

**Міністерство освіти і науки України  
Національний технічний університет України «КПІ» імені Ігоря Сікорського  
Кафедра обчислювальної техніки ФІОТ**

**ЗВІТ  
з лабораторної роботи №5  
з навчальної дисципліни «Data Science Technology»**

**Тема:**

**РЕАЛІЗАЦІЯ МЕТОДІВ МАШИННОГО НАВЧАННЯ (MACHINE LEARNING  
(ML))**

**Виконав:**

Студент 3 курсу кафедри ОТ ФІОТ,  
Навчальної групи ІМ-13  
Тавлуй Д. О.

**Перевірив:**

Професор кафедри ОТ ФІОТ  
Писарчук О.О.

**Київ 2023**

## **I. Мета:**

Виявити дослідити та узагальнити особливості аналізу даних з використанням методів та технологій машинного навчання (Machine Learning (ML)).

## **II. Завдання:**

**Завдання II рівня складності 8 балів:** реалізувати на вибір ДВІ з п'яти сформованих груп технічних вимог.

Я обрав варіант 1 та 3 групи вимог

### **Група технічних вимог\_1:**

Реалізувати кластеризацію вхідних даних, отриманих Вами у ході виконання Дз\_1, модельних та (або) реальних – на власний вибір. Методи Machine Learning з переліку: kmeans (k-середніх); Support Vector Machine (машина опорних векторів); k-nearest neighbors (найближчих сусідів); ієрархічна кластеризація – для кластеризації обраних даних обрати самостійно. Провести аналіз та пояснення отриманих результатів, сформулювати висновки.

### **Група технічних вимог\_3:**

Підрахувати кількість об'єктів на обраному цифровому зображенні. Об'єкти, що підлягають обрахунку обрати самостійно. Зміст етапів попередньої обробки зображень (корекція кольору, фільтрація, векторизація, кластеризація) має бути результатом R&D процесів, що конкретизується обраним зображенням і об'єктами для підрахунку. Провести аналіз отриманих результатів, сформулювати висновки.

## **III. Результати виконання лабораторної роботи.**

Після запуску головного файлу відкриється вікно з вибором групи технічних вимог лабораторної:

Оберіть завдання лабораторної:

1 - Кластеризація даних

2 - Підрахувати кількість об'єктів на фото

Перший режим виконує умову першої групи технічних вимог – Кластеризацію вхідних даних.

Другий режим – третю умову технічних вимог. Далі детальніше опишу ці режими.

## Результати архітектурного проектування та їх опис

Я обрав модульну архітектуру, щоб програми достатньо самостійні, тому я вирішив зробити кожну в окремому файлі – `clustering.py` та `object_counter.py`. Також є головний файл `main.py`.

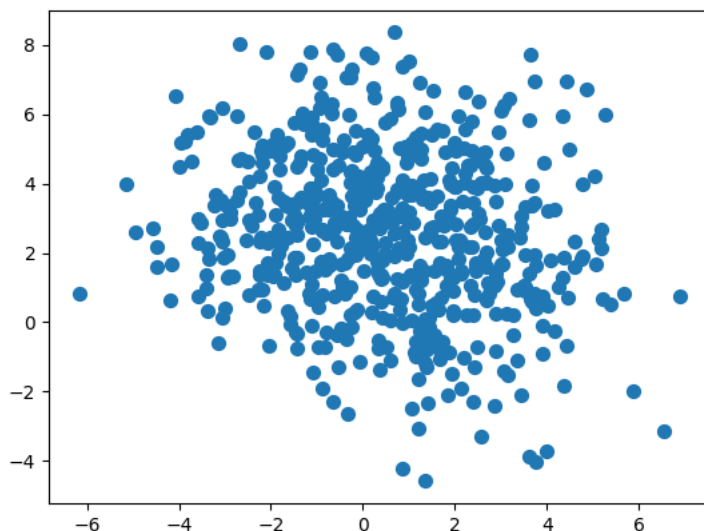
### Опис структури проекту програми

У файлі `clustering.py` створюється функція для побудови кластерів за допомогою `k-means`, далі створена функція яка їх викликає за заданими параметрами. У файлі `object_counter.py` міститься функція яка повинна підраховувати об'єкти. Далі ці функції з двох файлів імпортуються у файл `main.py`, де і відбувається їх виклик.

