CSE/ISyE 6740
Computational Data Analysis

Clustering

08/20/2025

Kai Wang, Assistant Professor in Computational Science and Engineering
kwang692@gatech.edu

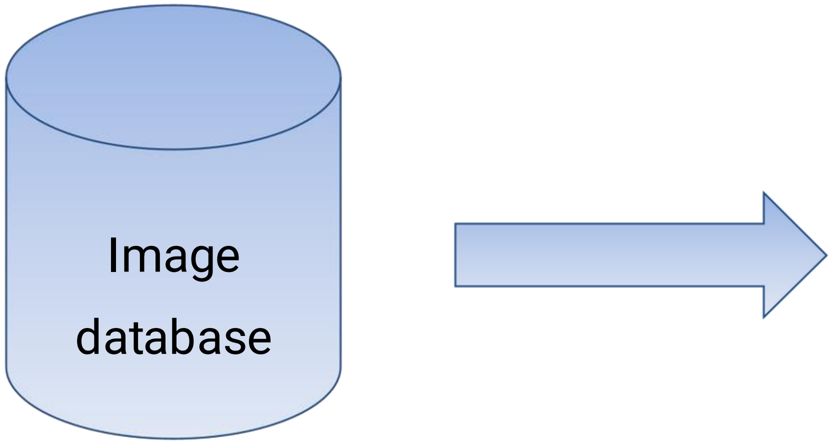
Georgia Tech

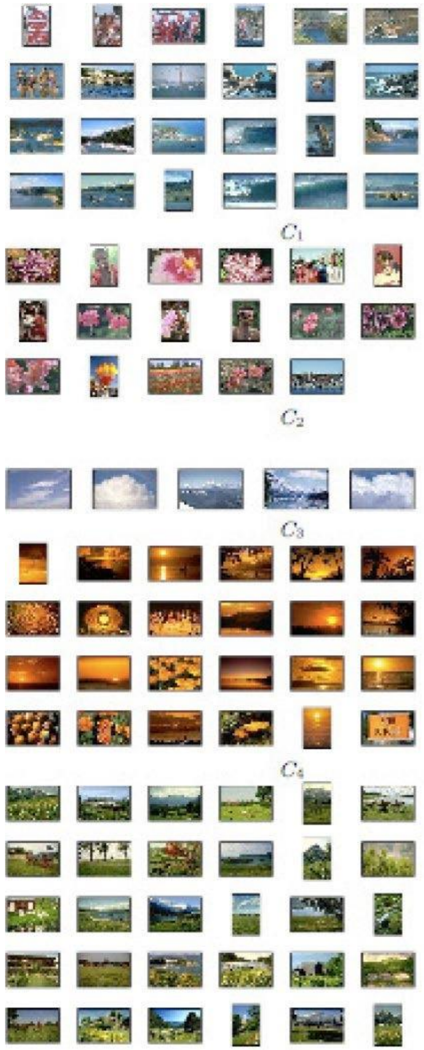
Outline

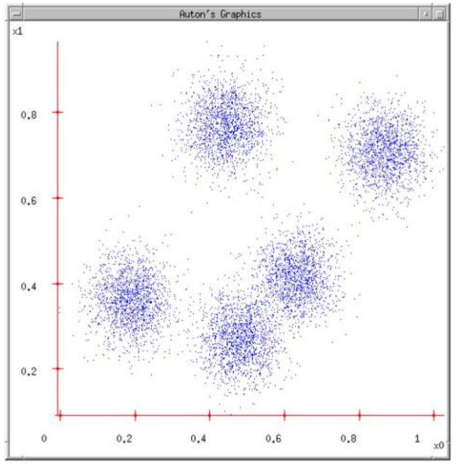
- **Unsupervised learning**
  - **Clustering**
    - Formal statement of clustering problem
    - K-means algorithm
    - Hardness
    - Distance
    - Generalized k-means
  - **Spectral clustering**
    - How to handle network information?
    - Adjacency matrix, graph Laplacian, and eigenvectors as representation!

Clustering

Clustering Images





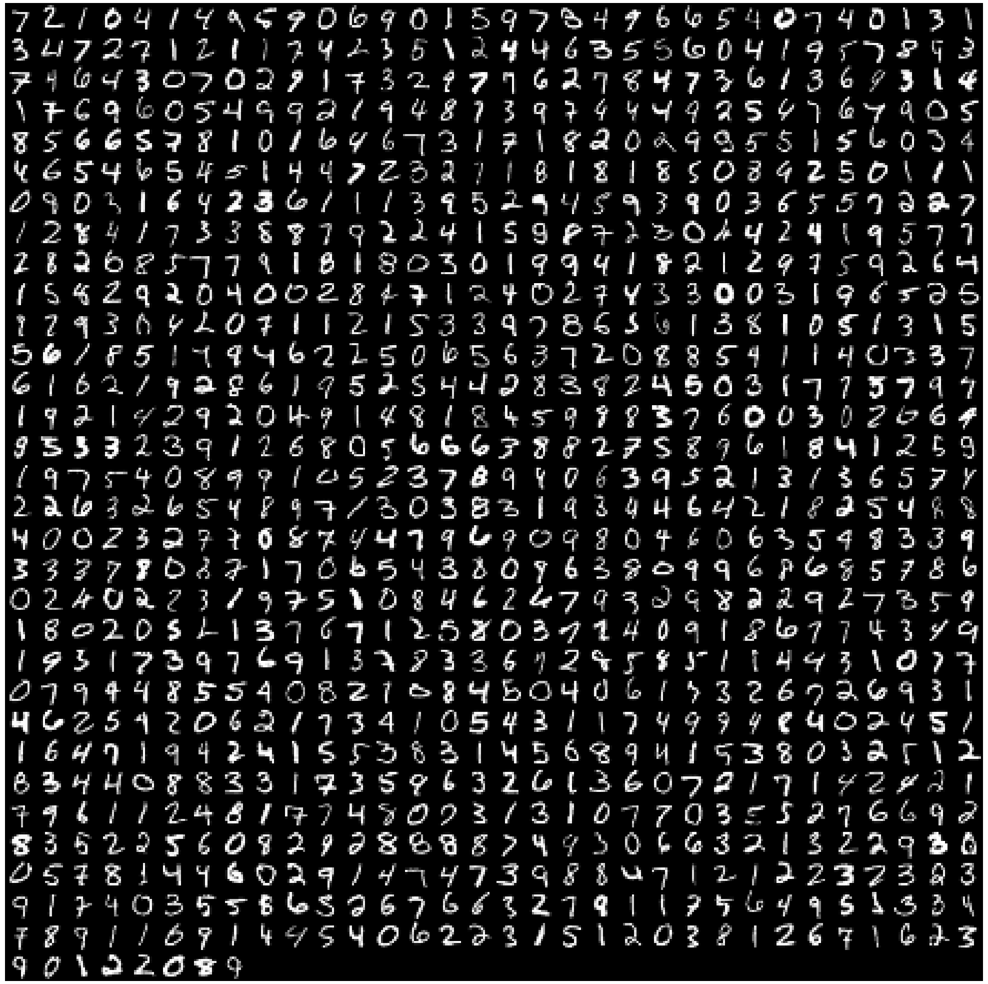


- Goal of clustering: Divide object into groups, and objects within a group are more similar than those outside the group

