Виконав:

студент групи КМ-32

Друзенко Є.Р.

**Звіт по виконанню завдання**

**Мета завдання:**  
Зчитати датасет з файлу, знайти опуклу оболонку для заданих точок, відобразити її разом з точками вихідного датасету на графіку та зберегти результати у графічному файлі.

**Опис виконання**

1. **Зчитування даних з файлу:** Для зчитування координат точок використано стандартну функцію Python для роботи з файлами. Дані зчитуються з файлу "DS5.txt", де кожен рядок містить координати двох чисел (x, y). За допомогою методу open() відкривається файл для читання, а метод split() розділяє кожен рядок на окремі елементи, які конвертуються в числа типу float.
2. **Визначення опуклої оболонки:** Для обчислення опуклої оболонки застосовано бібліотеку scipy.spatial. Конкретно, використано клас ConvexHull, який реалізує алгоритм для пошуку опуклої оболонки множини точок. Обчислені вершини оболонки зберігаються у списку hull\_points, а за допомогою атрибута simplices класу ConvexHull визначаються відрізки оболонки, які потім використовуються для побудови графіку.
3. **Візуалізація результатів:** Для побудови графіків використано бібліотеку matplotlib.pyplot. Встановлено розміри полотна 960x540 пікселів за допомогою параметра figsize. Точки з початкового датасету відображаються як сині крапки на графіку за допомогою функції scatter(). Опукла оболонка з'єднується відрізками, для чого використано функцію plot(). Крім того, для кращого відображення на графіку було налаштовано рівні масштаби осей за допомогою методу set\_aspect().
4. **Збереження результатів:** Після побудови графіку результат зберігається у графічному файлі "output\_with\_hull.png" за допомогою функції savefig(). Також, точки опуклої оболонки зберігаються в окремому текстовому файлі "convex\_hull.txt", де кожна точка записується в окремий рядок.

**Використані бібліотеки та методи:**

1. **matplotlib.pyplot**:
   * plt.scatter(): для відображення точок датасету.
   * plt.plot(): для відображення відрізків опуклої оболонки.
   * plt.xlim() і plt.ylim(): для налаштування меж осей.
   * plt.savefig(): для збереження графіка у файл.
2. **scipy.spatial.ConvexHull**:
   * ConvexHull(): для обчислення опуклої оболонки множини точок.
   * hull.simplices: для отримання відрізків оболонки, які з'єднують пари точок.
   * hull.vertices: для отримання координат вершин оболонки.
3. **Вбудовані функції Python**:
   * open(): для відкриття файлів.
   * split(): для розділення рядків на компоненти.
   * float(): для конвертації рядків у числа.

**Результат виконання:**

1. Графік з точками датасету та опуклою оболонкою був успішно побудований та збережений у файл "output\_with\_hull.png".
2. Координати точок опуклої оболонки збережено у текстовому файлі "convex\_hull.txt".

