

การทดลองจัดกลุ่มข้อมูล 2 มิติ ด้วยวิธี k-Means และ Hierarchical Clustering

ชุดข้อมูลที่ 1

data2Dset1.csv

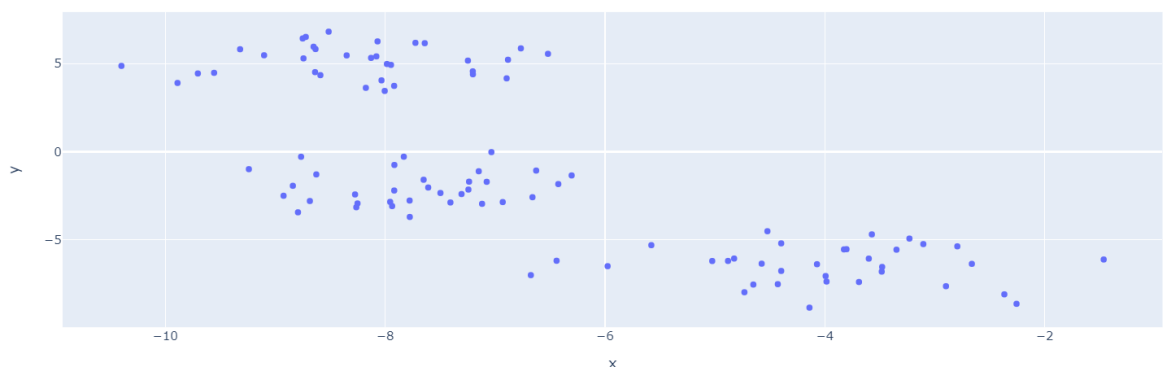
1. เขียนโปรแกรมภาษาไพทอน อ่านข้อมูลจากไฟล์ชุดข้อมูลที่กำหนดให้ ส่วนของโค้ดที่แนะนำให้ใช้เพื่ออ่านข้อมูล

```
data = pd.read_csv("data2Dset1.csv",header=None)
data.columns = ["x","y"]
data.head()
```

	x	y
0	-4.575007	-6.364897
1	-7.202692	4.560245
2	-7.148368	-1.115191
3	-7.915773	-0.757674
4	-7.118251	-2.965019

2. เขียนโปรแกรม plot จุดข้อมูล

```
import plotly.express as px
fig = px.scatter(data, x="x", y="y")
fig.show()
```



3. เขียนโปรแกรมจัดกลุ่มชุดข้อมูลที่อ่านเข้ามา โดยใช้วิธี k-Means

a. ให้เลือกใช้ k ที่ต่างกัน 3 ค่า

K = 2 ;

```
model_kmeans = cluster.KMeans(n_clusters=2)
model_kmeans.fit(data)
data['cluster_id'] = model_kmeans.labels_
centroids = model_kmeans.cluster_centers_
print(centroids)
```

```
[[-5.84938602 -4.29998479]
 [-8.24132694  5.12663729]]
```

K = 3 ;

```
model_kmeans = cluster.KMeans(n_clusters=3)
model_kmeans.fit(data)
data['cluster_id'] = model_kmeans.labels_
centroids = model_kmeans.cluster_centers_
print(centroids)
```

```
[[-4.0259642  -6.46730659  0.          ]
 [-8.24132694  5.12663729  1.          ]
 [-7.72806305 -2.06698658  0.          ]]
```

K = 4 ;

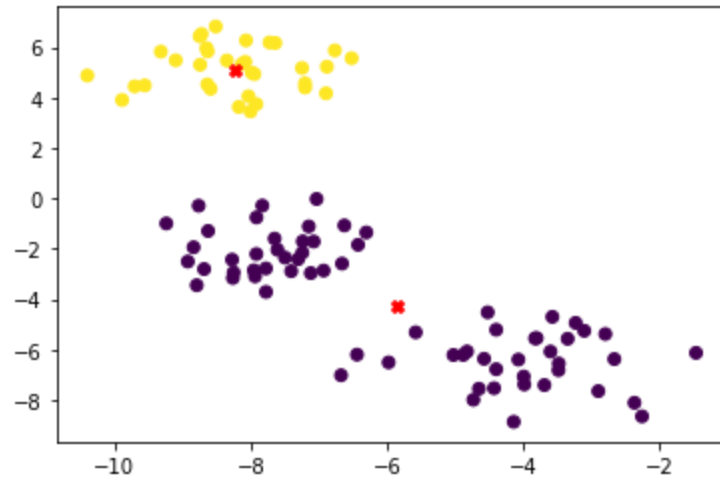
```
model_kmeans = cluster.KMeans(n_clusters=4)
model_kmeans.fit(data)
data['cluster_id'] = model_kmeans.labels_
centroids = model_kmeans.cluster_centers_
print(centroids)
```

```
[[-8.24132694e+00  5.12663729e+00  1.00000000e+00]
 [-3.36095853e+00 -5.43813035e+00  0.00000000e+00]
 [-7.72806305e+00 -2.06698658e+00  2.00000000e+00]
 [-4.38869457e+00 -7.02873000e+00  2.22044605e-16]]
```

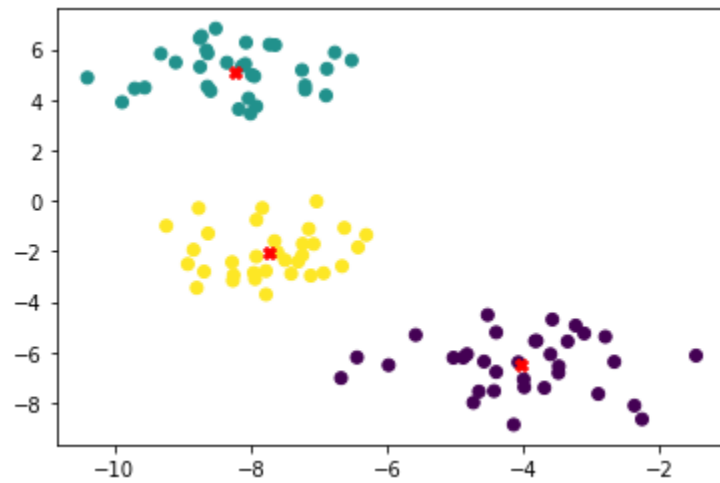
```
plt.scatter(data['x'],data['y'],c=data['cluster_id'])
plt.scatter(centroids[:,0],centroids[:,1],marker='X',c='r')
plt.show()
```

b. plot ผลการจัดกลุ่มที่ได้ สำหรับแต่ละค่า k

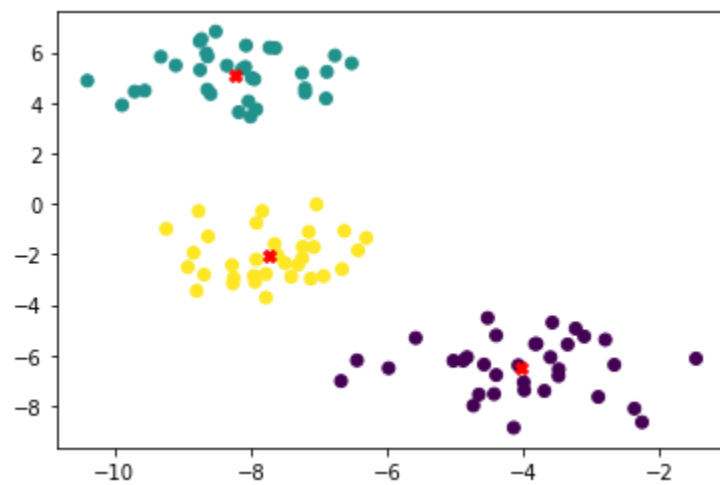
k = 2 ;



$k = 3$;



$k = 4$;



- a. วิเคราะห์ผลการทดลองที่ได้ (ควรมี $k \geq 1$ ค่าที่นักศึกษาคิดว่าดีที่สุดสำหรับชุดข้อมูลดังกล่าว)

เลือกค่า $K = 3$ เพราะเห็นว่าจะที่ plot มาข้อมูลจะเกาะกลุ่มอยู่ประมาณ 3 กลุ่มใหญ่ๆ

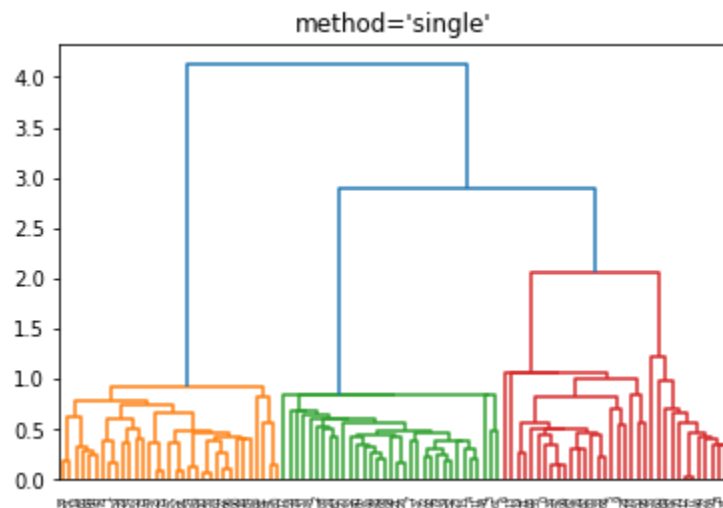
ถ้าเป็น 4 ตัวข้อมูล สีฟ้ากับสีม่วงจะอยู่ใกล้กันมาก และ ถ้าเป็น 2 สีม่วงมันจะสามารถแบ่งได้อีก 2 กลุ่มใหญ่จึงคิดว่า $K = 3$ ดีที่สุด

