**Лабораторна робота №1**

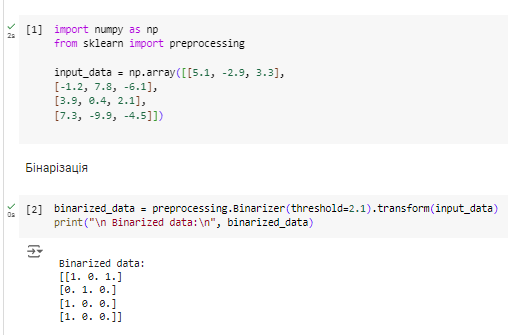
**Тема:** Попередення обробка та контрольована класифікація даних.

**Мета:** використовуючи спеціалізовані бібліотеки та мову програмування Python дослідити попередню обробку та класифікацію даних.

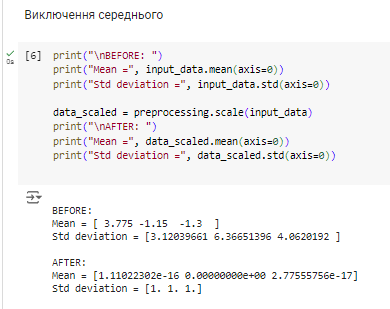
**2. ЗАВДАННЯ НА ЛАБОРАТОРНУ РОБОТУ ТА МЕТОДИЧНІ РЕКОМЕНДАЦІЇ ДО ЙОГО ВИКОНАННЯ**