Cluster Other Things…

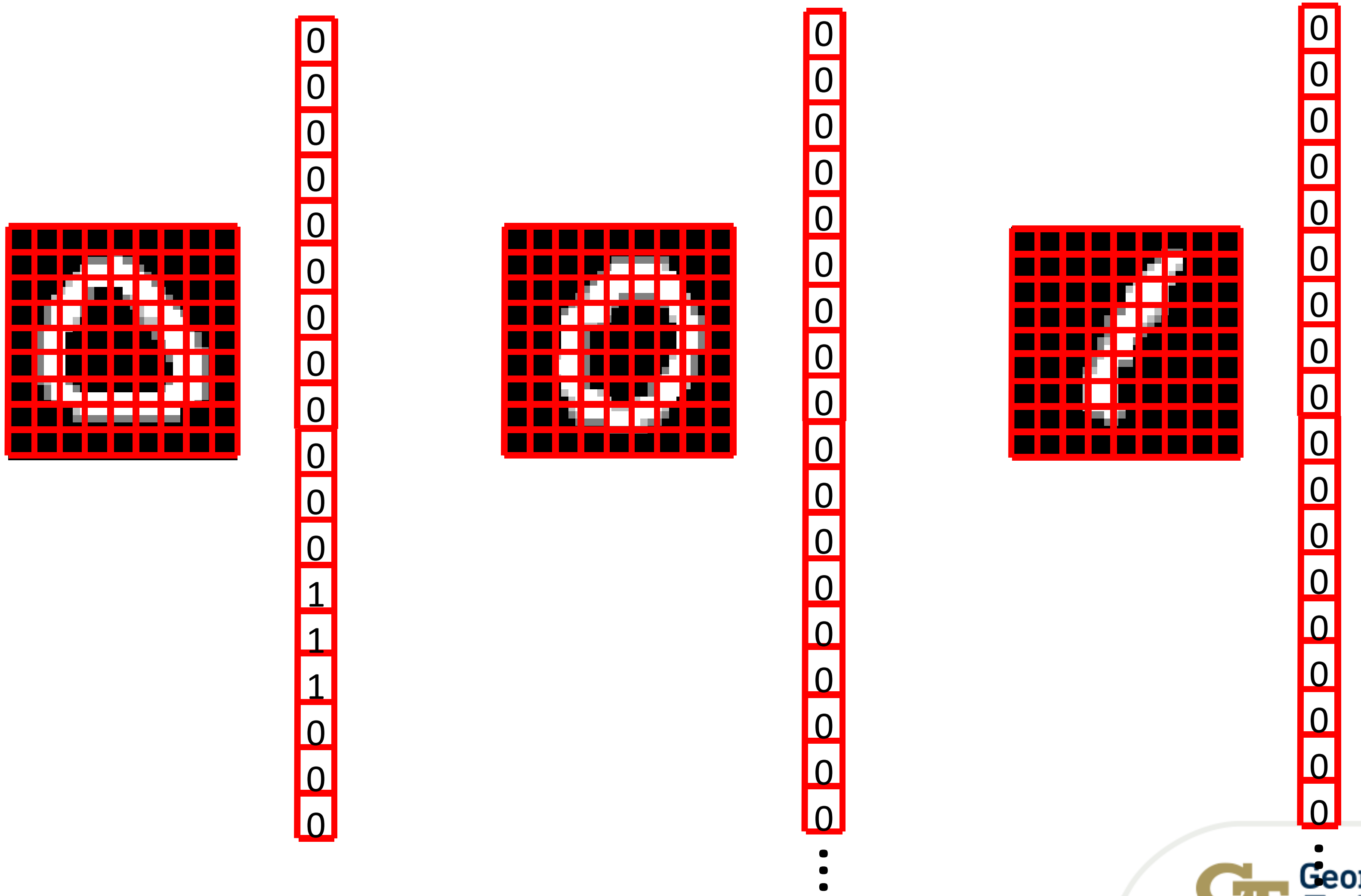
Piotr
Pyotr
Petros
Pietro
Pedro
Pierre
Piero
Peter
Peder
Peka
Peadar

Cluster Handwritten Images

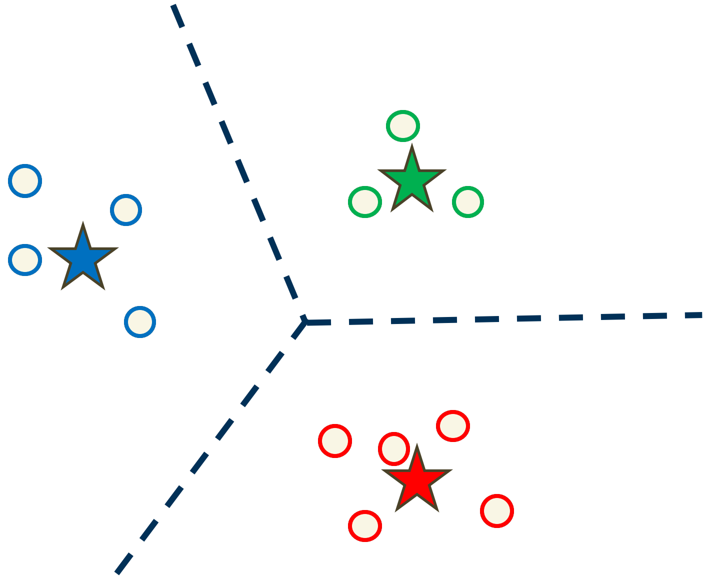


How to Represent Objective?

- Represent image as a vector of pixels, and each pixel as a value between 0 and 1.



Formal Statement of Clustering Problem



- Given n data points: $\{x^1, x^2, \dots, x^n\} \in \mathbb{R}^d$
- Find k cluster centers: $\{c^1, c^2, \dots, c^k\} \in \mathbb{R}^d$
- Assign each data point to one cluster: $\pi(i) \in \{1,2,\dots,k\}$
- Such that the averaged square distances from each data point to its respective cluster center is small, i.e.,

$$\min_{c,\pi} \frac{1}{n} \sum_{i=1}^{n} \left\| x^i - c^{\pi(i)} \right\|^2$$

K-means Algorithm

1. define "k"'s value
2. randomly get k points as cluster centers.
3. assign each data points.

- Initialize k cluster centers $\{c^1, c^2, \dots, c^k\}$ randomly
- Do
  - Decide the cluster memberships of each data point, x^i, by assigning it to the nearest cluster center (cluster assignment)

$$\pi(i) = argmin_{j=1,2,\dots k} \left\| x^i - c^j \right\|^2$$

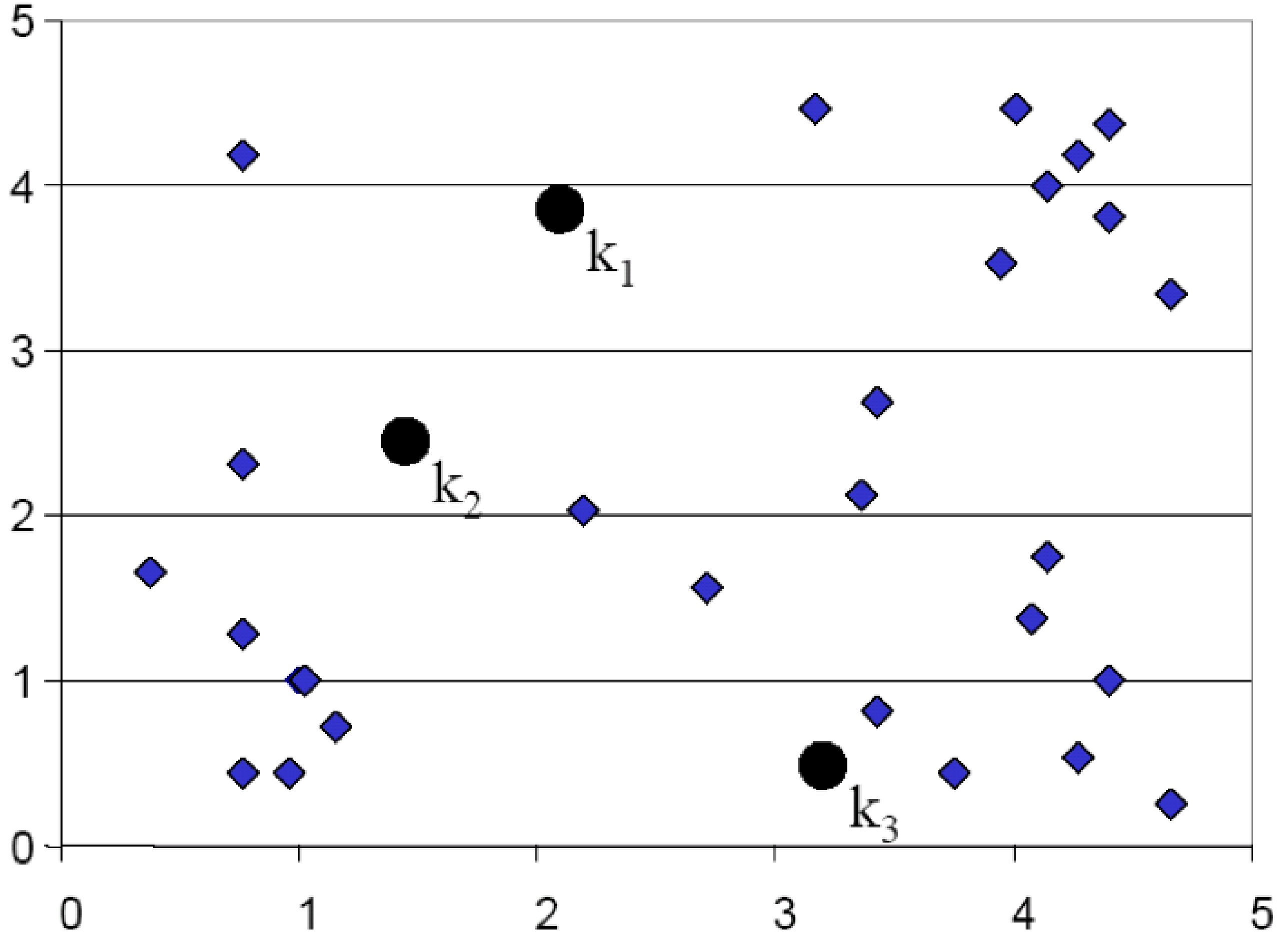
to the nearest cluster center.