4. เขียนโปรแกรมจัดกลุ่มชุดข้อมูลที่อ่านเข้ามา โดยใช้วิธี Hierarchical Clustering

- ให้เลือกใช้ method ที่ต่างกัน 3 แบบ แสดง dendrogram ที่ได้แต่ละแบบ

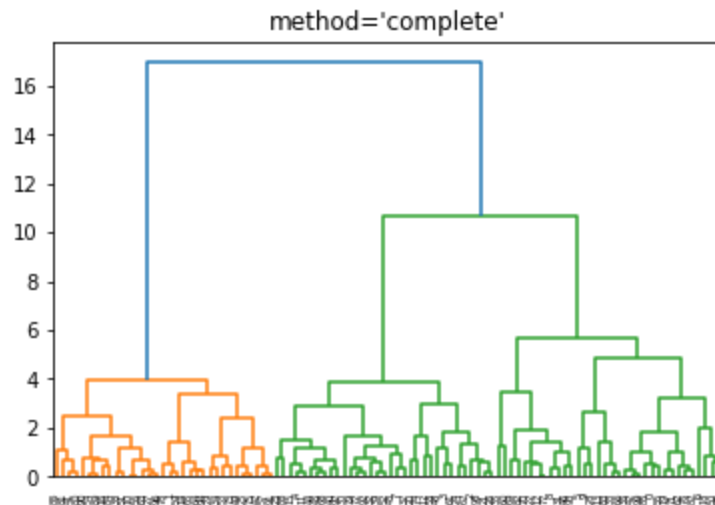
Single ;

```
linkage_data = linkage(data,method='single',metric='euclidean')
dendrogram(linkage_data)
plt.title("method='single'")
plt.show()
```



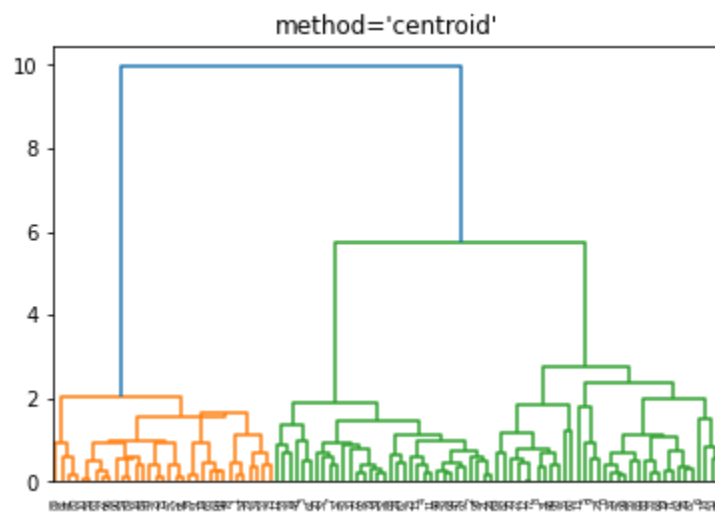
Complete ;

```
linkage_data = linkage(data,method='complete',metric='euclidean')
dendrogram(linkage_data)
plt.title("method='complete'")
plt.show()
```



Centroid ;

```
linkage_data = linkage(data,method='centroid',metric='euclidean')
dendrogram(linkage_data)
plt.title("method='centroid'")
plt.show()
```



- b. เลือก cut-off โดยกำหนด criterion='distance' และให้นักศึกษาเลือกระนาบค่า t ที่คิดว่า เหมาะสม สำหรับแต่ละ dendrogram ที่ได้ในข้อ a)

Single ; $t = 2.5$

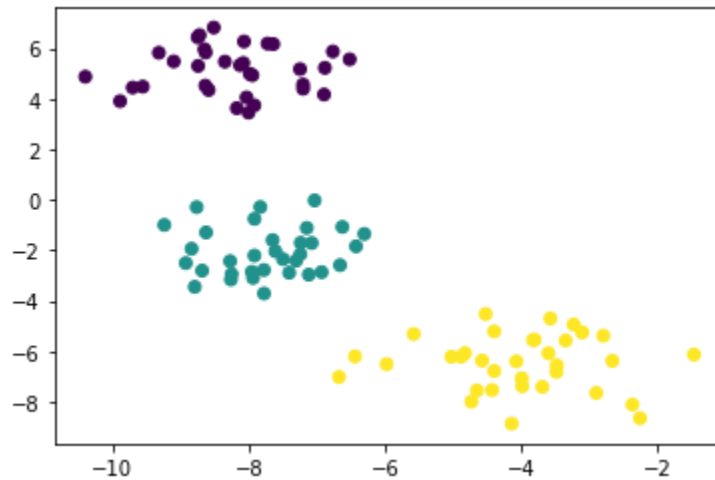
Complete ; $t = 6$

Centroid ; $t = 3$

c. Plot ผลการการจัดกลุ่ม ที่ได้แต่ละแบบในข้อ b)

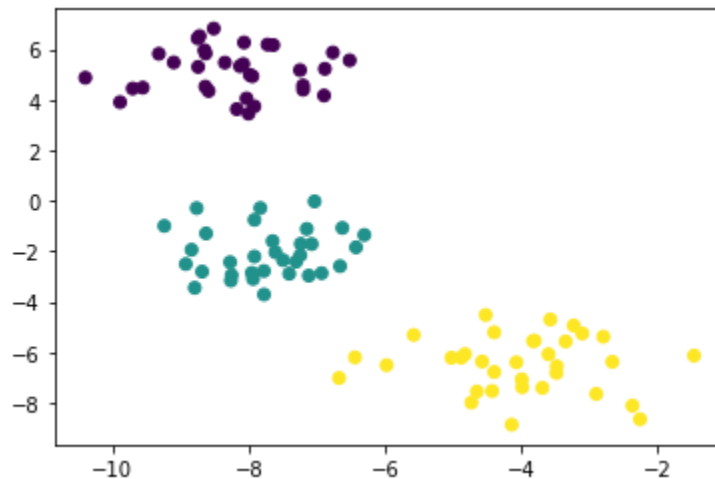
Single ;

```
cluster_id = fcluster(linkage_data,t=2.5,criterion='distance')
plt.scatter(data["x"],data["y"],c=cluster_id)
plt.show()
```



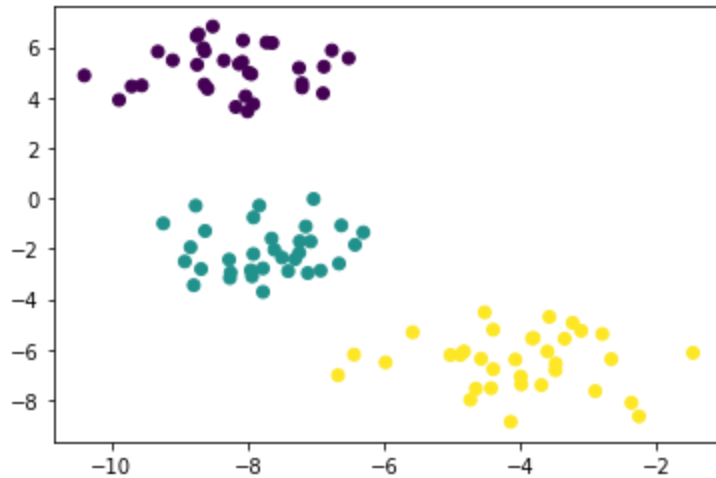
Complete ;

```
cluster_id = fcluster(linkage_data,t=6,criterion='distance')
plt.scatter(data["x"],data["y"],c=cluster_id)
plt.show()
```



Centroid ;

```
cluster_id = fcluster(linkage_data,t=3,criterion='distance')
plt.scatter(data["x"],data["y"],c=cluster_id)
plt.show()
```



5. เขียนบรรยายสรุปผลการทดลอง แสดงความคิดเห็น วิธีใด เหมาะกับ ชุดข้อมูลแบบไหน แต่ละวิธีมีข้อดี/ ข้อเสีย อย่างไร

data2Dset1.csv ส่วนตัวมองว่า ใช้ได้ทั้ง 2 วิธีเลย แต่ถ้าเป็น **K-mean** น่าจะมองง่ายกว่า เพราะถ้าดูจาก การ plot ในข้อ 2 แล้วจะทำให้รู้ว่า ค่า K น่าจะเป็น 3 เลย เพราะ ข้อมูลกระจายเป็นกลุ่มๆเลย แถบไม่มีข้อมูลไหนที่อยู่ติดกัน จนแยกไม่ค่อยได้เลย ส่วน **Hierarchical Clustering** อาจต้องเสียเวลามาหาค่า T ไว้ cutting ก่อนจึงค่อยแยก ข้อมูล แต่มันก็ไม่ต่างกันมากเท่าไรครับ

