# 第九讲：无监督学习-聚类算法

1、使用 k-means 算法，将如下八个点划分到三个聚类中。

八个点坐标： A1=(2,10), A2=(2,5), A3=(8,4), A4=(5,8), A5=(7,5), A6=(6,4), A7=(1,2), A8=(4,9)

各点间的欧式距离：

| | A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 |
|---|---|---|---|---|---|---|---|---|
| A1 | 0 | $\sqrt{25}$ | $\sqrt{36}$ | $\sqrt{13}$ | $\sqrt{50}$ | $\sqrt{52}$ | $\sqrt{65}$ | $\sqrt{5}$ |
| A2 | | 0 | $\sqrt{37}$ | $\sqrt{18}$ | $\sqrt{25}$ | $\sqrt{17}$ | $\sqrt{10}$ | $\sqrt{20}$ |
| A3 | | | 0 | $\sqrt{25}$ | $\sqrt{2}$ | $\sqrt{2}$ | $\sqrt{53}$ | $\sqrt{41}$ |
| A4 | | | | 0 | $\sqrt{13}$ | $\sqrt{17}$ | $\sqrt{52}$ | $\sqrt{2}$ |
| A5 | | | | | 0 | $\sqrt{2}$ | $\sqrt{45}$ | $\sqrt{25}$ |
| A6 | | | | | | 0 | $\sqrt{29}$ | $\sqrt{29}$ |
| A7 | | | | | | | 0 | $\sqrt{58}$ |
| A8 | | | | | | | | 0 |

三个聚簇的初始中心点为 A1，A4 和 A7，运行一遍 k-mean 算法后，给出新的聚类归属关系，以及新的聚类中心点。 可以画出八个点的示意图。

A1 到三个聚簇的初始中心 A1，A4 和 A7 的距离分别为：0，sqrt（13）,sqrt(65)

Min = 0 则归类到 A1

A2 到三个聚簇的初始中心 A1，A4 和 A7 的距离分别为：5，sqrt（37）,sqrt(10)

Min = sqrt（10） 则归类到 A7

A3 到三个聚簇的初始中心 A1，A4 和 A7 的距离分别为：6，5,sqrt(53)

Min = 5 则归类到 A4

A4 到三个聚簇的初始中心 A1，A4 和 A7 的距离分别为：sqrt（13），0,sqrt(52)

Min = 0 则归类到 A4

A5 到三个聚簇的初始中心 A1，A4 和 A7 的距离分别为：sqrt（50）， sqrt（13）,sqrt(45)

Min = sqrt（13） 则归类到 A4

A6 到三个聚簇的初始中心 A1，A4 和 A7 的距离分别为：sqrt（52），sqrt（17）,sqrt(29)

Min = sqrt（17） 则归类到 A4

A7 到三个聚簇的初始中心 A1，A4 和 A7 的距离分别为：sqrt（65），sqrt（52),0

Min = 0 则归类到 A7

A8 到三个聚簇的初始中心 A1，A4 和 A7 的距离分别为：sqrt（5），sqrt（2）,sqrt(58)

Min = sqrt（2） 则归类到 A4

新的聚归类：

A1 有 1 个属于它的点:A1;

A4 有 5 个属于它的点:A8,A6,A5,A4,A3;

A7 有 2 个属于它的点:A7,A2;

八个点坐标： A1=(2,10), A2=(2,5), A3=(8,4), A4=(5,8), A5=(7,5), A6=(6,4), A7=(1,2), A8=(4,9)

新的中心：

A1 只有一个则；

(2,10)

A4 = (sum(y)/n,sum(x)/n );

A4(6,6)

