**ЛАБОРАТОРНА РОБОТА №1**

**ПОПЕРЕДНЯ ОБРОБКА ТА КОНТРОЛЬОВАНА КЛАСИФІКАЦІЯ ДАНИХ**

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

**Хід роботи**

**GitHub репозиторій:** https://github.com/AlexanderHorielko/SAI\_Horielko\_PI-59

**Варіант №5**

Завдання 1

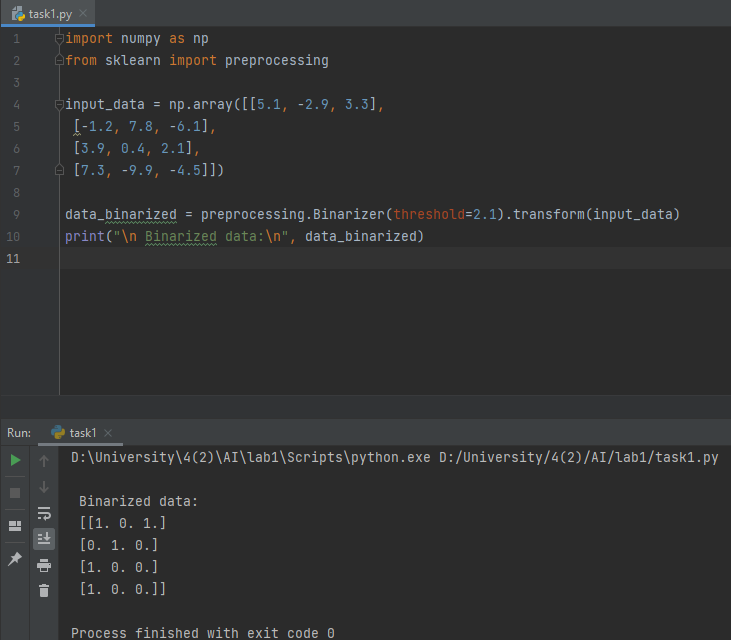


Рис. 1.1.1 Бінарізація

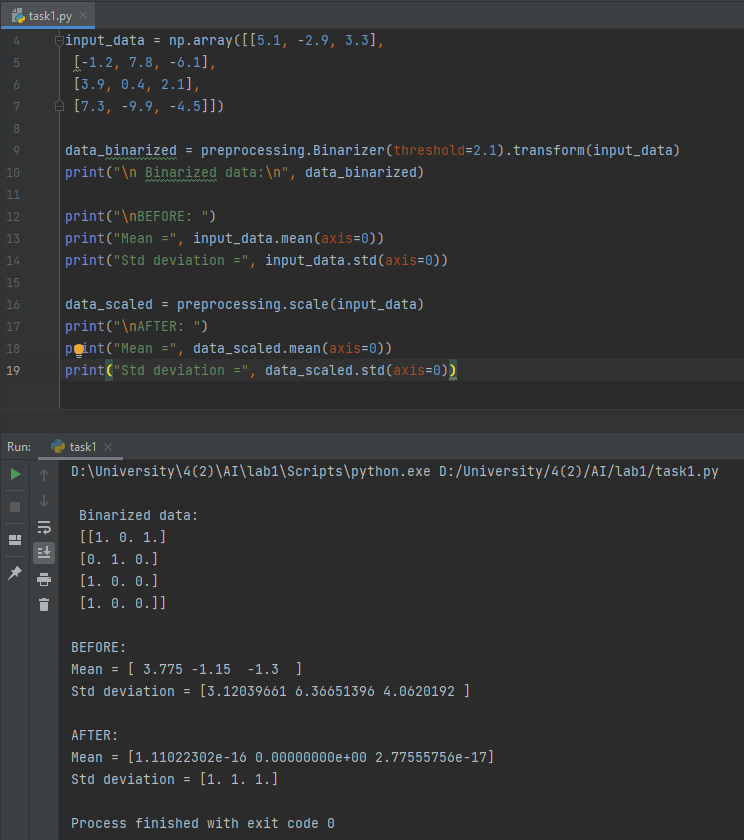


Рис. 1.1.2 Виключення середнього

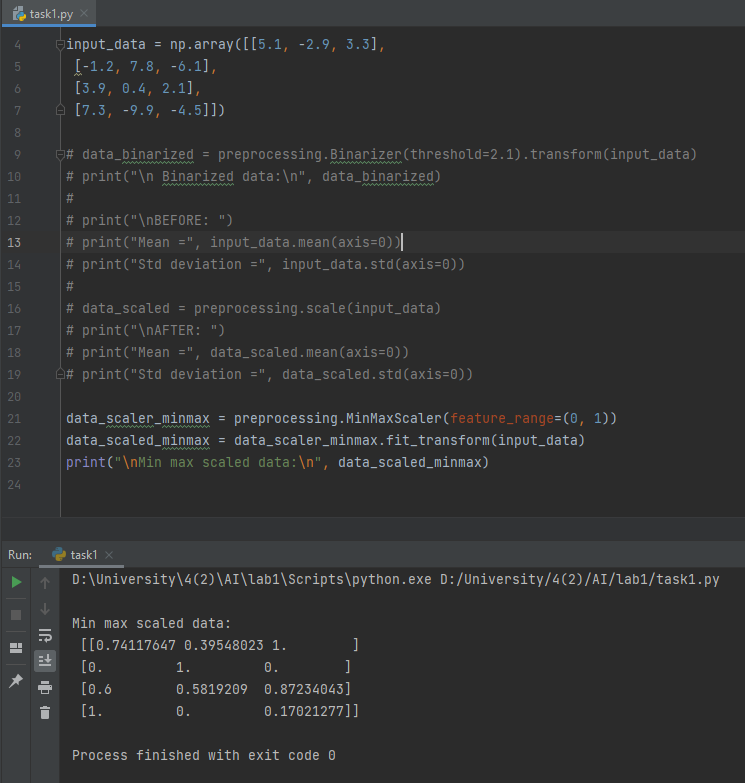