ชุดข้อมูลที่ 2

data2Dset2.csv

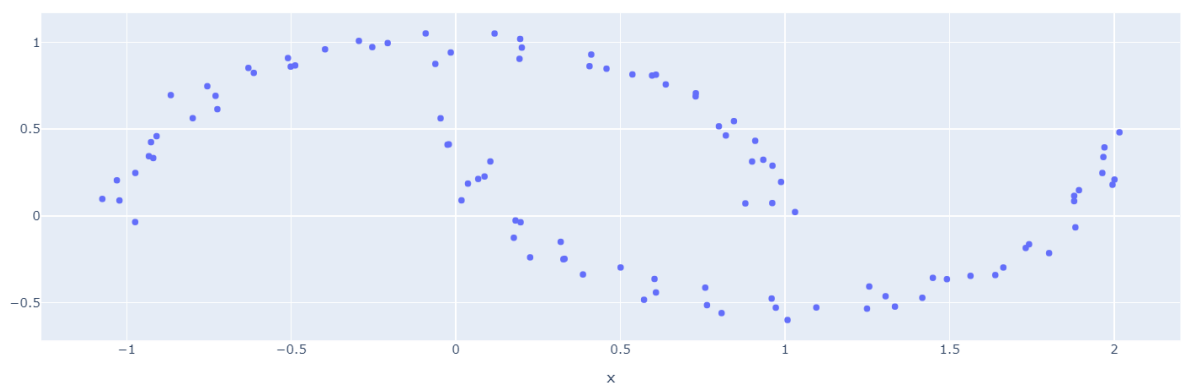
1. เขียนโปรแกรมภาษาไพทอน อ่านข้อมูลจากไฟล์ชุดข้อมูลที่กำหนดให้ ส่วนของโค้ดที่แนะนำให้ใช้เพื่ออ่านข้อมูล

```
data = pd.read_csv("data2Dset2.csv", header=None)
data.columns = ["x", "y"]
data.head()
```

	x	y
0	1.967099	0.339064
1	0.762843	-0.513650
2	-1.029709	0.205156
3	0.637710	0.756872
4	2.000786	0.209418

2. เขียนโปรแกรม plot ชุดข้อมูล

```
import plotly.express as px
fig = px.scatter(data, x="x", y="y")
fig.show()
```



3. เขียนโปรแกรมจัดกลุ่มชุดข้อมูลที่อ่านเข้ามา โดยใช้วิธี k-Means

- a. ให้เลือกใช้ k ที่ต่างกัน 3 ค่า

K = 2 ;


```

model_kmeans = cluster.KMeans(n_clusters=2)
model_kmeans.fit(data)
data['cluster_id'] = model_kmeans.labels_
centroids = model_kmeans.cluster_centers_
print(centroids)

```

```

[[ 1.18780819 -0.07982386]
 [-0.23144767  0.57845957]]

```

K = 3 ;

```

model_kmeans = cluster.KMeans(n_clusters=3)
model_kmeans.fit(data)
data['cluster_id'] = model_kmeans.labels_
centroids = model_kmeans.cluster_centers_
print(centroids)

```

```

[[ 7.67568428e-01 -5.51056610e-02 -1.11022302e-16]
 [-2.31447668e-01  5.78459572e-01  1.00000000e+00]
 [ 1.69940443e+00 -1.09915574e-01 -1.11022302e-16]]

```

K = 4 ;

```

model_kmeans = cluster.KMeans(n_clusters=4)
model_kmeans.fit(data)
data['cluster_id'] = model_kmeans.labels_
centroids = model_kmeans.cluster_centers_
print(centroids)

```

```

[[ 1.52935308e-01  6.06910519e-01  1.00000000e+00]
 [ 1.69940443e+00 -1.09915574e-01  2.00000000e+00]
 [ 7.67568428e-01 -5.51056610e-02  4.44089210e-16]
 [-7.88802984e-01  5.37205698e-01  1.00000000e+00]]

```

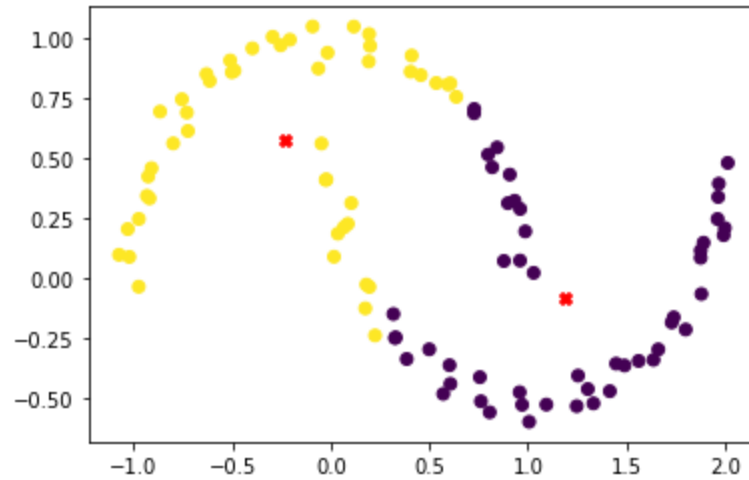
b. plot ผลการจัดกลุ่มที่ได้ สำหรับแต่ละค่า k

```

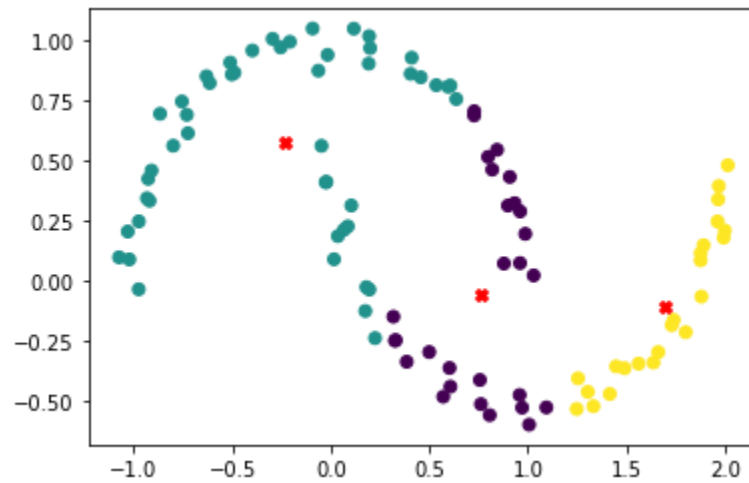
plt.scatter(data['x'],data['y'],c=data['cluster_id'])
plt.scatter(centroids[:,0],centroids[:,1],marker='X',c='r')
plt.show()

```

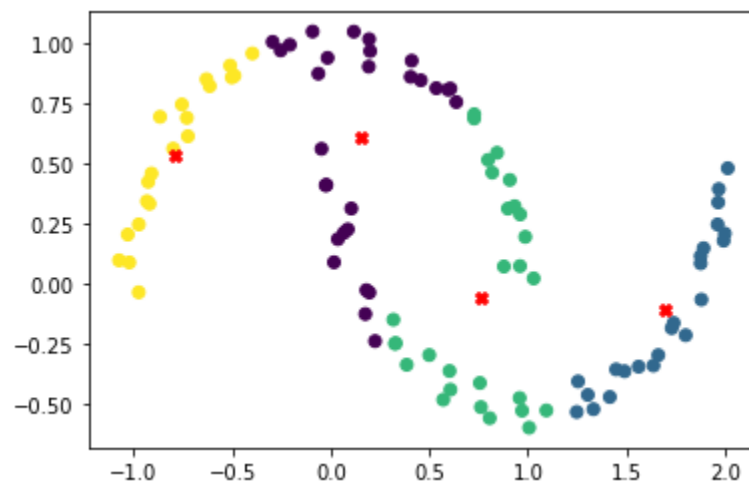
k = 2 ;



$k = 3$;



$k = 4$;



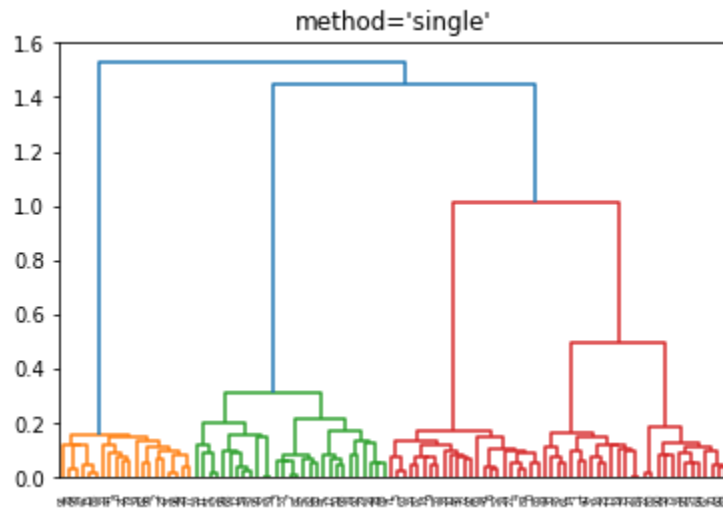
c. วิเคราะห์ผลการทดลองที่ได้ (ควรมี $k = 1$ ค่าที่นักศึกษาคิดว่าดีที่สุดสำหรับชุดข้อมูลดังกล่าว)

