Міністерство освіти і науки України

Львівський національний університет імені Івана Франка

Факультет електроніки та комп’ютерних технологій

Кафедра системного проектування

Звіт

Про виконання лабораторної роботи №4

З курсу «Системи машинного навчання»

### Класифікаційні моделі

**Виконала:**

Студентка групи ФЕС-32

Філь Дарина

**Перевірив:**

Доцент Колич І.І.

Львів 2024

**Мета:** Навчитися будувати та оцінювати класифікаційні моделі.

**Інструменти:** Python, Scikit-learn, Matplotlib, Seaborn.

**Теоретичні відомості**

**Логістична регресія**

Логістична регресія - це статистичний метод для бінарної класифікації, який оцінює ймовірність того, що заданий вхід належить до певного класу.

**Формула логістичної регресії**

Логістична регресія використовує логістичну функцію (сигмоїдну), щоб перетворити лінійну комбінацію незалежних змінних на ймовірності:

де:

* залежна змінна;
* вільний член (intercept);
* коефіцієнти регресії для кожної незалежної змінної;
* незалежні змінні;

**Переваги логістичної регресії**

* Легко інтерпретована: результати можна інтерпретувати як ймовірності;
* Обчислювально ефективна: може бути швидко навчена навіть на великих наборах даних;
* Відмінно працює на лінійно роздільних даних: добре підходить, якщо класи можуть бути розділені лінійно;

**Недоліки логістичної регресії**

* Лінійність: не в змозі моделювати складні нелінійні взаємозв’язки;
* Чутливість до шуму: може бути чутливою до випадкових шумів у даних;

**Дерева рішень**

Дерева рішень - це ієрархічна модель класифікації, яка використовує правила ухвалення рішень, побудовані на основі ознак даних, щоб передбачити результат.

**Побудова дерева рішень**

Дерева рішень використовують рекурсивне розбиття, щоб створити модель у вигляді дерева. На кожному вузлі дерева вибирається ознака, за якою дані будуть розбиті на дві або більше підмножини.

**Переваги дерев рішень**

* Легко інтерпретовані: моделі у вигляді дерева можуть бути легко візуалізовані та інтерпретовані;
* Не потребують масштабування даних: дерева рішень управління з ознаками, які не потребують нормалізації або стандартизації;
* Працюють з категоріальними змінними: можуть легко працювати з категоріальними даними без потреби в додаткових перетвореннях;

**Недоліки дерев рішень**

* Перенавчання: дерева рішень часто схильні до перенавчання, особливо коли вони є глибокими;
* Чутливість до змін у даних: незначні зміни в даних можуть призвести до побудови зовсім іншого дерева;

**Оцінка моделей**

**Точність (Accuracy)**

**Визначення:** Точність (Accuracy) є часткою правильно передбачених випадків серед усіх випадків.

де:

* TP (True Positive) — кількість правильно передбачених позитивних випадків;
* TN (True Negative) — кількість правильно передбачених негативних випадків;
* FP (False Positive) — кількість неправильно передбачених позитивних випадків;
* FN (False Negative) — кількість неправильно передбачених негативних випадків;

**Переваги**

* Легкість інтерпретації;

**Недоліки**

* Не підходить для незбалансованих наборів даних;

**Влучність (Precision)**

Визначення: Частка правильно передбачених позитивних випадків серед усіх випадків, які модель передбачила як позитивні.

**Переваги**

* Важлива, коли помилкові позитивні передбачення мають високі витрати;

**Недоліки**

* Не враховує помилкові негативні передбачення;

**Повнота (Recall)**

Визначення: Частка правильно передбачених позитивних випадків серед усіх фактичних позитивних випадків.

**Переваги**

* Важлива, коли помилкові негативні передбачення мають високі витрати;

**Недоліки**

* Не враховує помилкові позитивні передбачення;

F1-score

Визначення: Гармонічне середнє між Precision та Recall.

**Переваги**

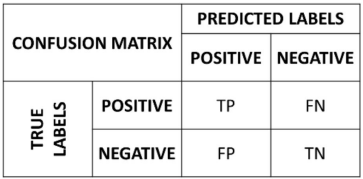
* Комбінує Precision та Recall, забезпечуючи баланс між ними;

**Недоліки**

* Не завжди легко інтерпретувати;

**Матриця невідповідностей (Confusion Matrix)**

Визначення: Матриця невідповідностей є таблицею, яка дозволяє візуалізувати продуктивність алгоритму класифікації. Вона відображає кількість правильних та неправильних передбачень, розподілених за кожним класом.