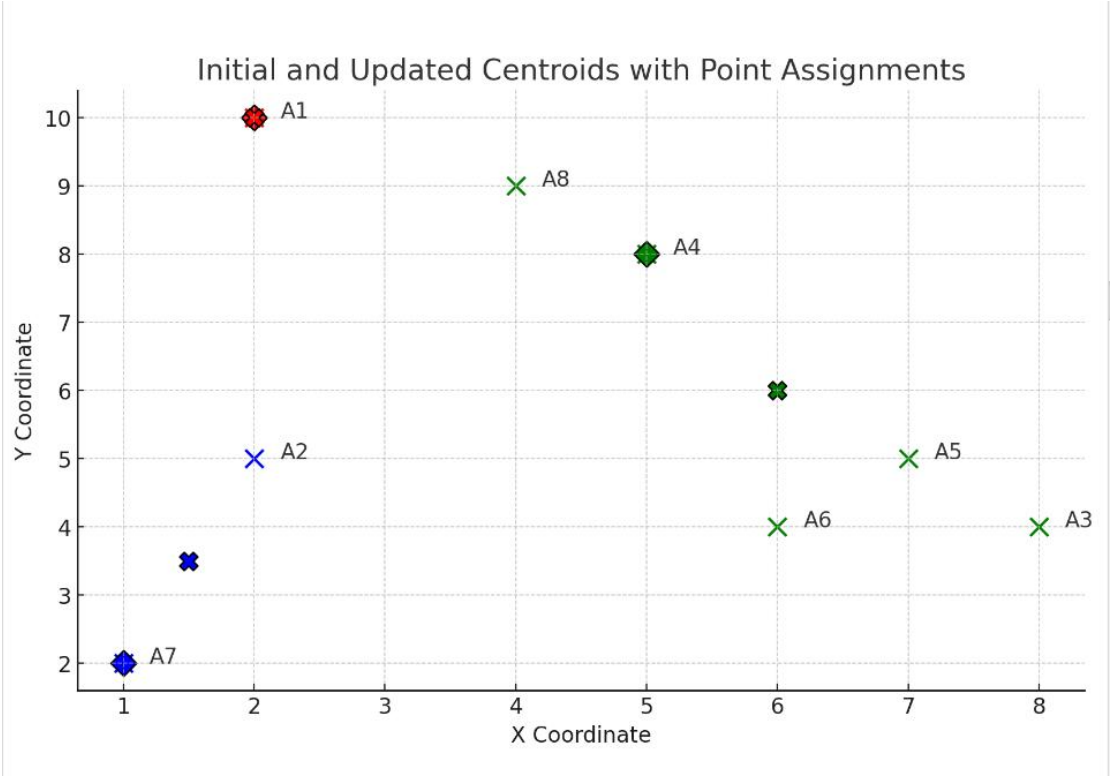
x = (4+6+7+5+8)/5 = x(6)

y = (9+4+5+8+4)/5 = y(6)

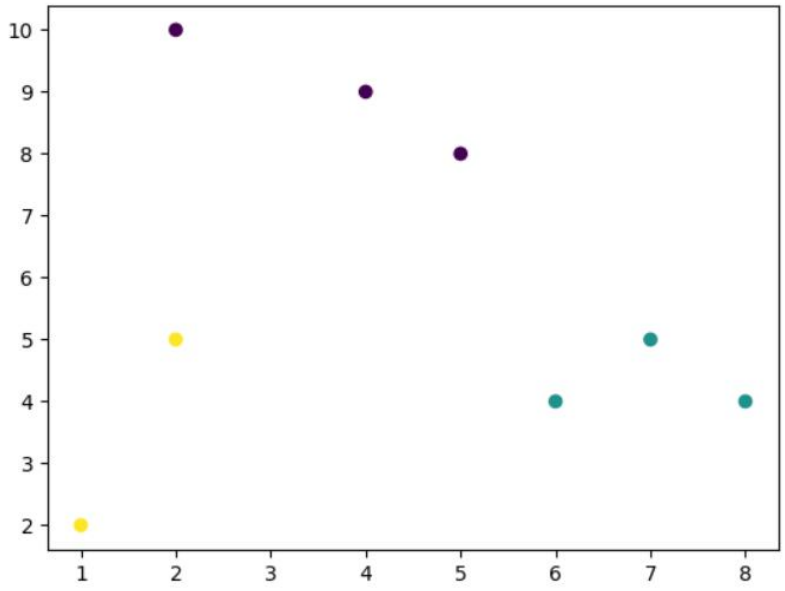
A7

x = 3/2 = x(1.5) y = 3+5/2= y(3.5)

(1.5,3.5)


Initial and Updated Centroids with Point Assignments
A1
A8
A4
A2
A5
A6
A3
A7
Y Coordinate
X Coordinate
10
9
8
7
6
5
4
3
2
1
2
3
4
5
6
7
8

```
#Three lines to make our compiler able to draw:
import sys
import matplotlib
matplotlib.use('Agg')

import matplotlib.pyplot as plt
from sklearn.cluster import KMeans

x = [2,2,8,5,7,6,1,4]
y = [10,5,4,8,5,4,2,9]

data = list(zip(x, y))

kmeans = KMeans(n_clusters=3)
kmeans.fit(data)

plt.scatter(x, y, c=kmeans.labels_)
plt.show()

#Two  lines to make our compiler able to draw:
plt.savefig(sys.stdout.buffer)
sys.stdout.flush()
```


10
9
8
7
6
5
4
3
2
1
2
3
4
5
6
7
8