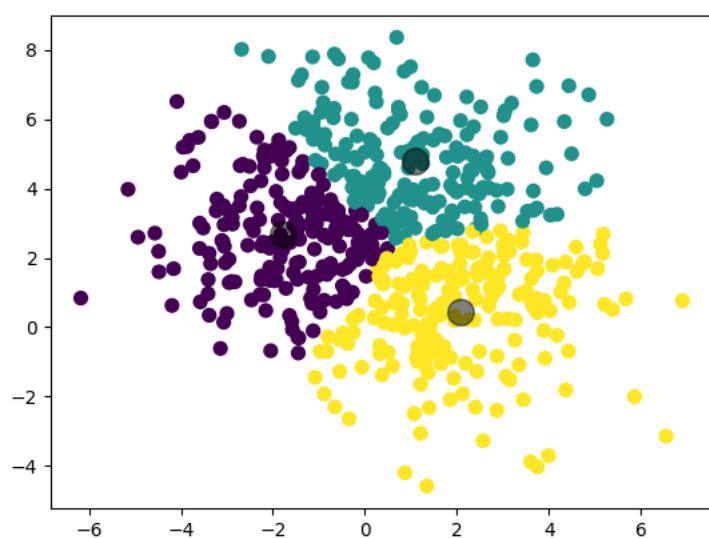
## Результати роботи програми відповідно до завдання

### Група технічних вимог\_1:

Перш за все, я створив штучний датасет, використовуючи для нього `k-means`

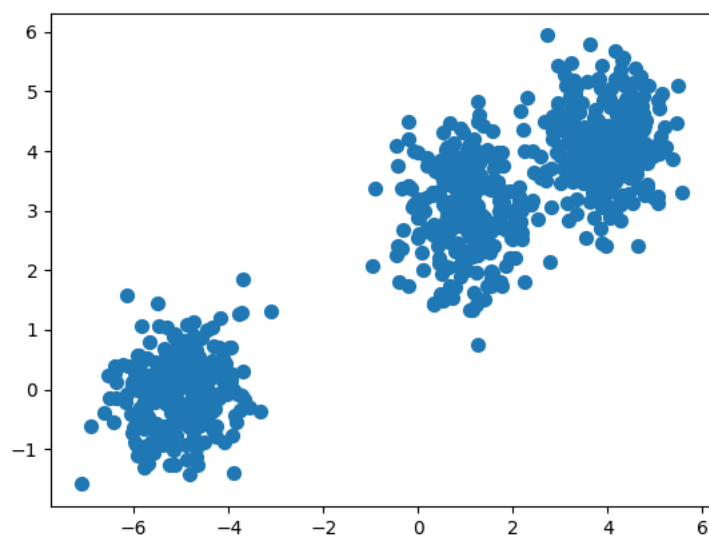


Датасет при 600 екземплярах та середньоквадратичному відхиленні 1.8

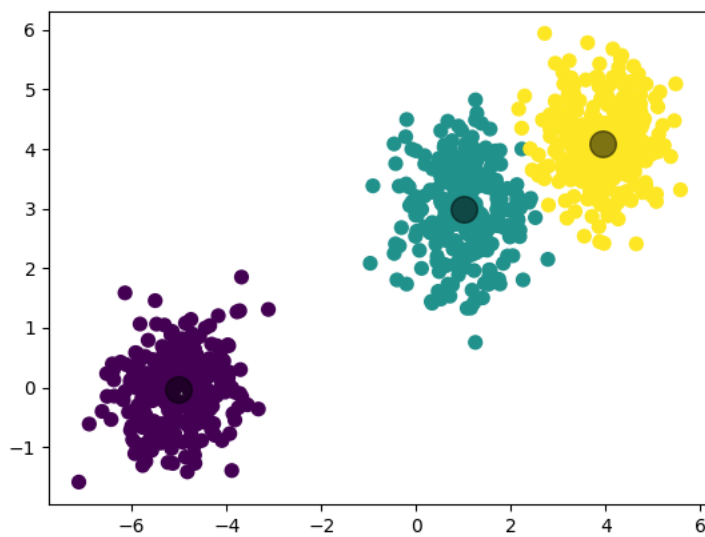


На перший погляд, центри здаються досить точними

Далі я створив датасет з вже заданими центрами



Тут 800 екземплярів та середньоквадратичне відхилення дорівнює 0.7



Тепер можна порівняти задані центри та дійсні

```
Expected centers: [[-5, 0], [1, 3], [4, 4]]
```

```
Actual centers: [[-5.02524766 -0.02185852]
```

```
[ 1.00474014  2.98587882]
```