Рис. 1.1.3 Масштабування

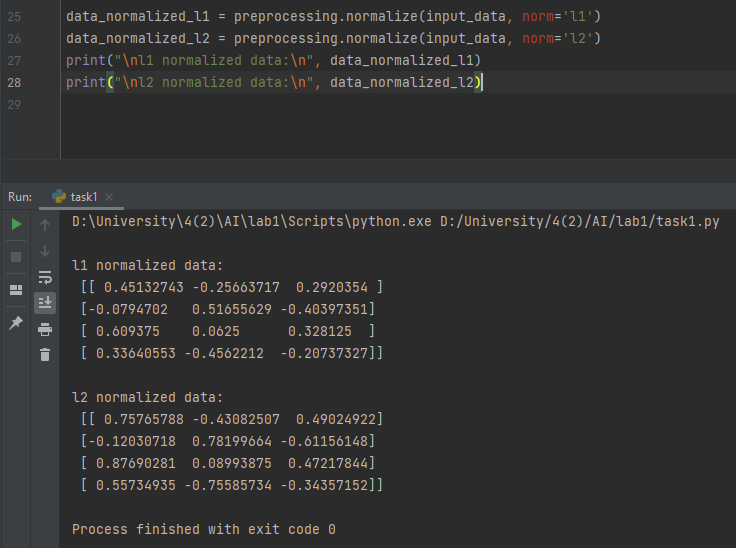


Рис. 1.1.4 Нормалізація

L1-нормалізація використовує метод найменших абсолютних відхилень (Least Absolute Deviations), що забезпечує рівність 1 суми абсолютних значень в кожному ряду. L2-нормалізація використовує метод найменших квадратів, що забезпечує рівність 1 суми квадратів 4 значень. Тому можна зробити висновки, що L1 нормалізація є більш надійною у порівняні з L2.

