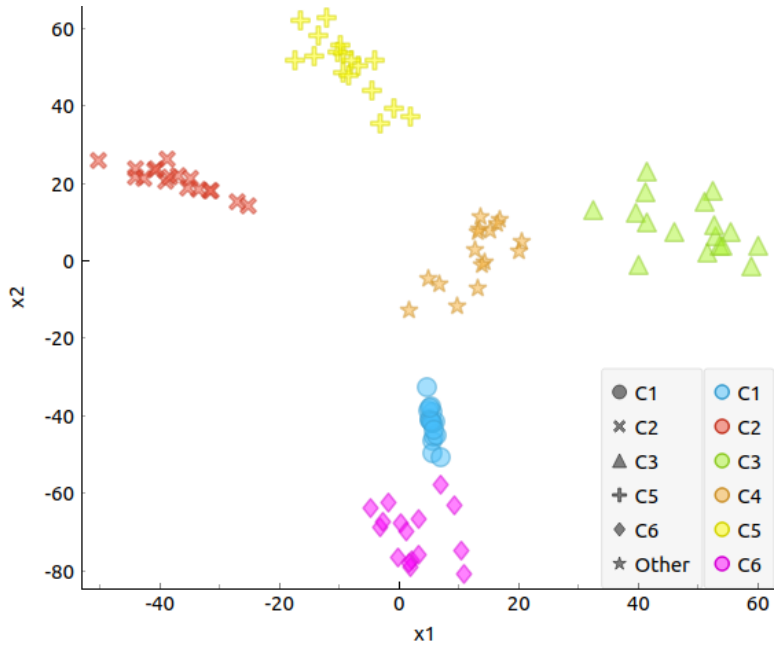


Рисунок 7.7 — Діаграма розсіювання даних, кластеризованих методом k-середніх

Для здійснення кластеризації методом ієрархічної кластеризації необхідно побудувати матрицю відстаней між всіма парами об’єктів набору даних. Для цього скористуємось віджетом Distances.

Використовуючи віджет Hierarchical Clustering, побудуємо дендрограму для набору даних. За дендрограмою визначимо доцільну кількості кластерів (рисунок 7.8).