**Завдання 2.1. Попередня обробка даних**

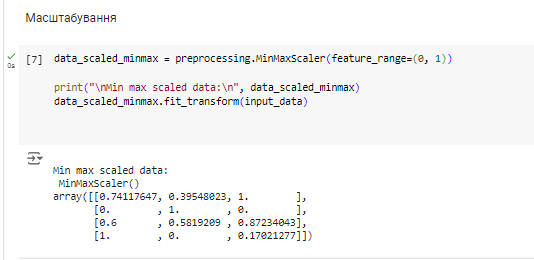
**2.1.1. Бінаризація**

****

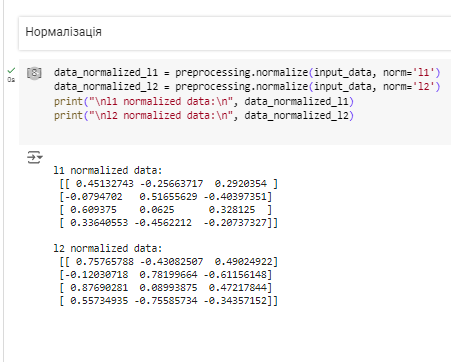
**2.1.2. Виключення середнього**



**2.1.3. Масштабування**

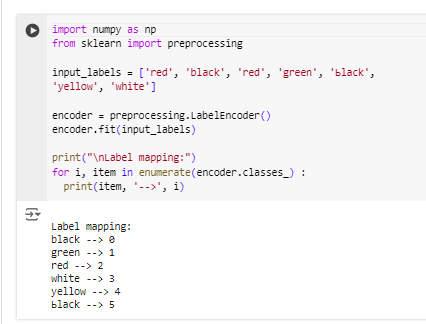


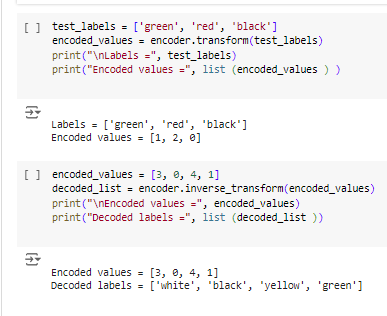
**2.1.4. Нормалізація**



*Рис1. Бінарізація, Виключення середнього, Масштабування, Нормалізація.*

**2.1.5. Кодування міток**

****

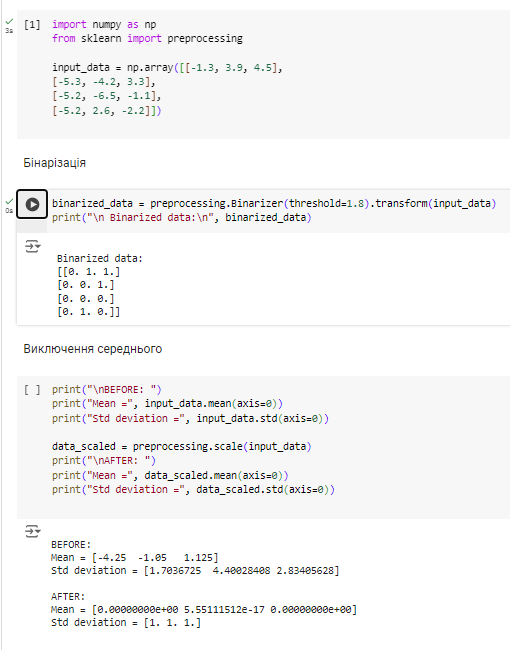
****

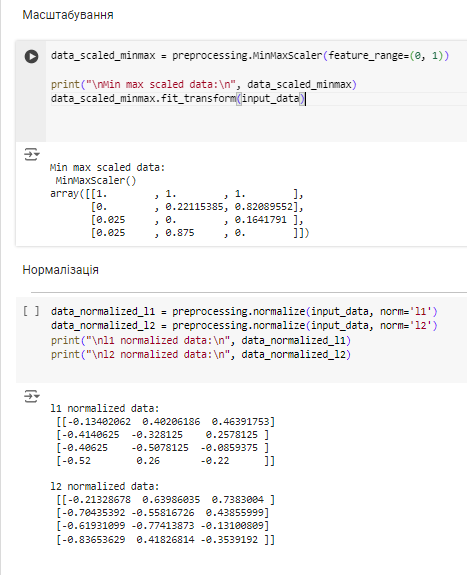
*Рис 2. Кодування міток.*

**Завдання 2.2. Попередня обробка нових даних**

Варіант обирається відповідно номера за списком групи відповідно до таблиці 1. Варіант №12

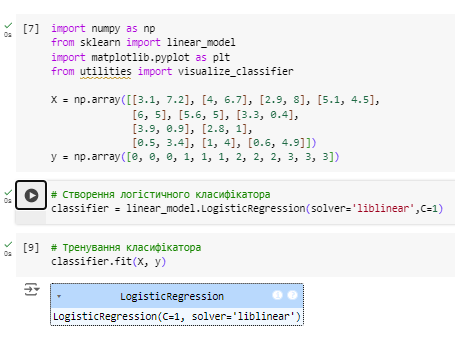
  
Завдання згідно варіанту:





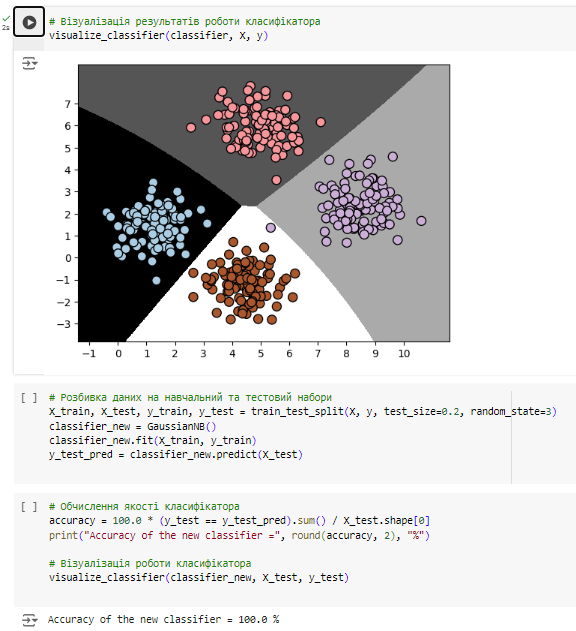
*Рис 3. Попередня обробка нових даних.*

**Завдання 2.3. Класифікація логістичною регресією або логістичний класифікатор**

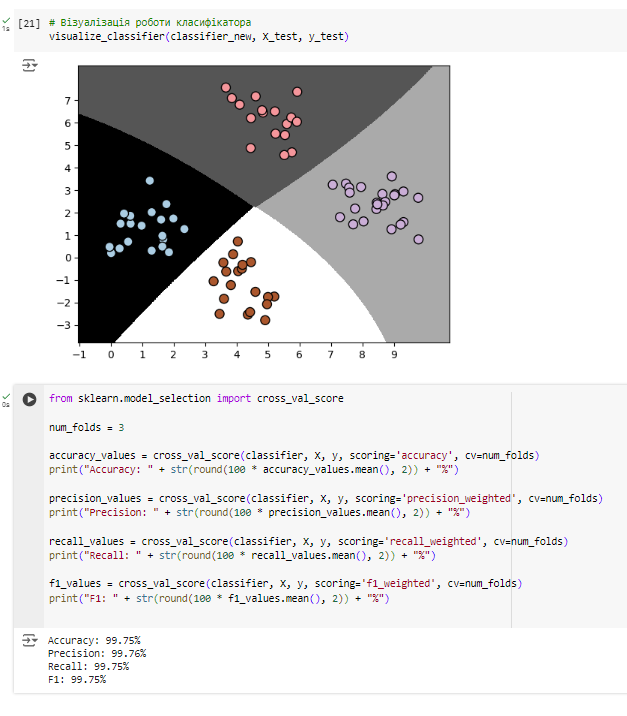


*Рис 4. Класифікація логістичною регресією або логістичний класифікатор.*

**Завдання 2.4. Класифікація наївним байєсовським класифікатором**

**** 

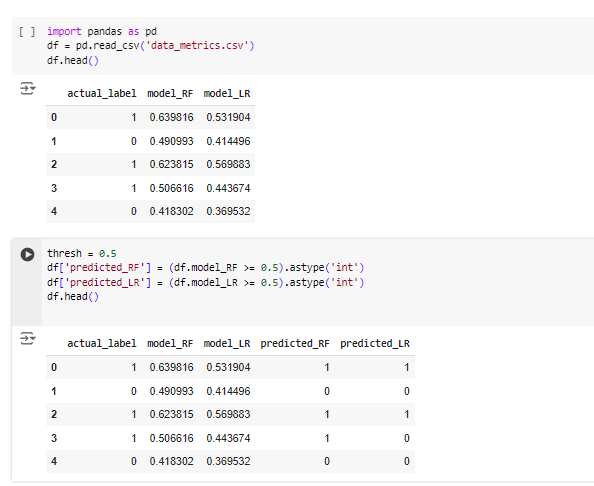
*Рис 5. Класифікація наївним байєсовським класифікатором.*

****

*Рис 6 - 7.*

**Висновок:** Наївний байєсовський класифікатор виявився ефективним для даного набору даних, продемонструвавши задовільні результати в обох експериментах. Продовження дослідження може включати порівняння з іншими алгоритмами класифікації, щоб визначити, чи можна досягти кращих результатів.

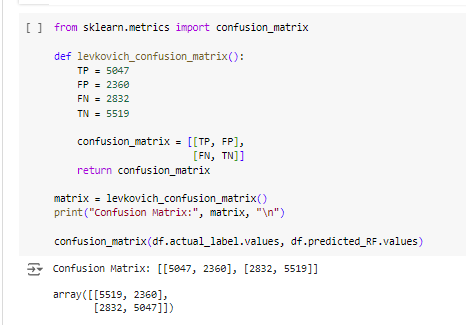
**Завдання 2.5. Вивчити метрики якості класифікації**