  - Adjust the cluster center (center adjustment)

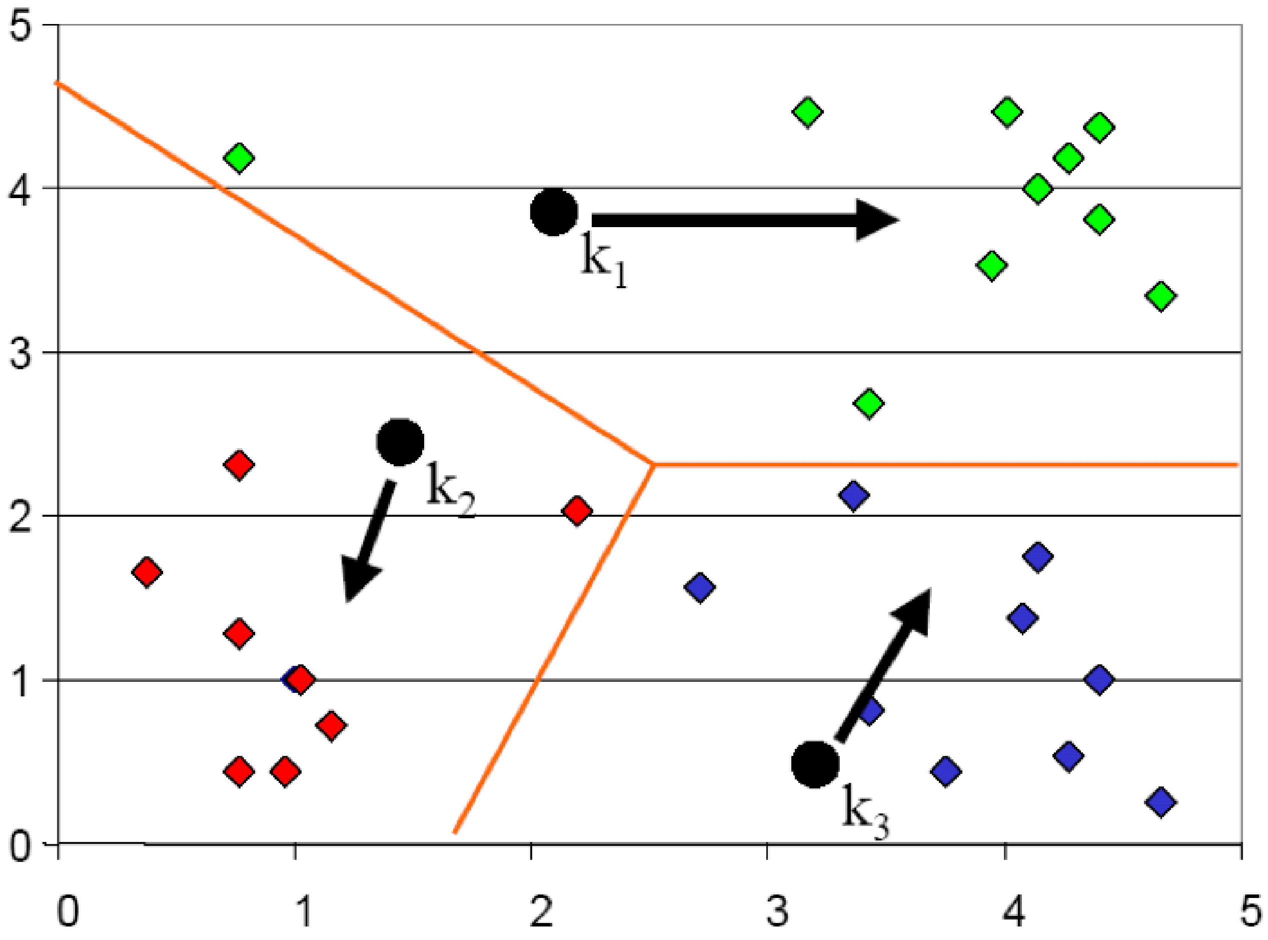
$$c^j = \frac{1}{|\{i: \pi(i) = j\}|} \sum_{i:\pi(i)=j} x^i$$

- While(any cluster center has been changed)

