

Unsupervised learning: basics

CLUSTER ANALYSIS IN PYTHON



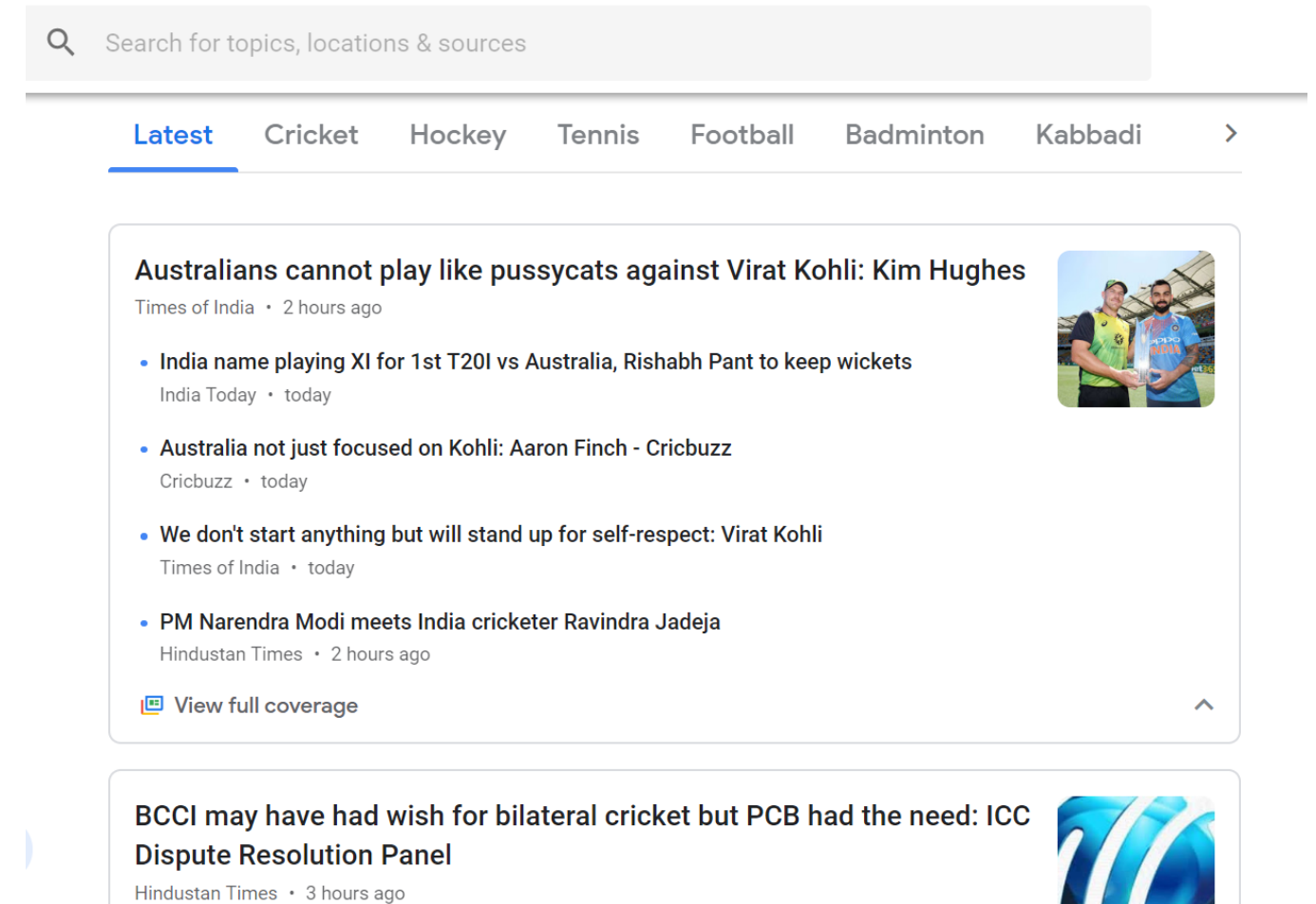
Shaumik Daityari
Business Analyst

Everyday example: Google news

- How does Google News classify articles?
- Unsupervised Learning Algorithm: Clustering
- Match frequent terms in articles to find similarity

google

= >



The screenshot shows the Google News interface for cricket. At the top is a search bar with the placeholder text "Search for topics, locations & sources". Below the search bar is a navigation bar with tabs for "Latest", "Cricket", "Hockey", "Tennis", "Football", "Badminton", and "Kabbadi". The "Latest" tab is selected. The main content area displays two news articles. The first article is titled "Australians cannot play like pussycats against Virat Kohli: Kim Hughes" and is from the "Times of India", published 2 hours ago. It includes a list of related headlines: "India name playing XI for 1st T20I vs Australia, Rishabh Pant to keep wickets" (India Today, today), "Australia not just focused on Kohli: Aaron Finch - Cricbuzz" (Cricbuzz, today), "We don't start anything but will stand up for self-respect: Virat Kohli" (Times of India, today), and "PM Narendra Modi meets India cricketer Ravindra Jadeja" (Hindustan Times, 2 hours ago). There is a "View full coverage" link and an upward arrow icon. The second article is titled "BCCI may have had wish for bilateral cricket but PCB had the need: ICC Dispute Resolution Panel" and is from the "Hindustan Times", published 3 hours ago. It features a blue and white graphic of a cricket ball.

Labeled and unlabeled data

Data with no labels

X,Y 가 list가 가
point
label data feature

- Point 1: (1, 2)
- Point 2: (2, 2)
- Point 3: (3, 1)

Data with labels

Danger Zone, Normal Zone data point
label

- Point 1: (1, 2), Label: Danger Zone
- Point 2: (2, 2), Label: Normal Zone
- Point 3: (3, 1), Label: Normal Zone

What is unsupervised learning?

- A group of machine learning algorithms that find patterns in data
- Data for algorithms has not been labeled, classified or characterized
- The objective of the algorithm is to interpret any structure in the data
- Common unsupervised learning algorithms: clustering, neural networks, anomaly detection

point

clustering,

. clustering

What is clustering?

- The process of grouping items with similar characteristics
- Items in groups similar to each other than in other groups
- Example: distance between points on a 2D plane



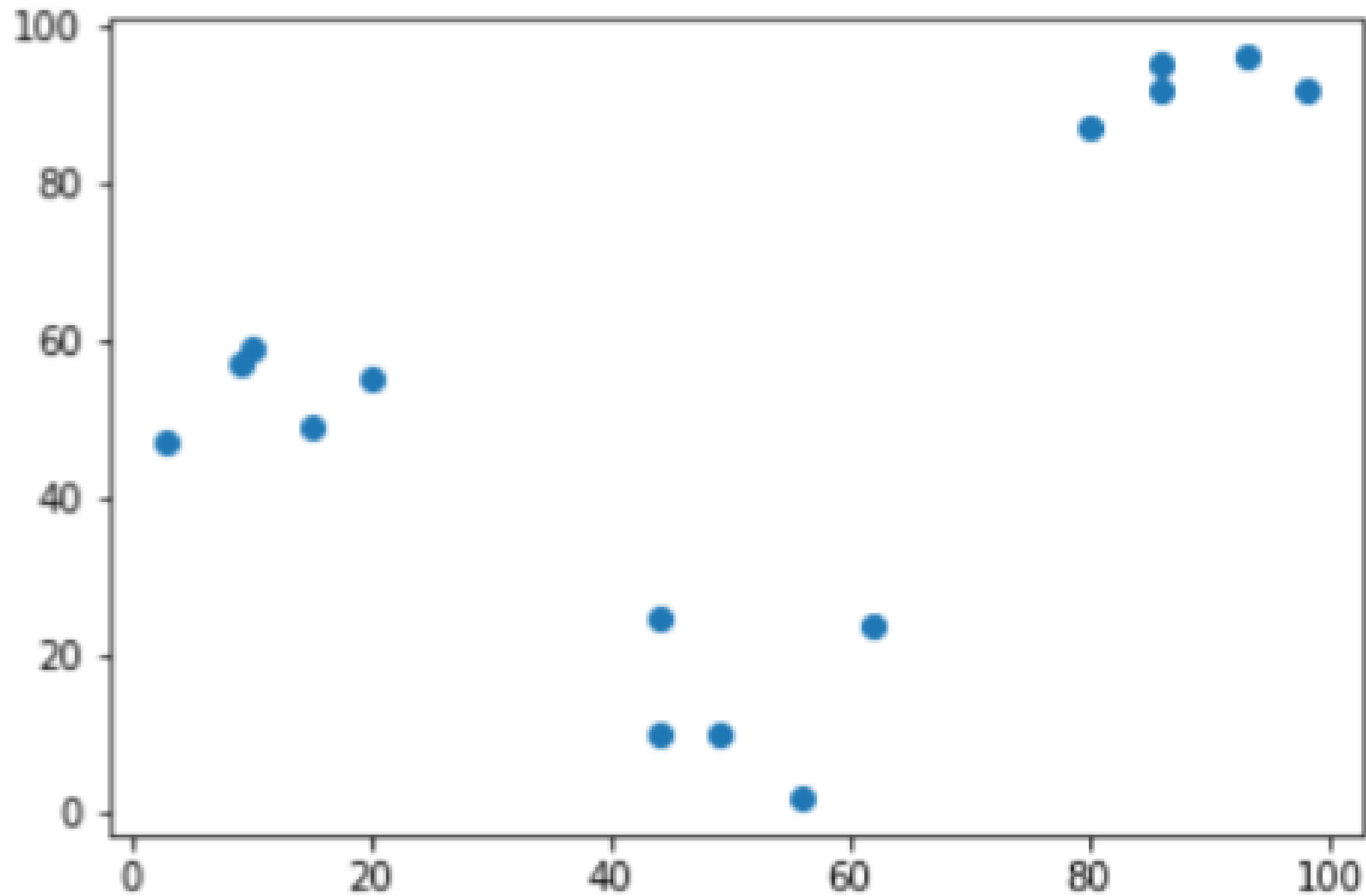
Plotting data for clustering - Pokemon sightings