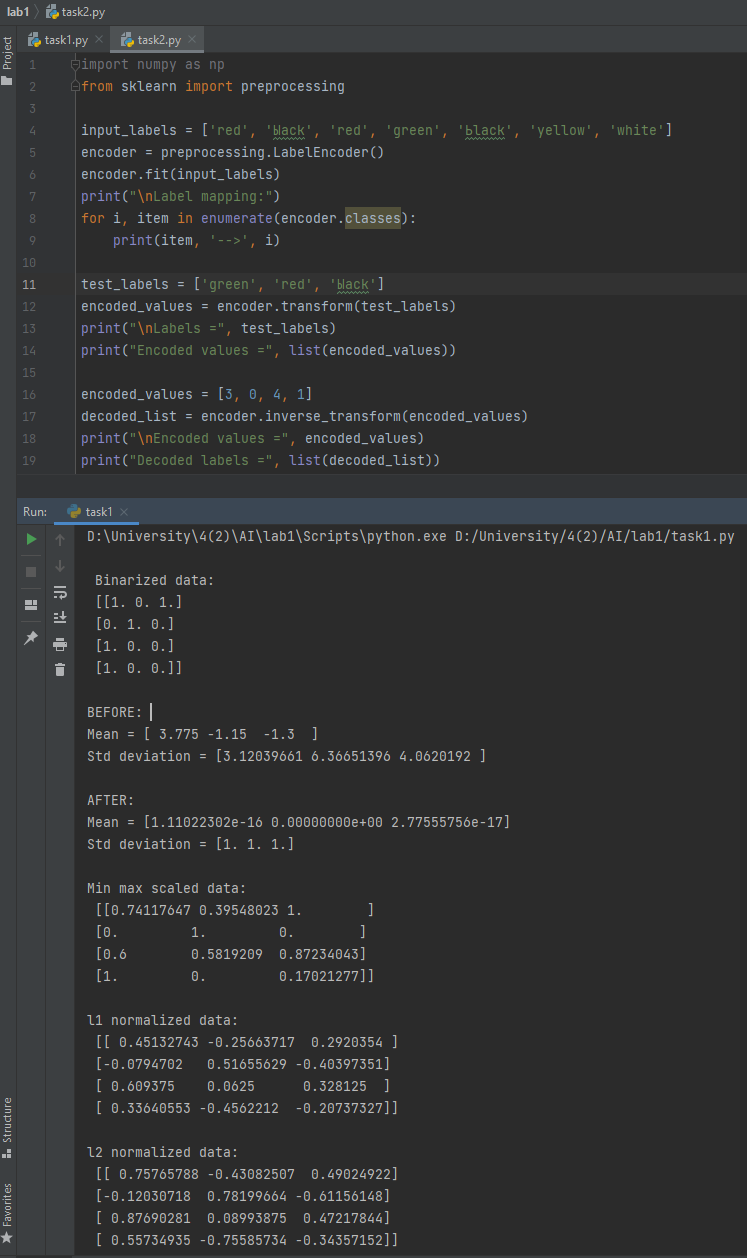


Рис. 1.1.5 Кодування міток

Масив Input\_labels був пересортований за алфавітним порядком, та був проіндексований від 0 до 4. Наступна частина коду демонструє роботу

кодувальника, (слова заміняються числами). Третя частина коду демонструє зворотню процедуру.