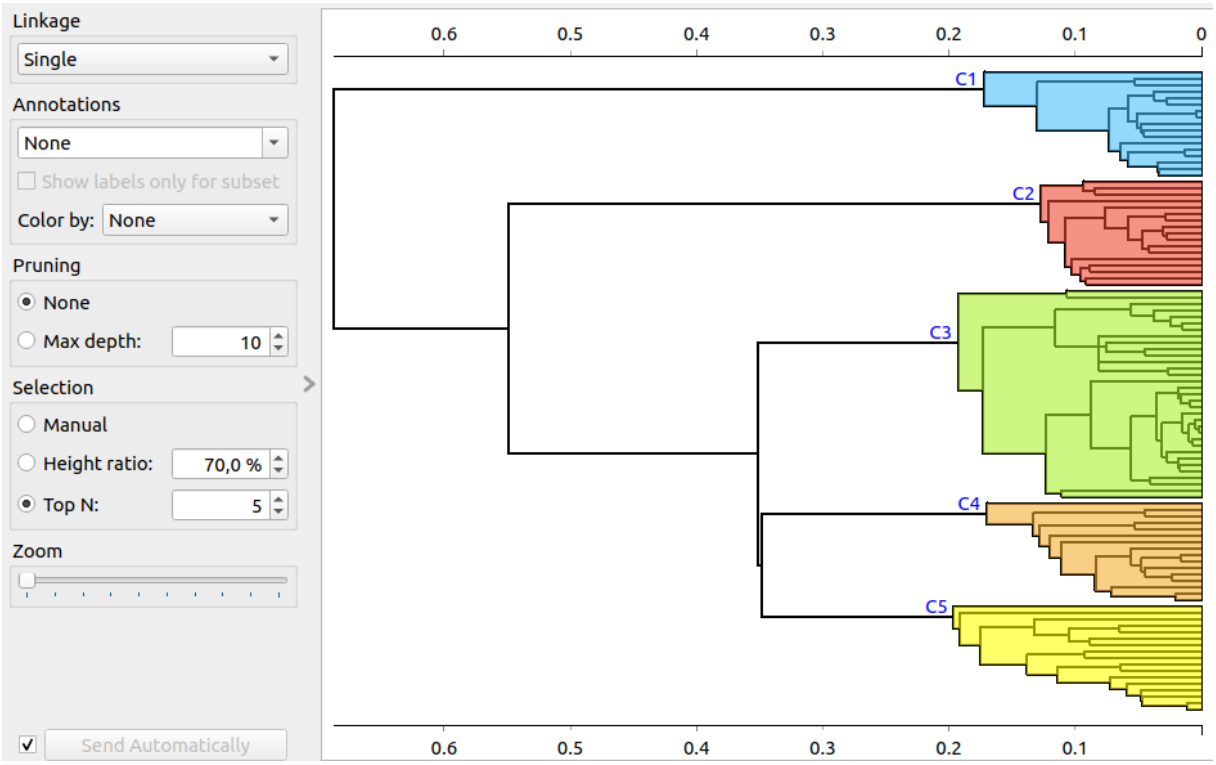


Рисунок 7.8 — Дендрограма побудована у результаті роботи методу ієрархічної кластеризації

Використовуючи віджет Scatter Plot (2), візуалізуємо результати ієрархічної кластеризації (рисунок 7.9).

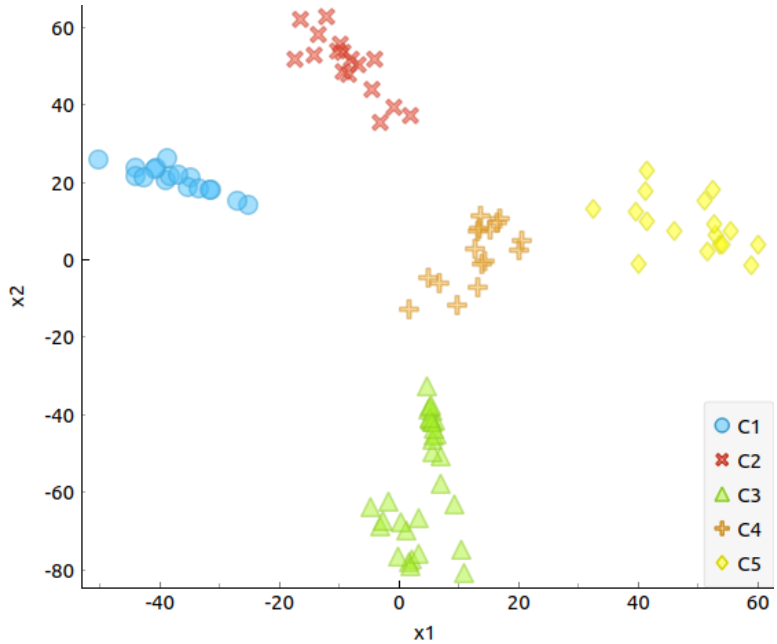


Рисунок 7.9 — Діаграма розсіювання даних, кластеризованих методом ієрархічної кластеризації

Порівнюючи результати кластеризації, можна зробити висновок, що метод k-means створює чітко відокремлені та компактні кластери, тоді як ієрархічна кластеризація формує кластери більш різноманітні за формою та щільністю. Крім того, ієрархічна кластеризація виявила менше кластерів і перерозподілила деякі дані.

Висновки. У роботі проведено аналіз даних за допомогою методів кластеризації, таких як метод k-середніх та ієрархічна кластеризація. За допомогою візуалізацій та оцінки якості кластеризації визначено оптимальні параметри для кожного методу. Результати кластеризації дають змогу виділити групи схожих об’єктів та зробити висновки про структуру даних.

Запитання для самоконтролю

1 Що таке завдання кластеризація у машинному навчанні?

2 Яка різниця між методом k-середніх та ієрархічною кластеризацією?

3 Як оцінити якість кластеризації?

4 Що таке дендрограма?

5 Які проблеми можуть виникнути при виборі значення k для методу k-середніх?

6 Як працює метод k-середніх?

7 У чому полягає відмінність між агломеративною та розділювальною ієрархічною кластеризацією?

8 Які метрики відстані можна використовувати у кластеризації?

9 Як вибір метрики відстані впливає на результати кластеризації?

10 Що таке метод силуетного коефіцієнта?

11 Чим відрізняється кластеризація від класифікації?

12 Які практичні застосування кластеризації в реальних задачах?

СПИСОК ЛІТЕРАТУРИ

1 Download Orange. Orange Data Mining. URL: https://orangedatamining.com/download (дата звернення: 12.01.2024).

2 draw.io. *draw.io*. URL: https://www.drawio.com/ (дата звернення: 15.06.2024).