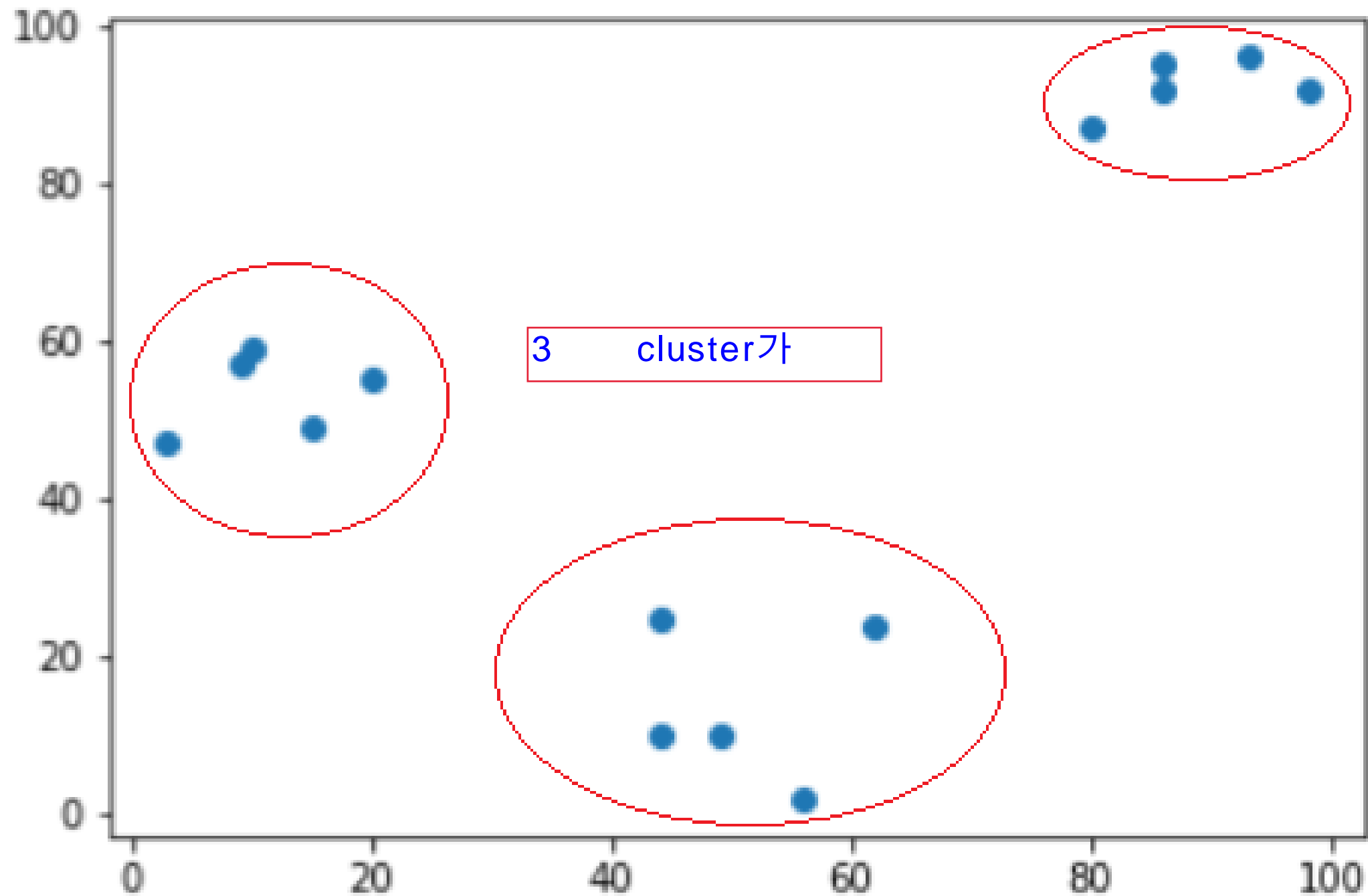
```
from matplotlib import pyplot as plt
```

```
x_coordinates = [80, 93, 86, 98, 86, 9, 15, 3, 10, 20, 44, 56, 49, 62, 44]
```

```
y_coordinates = [87, 96, 95, 92, 92, 57, 49, 47, 59, 55, 25, 2, 10, 24, 10]
```

```
plt.scatter(x_coordinates, y_coordinates)  
plt.show()
```





Up next - some practice

CLUSTER ANALYSIS IN PYTHON

Basics of cluster analysis

CLUSTER ANALYSIS IN PYTHON

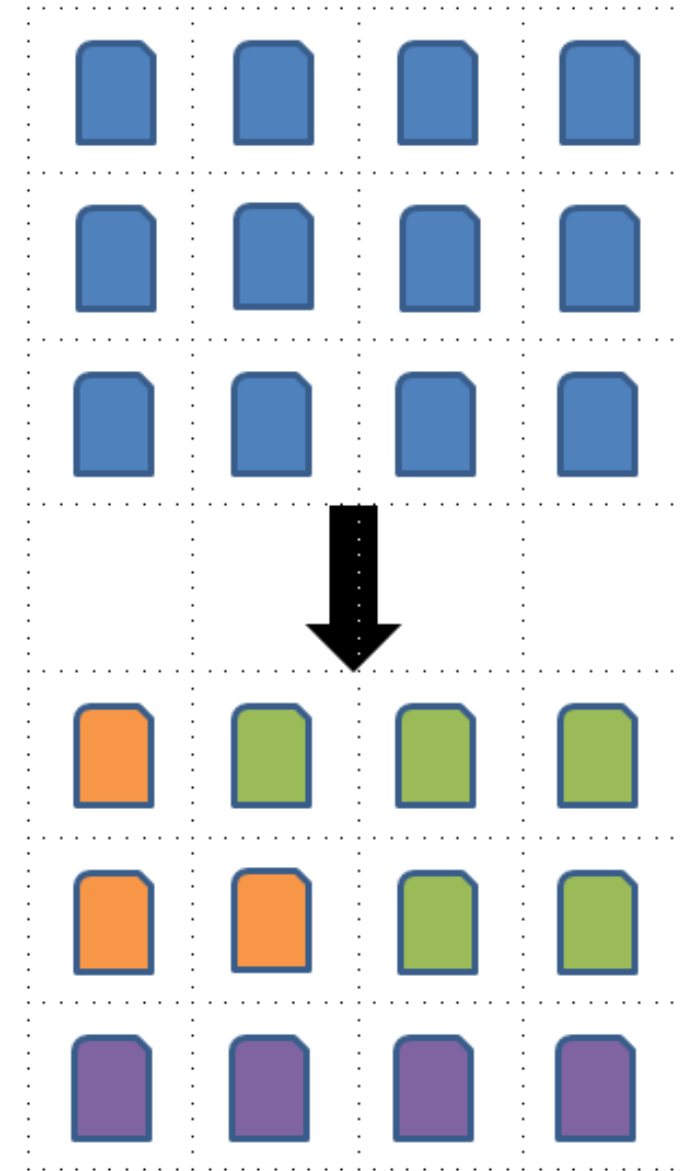


Shaumik Daityari
Business Analyst

What is a cluster?