**Переваги**

* Забезпечує повний огляд продуктивності моделі;
* Допомагає виявити патерни помилок;

**Недоліки**

* Не дає безпосереднього уявлення про баланс між Precision та Recall;

**Набір даних Iris**

Набір даних Iris є класичним набором даних для задач класифікації. Він містить інформацію про довжину та ширину чашолистків і пелюсток для трьох видів ірисів (Iris setosa, Iris versicolor, Iris virginica).

**Опис набору даних**

* **Кількість зразків:** 150
* **Кількість ознак:** 4
* **Кількість класів:** 3 (три різні види ірисів)

**Ознаки**

1. **Sepal length (cm):** довжина чашолистка
2. **Sepal width (cm):** ширина чашолистка
3. **Petal length (cm):** довжина пелюстки
4. **Petal width (cm):** ширина пелюстки

**Класи**

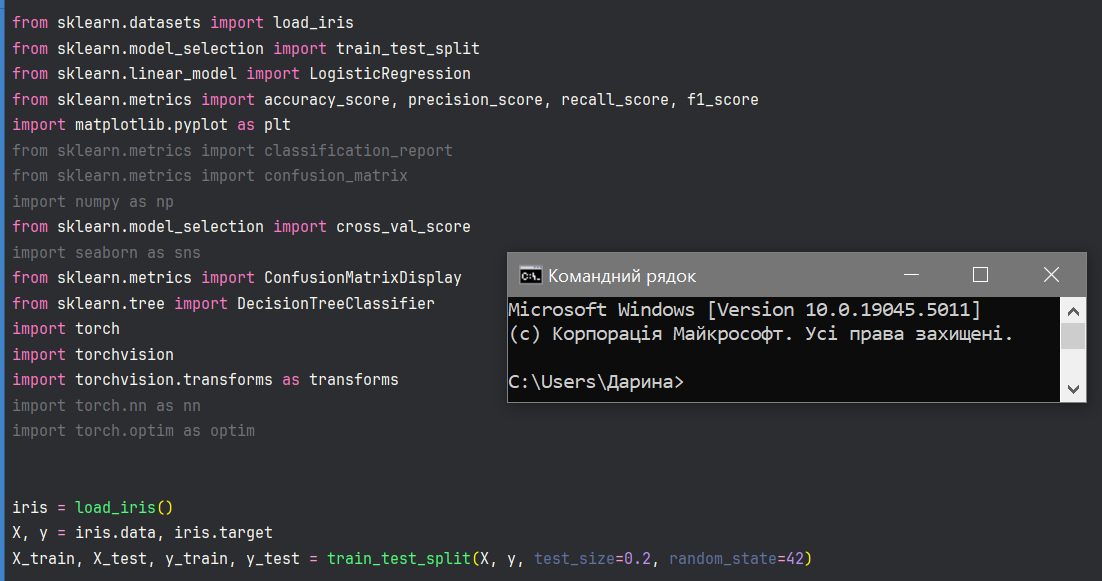
Цільова змінна (клас) вказує на вид ірису та має три можливі значення:

* 0: Iris setosa
* 1: Iris versicolor
* 2: Iris virginica

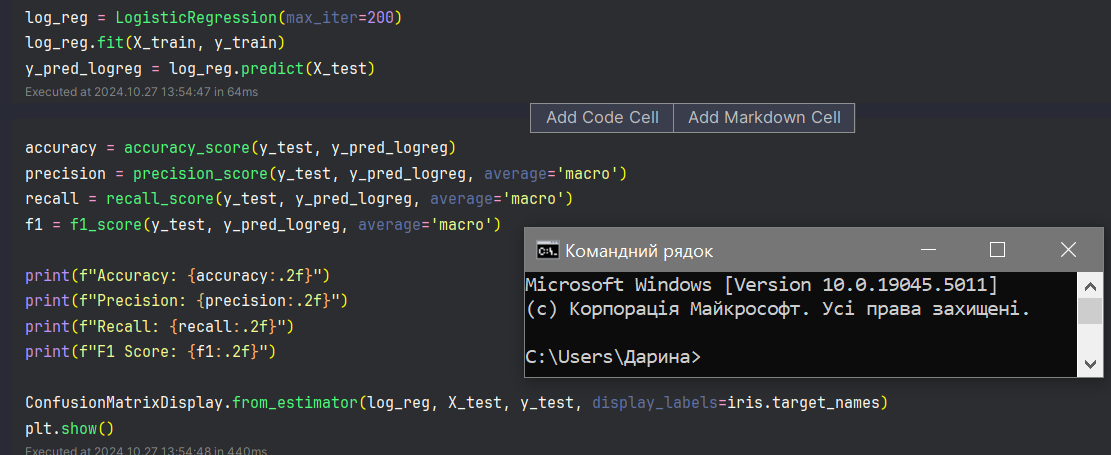
**Хід роботи**

**ЗАВДАННЯ**

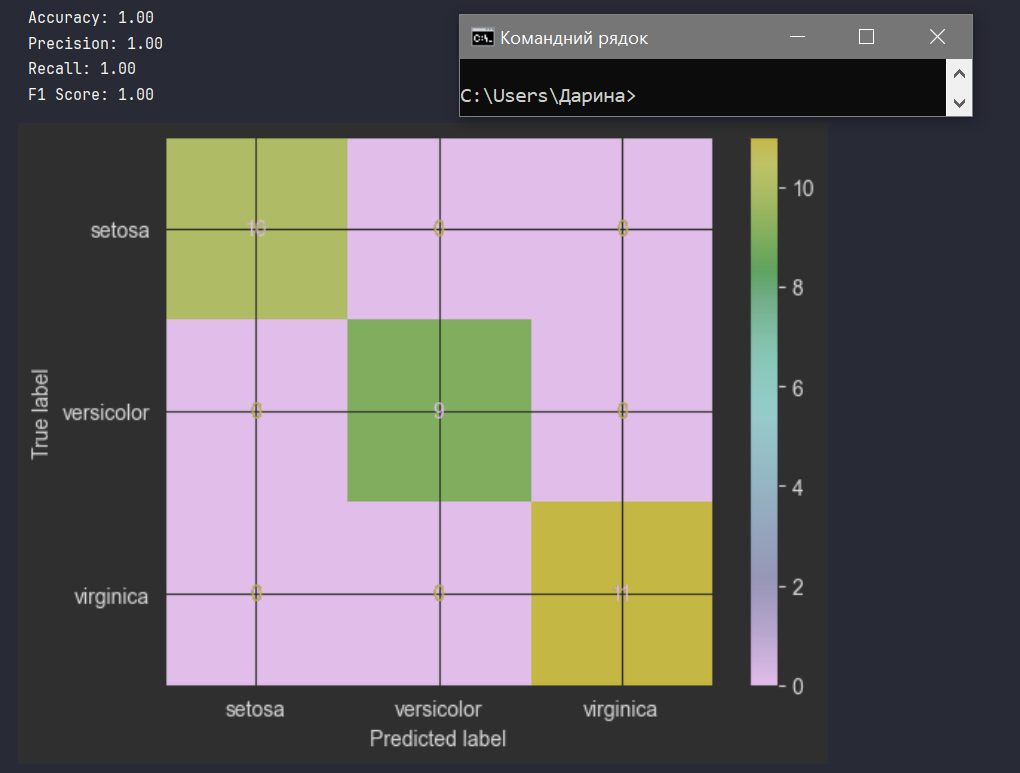
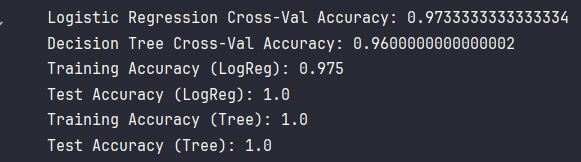
1. **Підготовка даних** 
   1. Використайте набір даних Iris.
   2. Розділіть дані на тренувальний (80%) та тестовий (20%) набори.
2. **Логістична регресія** 
   1. Навчіть модель логістичної регресії на тренувальних даних.
   2. Виконайте прогнозування на тестових даних.
   3. Оцініть результати за допомогою метрик якості моделі: точність (Accuracy), Precision, Recall, F1-score.
   4. Візуалізуйте матрицю невідповідностей.
3. **Дерева рішень** 
   1. Навчіть модель дерева рішень на тренувальних даних.
   2. Виконайте прогнозування на тестових даних.
   3. Оцініть результати за допомогою метрик якості моделі: точність (Accuracy), Precision, Recall, F1-score.
   4. Візуалізуйте матрицю невідповідностей.
4. **Повторити навчання та оцінку для Fashion-MNIST набору даних** 
   1. Рекомендовано використовувати PyTorch backend для Keras.
   2. Порівняти зміни в моделях при використанні іншого набору
5. **Оформити звіт**