เลือกค่า $K = 4$ เพราะเนื่องจากข้อมูลมีความใกล้เคียงหรือเกาะกลุ่มกันสูง จึงต้องทำการ ให้ ค่า K ที่เยอะ
เพื่อจะได้ แบ่ง clustering เป็นส่วนๆได้ เพื่อนำมาวิเคราะห์ได้ง่าย

4. เขียนโปรแกรมจัดกลุ่มชุดข้อมูลที่อ่านเข้ามา โดยใช้วิธี Hierarchical Clustering
 - a. ให้เลือกใช้method ที่ต่างกัน 3 แบบ แสดง dendrogram ที่ได้แต่ละแบบ

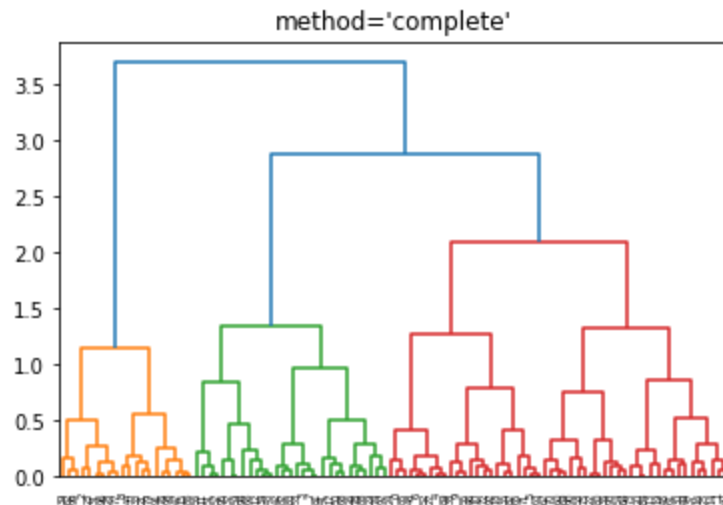
Single ;

```
linkage_data = linkage(data,method='single',metric='euclidean')
dendrogram(linkage_data)
plt.title("method='single'")
plt.show()
```



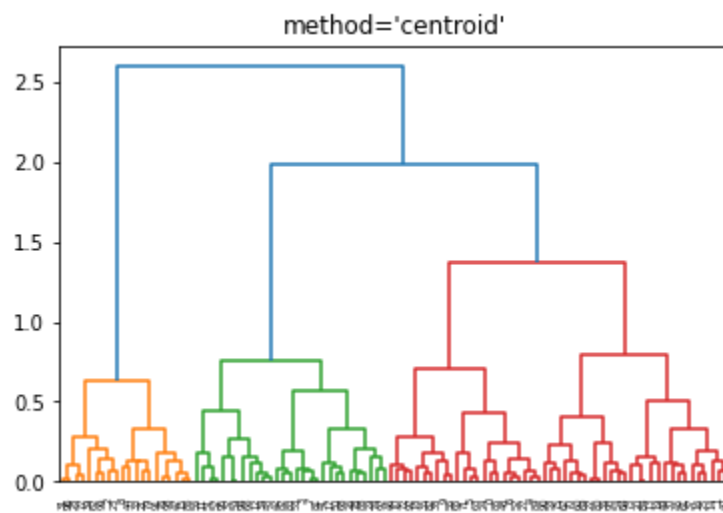
Complete ;

```
linkage_data = linkage(data,method='complete',metric='euclidean')
dendrogram(linkage_data)
plt.title("method='complete'")
plt.show()
```



Centroid ;

```
linkage_data = linkage(data,method='centroid',metric='euclidean')
dendrogram(linkage_data)
plt.title("method='centroid'")
plt.show()
```



- b. เลือก cut-off โดยกำหนด criterion='distance' และให้นักศึกษาเลือกระนาบค่า t ที่คิดว่า เหมาะสม สำหรับแต่ละ dendrogram ที่ได้ในข้อ a)

Single ; $t = 0.6$

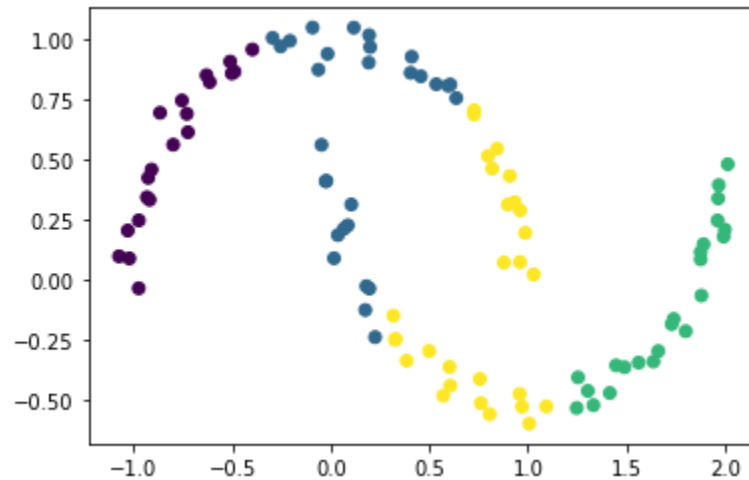
Complete ; $t = 1.5$

Centroid ; $t = 0.8$

c. Plot ผลการการจัดกลุ่ม ที่ได้แต่ละแบบในข้อ b)

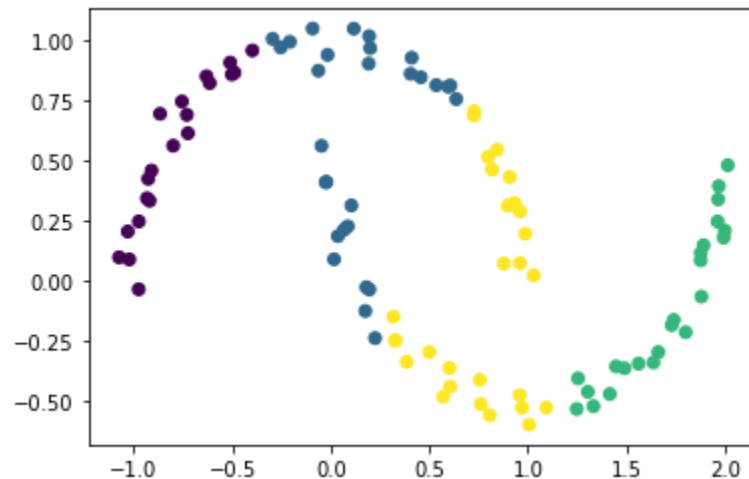
Single ;

```
cluster_id = fcluster(linkage_data,t=0.6,criterion='distance')
plt.scatter(data["x"],data["y"],c=cluster_id)
plt.show()
```



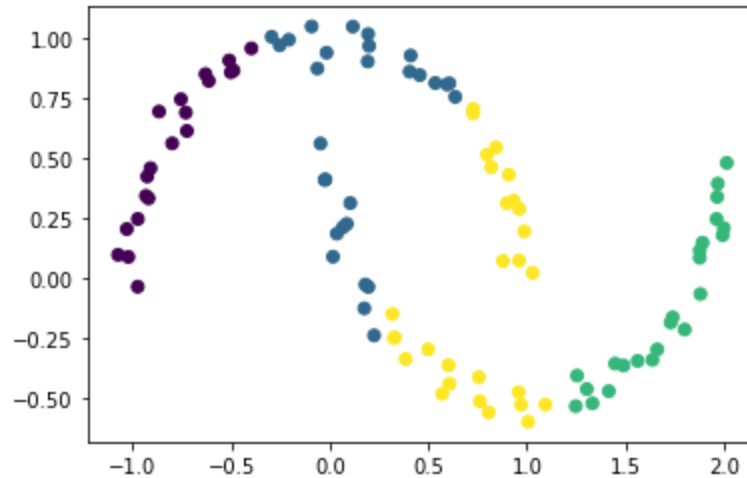
Complete ;

```
cluster_id = fcluster(linkage_data,t=1.5,criterion='distance')
plt.scatter(data["x"],data["y"],c=cluster_id)
plt.show()
```



Centroid ;

```
cluster_id = fcluster(linkage_data,t=0.8,criterion='distance')
plt.scatter(data["x"],data["y"],c=cluster_id)
plt.show()
```



5. เขียนบรรยายสรุปผลการทดลอง แสดงความคิดเห็น วิธีใด เหมาะกับ ชุดข้อมูลแบบไหน แต่ละวิธีมีข้อดี/ ข้อเสีย อย่างไร

data2Dset2.csv คิดว่าเหมาะกับวิธี **Hierarchical Clustering** เพราะเนื่องจาก ข้อมูลนั้นมีลักษณะใกล้เคียงกันสูง ทำให้ไม่สามารถกำหนดค่า K ได้ เพราะมันใกล้เคียงกันมาก ถ้า ให้ค่า $K=2$ ข้อมูลจะใหญ่ไปหรือถ้าแค่ 3 ข้อมูลก็ไม่ค่อยแบ่งเป็นกลุ่มๆสักเท่าไร จึงคิดว่าวิธีที่ดีที่สุดคือ วิธี **Hierarchical Clustering** เพราะ การหาค่า T จากกราฟนั้นค่อนข้างที่จะมองหรือกะง่ายและทำให้ข้อมูลแบ่งเป็นส่วนๆได้ด้วย