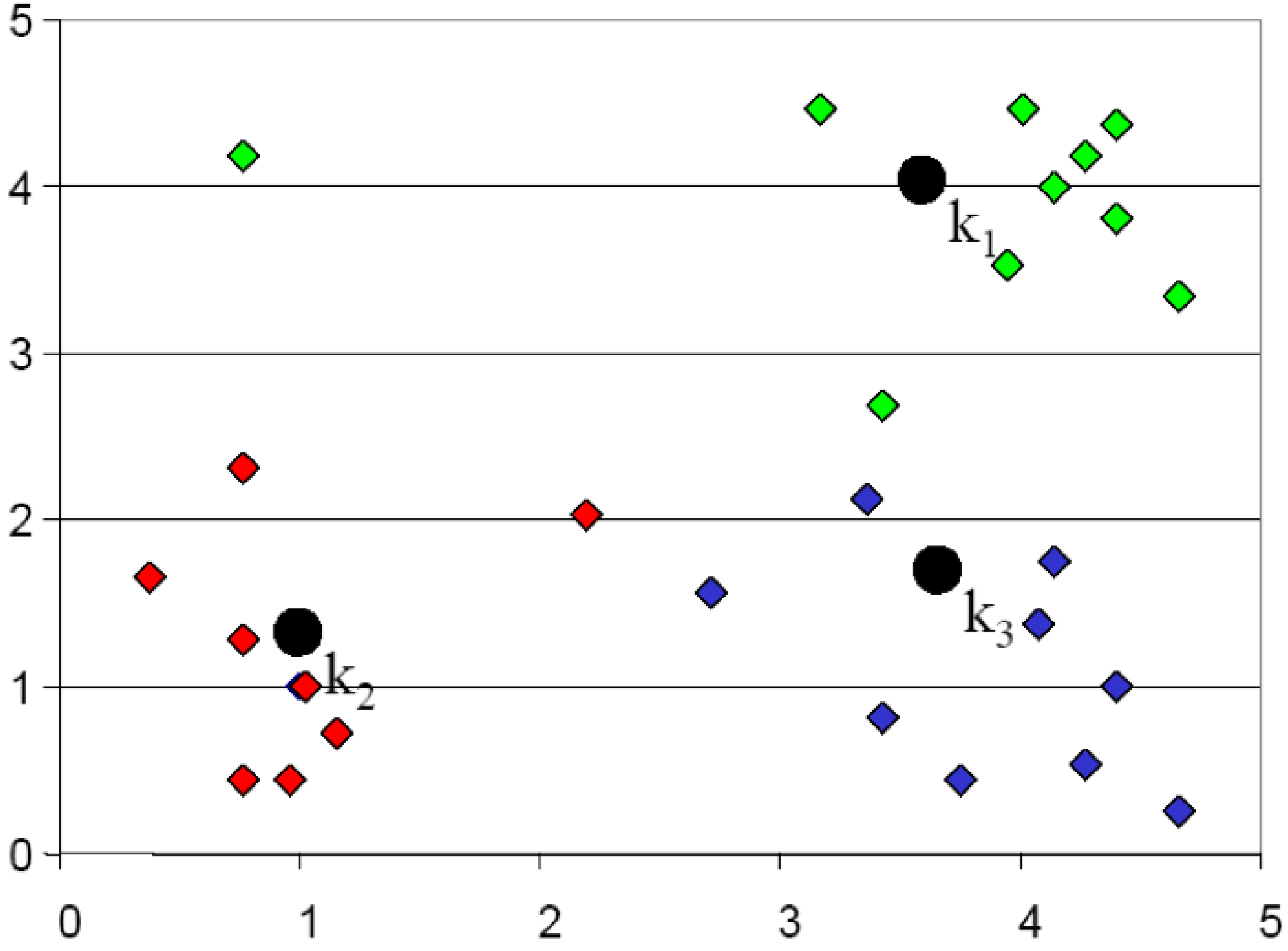
K-means Step 1



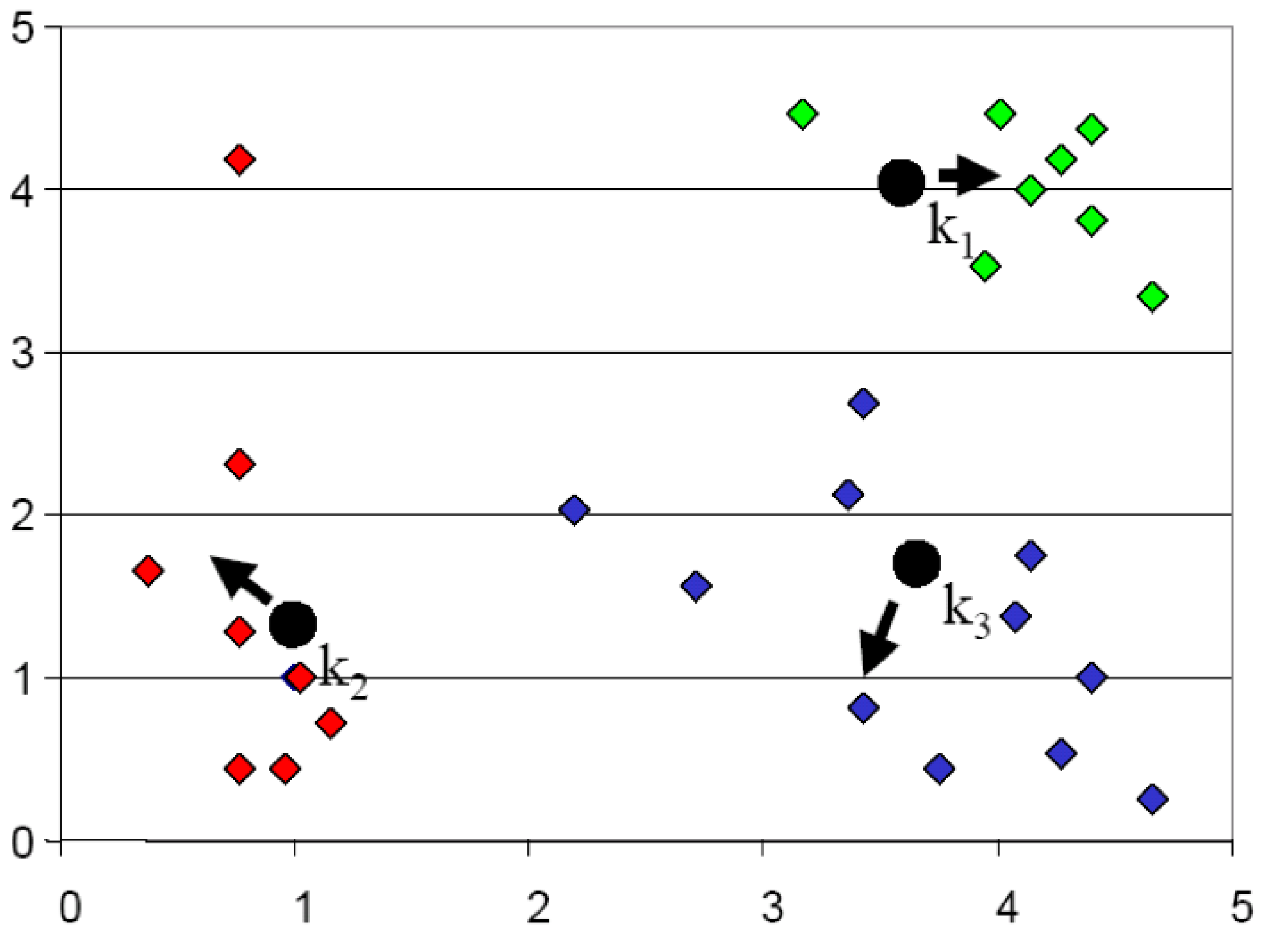
K-means Step 2



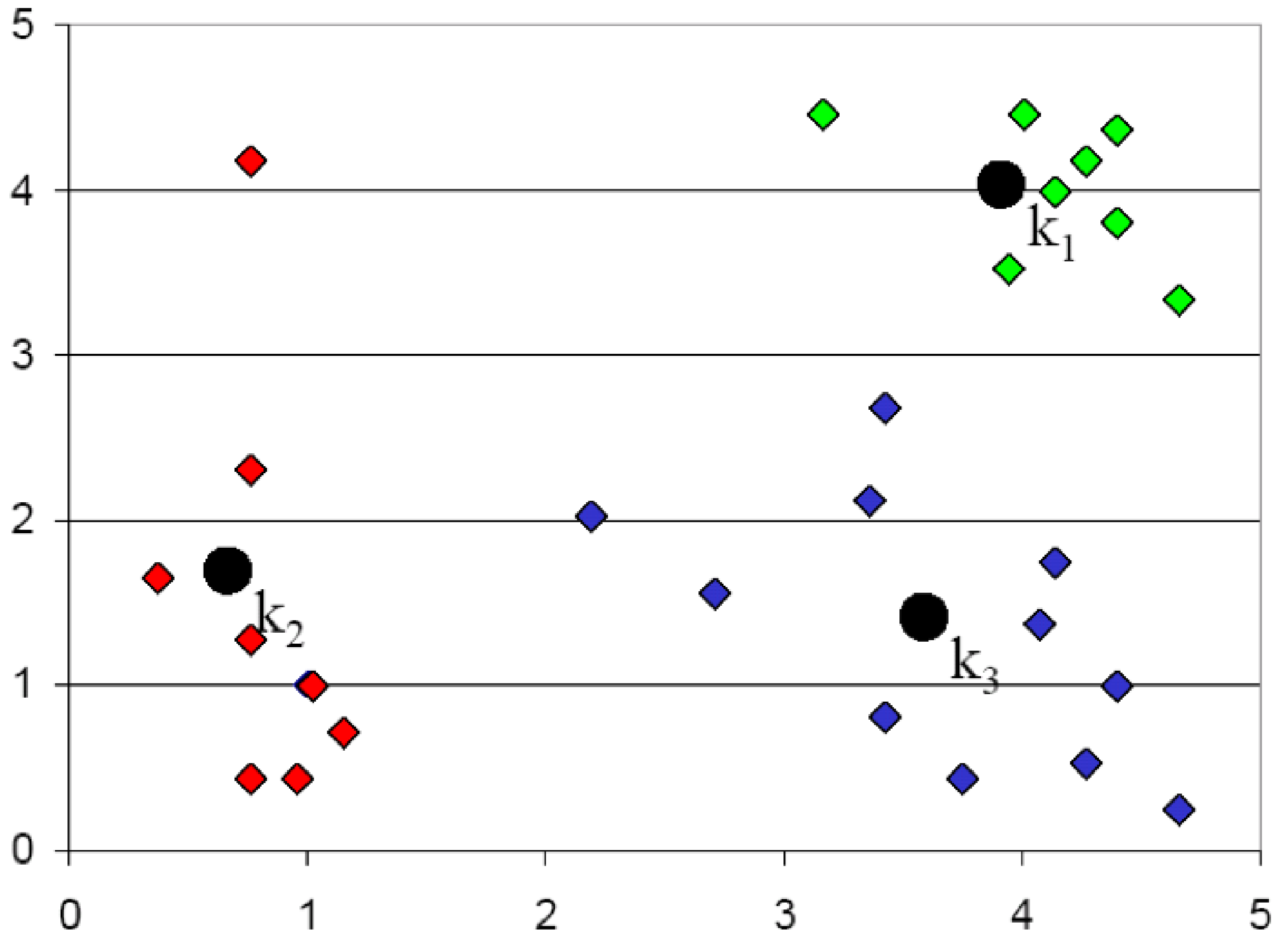
K-means Step 3



K-means Step 4



K-means Step 5



Questions

- Will different initializations lead to different results?
  - Yes
  - No
  - Sometimes

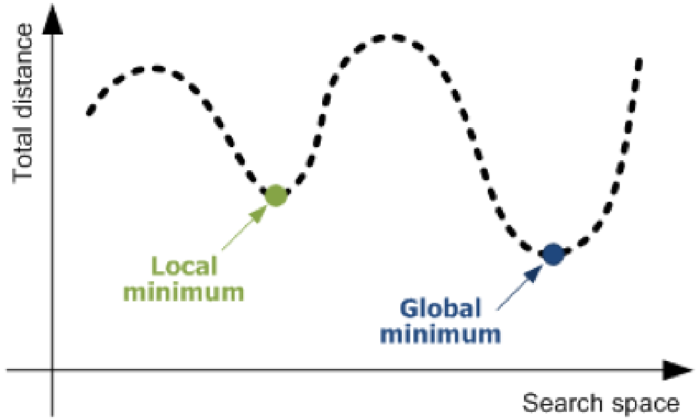
- Will the algorithm always stop after some iterations?
  - Yes
  - No (we have to set a maximum number of iterations)
  - Sometimes

Clustering is NP-hard in General

- Find k cluster centers, $\{c^1, c^2, \dots, c^k\} \in \mathbb{R}^d$, and assign each data point i to one cluster, $\pi(i) \in \{1,2,\dots,k\}$, to minimize

$$\min_{c,\pi} \frac{1}{n} \sum_{i=1}^{n} \left\| x^i - c^{\pi(i)} \right\|^2$$

- A search problem over the space of discrete assignments
  - For all n data points together, there are k^n possibility.
  - The cluster assignment determines cluster centers, and vice versa.