**Завдання 2**: Попередня обробка нових даних



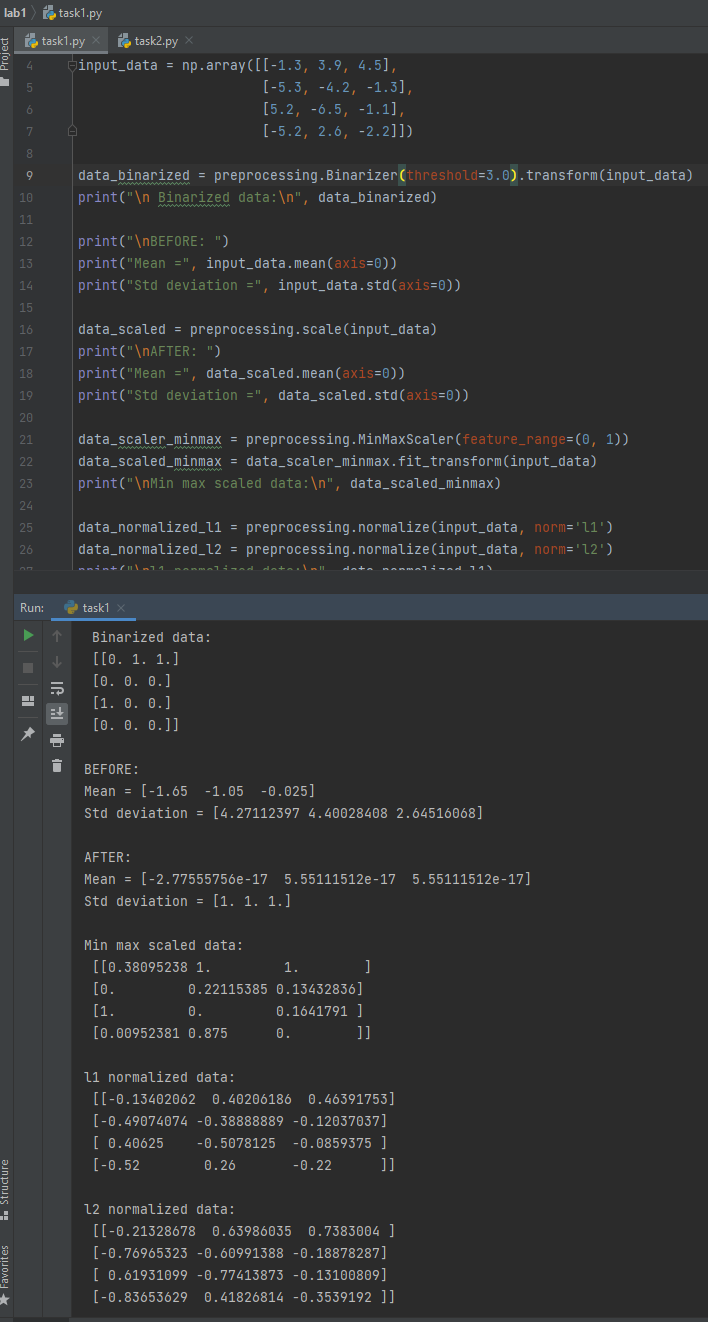


Рис. 1.3 Результат

**Завдання 3**: Класифікація логістичною регресією або логістичний класифікатор

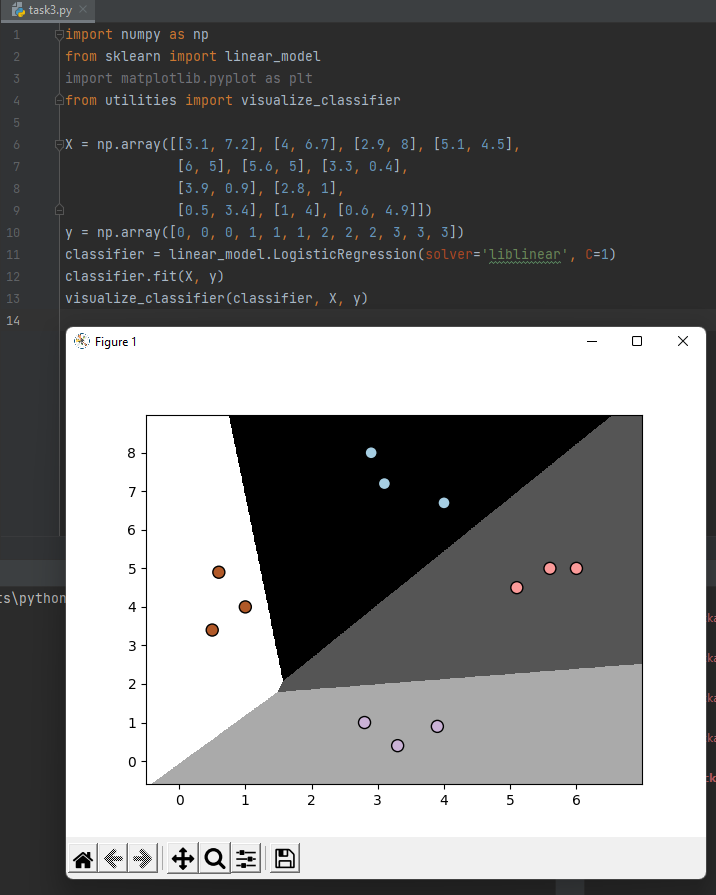


Рисунок 1.4 Візуалізація класифікації логістичною регресією

**Завдання 4**: Класифікація наївним байєсовським класифікатором

Обидва прогони дали ідентичний результат, оскільки генерувались однакові набори даних для навчання й тестування.