- A group of items with similar characteristics
- Google News: articles where similar words and word associations appear together
- Customer Segments

```
cluster
-      가      cluster
- google
- clustering
```

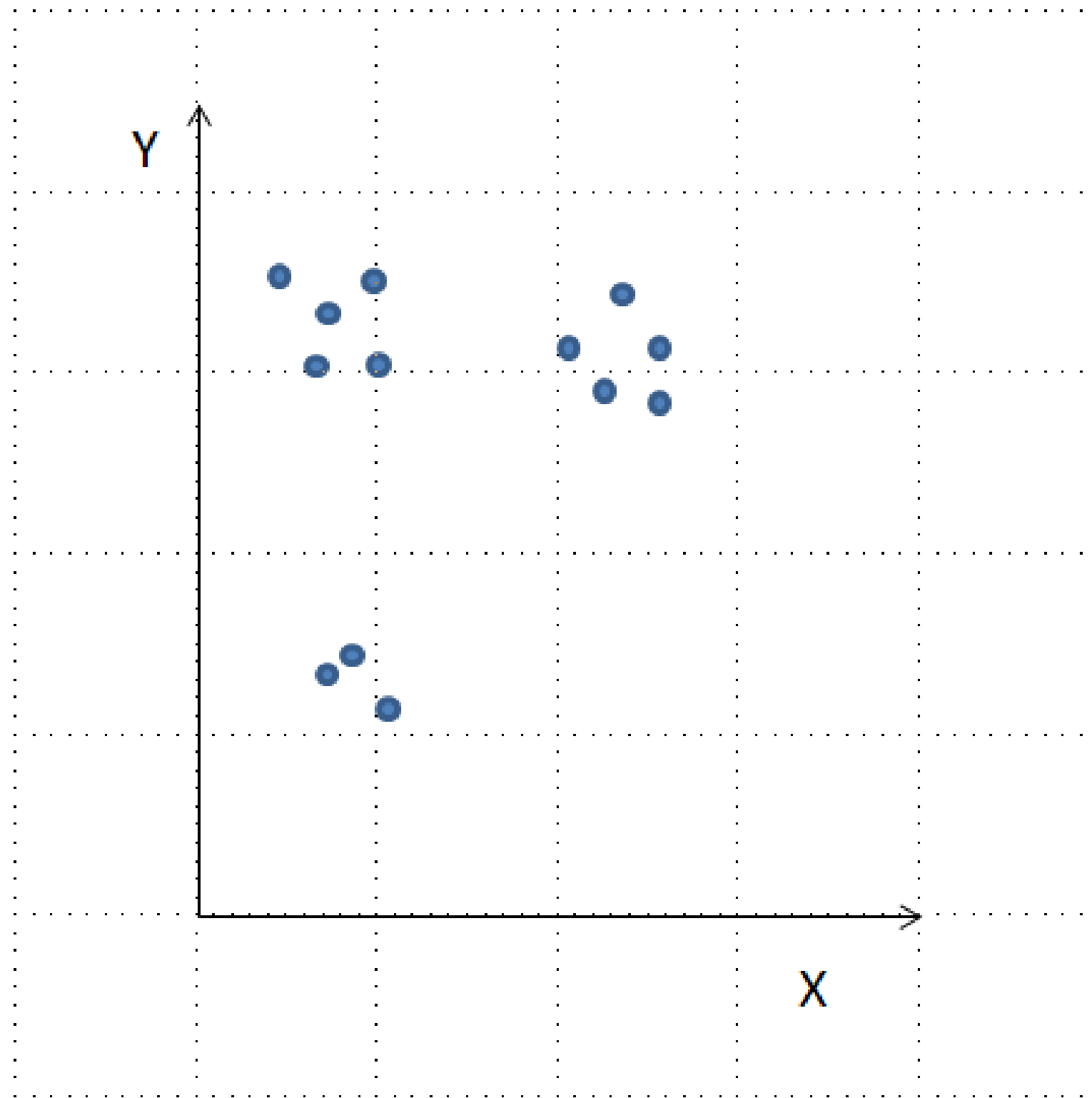


Clustering algorithms

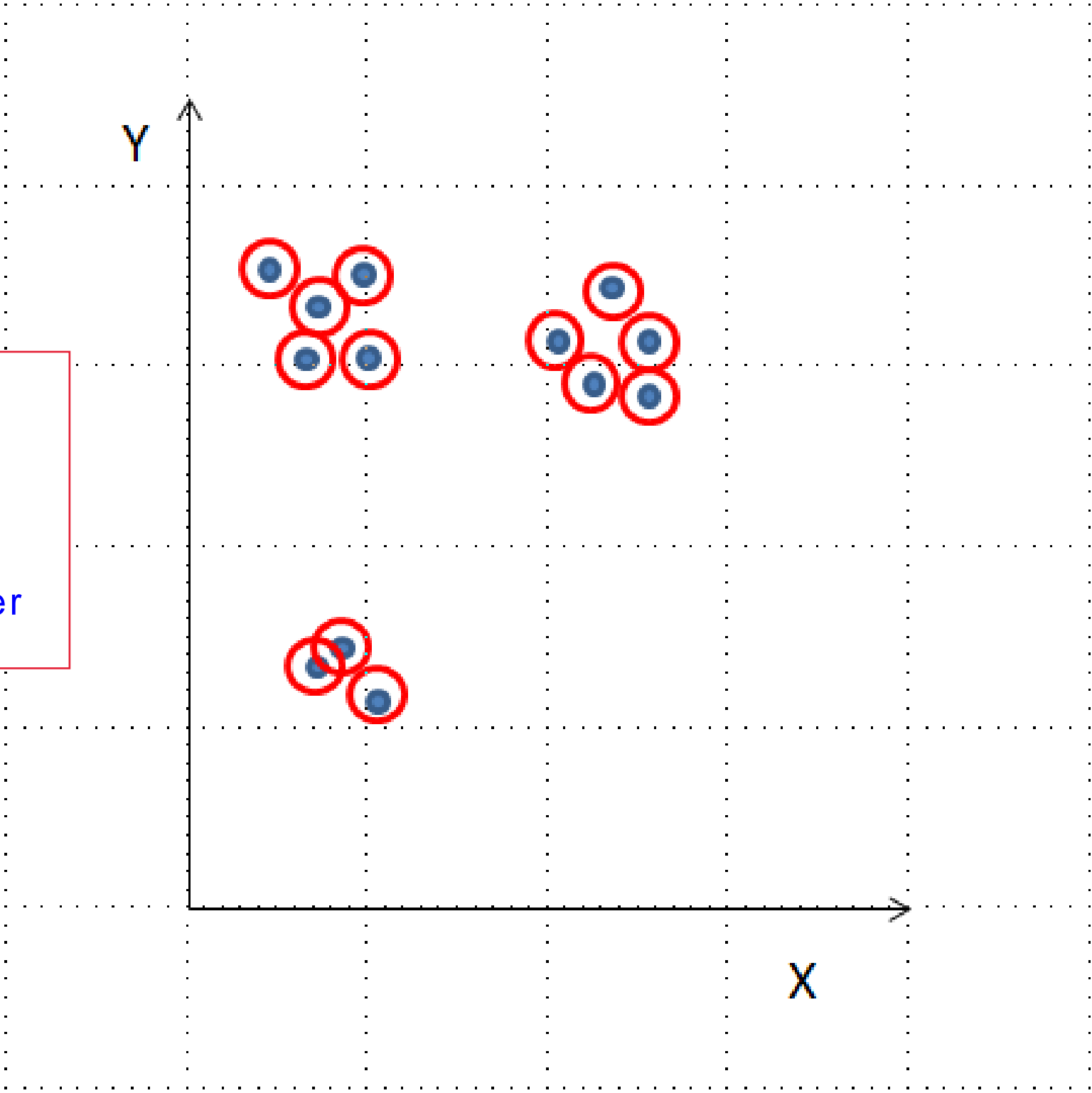
- Hierarchical clustering
- K means clustering
- Other clustering algorithms: DBSCAN, Gaussian Methods

DBSCAN
가
가
k - , .