****

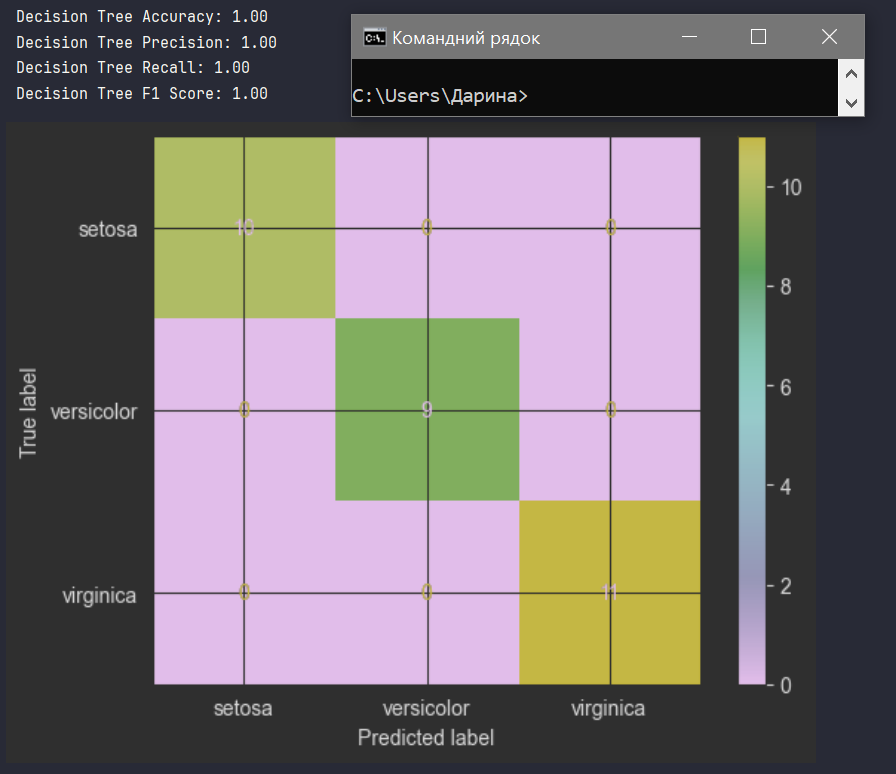
**Рис. 1** Завантажування бібліотек та поділ даних