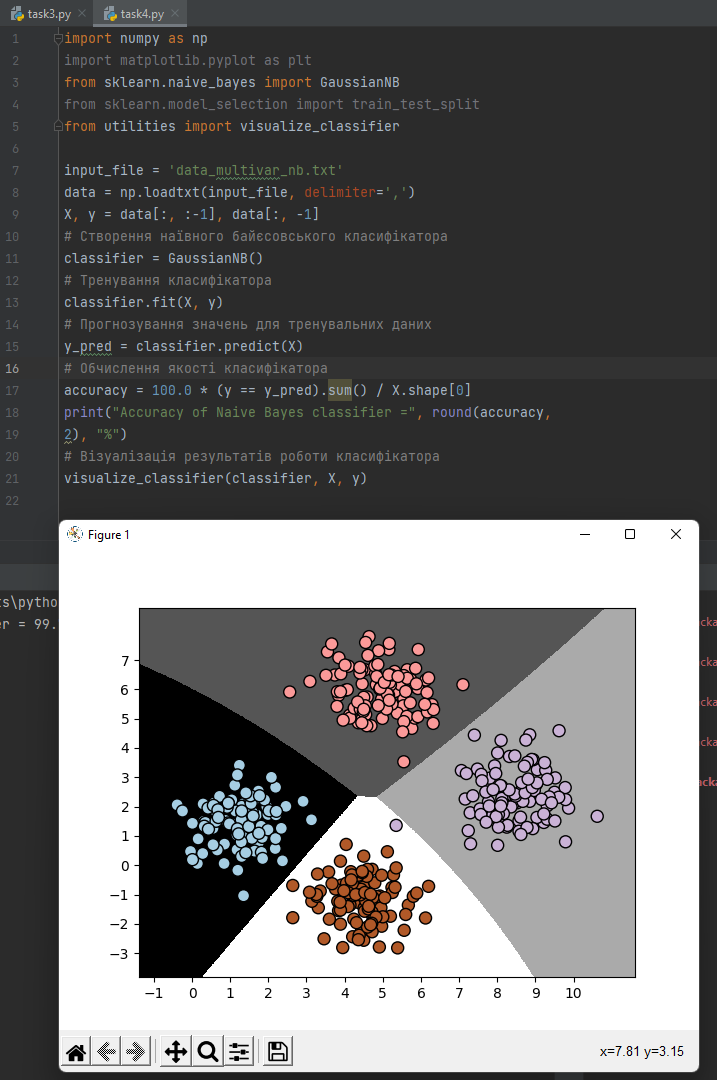


Рисунок 1.5 Класифікація наївним байєсовським класифікатором

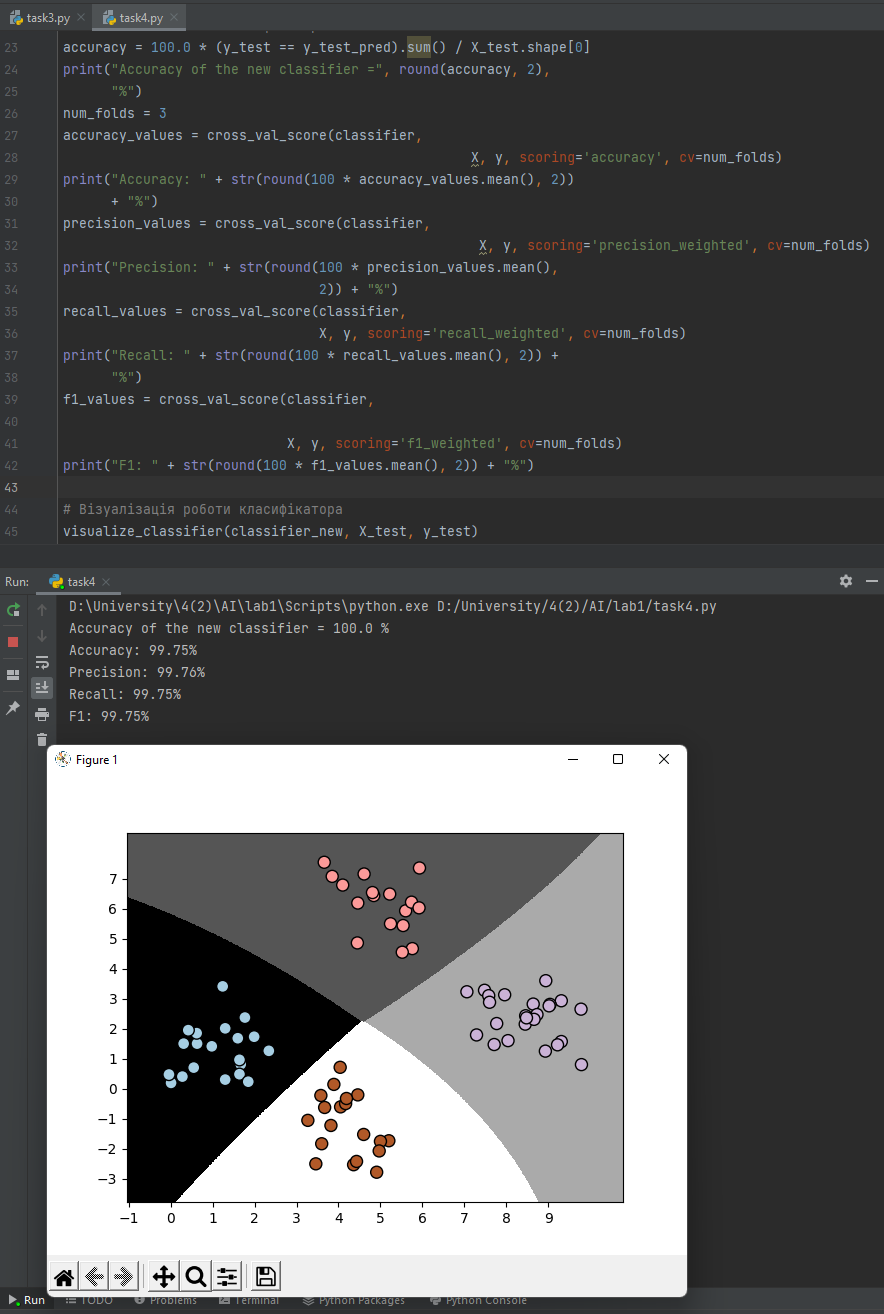


Рисунок 1.6. Класифікація наївним байєсовським класифікатором з обчисленням якості, точності та повноти