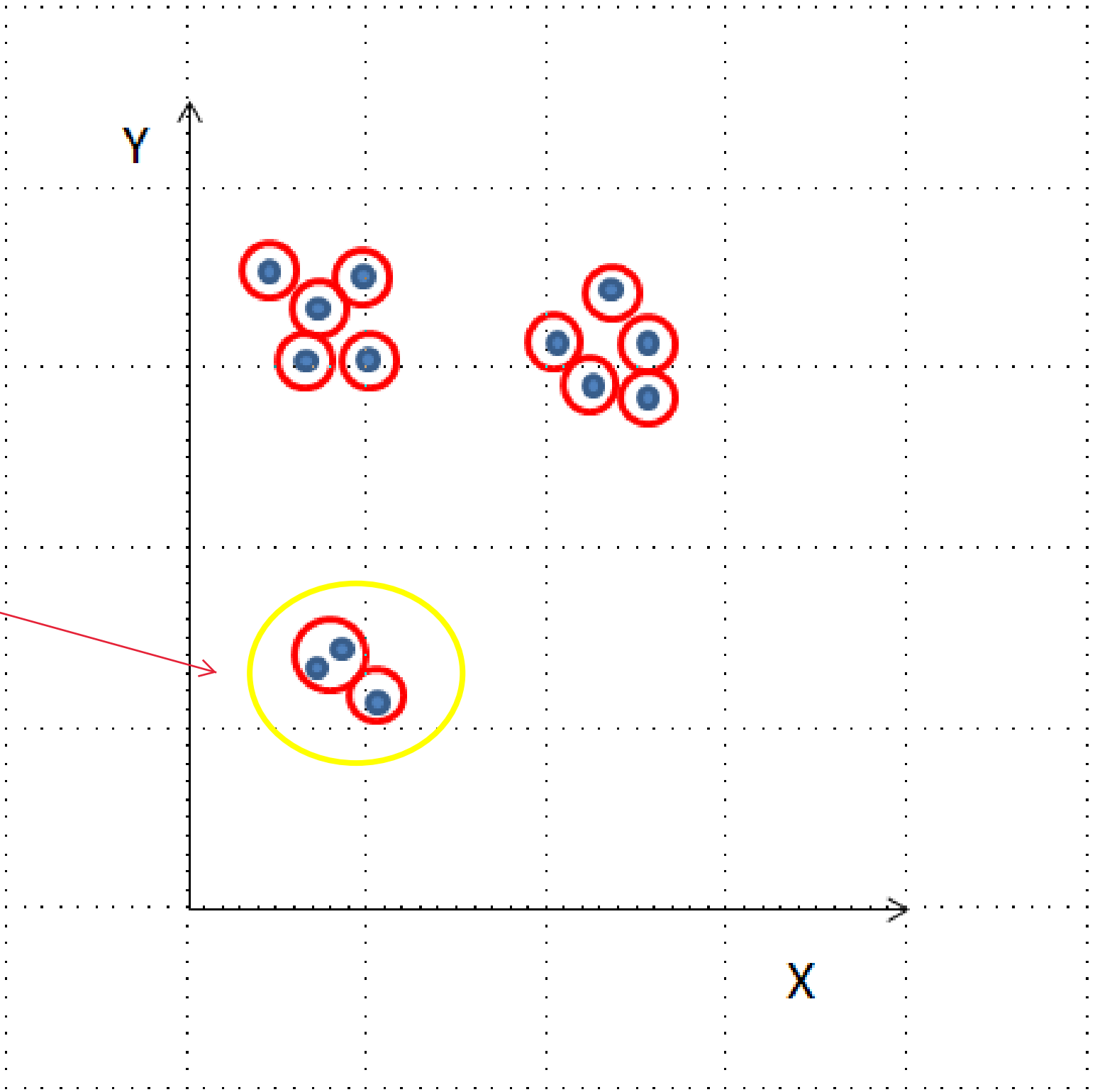
clustering



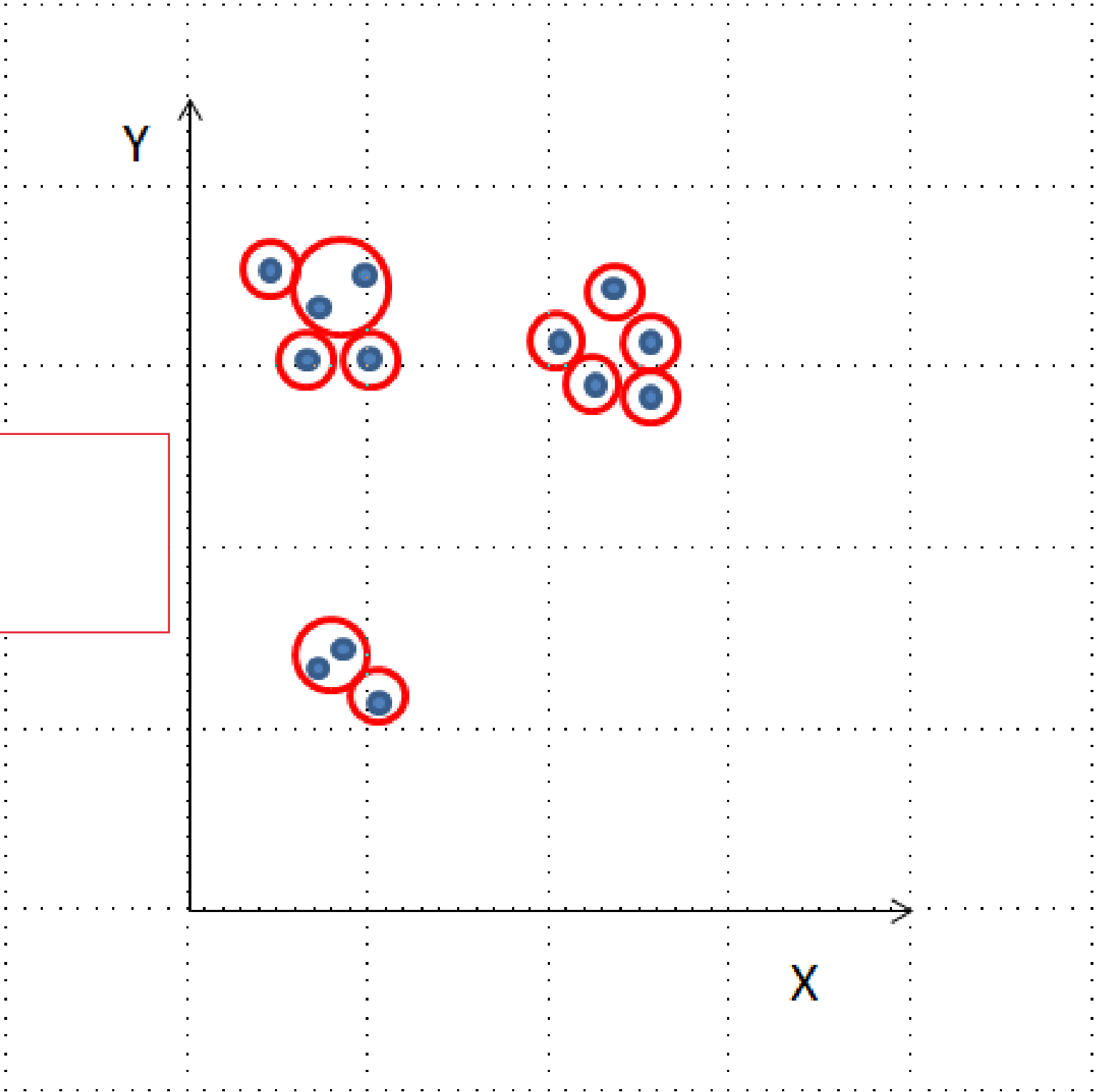
1. point cluster
cluster center point cluster
cluster x y
가 cluster cluster