3 How to Create a Gantt Chart in Google Sheets, 2020. *YouTube*. URL: https://youtu.be/8eKk0M2zGIk.

4 Jira. *Atlassian*. URL: https://www.atlassian.com/software/jira (дата звернення: 01.09.2023).

5 Online Gantt. URL: https://www.onlinegantt.com/#/gantt.Zupan B., Demsar J. Introduction to Data Mining. *University of Ljubljana Faculty of Computer and Information Science*. URL: https://file.biolab.si/notes/2018-05-intro-to-datamining-notes.pdf (дата звернення: 12.01.2024).

6 Polynomial Regression. Orange Data Mining. URL: https://orangedatamining.com/widget-catalog/educational/polynomial-regression (дата звернення: 12.01.2024).

7 Машинне навчання: навч. посіб. / Т. М. Басюк та ін. Львів: Новий світ-2000, 2019, 329 с.

8 Блага Н. В. Управління проєктами: навч. посіб. Львів: Львів. держ. ун-т внутр. справ, 2021. 152 с.

9 Гавриленко О. В. Аналіз даних в інформаційно-управляючих системах: навч. посіб. Київ: КПІ ім. Ігоря Сікорського, 2023. 205 с.

10 Данченко О. Практичні аспекти реінжинірингу бізнес-процесів: навч. посіб. Київ, 2013. 239 с.

11 Доценко С. І. Організація та системи керування базами даних: навч. посіб. Харків: УкрДУЗТ, 2023. 117 с.

12 Іванюк О. І. Варіанти вихідних даних до роботи 1 з циклу інформаційних систем та технологій. *GitHub*. URL: https://github.com/oleksa-iv/ist-course/tree/master/task_1_gantt (дата звернення: 15.05.2024).

13 Іванюк О. І. Варіанти вихідних даних до роботи 2 з циклу інформаційних систем та технологій. *GitHub*. URL: https://github.com/oleksa-iv/ist-course/tree/master/task_2_kanban (дата звернення: 15.05.2024).

14 Іванюк О. І. Варіанти вихідних даних до роботи 3 з циклу інформаційних систем та технологій. *GitHub*. URL: https://github.com/oleksa-iv/ist-course/tree/master/task_3_erd (дата звернення: 15.05.2024).

15 Іванюк О. І. Варіанти вихідних даних до роботи 4 з циклу інформаційних систем та технологій. *GitHub*. URL: https://github.com/oleksa-iv/ist-course/tree/master/task_4_idef0 (дата звернення: 15.05.2024).

16 Іванюк О. І. Варіанти вихідних даних до роботи 5 з циклу інформаційних систем та технологій. *GitHub*. URL: https://github.com/oleksa-iv/ist-course/tree/master/task_5_regression (дата звернення: 15.05.2024).

17 Іванюк О. І. Варіанти вихідних даних до роботи 6 з циклу інформаційних систем та технологій. *GitHub*. URL: https://github.com/oleksa-iv/ist-course/tree/master/task_6_classification (дата звернення: 15.05.2024).

18 Іванюк О. І. Варіанти вихідних даних до роботи 7 з циклу інформаційних систем та технологій. *GitHub*. URL: https://github.com/oleksa-iv/ist-course/tree/master/task_7_clustering (дата звернення: 15.05.2024).

19 Лупан І. Інтелектуальний аналіз даних Data Mining: навч.-метод. посіб. Кропивницький: ФОП Піск М. А., 2022. 112 с.

20 Мінухін С. В., Беседовський О. М., Знахур С. В. Методи і моделі проектування на основі сучасних CASE-засобів: навч. посіб. Харків: ХНЕУ, 2008. 272 с.

МЕТОДИЧНІ ВКАЗІВКИ

до практичних занять

з дисципліни

«*ІНФОРМАЦІЙНІ СИСТЕМИ І ТЕХНОЛОГІЇ*»

Відповідальний за випуск Іванюк О. І.

Підписано до друку 17.06.2024 р.
Умовн. друк. арк. 3,75. Тираж . Замовлення № .
Видавець та виготовлювач Український державний університет залізничного транспорту,
61050, Харків-50, майдан Фейєрбаха,7.
Свідоцтво суб'єкта видавничої справи ДК № 6100 від 21.03.2018 р.