ชุดข้อมูลที่ 3

data2Dset3.csv

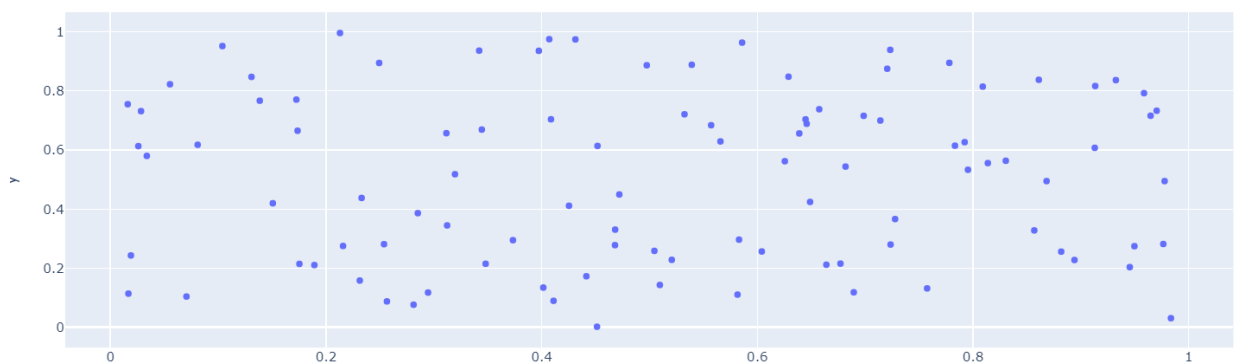
1. เขียนโปรแกรมภาษาไพทอน อ่านข้อมูลจากไฟล์ชุดข้อมูลที่กำหนดให้ ส่วนของโค้ดที่แนะนำให้ใช้เพื่ออ่านข้อมูล

```
data = pd.read_csv("data2Dset3.csv", header=None)
data.columns = ["x", "y"]
data.head()
```

	x	y
0	0.028405	0.731484
1	0.471940	0.449512
2	0.970259	0.732859
3	0.070531	0.104092
4	0.539139	0.888368

2. เขียนโปรแกรม plot จุดข้อมูล

```
import plotly.express as px
fig = px.scatter(data, x="x", y="y")
fig.show()
```



3. เขียนโปรแกรมจัดกลุ่มชุดข้อมูลที่อ่านเข้ามา โดยใช้วิธี k-Means
 - a. ให้เลือกใช้ k ที่ต่างกัน 3 ค่า
 $K = 2$;

```

model_kmeans = cluster.KMeans(n_clusters=2)
model_kmeans.fit(data)
data['cluster_id'] = model_kmeans.labels_
centroids = model_kmeans.cluster_centers_
print(centroids)

```

```

[[0.53548368 0.73937577]
 [0.48716765 0.23323258]]

```

K = 3 ;

```

model_kmeans = cluster.KMeans(n_clusters=3)
model_kmeans.fit(data)
data['cluster_id'] = model_kmeans.labels_
centroids = model_kmeans.cluster_centers_
print(centroids)

```

```

[[7.53720560e-01 7.12041439e-01 2.77555756e-16]
 [4.87167648e-01 2.33232575e-01 1.00000000e+00]
 [2.31849764e-01 7.77406137e-01 1.11022302e-16]]

```

K = 4 ;

```

model_kmeans = cluster.KMeans(n_clusters=4)
model_kmeans.fit(data)
data['cluster_id'] = model_kmeans.labels_
centroids = model_kmeans.cluster_centers_
print(centroids)

```

```

[[0.77315914 0.23617367 1.         ]
 [0.75372056 0.71204144 0.         ]
 [0.23184976 0.77740614 2.         ]
 [0.31352996 0.23144691 1.         ]]

```

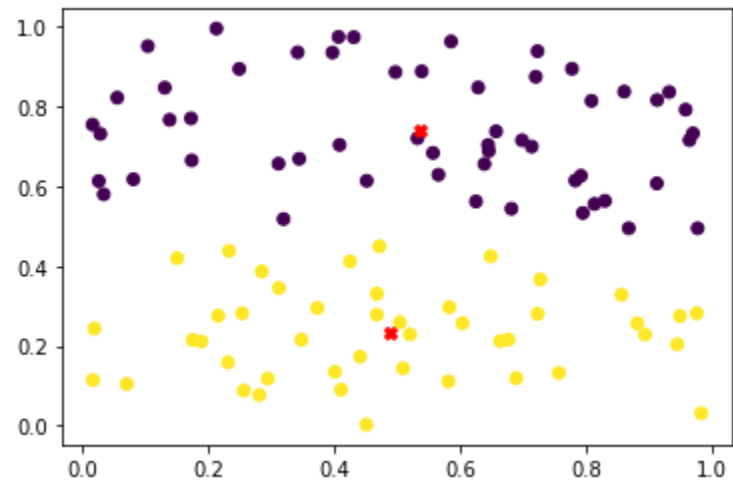
```

plt.scatter(data['x'],data['y'],c=data['cluster_id'])
plt.scatter(centroids[:,0],centroids[:,1],marker='X',c='r')
plt.show()

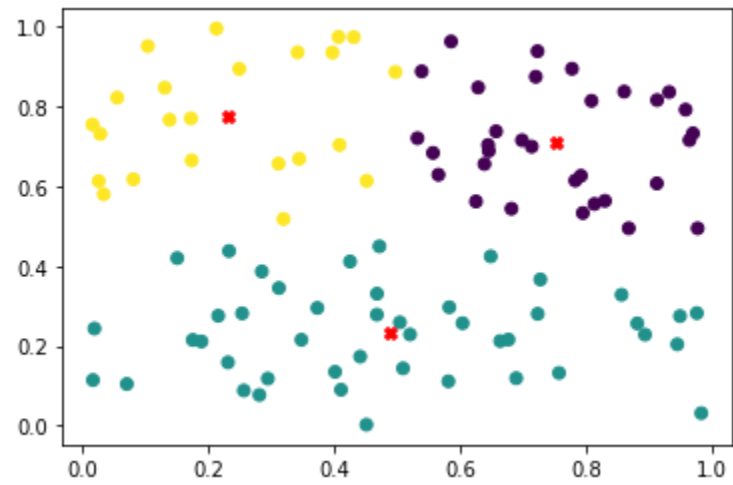
```

b. plot ผลการจัดกลุ่มที่ได้ สำหรับแต่ละค่า k

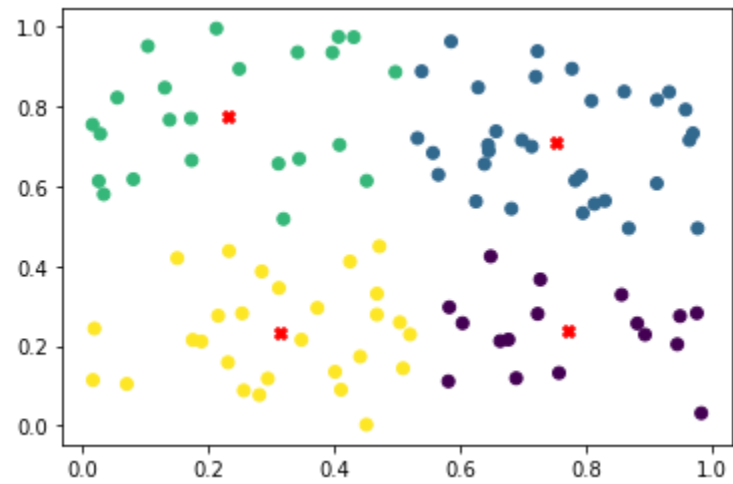
k = 2 ;



k = 3 ;



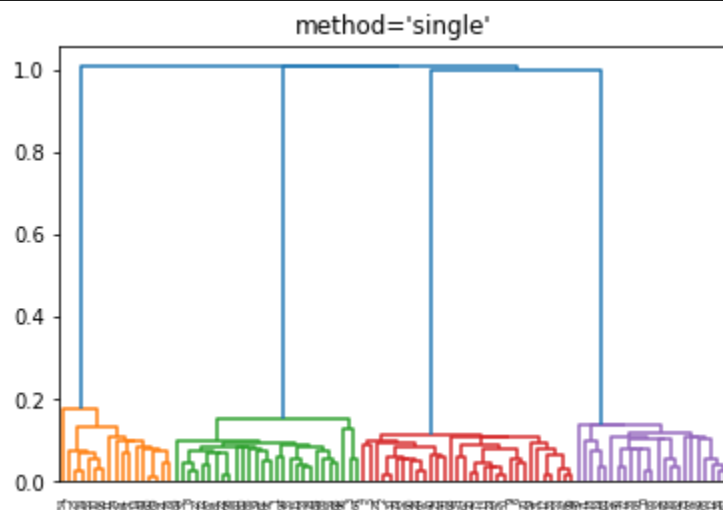
k = 4 ;