**Завдання 5**: Вивчити метрики якості класифікації

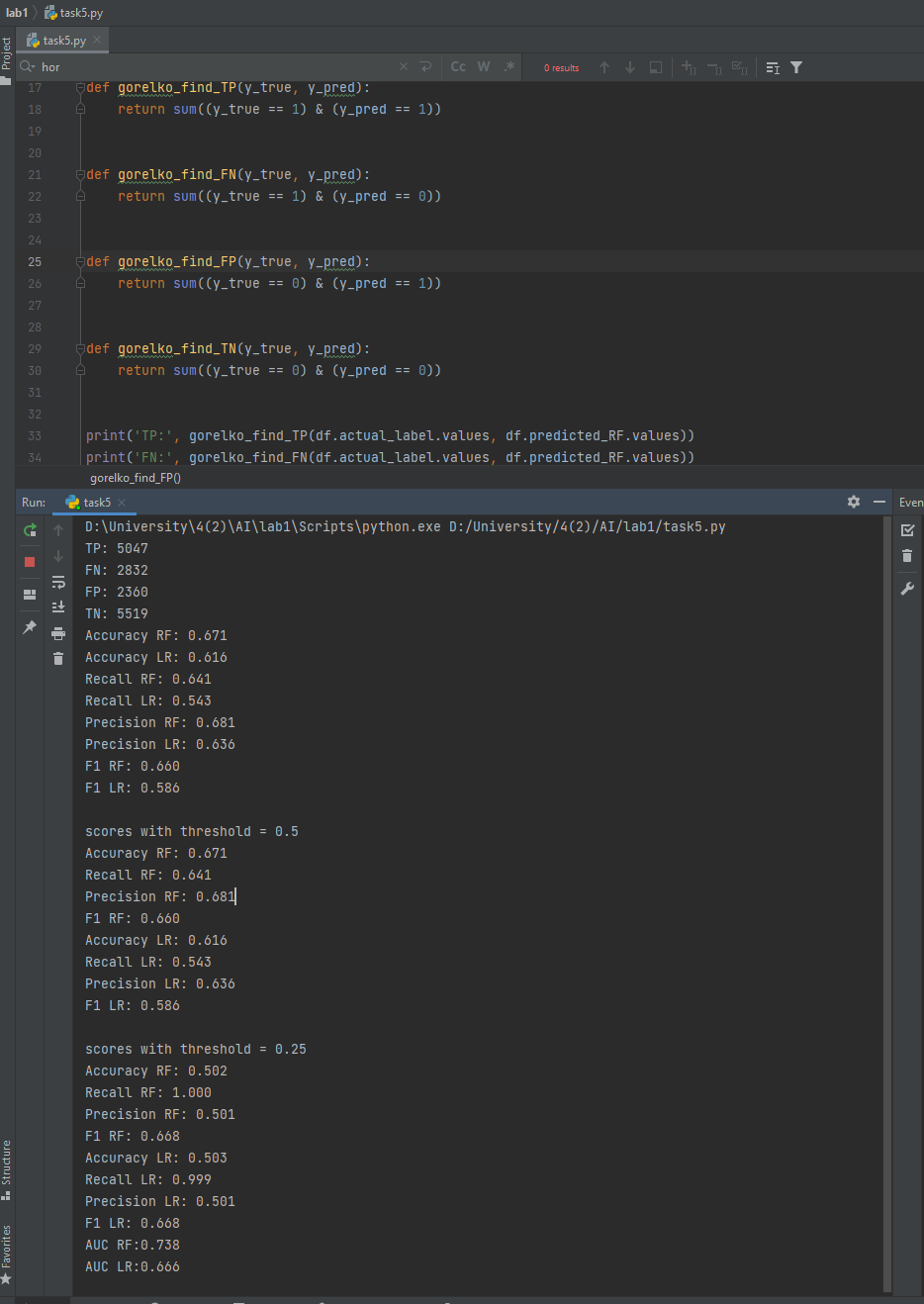


Рисунок 1.7. Порівння моделей RF та LF на кроках 0.25 та 0.5

При порозі 0.5 якість та точність значно вищі, у разі використання моделі RF, тому, як на мене вона