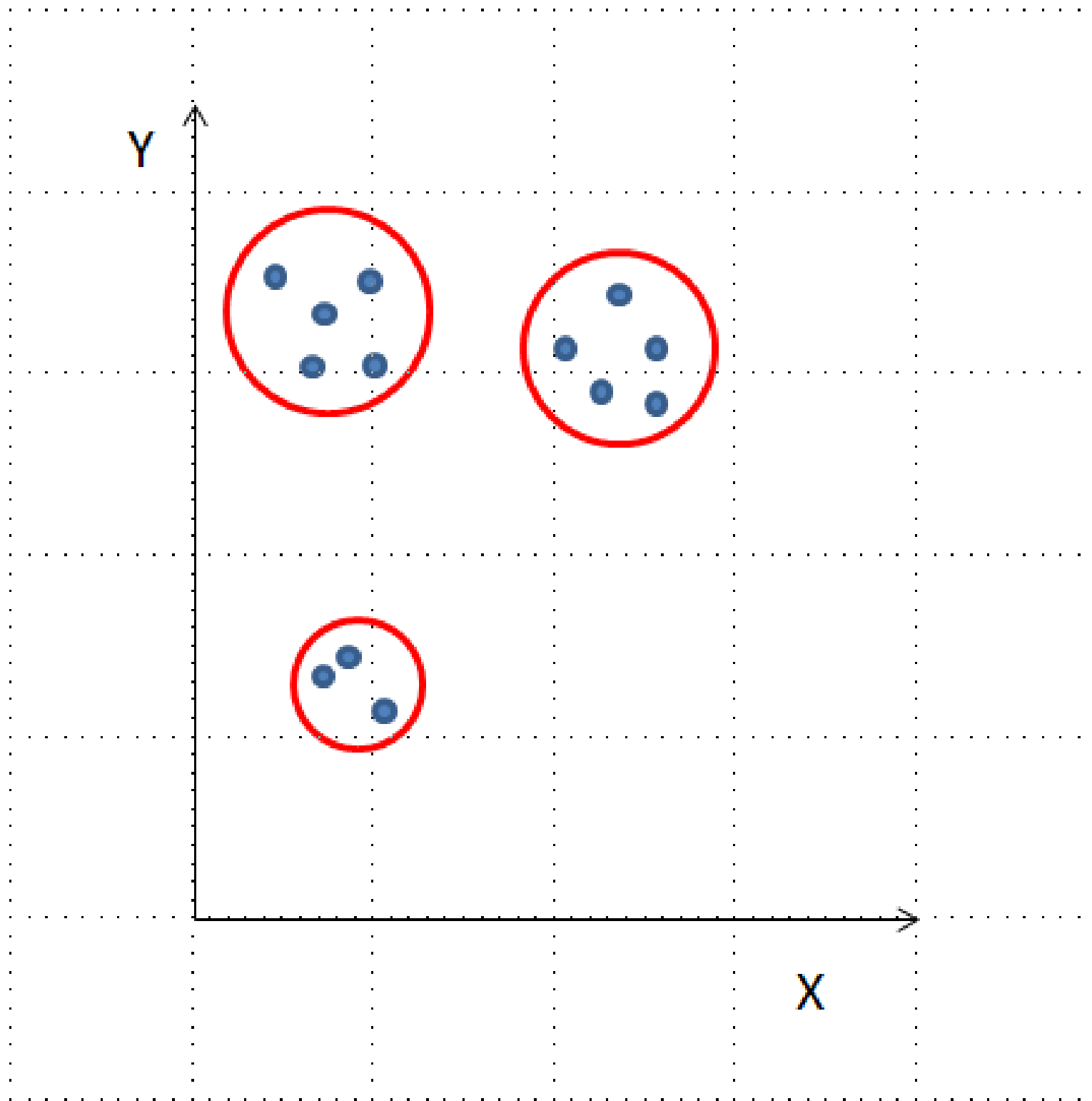
2.
가 가 가
cluster
cluster center가



가 가 cluster
process cluster
cluster



3 cluster



Hierarchical clustering in SciPy

```
from scipy.cluster.hierarchy import linkage, fcluster
from matplotlib import pyplot as plt
import seaborn as sns, pandas as pd
```

```
x_coordinates = [80.1, 93.1, 86.6, 98.5, 86.4, 9.5, 15.2, 3.4,
                 10.4, 20.3, 44.2, 56.8, 49.2, 62.5, 44.0]
y_coordinates = [87.2, 96.1, 95.6, 92.4, 92.4, 57.7, 49.4,
                 47.3, 59.1, 55.5, 25.6, 2.1, 10.9, 24.1, 10.3]

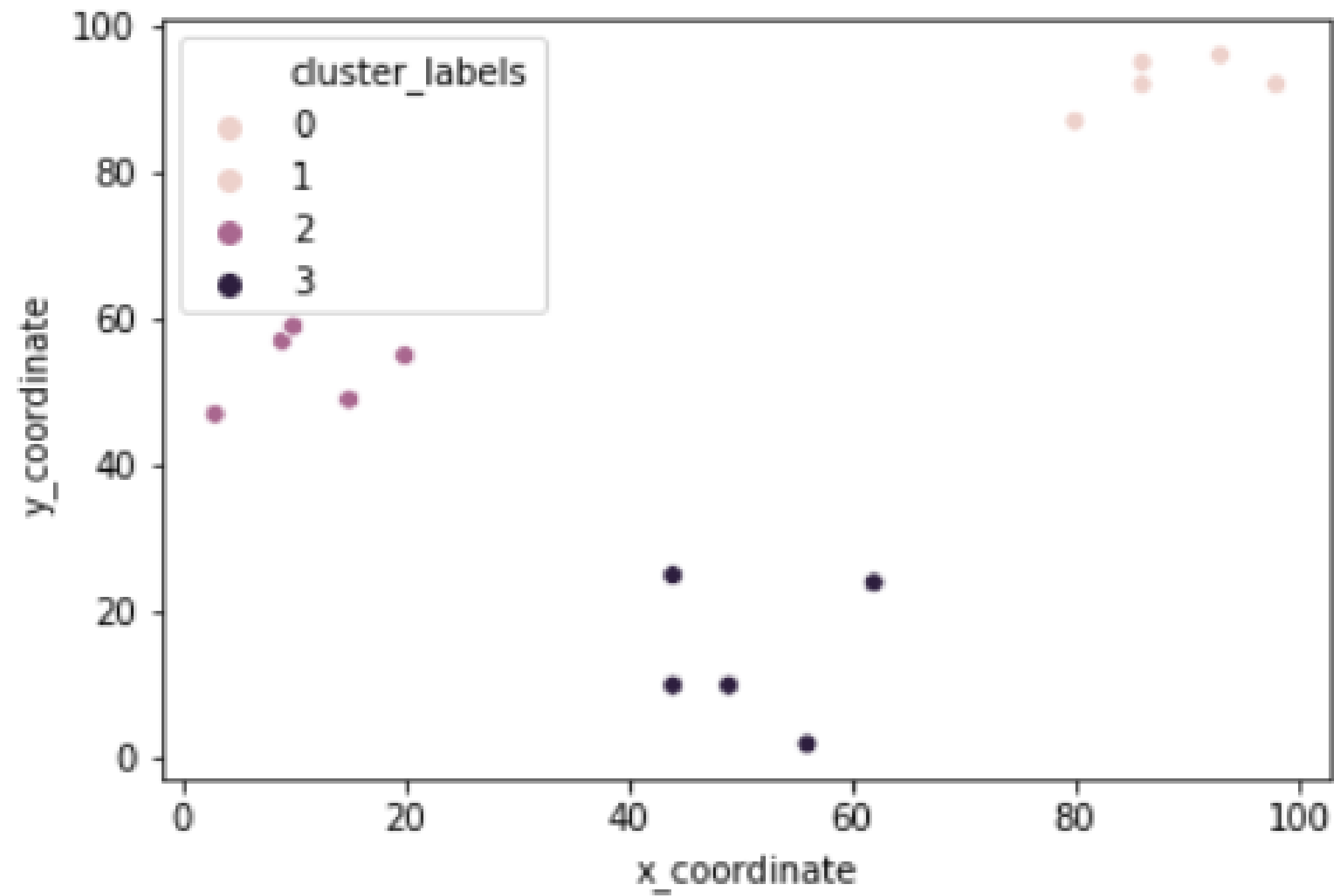
df = pd.DataFrame({'x_coordinate': x_coordinates,
                   'y_coordinate': y_coordinates})
```

```
Z = linkage(df, 'ward')
df['cluster_labels'] = fcluster(Z, 3, criterion='maxclust')
```