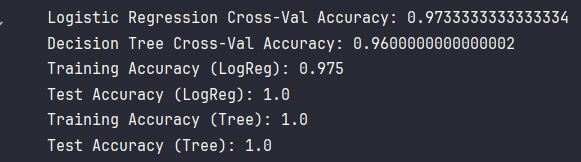
****

**Рис. 2** Тренування моделі логістичної регресії

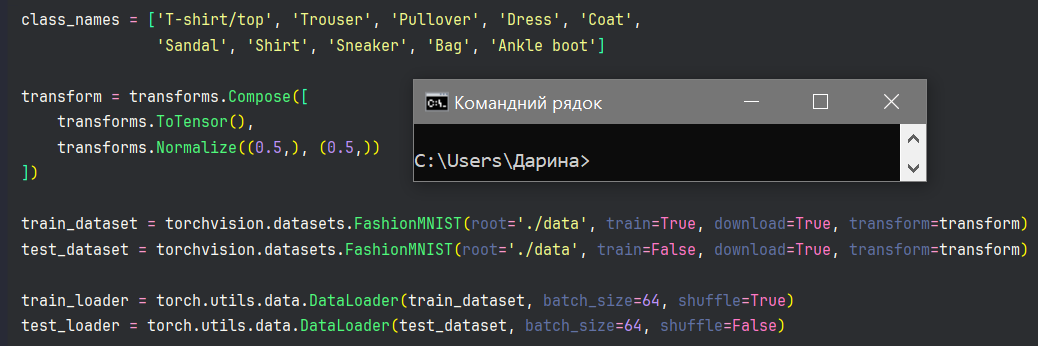
**  
  
**

**Рис. 3-4** Результати тренування моделі логістичної регресії та її оцінка показують, що модель, натренована на датасеті Iris, має фактично ідеальну точність і F1-score. Це може свідчити про можливе перенасичення даними. Для перевірки цього було проведено крос-валідацію, яка також показала фактично ідеальну точність, що свідчить про коректність результатів тренування моделі та відсутність перенасичення. Висока точність пояснюється тим, що датасет Iris є досить маленьким у порівнянні з іншими, тому модель легше класифікує дані  
 ****

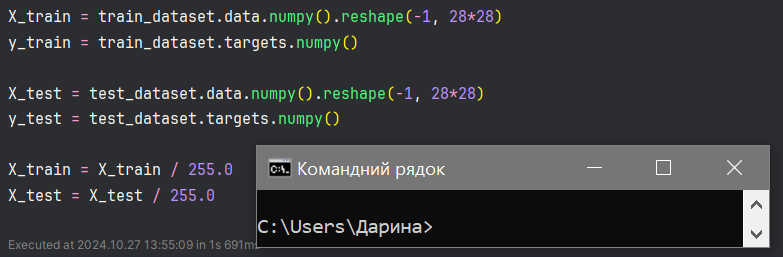
**Рис. 5** Тренування моделі дерева рішень ****

****

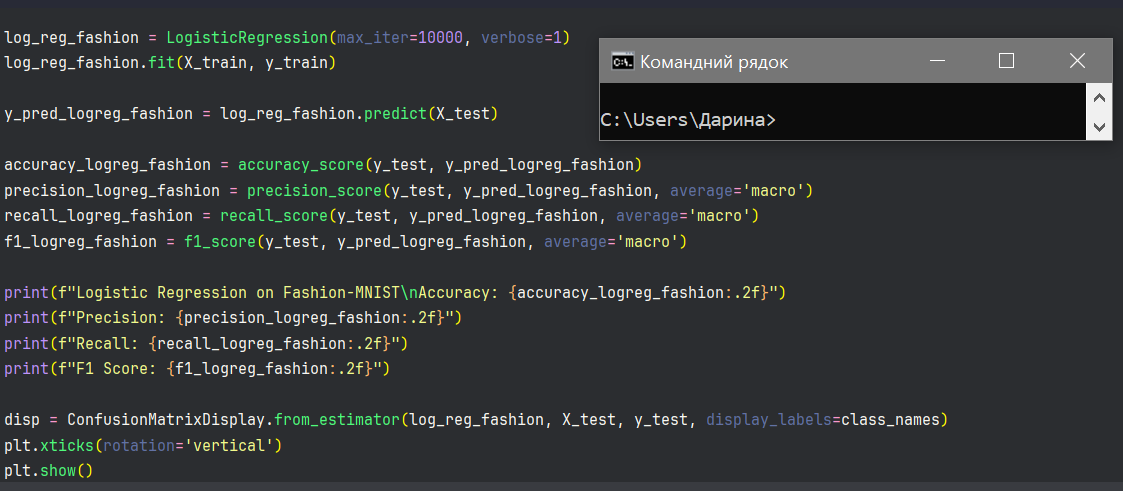
**Рис. 6-7** Схожа ситуація спостерігається і з моделлю дерева рішень, де точність тренування є ідеальною. Це також можна пояснити малим розміром датасету, що підтверджується результатами крос-валідації

****

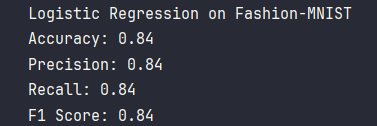
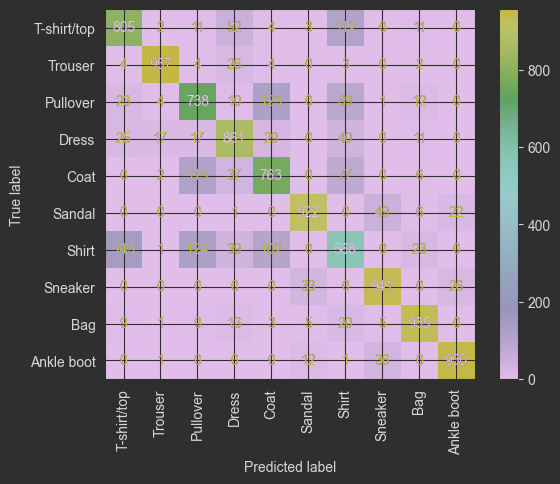
**Рис. 8** Назва класів у дата-сеті FashionMNIST та поділ його на тренувальний та тестувальний

