**Lab 5 วิชา การเรียนรู้แบบไม่มีผู้สอนของเครื่อง (Unsupervised Machine Learning)**

**หัวข้อ**

1. Gaussian Mixture Model: Gaussian Distribution
2. DBSCAN

**คำสั่งส่วนที่ 1**

1. *การทดลองในครั้งนี้ อนุญาตให้ใช้ Library ได้ทั้งหมด*
2. ใช้ชุดข้อมูล load\_iris()

iris = datasets.load\_iris()  
X = iris.data  
y = iris.target

1. เลือกข้อมูลแอททริบิวท์ Sepal\_length จากข้อมูลตัวอย่าง เพื่อสร้างกราฟแสดงความถี่ (Histogram) ของ Sepal\_length ของดอกไอริสทั้งหมด
2. สร้างกราฟแสดงความถี่ของ Sepal\_length ของดอกไอริสแต่ละประเภท (ได้ผลลัพธ์เป็น 3 กราฟ)
3. บันทึกกราฟลงในส่วนบันทึกผลการทดลองส่วนที่ 1
4. \*\*(Optional) ลองใช้ Guassian Mixture Model ในการจำแนกข้อมูล load\_iris (ไม่ต้องบันทึกผล)

**คำสั่งส่วนที่ 2**

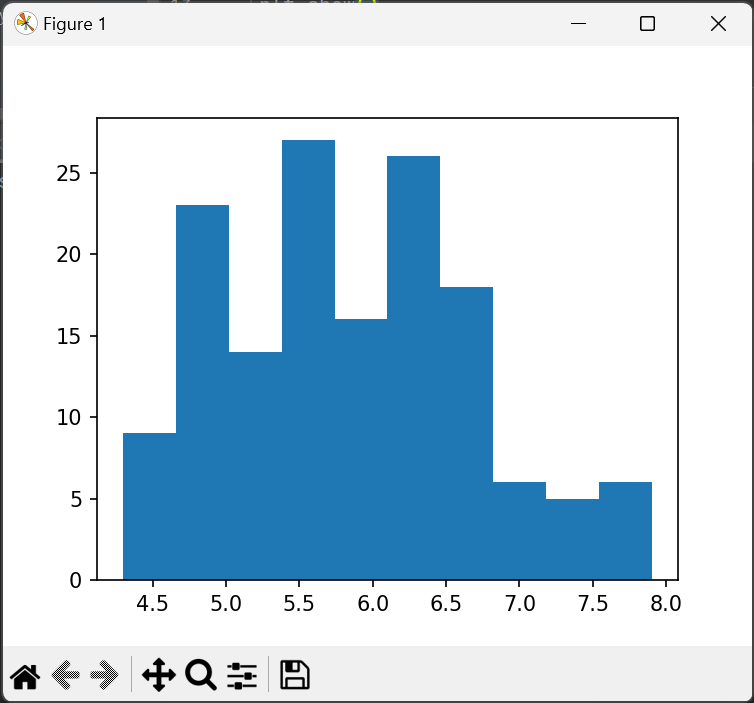
1. ตรวจสอบหมายเลข Label ของข้อมูลแต่ละตัว บันทึกผลลงในช่อง target
2. ใช้ DBSCAN ในการวิเคราะห์คลัสเตอร์ของ load\_iris() โดยใช้ข้อมูลทั้ง 100% ในการเทรน โดยไม่กำหนด target ให้ model (เทรนโดยใช้แค่ data) บันทึกผลลัพธ์ของการทำคลัสเตอริ่ง
3. บันทึกผลการทดลองลงในตารางบันทึกผลส่วนที่ 2 ให้สมบูรณ์
4. รายละเอียดการบันทึกผล
   1. บันทึกผลลงในใบงาน (หน้า 2 เป็นต้นไป) ไม่ต้องกังวลเกี่ยวกับแบบฟอร์มหรือจำนวนหน้า แม้ว่า ตารางจะตกหน้า แยกออกจากกัน หรือทำให้มีพื้นที่ว่างในบางหน้ามากเกินไป ก็ไม่มีปัญหา
   2. เซฟไฟล์นี้ ตั้งชื่อว่า Lab05\_result\_รหัสนักศึกษา.docx ส่งพร้อมโค้ด ตั้งชื่อว่า Lab05\_code\_รหัสนักศึกษา.py (แยกไฟล์กันมา)
   3. ส่งไฟล์โดยอัปโหลดไปยัง <https://silpakorn-my.sharepoint.com/:f:/g/personal/aonpong_p_su_ac_th/EmRn8pOWnXJOvxtOcoRs98IBI6ex0H6-AwN_m-BxJYG3gg?e=XSs2Zt>

\* พยายามควบคุมพารามิเตอร์ โค้ดที่ส่งจะต้องไม่มี Warning (เช่น Convergence Warning หรือ Warning อื่น ๆ)