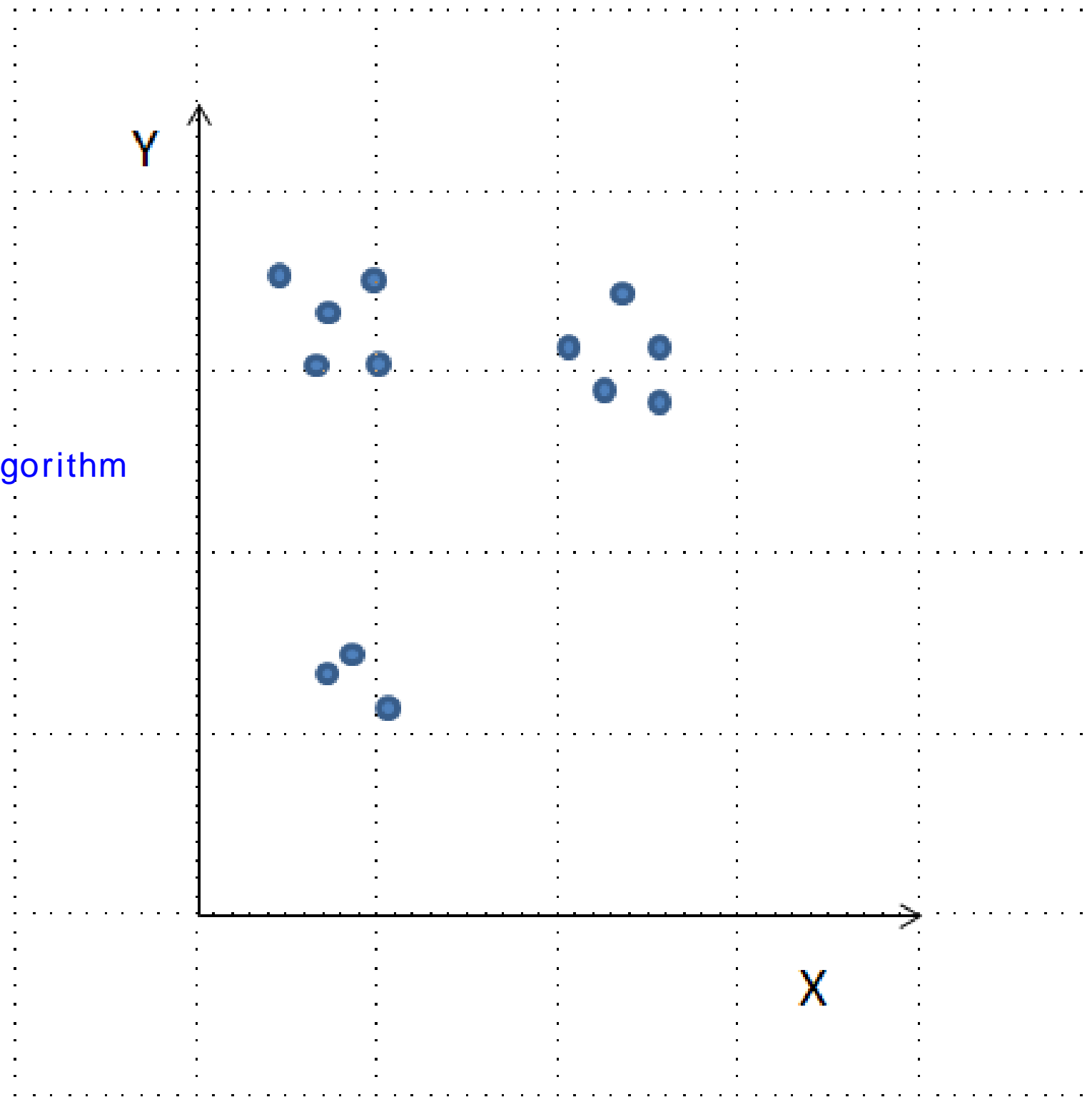
- 1.
2. fcluster method cluster
cluster label dataframe columns

```
sns.scatterplot(x='x_coordinate', y='y_coordinate',  
                hue='cluster_labels', data = df)  
  
plt.show()
```