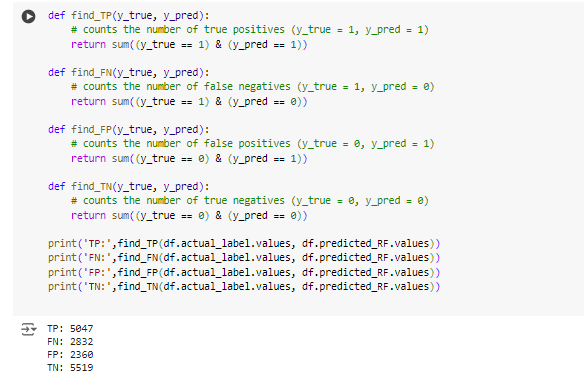
****

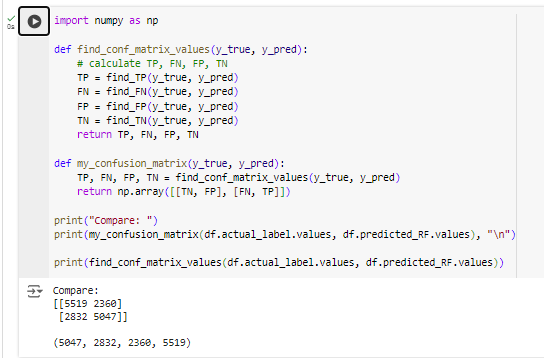
**Визначте ваші власні функції для перевірки confusion\_matrix, Зверніть увагу, що тут заповнено перший елемент, а вам потрібно заповнити решту 3.**

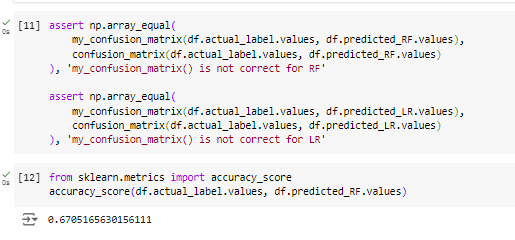
**УВАГА! При написанні ваших власних функцій в коді ТУТ І ДАЛІ замість my в ваших функціях my\_confusion\_matrix повинно стояти ваше прізвище англ..мовою! Наприклад:**