- a. วิเคราะห์ผลการทดลองที่ได้ (ควรมี **k 1** ค่าที่นักศึกษาคิดว่าดีที่สุดสำหรับชุดข้อมูลดังกล่าว)
เลือกค่า **K = 4** เพราะข้อมูลมีการกระจายตัวที่เยอะมาก ถ้าเป็น **K = 2** หรือ **K = 3** ข้อมูลบางข้อมูลจะใหญ่เกินไปและส่วนแบ่งได้อีก
4. เขียนโปรแกรมจัดกลุ่มชุดข้อมูลที่อ่านเข้ามา โดยใช้วิธี **Hierarchical Clustering**
 - a. ให้เลือกใช้ **method** ที่ต่างกัน 3 แบบ แสดง **dendrogram** ที่ได้แต่ละแบบ

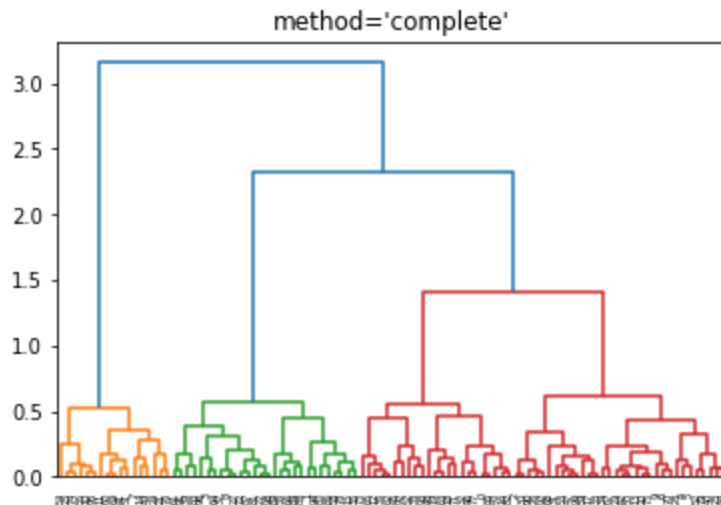
Single ;

```
linkage_data = linkage(data,method='single',metric='euclidean')
dendrogram(linkage_data)
plt.title("method='single'")
plt.show()
```



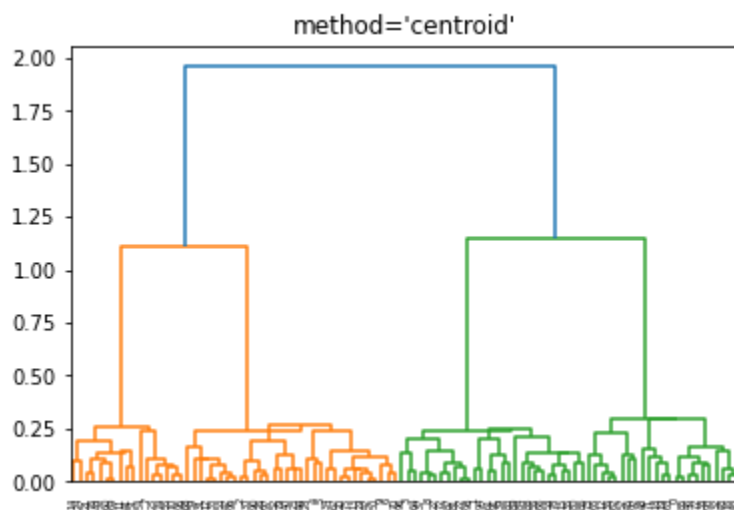
Complete ;

```
linkage_data = linkage(data,method='complete',metric='euclidean')
dendrogram(linkage_data)
plt.title("method='complete'")
plt.show()
```



Centroid ;

```
linkage_data = linkage(data,method='centroid',metric='euclidean')
dendrogram(linkage_data)
plt.title("method='centroid'")
plt.show()
```



- b. เลือก cut-off โดยกำหนด criterion='distance' และให้นักศึกษาเลือกระนาบค่า t ที่คิดว่า เหมาะสม สำหรับแต่ละ dendrogram ที่ได้ในข้อ a)

Single ; $t = 0.6$

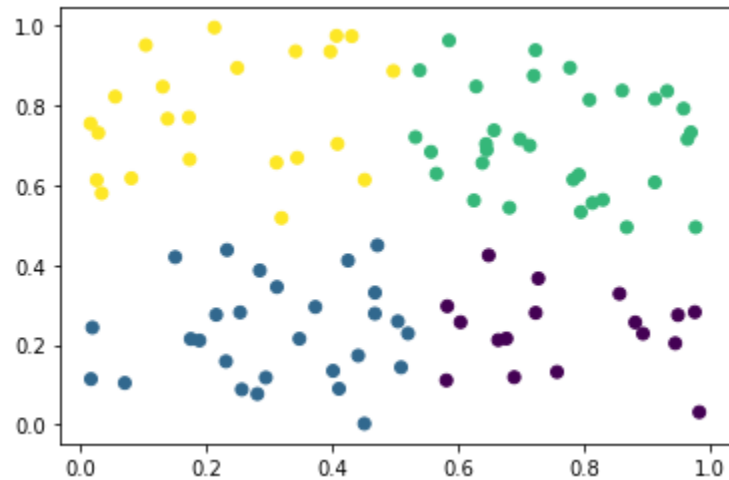
Complete ; $t = 1.0$

Centroid ; $t = 0.8$

c. Plot ผลการการจัดกลุ่ม ที่ได้แต่ละแบบในข้อ b)

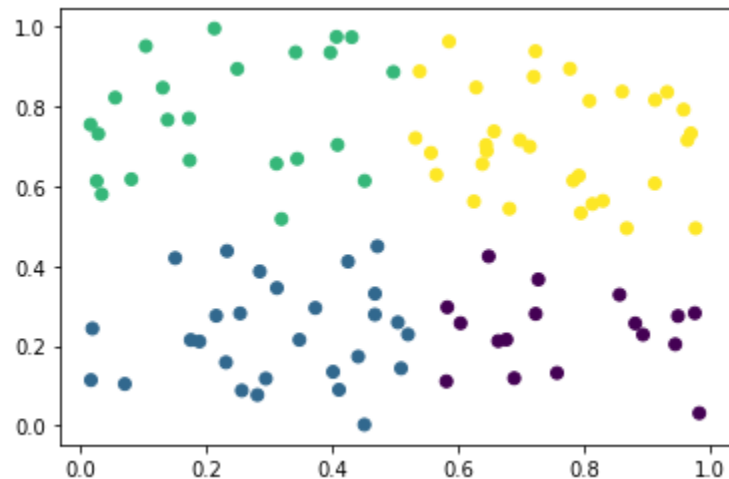
Single ;

```
cluster_id = fcluster(linkage_data,t=2.5,criterion='distance')
plt.scatter(data["x"],data["y"],c=cluster_id)
plt.show()
```



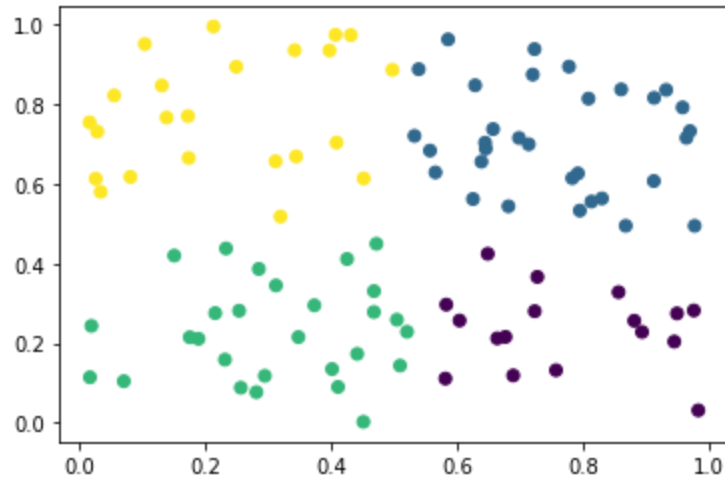
Complete ;

```
cluster_id = fcluster(linkage_data,t=1,criterion='distance')
plt.scatter(data["x"],data["y"],c=cluster_id)
plt.show()
```



Centroid ;

```
cluster_id = fcluster(linkage_data,t=0.8,criterion='distance')
plt.scatter(data["x"],data["y"],c=cluster_id)
plt.show()
```



5. เขียนบรรยายสรุปผลการทดลอง แสดงความคิดเห็น วิธีใด เหมาะกับ ชุดข้อมูลแบบไหน แต่ละวิธีมีข้อดี/ ข้อเสียอย่างไร

data2Dset3.csv เหมาะกับการใช้ Hierarchical Clustering เพราะเราไม่สามารถที่จะแบ่ง กลุ่มข้อมูลออกเป็นแต่ละ Cluster ได้ด้วยตาเปล่า จึงทำให้การใช้ K-mean ค่อนข้างยากและแบ่งไม่ค่อยละเอียดเท่า Hierarchical Clustering กับข้อมูลที่มีการกระจายตัวสูง