Convergence of k-means Clustering

- Will k-means objective oscillate?

$$\frac{1}{n}\sum_{i=1}^{n}\left\|x^i - c^{\pi(i)}\right\|^2$$

- The minimum value of the objective is finite.

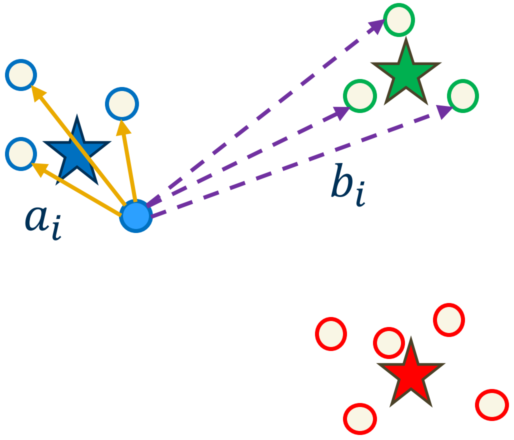
- Each iteration of k-means algorithm decrease the objective.
    - Cluster assignment step decreases the objective
        - $\pi(i) = argmin_{j=1,2,\ldots k}\left\|x^i - c^j\right\|^2$
    - Center assignment step decreases the objective
        - $c^j = \frac{1}{|\{i:\pi(i)=j\}|}\sum_{i:\pi(i)=j} x^i = \operatorname{argmin}_c \sum_{i:\pi(i)=j}\left\|x^i - c\right\|^2$

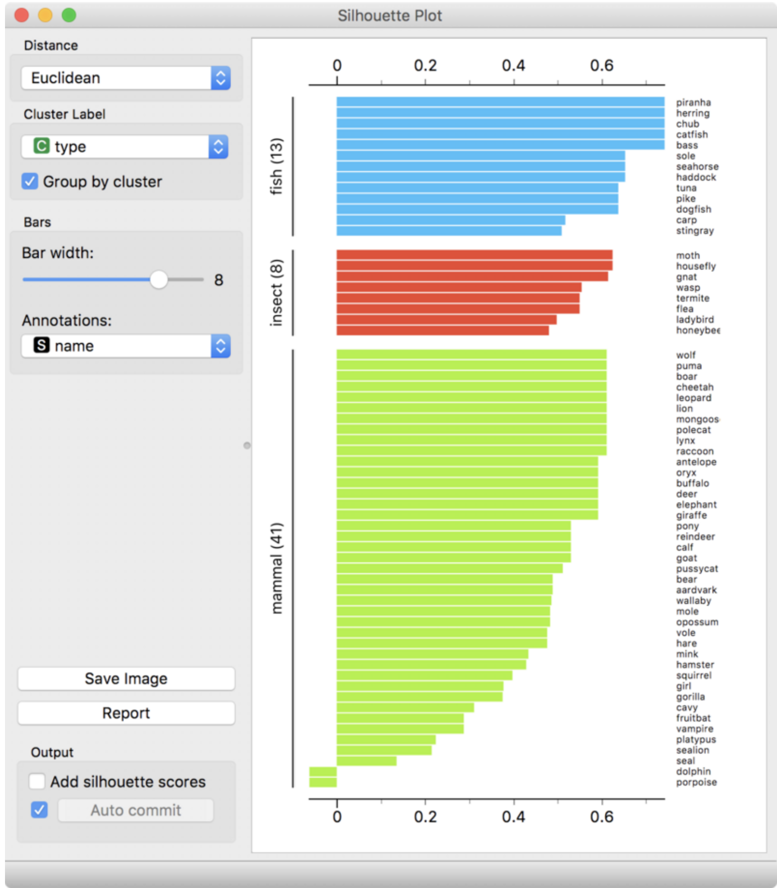
How Many Clusters?

- Fixed a-priori? Data-driven approach?

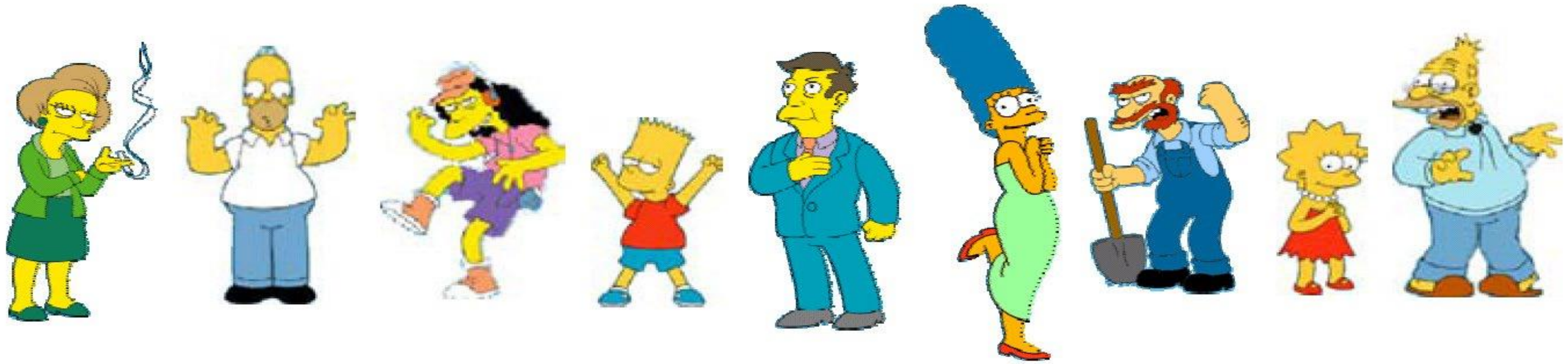
- Silhouette value: $S_i = \frac{b_i - a_i}{\max(a_i, b_i)} \in [-1,1]$ (one heuristic)
  - **Distance to closest cluster** b_i: the minimum average distance from the i-th point to points in a different cluster, minimized over clusters.
  - **In-cluster distance** a_i: the average distance from the i-th point to other points in the same cluster as i.

- No gold standard method
  - Often determined by trial-and-error

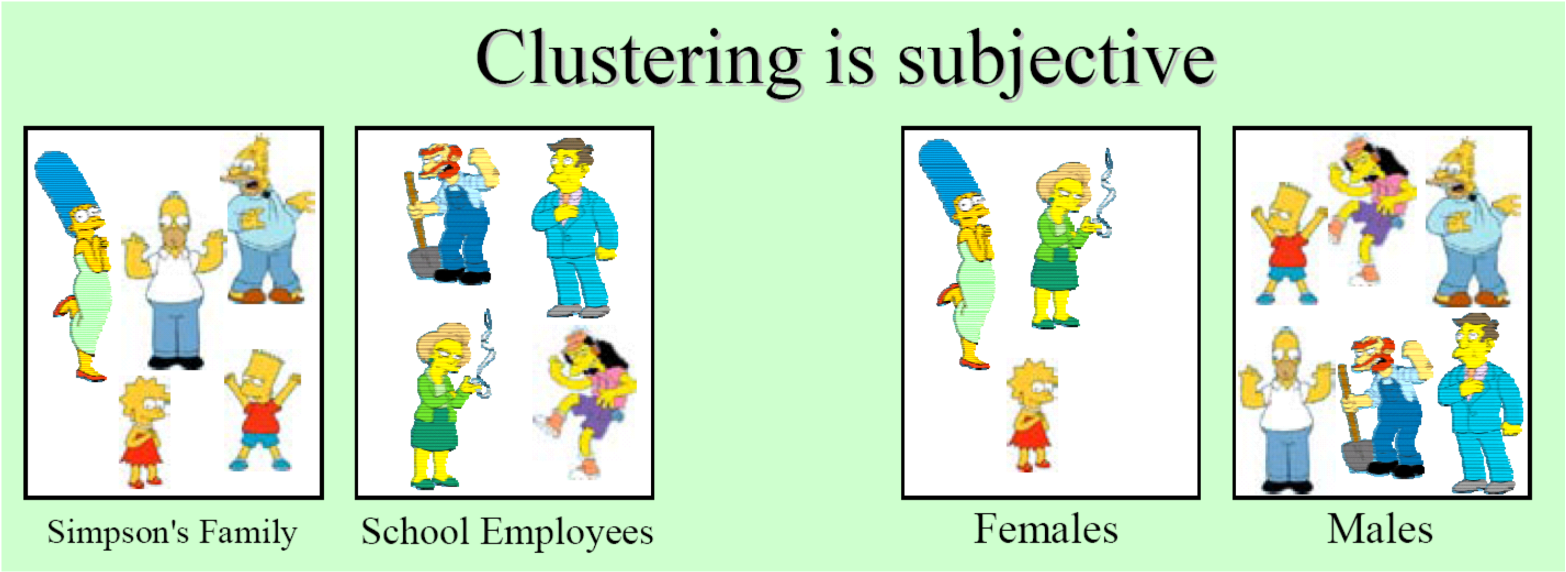




Are These Everything About Clustering?