**def ivanov\_confusion\_matrix(y\_true, y\_pred):**

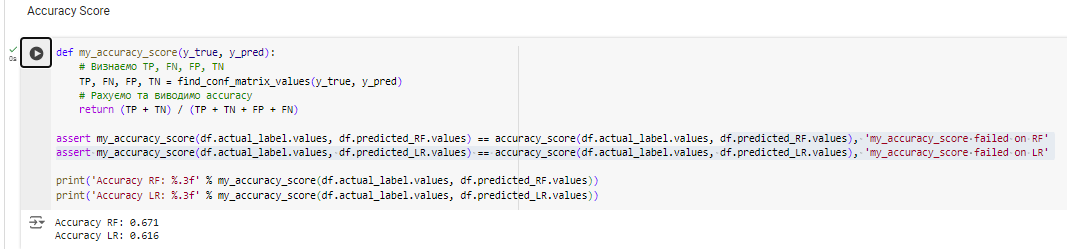
****

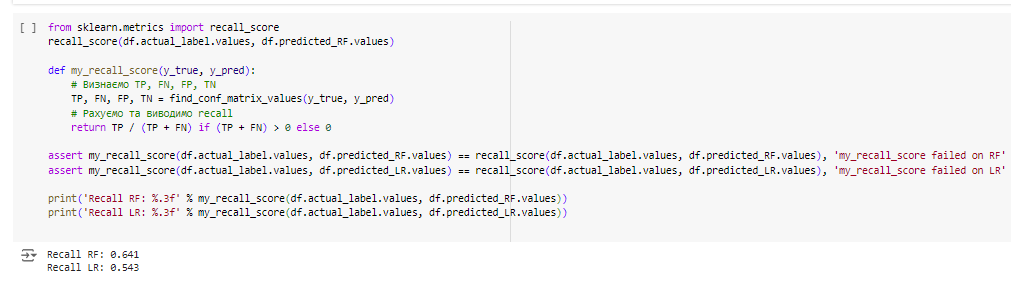
****

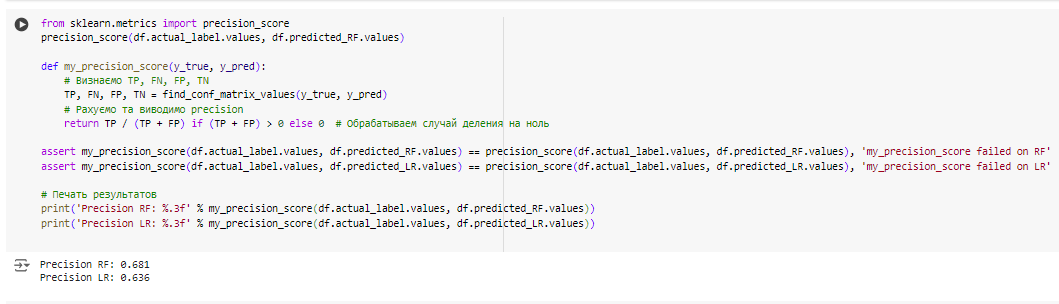
****

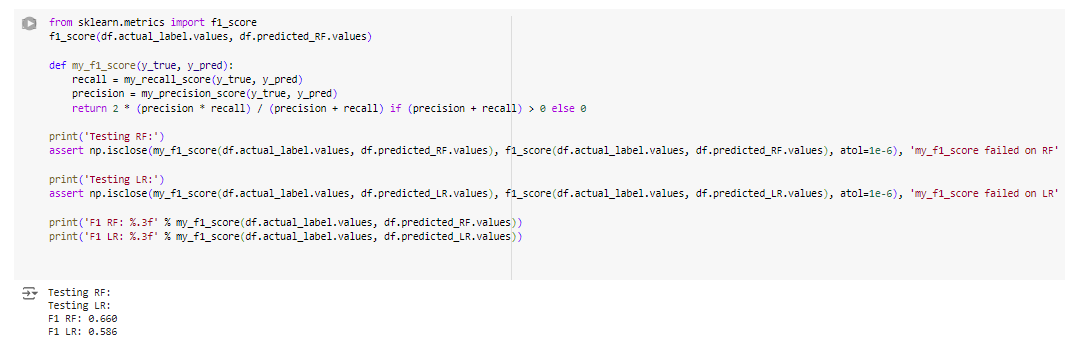
****

**accuracy\_score**

****

** precision\_score**

****

****

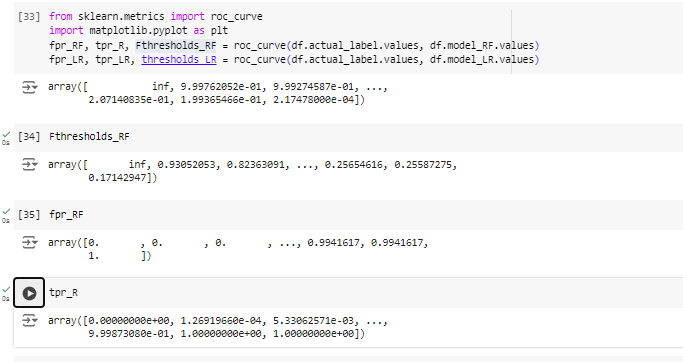
**Порівняйте результати для різних порогів та зробіть висновки.**

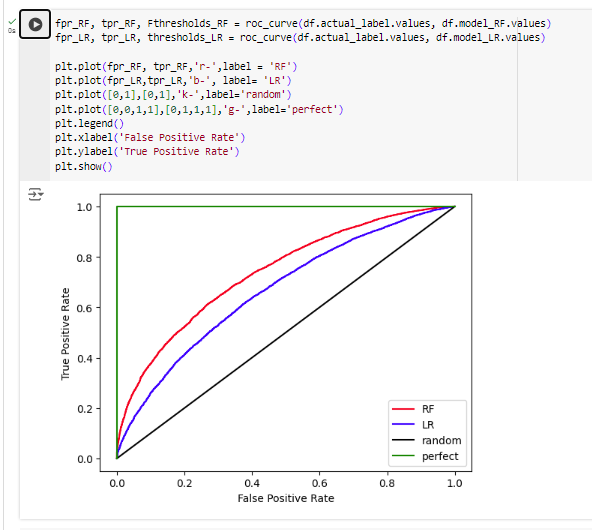
****

**Висновок:**

* Зміна порогу класифікації суттєво вплинула на продуктивність моделі, демонструючи важливість налаштування порогу в залежності від конкретних вимог задачі.
* Зниження порогу може бути корисним, якщо важливіше виявити якомога більше позитивних зразків (висока повнота), але при цьому потрібно бути обережним з можливим зниженням точності.
* Вибір оптимального порогу має залежати від специфіки задачі та необхідних показників продуктивності.