є більш оптимальною, але при порозі 0.25 LR модель справляється краще, тому остаточний вибір варто робити виходячи з вхідних даних.

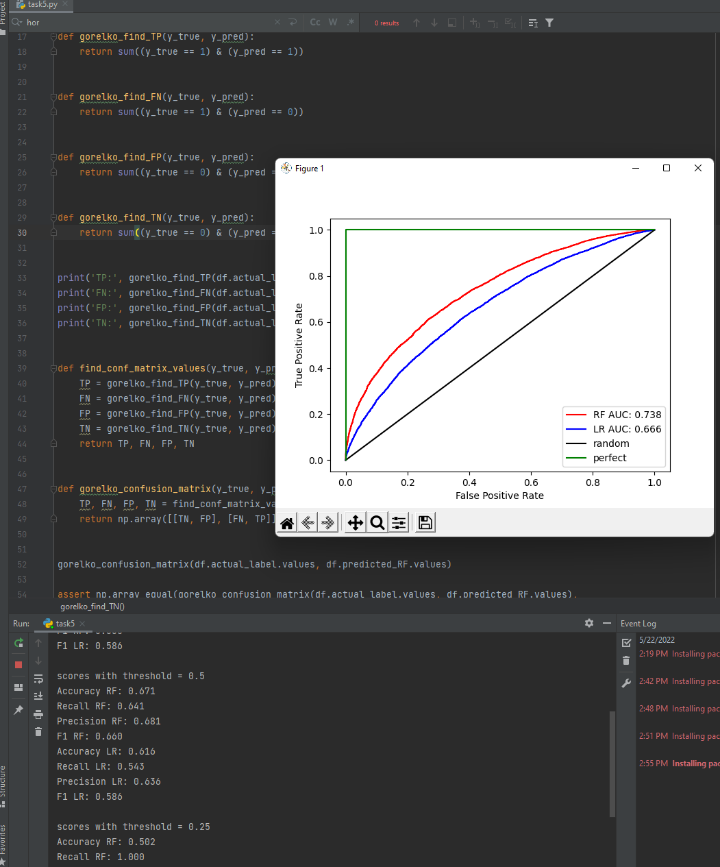
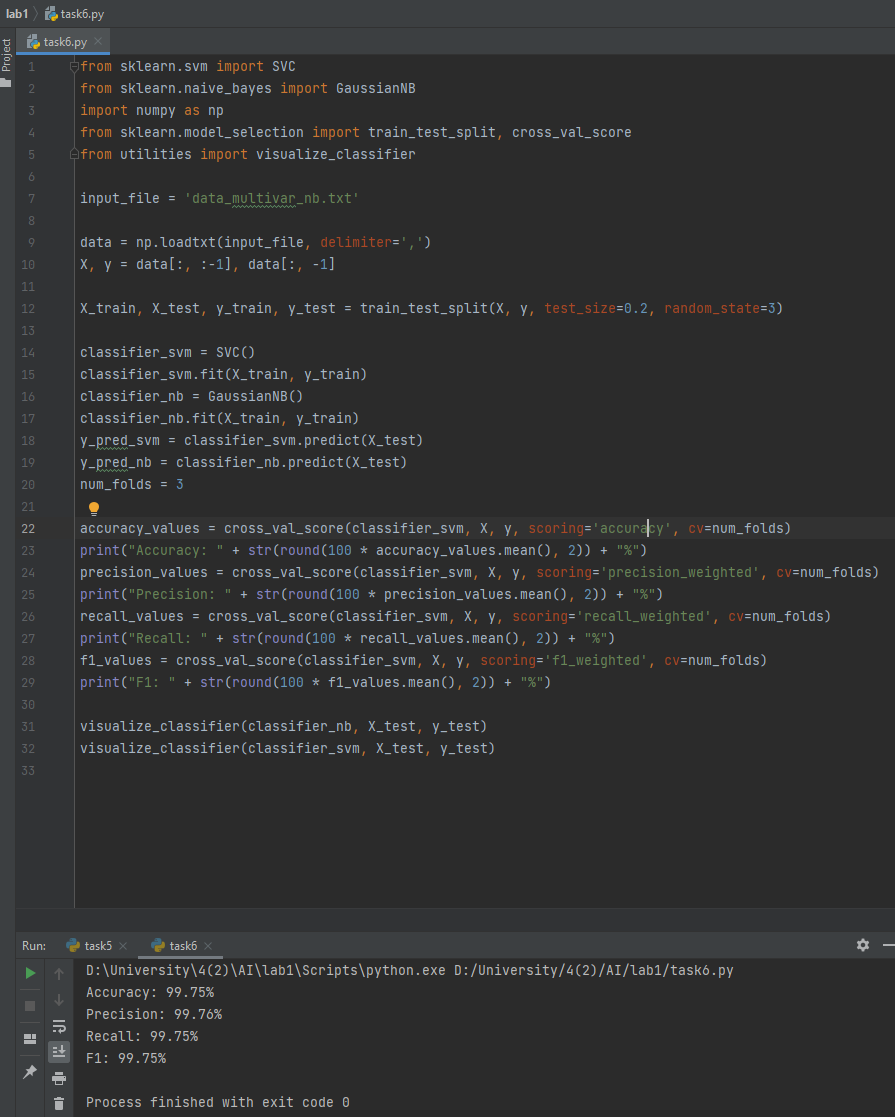
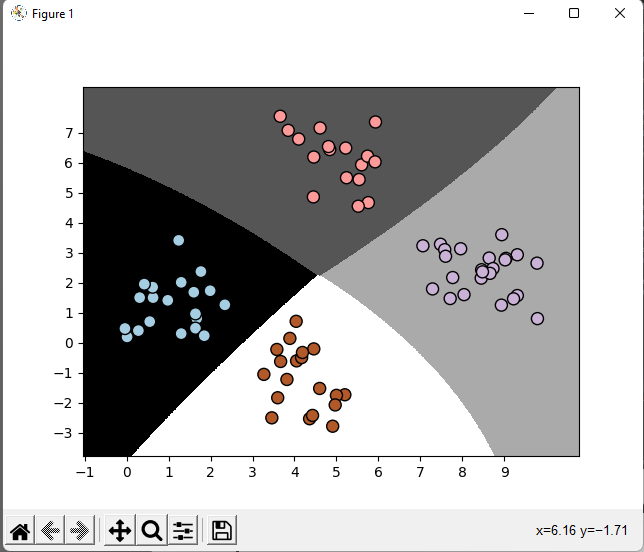