What is considered similar/dissimilar?



Object in Real Life

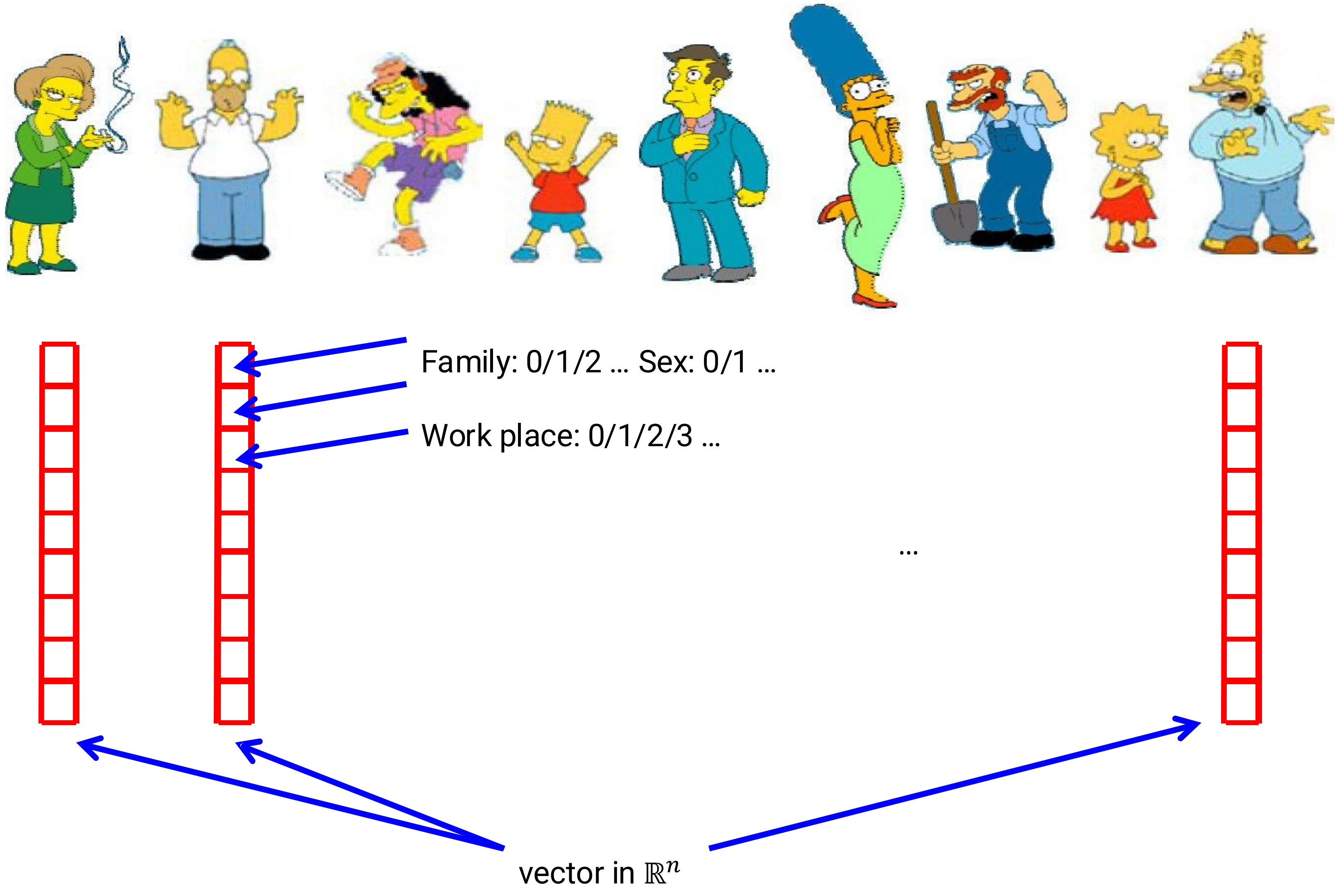
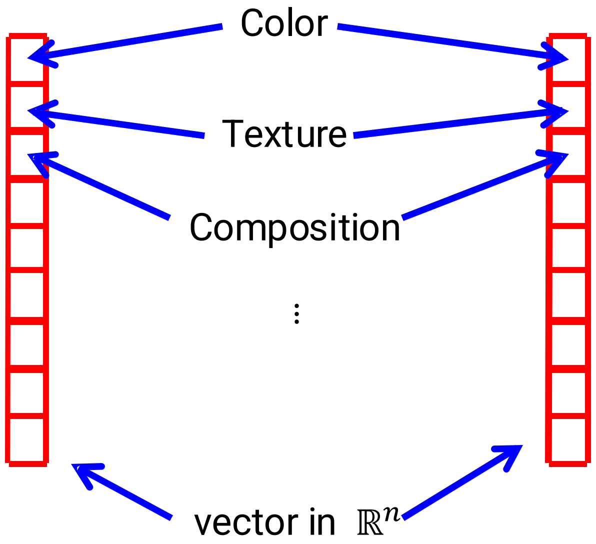


Image of Different Sizes



You Pick Your Similarity/Dissimilarity

What Similarity/Dissimilarity Function?

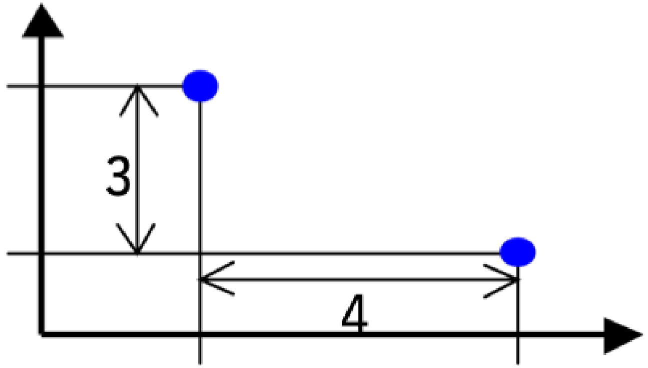
- Desired properties of dissimilarity (distance) function $d: X \times X \rightarrow \mathbb{R}^{+} \cup \{0\}$
  - **Symmetry**: $d(x, y) = d(y, x)$
    - Otherwise, you could claim "Alex looks like Bob, but Bob doesn't look like Alex".
  - **Positive separability**: $d(x, y) = 0$ if and only if $x = y$.
    - Otherwise, there are objects that are different, but you can't tell apart.
  - **Triangle inequality**: $d(x, y) \leq d(x, z) + d(z, y)$
    - Otherwise, you may have "Alex is very like Bob, Bob is very like Carl, but Alex is very unlike Carl."