labelcluster

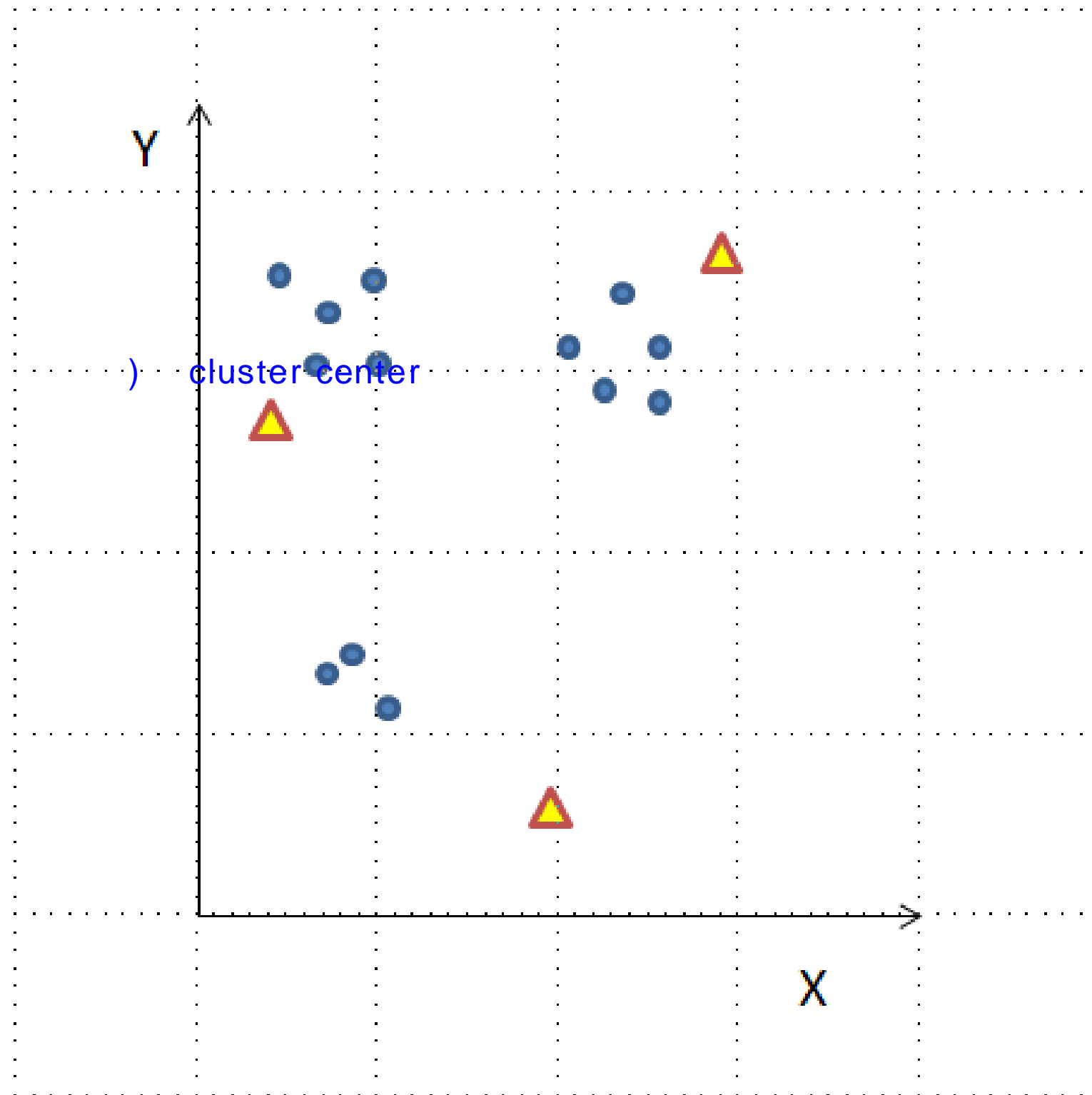


3 cluster
point set k - means clustering algorithm



1.3 cluster

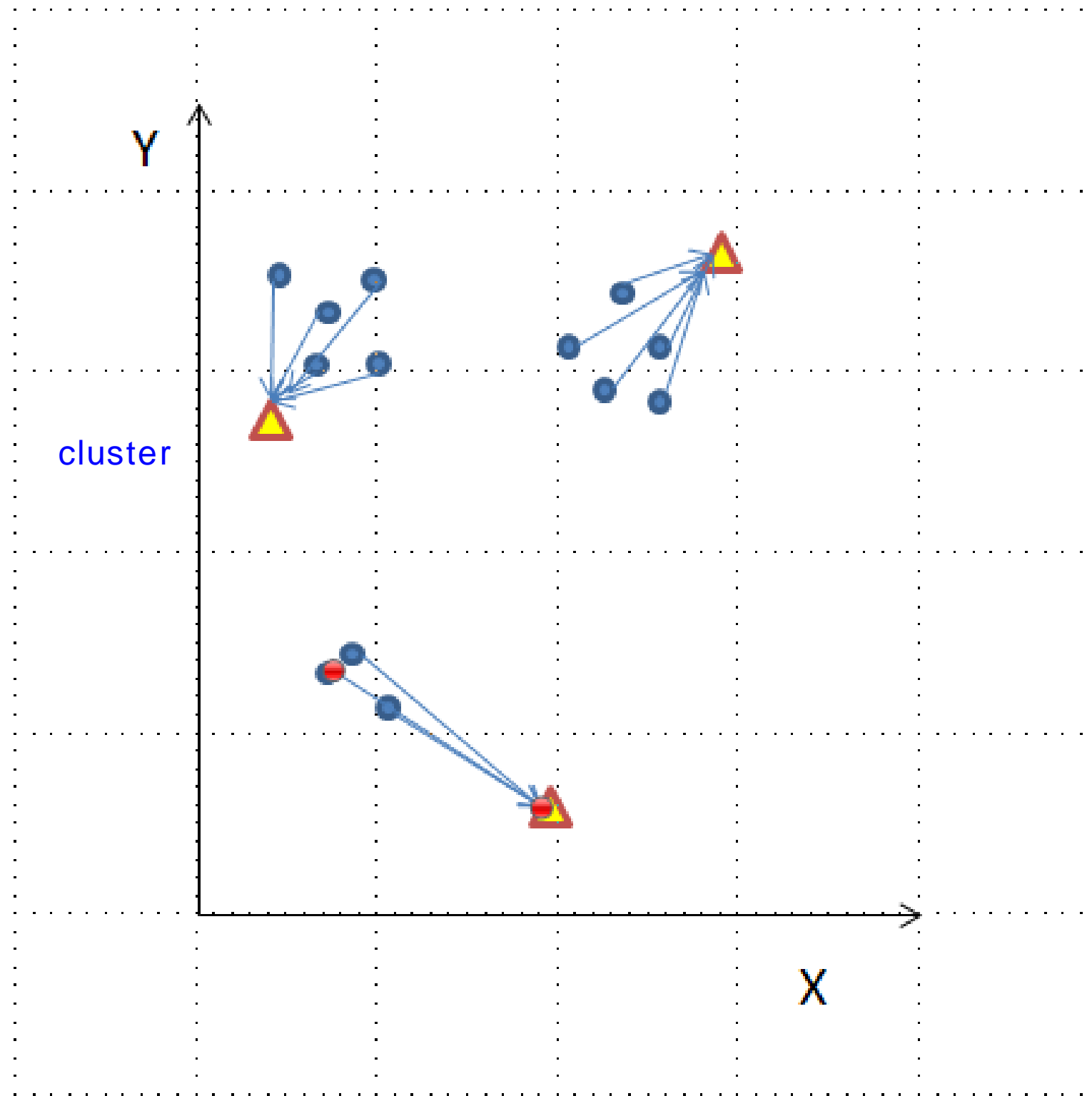
() cluster center



2. cluster center
point

가 가 가

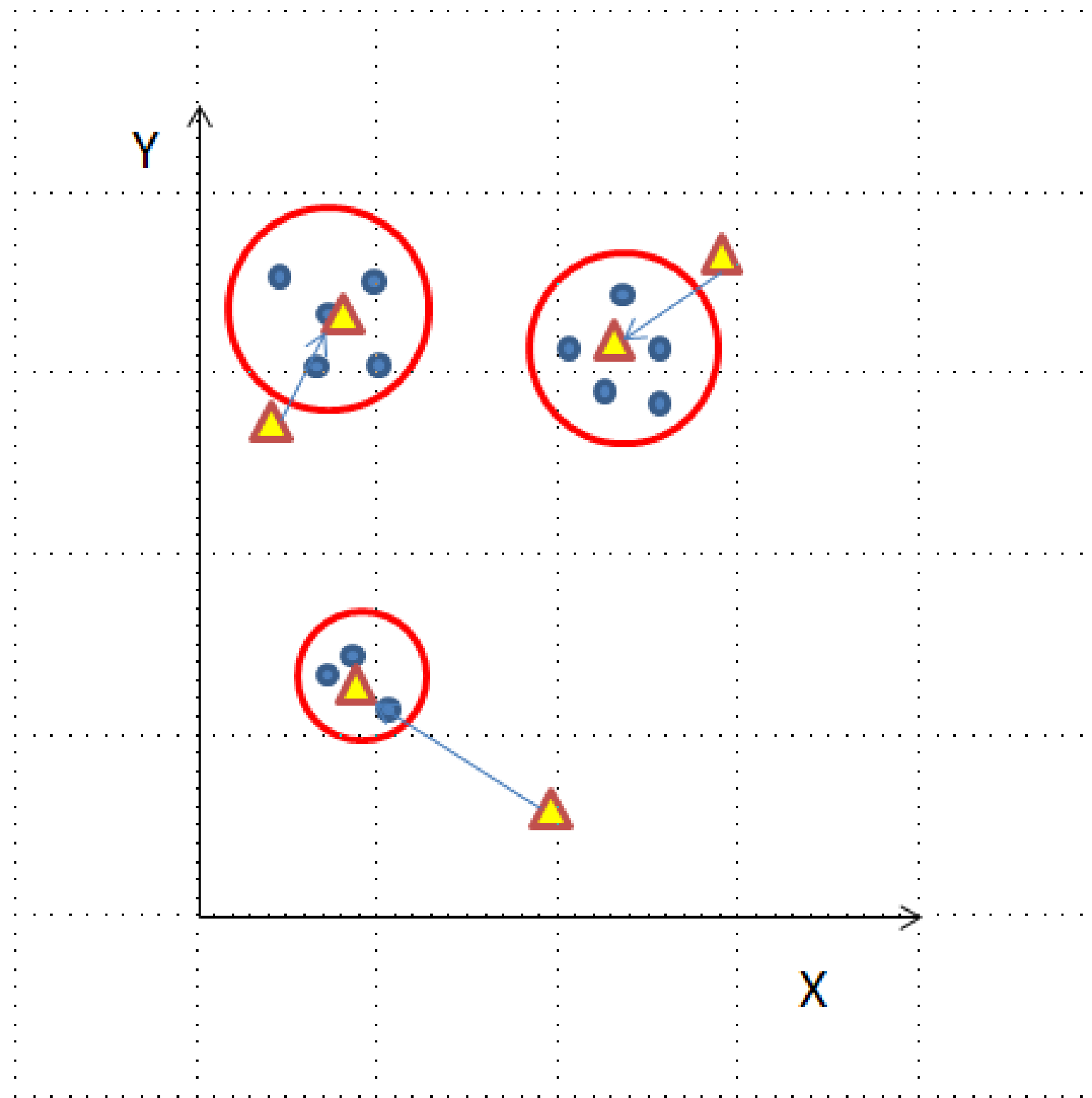
cluster



cluster center

cluster center

point



K-means clustering in SciPy

```
from scipy.cluster.vq import kmeans, vq
from matplotlib import pyplot as plt
import seaborn as sns, pandas as pd

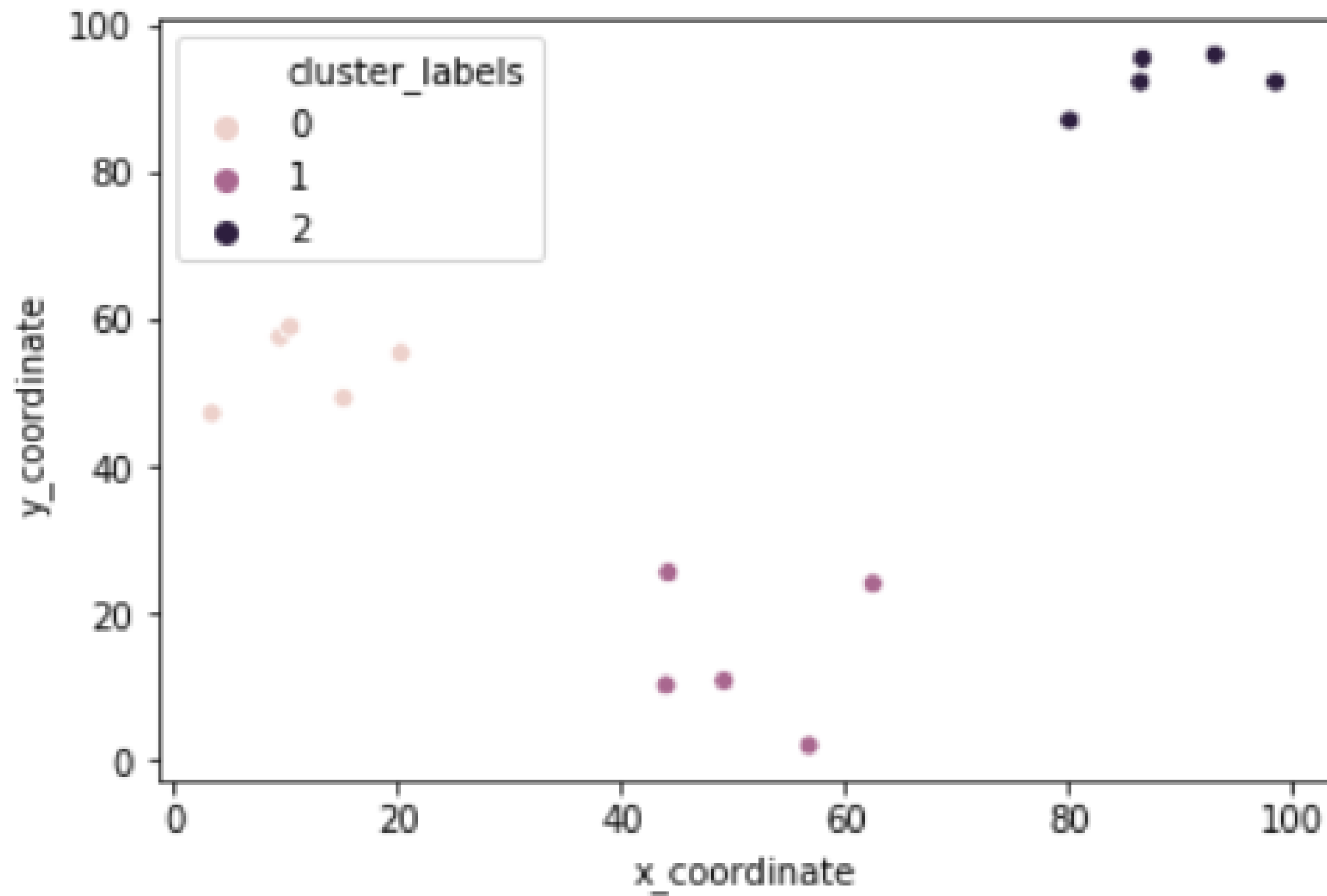
import random
random.seed((1000,2000))
```

```
x_coordinates = [80.1, 93.1, 86.6, 98.5, 86.4, 9.5, 15.2, 3.4,
                 10.4, 20.3, 44.2, 56.8, 49.2, 62.5, 44.0]
y_coordinates = [87.2, 96.1, 95.6, 92.4, 92.4, 57.7, 49.4,
                 47.3, 59.1, 55.5, 25.6, 2.1, 10.9, 24.1, 10.3]

df = pd.DataFrame({'x_coordinate': x_coordinates, 'y_coordinate': y_coordinates})
```

```
centroids, _ = kmeans(df, 3)
df['cluster_labels'], _ = vq(df, centroids)
```

```
sns.scatterplot(x='x_coordinate', y='y_coordinate',
                hue='cluster_labels', data = df)
plt.show()
```

Next up: hands-on exercises

CLUSTER ANALYSIS IN PYTHON

Data preparation for cluster analysis

CLUSTER ANALYSIS IN PYTHON



Shaumik Daityari
Business Analyst

Why do we need to prepare data for clustering?