Рисунок 1.8. Порівняння моделей за допомогою кривих ROC

**Завдання 6**: Розробіть програму класифікації даних





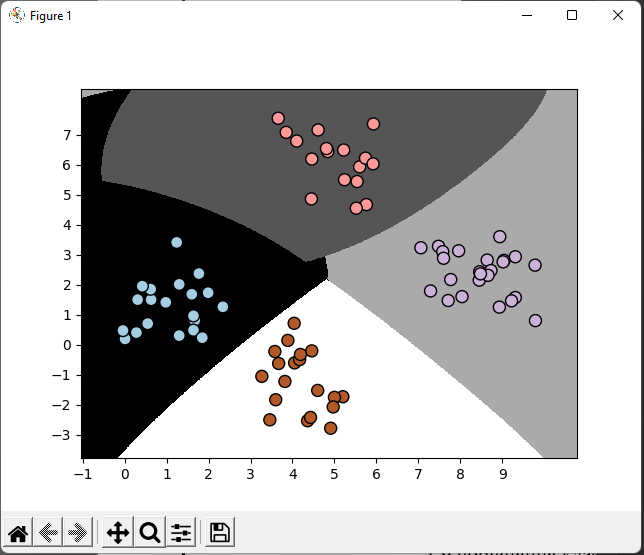


Рисунок 1.9 порівняння класифікаторів наївного байєса та SVM

**Висновок:** використовуючи спеціалізовані бібліотеки та мову програмування Python дослідив попередню обробку та класифікацію даних.