ชุดข้อมูลที่ 4

data2Dset4.csv

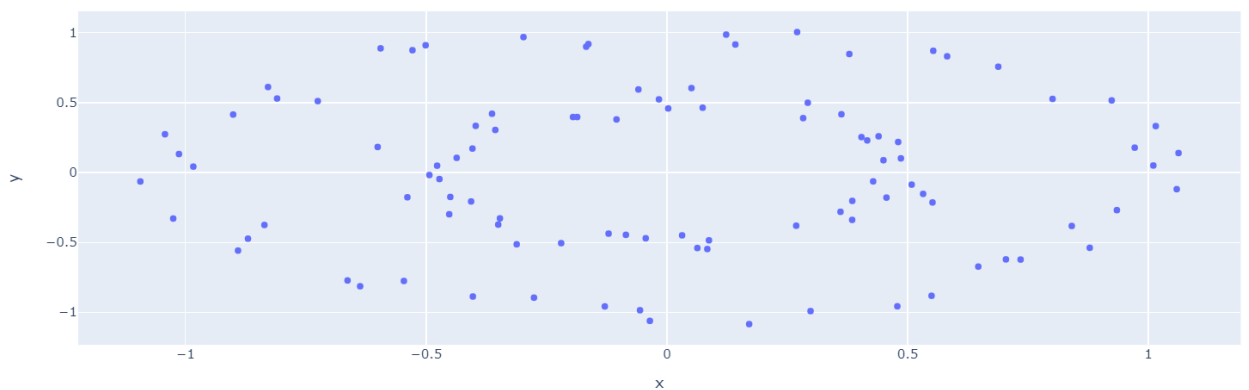
1. เขียนโปรแกรมภาษาไพทอน อ่านข้อมูลจากไฟล์ชุดข้อมูลที่กำหนดให้ ส่วนของโค้ดที่แนะนำให้ใช้เพื่ออ่านข้อมูล

```
data = pd.read_csv("data2Dset4.csv", header=None)
data.columns = ["x", "y"]
data.head()
```

	x	y
0	-0.016537	0.523408
1	0.971732	0.177843
2	-0.403983	0.171210
3	-0.406715	-0.207174
4	-0.055831	-0.985840

2. เขียนโปรแกรม plot จุดข้อมูล

```
import plotly.express as px
fig = px.scatter(data, x="x", y="y")
fig.show()
```



3. เขียนโปรแกรมจัดกลุ่มชุดข้อมูลที่อ่านเข้ามา โดยใช้วิธี k-Means

- a. ให้เลือกใช้ k ที่ต่างกัน 3 ค่า

K = 2 ;

```

model_kmeans = cluster.KMeans(n_clusters=2)
model_kmeans.fit(data)
data['cluster_id'] = model_kmeans.labels_
centroids = model_kmeans.cluster_centers_
print(centroids)

```

```

[[ 0.35567251  0.2954592 ]
 [-0.38961084 -0.3252557 ]]

```

K = 3 ;

```

model_kmeans = cluster.KMeans(n_clusters=3)
model_kmeans.fit(data)
data['cluster_id'] = model_kmeans.labels_
centroids = model_kmeans.cluster_centers_
print(centroids)

```

```

[[ 2.50569961e-02  6.88911462e-01 -3.33066907e-16]
 [-3.89610837e-01 -3.25255700e-01  1.00000000e+00]
 [ 6.61797986e-01 -6.88484582e-02 -2.77555756e-16]]

```

K = 4 ;

```

model_kmeans = cluster.KMeans(n_clusters=4)
model_kmeans.fit(data)
data['cluster_id'] = model_kmeans.labels_
centroids = model_kmeans.cluster_centers_
print(centroids)

```

```

[[ -9.13911135e-02 -6.89520810e-01  1.00000000e+00]
 [ 2.50569961e-02  6.88911462e-01  2.22044605e-16]
 [ 6.61797986e-01 -6.88484582e-02  2.00000000e+00]
 [-6.87830561e-01  3.90094112e-02  1.00000000e+00]]

```

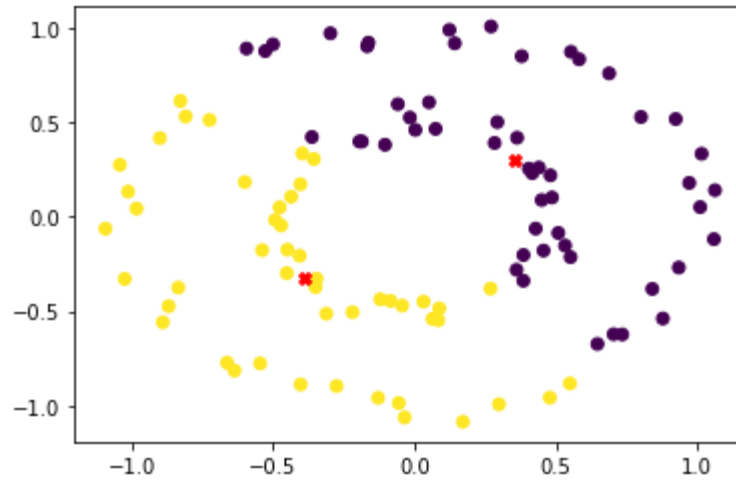
b. plot ผลการจัดกลุ่มที่ได้ สำหรับแต่ละค่า k

```

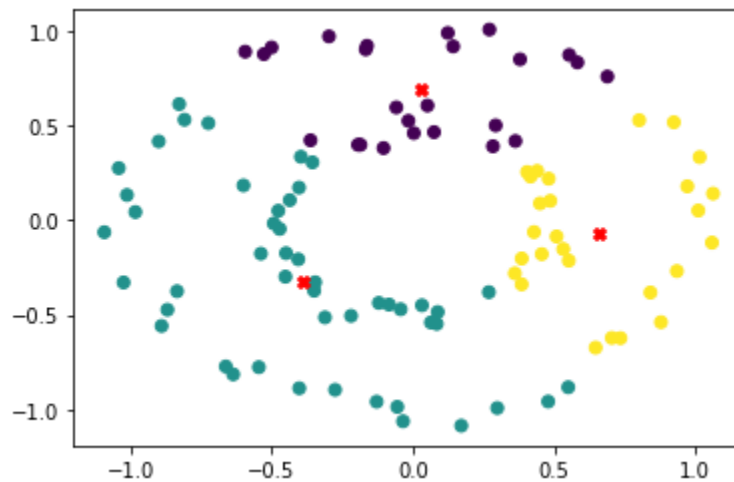
plt.scatter(data['x'],data['y'],c=data['cluster_id'])
plt.scatter(centroids[:,0],centroids[:,1],marker='X',c='r')
plt.show()

```

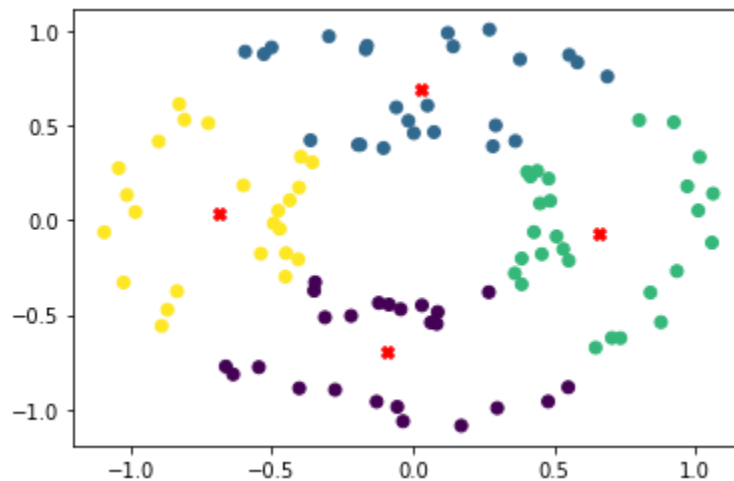
k = 2 ;



$k = 3$;



$k = 4$;



c. วิเคราะห์ผลการทดลองที่ได้ (ควรมี $k = 1$ ค่าที่นักศึกษาคิดว่าดีที่สุดสำหรับชุดข้อมูลดังกล่าว)

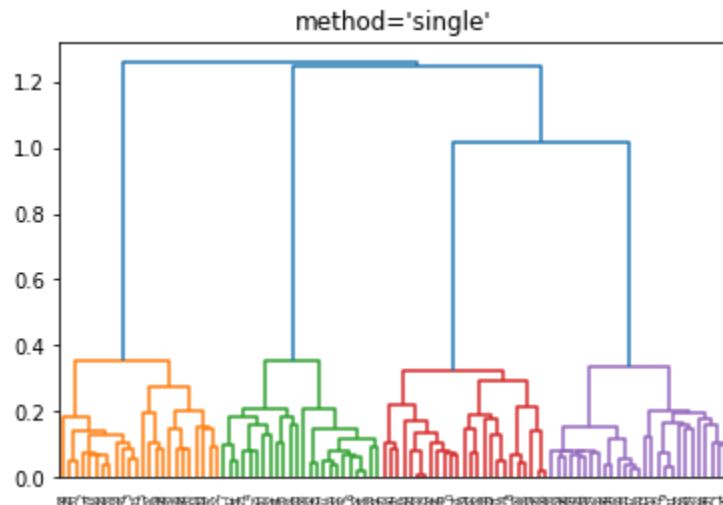
K = 4 เพราะข้อมูลมีลักษณะใกล้เคียงกันและยังมีการกระจายรอบๆ ถ้าเป็น 3 ก็มีสีฟ้าเขียว ที่มีลักษณะใหญ่และไม่ค่อยละเอียด หรือถ้าเป็น 2 ข้อมูลจะติดกันมาก ส่วนถ้าเป็น 5 ข้อมูลจะมีกลุ่ม 1 ที่ใกล้เคียงติดกันกับอีกตัวหนึ่งเลย