Distance Function for Vectors

- Suppose two data points, both in $\mathbb{R}^d$
  - $x = (x_1, x_2, \dots, x_d)^\top$
  - $y = (y_1, y_2, \dots, y_d)^\top$

- **Euclidean distance**: $d(x, y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$
- **Minkovski distance**: $d(x, y) = \sqrt[p]{\sum_{i=1}^{n}(x_i - y_i)^p}$ (or so call p norm)
  - Euclidean distance: $p = 2$
  - Manhattan distance: $p = 1,\ d(x, y) = \sum_i |x_i - y_i|$
  - "inf"-distance: $p = \infty,\ d(x, y) = \max_i |x_i - y_i|$

Distance Example



- Euclidean distance: $\sqrt{3^2 + 4^2} = 5$
- Manhattan distance: $4 + 3 = 7$
- "inf"-distance: $\max\{4,3\} = 4$

Hamming Distance

- Manhattan distance is also called ***Hamming distance*** when all the features are binary (or categorical)

- Count the number of differences between two binary (categorical) vectors

- Example, $x, y \in \mathbb{R}^{17}$

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
x	0	1	1	0	0	1	0	0	1	0	0	1	1	1	0	0	1
y	0	1	1	1	0	0	0	0	1	1	1	1	1	1	0	1	1

$$d(x, y) = 5$$

Edit Distance

- Transform one of the objects into the other, and measure how much effort it takes

$x =$ "INTENTION"

$y =$ "EXECUTION"

```
I N T E * N T I O N
| | | | | | | | | |
* E X E C U T I O N
d s s   i s
```

d: deletion (cost 5)
s: substitution (cost 1)
i: insertion (cost 2)

$$d(x, y) = 5 \times 1 + 1 \times 3 + 2 \times 1 = 10$$

Generalized K-means Clustering

- Initialize k cluster centers $\{c^1, c^2, \dots, c^k\}$ randomly

- Do
  - Decide the cluster memberships of each data point, x^i, by assigning it to the nearest cluster center (cluster assignment)

$$\pi(i) = argmin_{j=1,2,\dots k}\, d(x^i, c^j)$$

  - Adjust the cluster center (center adjustment)

$$c^j = argmin_{v \in \mathbb{R}^d} \sum_{i:\pi(i)=j} d(x^i, v)$$

Squared Euclidean distance

$$c^j = \frac{1}{|\{i: \pi(i) = j\}|} \sum_{i:\pi(i)=j} x^i$$

- While(any cluster center has been changed)

Spectral Clustering

Clustering Nodes in a Network

Visualization tools:

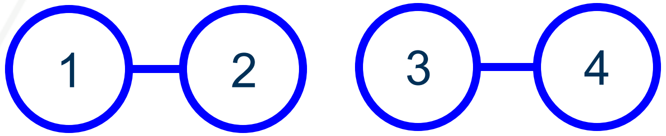
- [Networkx]()
- [Gephi]()
- …

No Feature? Find A Representation!

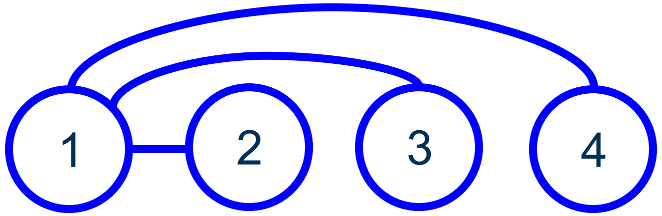
- If we have features associated to each node, you can use them to cluster nodes.
- However, if we don’t have features of the nodes in a network, we will need to find a representation to represent the nodes.
- **Adjacency matrix**, **graph Laplacian**, and their **eigenvectors** are good options for you to handle network data!

Spectral Clustering Algorithm

- **Step 1**: represent the graph as adjacency matrix A, and diagonal matrix D



$$A = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \quad D = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$



$$A = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \quad D = \begin{pmatrix} 3 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

- **Step 2**: form a special matrix $L = D - A$ (the graph Laplacian)

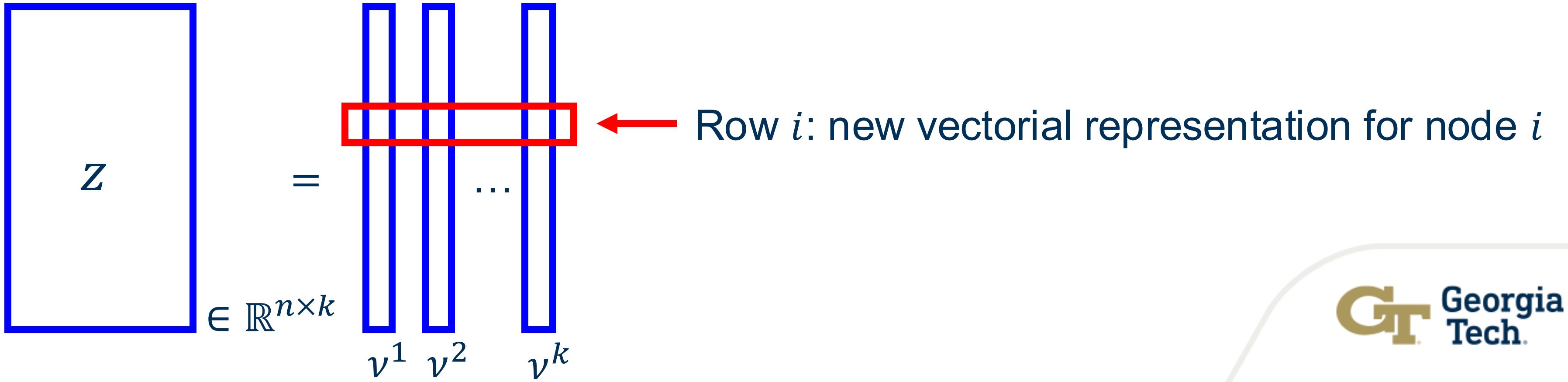
$$L = \begin{pmatrix} 1 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & -1 & 1 \end{pmatrix} \qquad L = \begin{pmatrix} 3 & -1 & -1 & -1 \\ -1 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{pmatrix}$$

Spectral Clustering Algorithm

- **Step 3**: compute k eigenvectors, $v^1, v^2, \dots, v^k$, of the Laplacian L, corresponding to the k-smallest eigenvalues ($k \ll n$).

$$Lv^i = \begin{pmatrix} 1 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & -1 & 1 \end{pmatrix} v^i = \lambda_i v^i$$

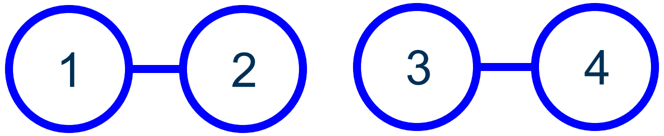
- **Step 4**: run k-means algorithm on $Z = (v^1, v^2, \dots, v^k)$ by treating each row as a new data point:



Why This Works?

- **Adjacency matrix** and **Graph Laplacian** include information about neighborhoods in the network (graph).

- Eigenvectors are a good way to represent the matrix and extract the most relevant information, i.e., **Principal Component Analysis**.
  - Eigenvectors tell you information about how nodes are clustered.
  - For connected graphs, eigenvectors with smaller eigenvalues can be used as a compact representation.



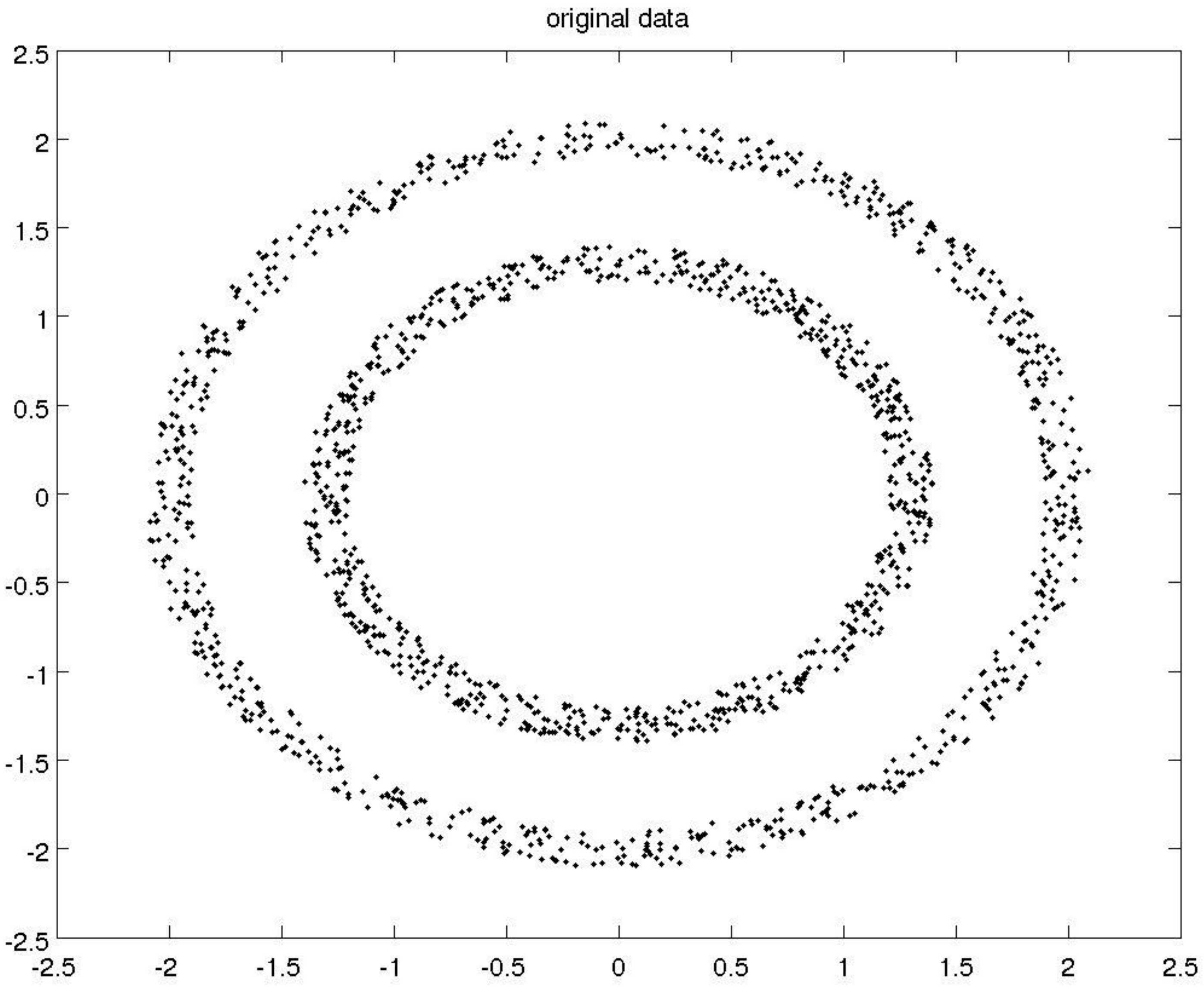
$$A = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