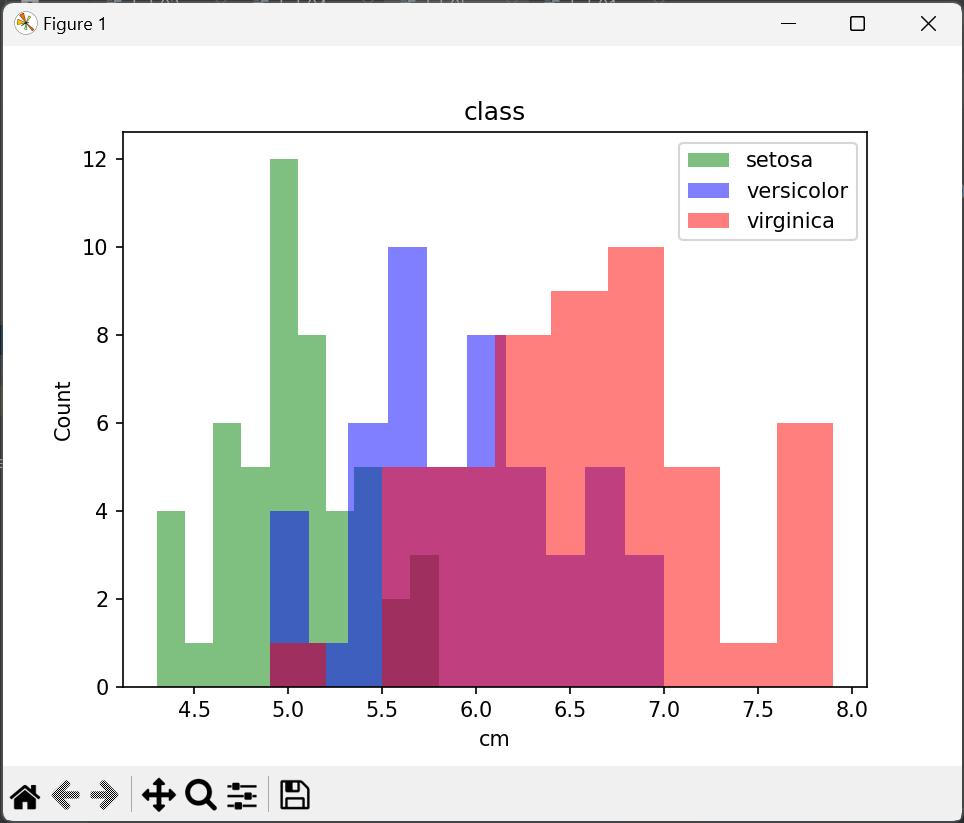
**630710440 พีท อ่อนทอง**

**บันทึกผลการทดลอง ส่วนที่ 1**

1. Sepal\_length ของข้อมูล load\_iris ทั้งหมด



1. Sepal\_length ของข้อมูล load\_iris (cluster 1)
2. Sepal\_length ของข้อมูล load\_iris (cluster 2)
3. Sepal\_length ของข้อมูล load\_iris (cluster 3)



**บันทึกผลการทดลอง ส่วนที่ 2**

**ผลการทดลอง DBSCAN**

|  |  |
| --- | --- |
| Clustering result | |
| DBSCAN | Target |
| 0 0 0 0 0 0 0 0 0 0 0 0 0 0 -1 -1 0 0 0 0 0 0 -1 0  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 -1 0 0 0 0 0 0  0 1  1 1 1 1 1 1 1 -1 1 1 -1 1 -1 1 -1 1 1 1 -1 1 2 1  2 1 1 1 1 1 1 -1 1 1 1 2 -1 -1 1 -1 1 1 1 1 1 -1 1 1  1 1 -1 1  -1 2 3 3 3 -1 -1 -1 -1 -1 3 3 3 2 -1 3 3 -1 -1 -1  3 2 -1 2 3 -1 2 2 3 -1 -1 -1 3 2 -1 -1 3 3 2 3 3 3 2 3  3 3 2 3 3 2 | 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,  0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,  0, 0, 0, 0, 0, 0,  1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,  1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,  1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,  2, 2, 2, 2, 2, 2, 2, 2, 2, 2,  2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,  2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2 |

clustering\_DBSCAN = DBSCAN(eps=0.4, min\_samples=5).fit(dataX)

\*เพราะคลัสเตอริ่ง เป็นวิธีการที่ไม่สนใจว่า label ของกลุ่มคืออะไร รู้เพียงแค่ว่าเป็นกลุ่มเดียวกันหรือไม่ ดังนั้น เราอาจพบเหตุการณ์ที่หมายเลข label ไม่ตรงกันได้บ่อยครั้ง “จากโมเดลที่เทรนมาด้วยกัน ถ้า label ที่ควรเป็นกลุ่มเดียวกันมีหมายเลขเดียวกัน ถือว่าใช้ได้”

\*\*ผลการทดลองในตารางนี้ เมื่อวางข้อมูลทั้งหมดลงมาแล้วอาจยาวหลายหน้า ไม่ต้องสนใจ วางได้เลย