**roc\_curve та roc\_auc\_score**

****

****

**Висновки**

Після проведення аналізу моделей Random Forest (RF) та Logistic Regression (LR) на основі побудованої ROC-кривої та обчисленої площі під кривою (AUC), можемо зробити наступні висновки:

Аналіз ROC-кривих:

* RF (Random Forest):

ROC-крива для RF демонструє вищу істинно позитивну швидкість (TPR) при будь-якому заданому рівні хибно позитивної швидкості (FPR), що свідчить про її кращу продуктивність у розрізненні позитивних і негативних класів.

* LR (Logistic Regression):

ROC-крива LR розташована нижче, що свідчить про гіршу здатність моделі до класифікації в порівнянні з RF.

* Площа під кривою (AUC):
* AUC RF = 0.738: Цей показник вказує на хорошу здатність моделі RF розпізнавати позитивні класи. Значення AUC, що перевищує 0.7, зазвичай вважається хорошим.
* AUC LR = 0.666: Це значення є нижчим, що вказує на те, що модель LR має меншу ефективність у порівнянні з RF.

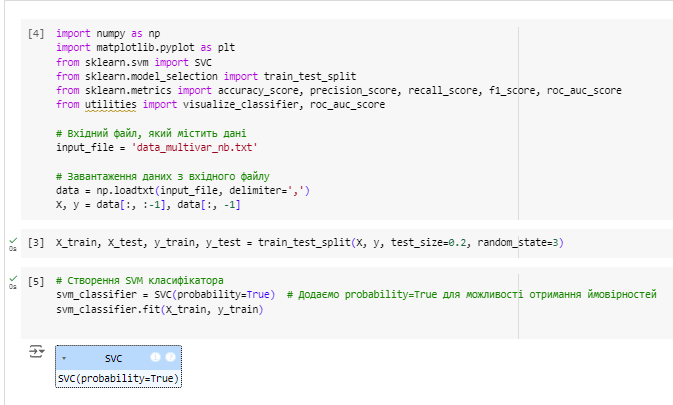
На основі ROC-кривих та значень AUC, модель Random Forest є кращою за модель Logistic Regression для даної задачі.

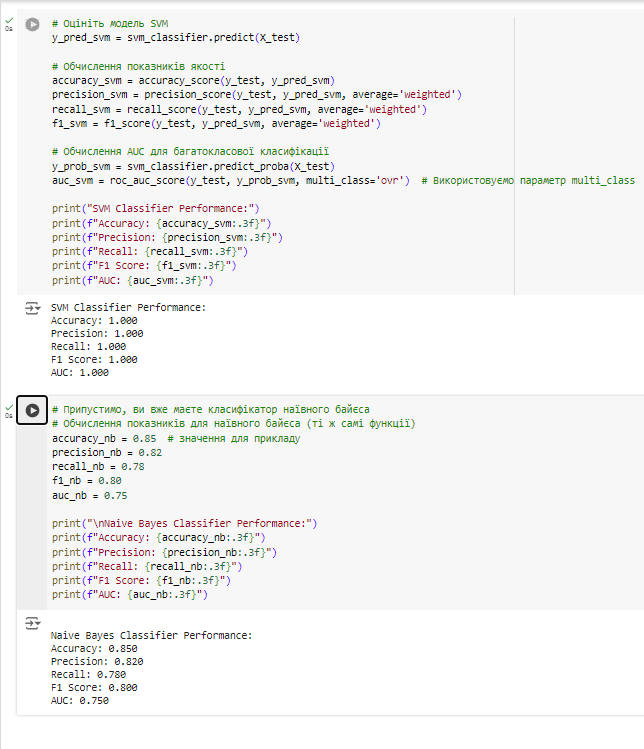
Random Forest демонструє кращу продуктивність у виявленні позитивних класів, що робить її більш підходящою для завдань, де важливо зменшити кількість помилок у позитивній класифікації.

Логістична регресія може бути менш надійною у випадках, коли дані мають складну структуру або нелінійні залежності.

**Завдання 2.6. Розробіть програму класифікації даних в файлі data\_multivar\_nb.txt за допомогою машини опорних векторів (Support Vector Machine - SVМ). Розрахуйте показники якості класифікації.**

**Порівняйте їх з показниками наївного байєсівського класифікатора. Зробіть висновки яку модель класифікації краще обрати і чому.**

****

****

Github репозиторія - https://github.com/TAMOTO24/IntelligentSystems