$$L = D - A = \begin{pmatrix} 1 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & -1 & 1 \end{pmatrix}$$

$$Lv^1 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & -1 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

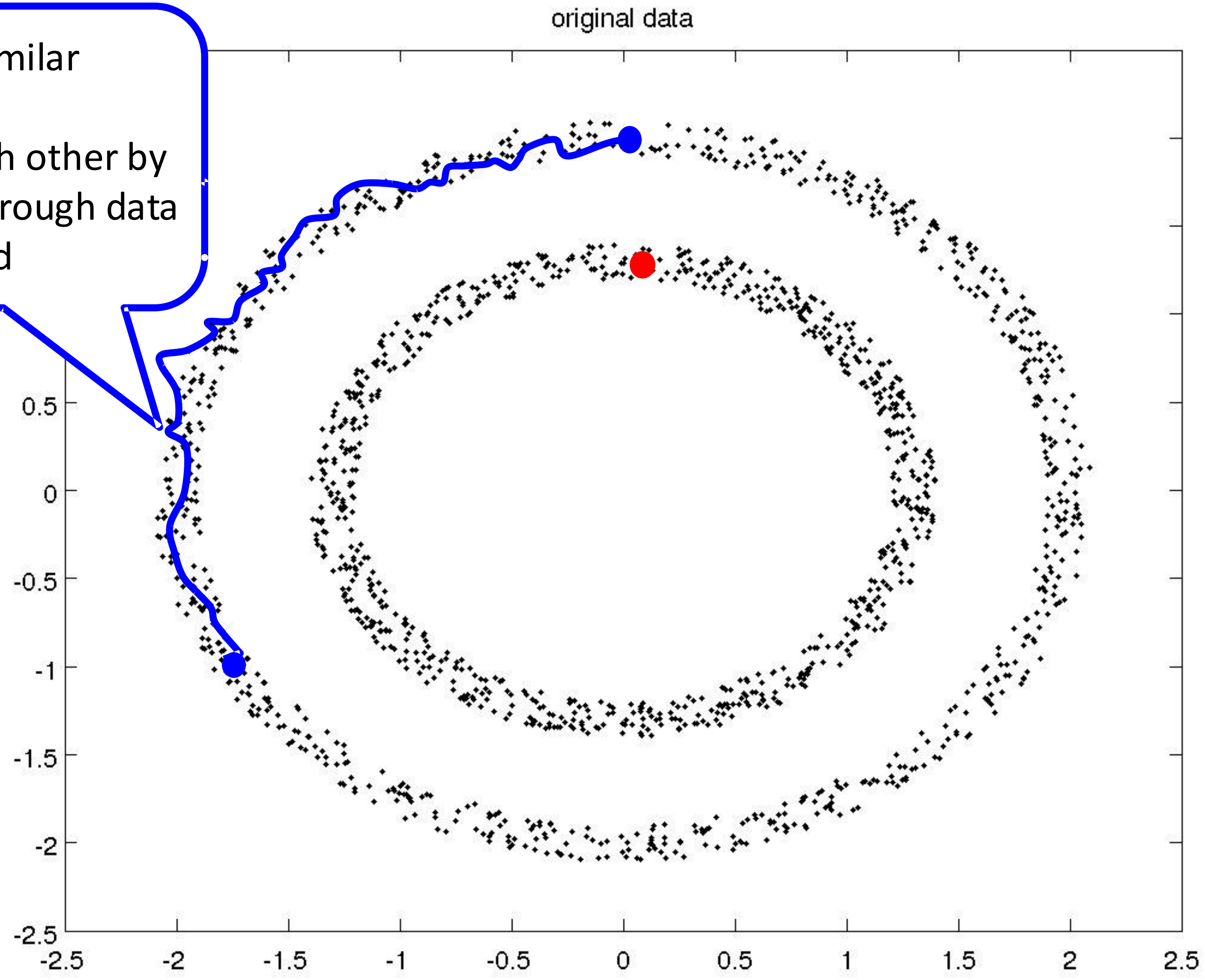
$$Lv^2 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & -1 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

How About this Dataset?

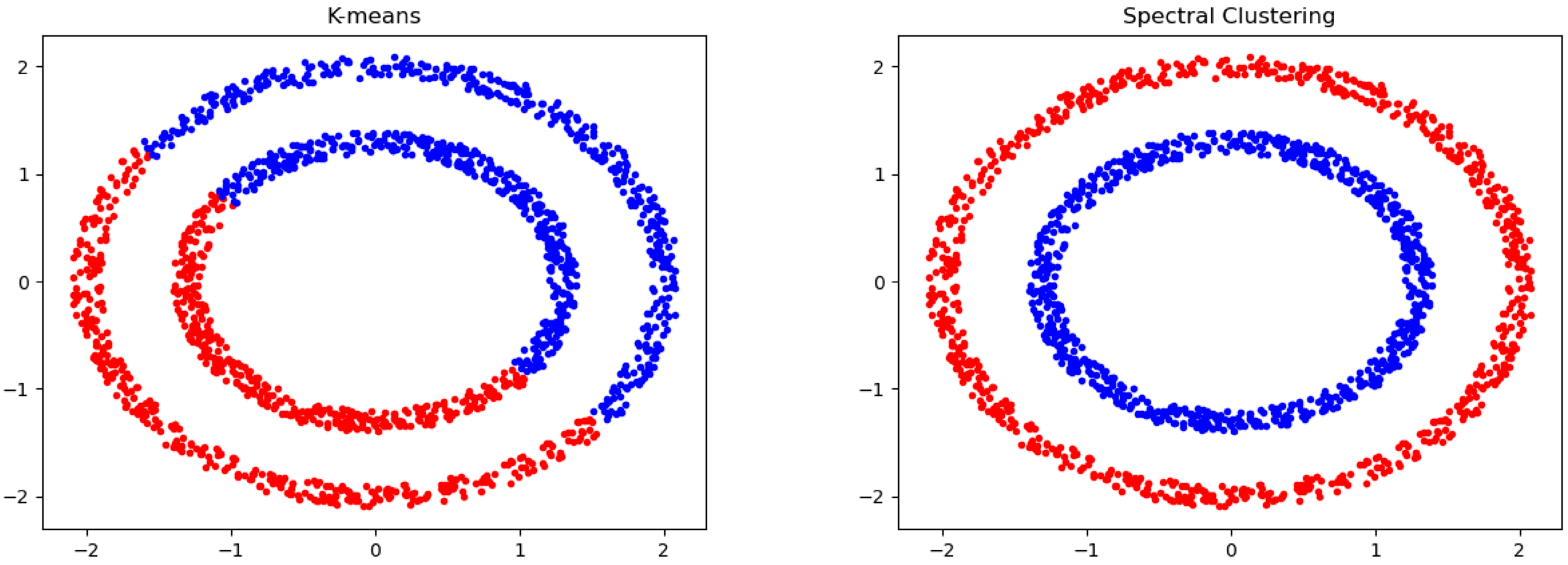


What’s a Reasonable Similarity Measure?

points similar
=
can reach each other by small jumps through data cloud



Comparison: K-means v.s. Spectral Clustering



Registration

- **Friday** is the registration deadline.
- Work on **Homework 0** earlier to check if you feel comfortable with the prerequisites, and get yourself familiar with Gradescope and autograder.
- If you decide to drop the course, please do so asap so that other people on the waitlist have time to register!
- See you next week!