- Variables have incomparable units (product dimensions in cm, price in \$)
- Variables with same units have vastly different scales and variances (expenditures on cereals, travel)
- Data in raw form may lead to bias in clustering
- Clusters may be heavily dependent on one variable
- Solution: normalization of individual variables

clustering data ?
- - 가 가
- 가
(ex -)
- raw data clustering 가
=>

Normalization of data

- data

Normalization: process of rescaling data to a standard deviation of 1

1

```
x_new = x / std_dev(x)
```

```
from scipy.cluster.vq import whiten
```

```
data = [5, 1, 3, 3, 2, 3, 3, 8, 1, 2, 2, 3, 5]
```

1.

whiten method

array 1

whiten

```
scaled_data = whiten(data)
```

```
print(scaled_data)
```

whiten method

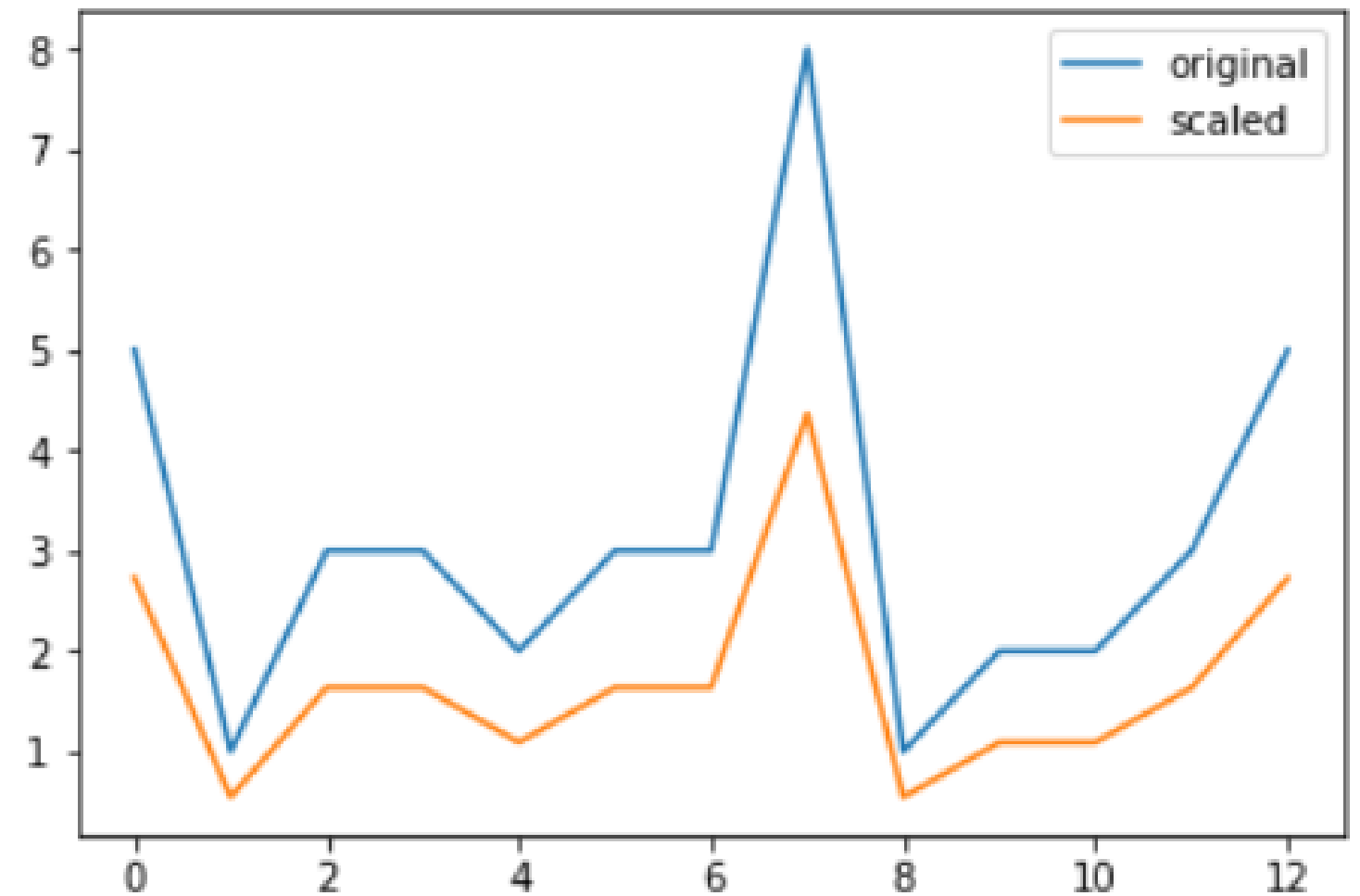
```
[2.73, 0.55, 1.64, 1.64, 1.09, 1.64, 1.64, 4.36, 0.55, 1.09, 1.09, 1.64, 2.73]
```

Illustration: normalization of data

```
# Import plotting library
from matplotlib import pyplot as plt

# Initialize original, scaled data
plt.plot(data, point, label="original")
plt.plot(scaled_data, label="scaled")

# Show legend and display plot
plt.legend()
plt.show()
```



x : point
y : original, scaled data

Next up: some DIY exercises

CLUSTER ANALYSIS IN PYTHON