4. เขียนโปรแกรมจัดกลุ่มชุดข้อมูลที่อ่านเข้ามา โดยใช้วิธี Hierarchical Clustering

- a. ให้เลือกใช้ method ที่ต่างกัน 3 แบบ แสดง dendrogram ที่ได้แต่ละแบบ

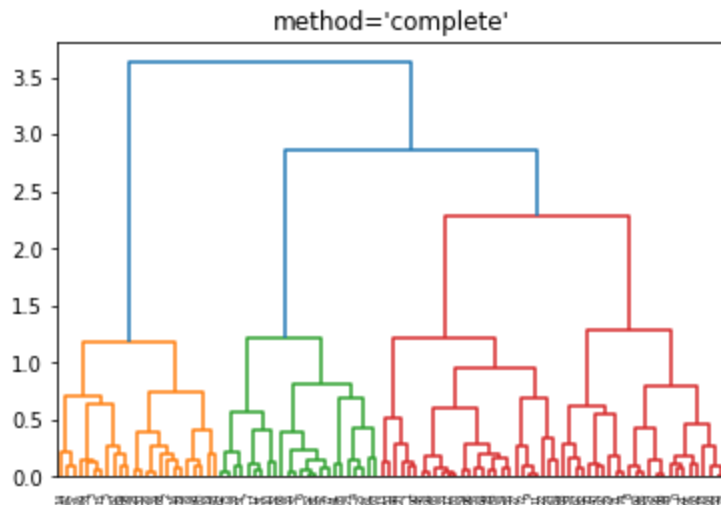
Single ;

```
linkage_data = linkage(data,method='single',metric='euclidean')
dendrogram(linkage_data)
plt.title("method='single'")
plt.show()
```



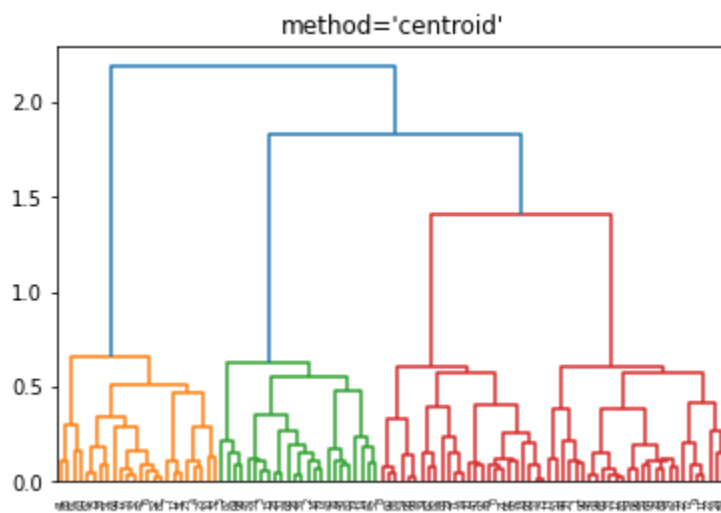
Complete ;

```
linkage_data = linkage(data,method='complete',metric='euclidean')
dendrogram(linkage_data)
plt.title("method='complete'")
plt.show()
```



Centroid ;

```
linkage_data = linkage(data,method='centroid',metric='euclidean')
dendrogram(linkage_data)
plt.title("method='centroid'")
plt.show()
```



- b. เลือก cut-off โดยกำหนด criterion='distance' และให้นักศึกษาเลือกระนาบค่า t ที่คิดว่า เหมาะสม สำหรับแต่ละ dendrogram ที่ได้ในข้อ a)

Single ; $t = 0.4$

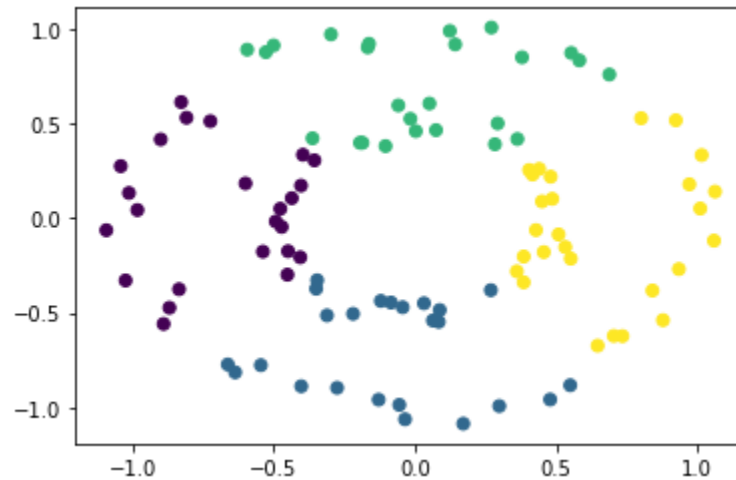
Complete ; $t = 1.5$

Centroid ; $t = 0.8$

c. Plot ผลการการจัดกลุ่ม ที่ได้แต่ละแบบในข้อ b)

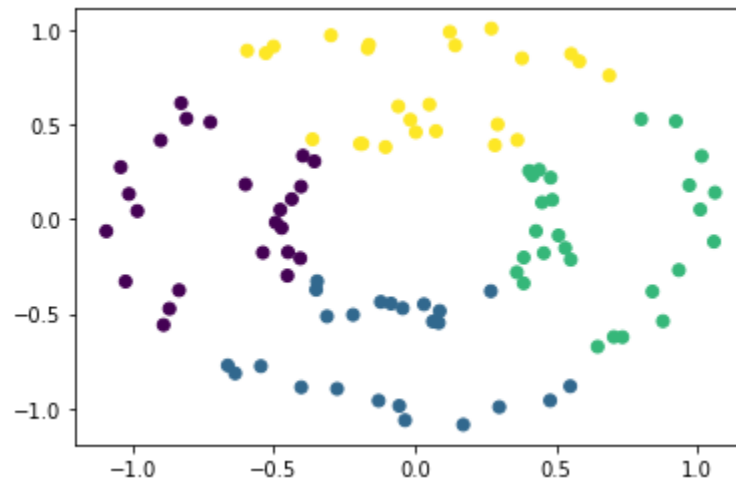
Single ;

```
cluster_id = fcluster(linkage_data,t=0.4,criterion='distance')
plt.scatter(data["x"],data["y"],c=cluster_id)
plt.show()
```



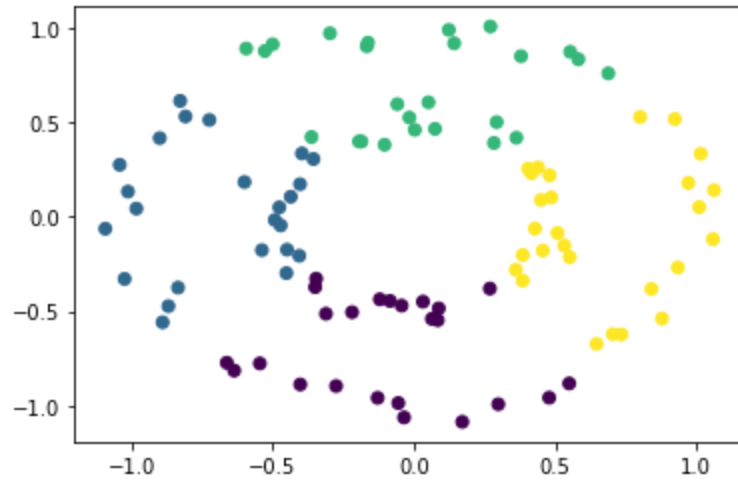
Complete ;

```
cluster_id = fcluster(linkage_data,t=1.5,criterion='distance')
plt.scatter(data["x"],data["y"],c=cluster_id)
plt.show()
```



Centroid ;

```
cluster_id = fcluster(linkage_data,t=0.8,criterion='distance')
plt.scatter(data["x"],data["y"],c=cluster_id)
plt.show()
```



5. เขียนบรรยายสรุปผลการทดลอง แสดงความคิดเห็น วิธีใด เหมาะกับ ชุดข้อมูลแบบไหน แต่ละวิธีมีข้อดี/ ข้อเสีย อย่างไร

data2Dset4.csv คิดว่าเหมาะกับวิธี **Hierarchical Clustering** เพราะเนื่องจาก ข้อมูลนั้นมีลักษณะใกล้เคียงกันสูง และมีการกระจายเป็นรอบๆ ทำให้ไม่สามารถกำหนดค่า **K** ได้ เพราะมันใกล้เคียงกัน จึงคิดว่าวิธีที่ดีคือ วิธี **Hierarchical Clustering** เพราะสามารถดูแล้วแบ่งตามกราฟได้เลย