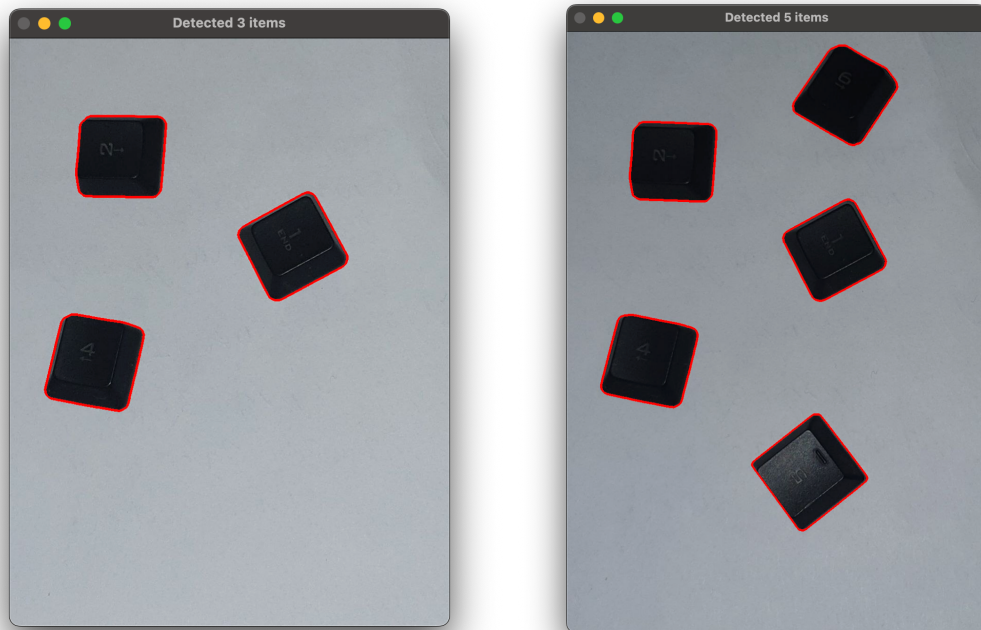
```
[ 3.93100845  4.0943124  ]]
```

Як можна побачити, центри визначені достатньо правильно, але є невелика похибка

### Група технічних вимог\_3:

Спочатку відбувається конвертація зображення у чорно-білий формат, далі інвертується значення пікселів. За допомогою `cv2.findContours` знаходяться контури на обробленому зображенні. Підраховується кількість об'єктів на зображенні, які мають площу більше 100 пікселів. Кожен знайдений об'єкт позначається червоним контуром.

Попередньо, я сфотографував 3 клавіші клавіатури, а також 5 клавіш.



Як видно, програма точно виділила їх та зверху видно правильну кількість клавiш на кожному фото.

**Програмний код, що забезпечує отримання результату**

#### **main.py**

```
from clustering import main_cluster
from object_counter import counter_image

if __name__ == "__main__":
    print('Оберiть завдання лабораторної:\n')
    print('1 - Кластеризація даних\n')
    print('2 - Підрахувати кількість об'єктів на фото\n')
    mode = int(input('Режим:'))

    if mode == 1:
        main_cluster()
    elif mode == 2:
        counter_image('keycaps.jpg')
        counter_image('keycaps-5.jpg')
```

#### **clustering.py**

```
import matplotlib.pyplot as plt
from sklearn.datasets import make_blobs
from sklearn.cluster import KMeans

def k_means(dataset):
    plt.scatter(dataset[:, 0], dataset[:, 1], s=50)
    plt.show()

    kmeans = KMeans(n_clusters=3)
    kmeans.fit(dataset)
```

```

y_kmeans = kmeans.predict(dataset)

plt.scatter(dataset[:, 0], dataset[:, 1], c=y_kmeans, s=50,
cmap='viridis')

centers = kmeans.cluster_centers_1
plt.scatter(centers[:, 0], centers[:, 1], c='black', s=200, alpha=0.5)
plt.show()

return centers

def main_cluster():
    dataset, _ = make_blobs(n_samples=600, cluster_std=1.8, random_state=0)
    k_means = kmeans(dataset)

    expected_centers = [[-5, 0], [1, 3], [4, 4]]
    dataset, _ = make_blobs(n_samples=800, centers=expected_centers,
cluster_std=0.7)
    actual_centers = k_means(dataset)

    print(f'Expected centers: {expected_centers}')
    print(f'Actual centers: {actual_centers}')
    return

```

## object\_counter.py

```

import cv2

def counter_image(filename):
    img = cv2.imread(filename)
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    min_threshold = 100
    _, threshold = cv2.threshold(gray, min_threshold, 255,
cv2.THRESH_BINARY)
    converted = cv2.bitwise_not(threshold)
    contours, _ = cv2.findContours(converted, cv2.RETR_TREE,
cv2.CHAIN_APPROX_SIMPLE)

    total = 0
    for contour in contours:
        area = cv2.contourArea(contour)
        if area > 100:
            total += 1
            cv2.drawContours(img, [contour], -1, (0, 0, 255), 3)

    cv2.imshow(f'Detected {total} items', img)
    cv2.waitKey(0)
    cv2.destroyAllWindows()

    return

```

## Висновки

У ході виконання цієї лабораторної роботи я здобув нові знання та навички в різних методах використання Machine Learning та Computer Vision. Я освоїв техніку роботи з кластеризацією за допомогою методу k-means і перевінив її ефективність. Також я оволодів вмінням розпізнавати конкретні об'єкти на фотографіях та визначати їхню кількість.