****

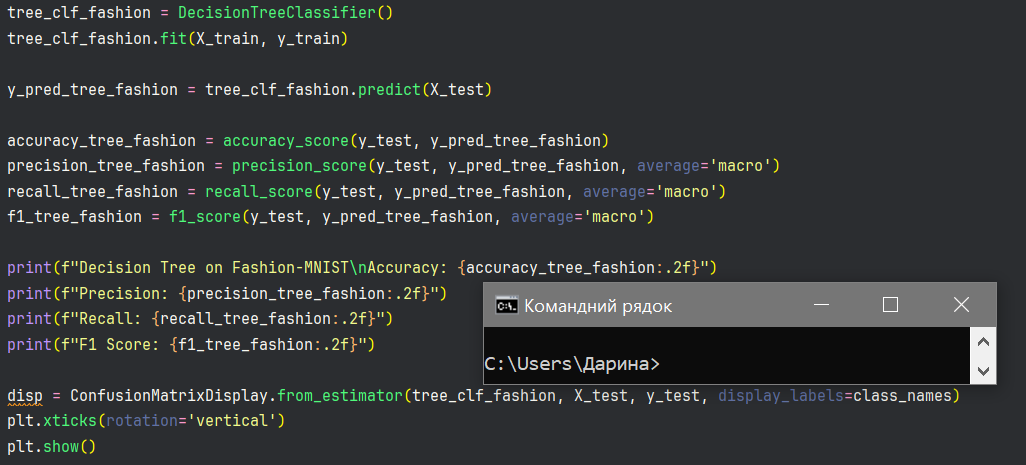
**Рис. 9** Трансформація зображень в дата-сеті на вектора та зведення значень до типу від 0 до 1

****

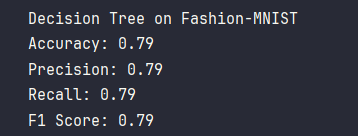
**Рис. 10** Тренування моделі лоістичної регресії для дата-сету FashionMNIST

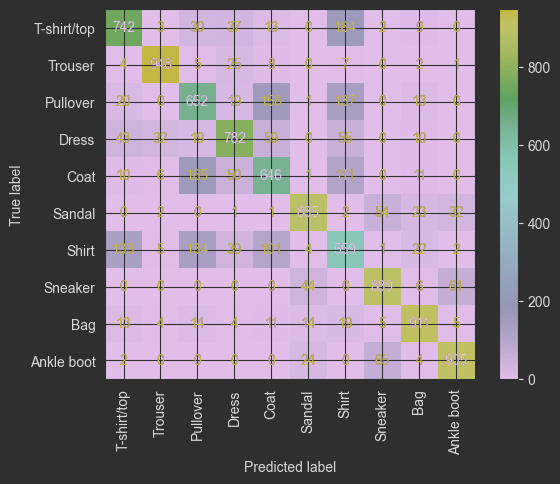
**  
  
**

**Рис. 11-12** Оцінка тренування моделі, можемо бачити, що результат вже далекі від ідеалу, цей результат можна назвати задовільнім

****

**Рис. 13** Тренування моделі дерева рішень для дата-сету FashionMNIST

****

****

**Рис. 14-15** Оцінка результатів класифікації дата-сету FashionMNIST на основі моделі дерева рішень, бачимо що є орієнтовна похибка в 0.21, що є задовільним результатом, але знову ж таки далеким від ідеалу.

**Висновок:** У цій лабораторній роботі я ознайомилася з методами класифікації, зокрема з логістичною регресією та деревами рішень. Я також мала змогу вивчити метрики оцінки моделей класифікації та матрицю невідповідностей, яка показує, які значення були класифіковані правильно, а які — ні, а також яким чином відбулася класифікація.