

Computational Data Analysis

Machine Learning

Yao Xie, Ph.D.

Associate Professor

Harold R. and Mary Anne Nash Early Career Professor
H. Milton Stewart School of Industrial and Systems
Engineering

Spectral Clustering



General formulation of clustering

- Given m data points, $\{x^1, x^2, \dots, x^m\} \in R^n$
- Find k cluster centers, $\{c^1, c^2, \dots, c^k\} \in R^n$
- And assign each data point i to one cluster, $\pi(i) \in \{1, \dots, k\}$
- Such that the sum of the squared distances from each data point to its respective cluster center is minimized

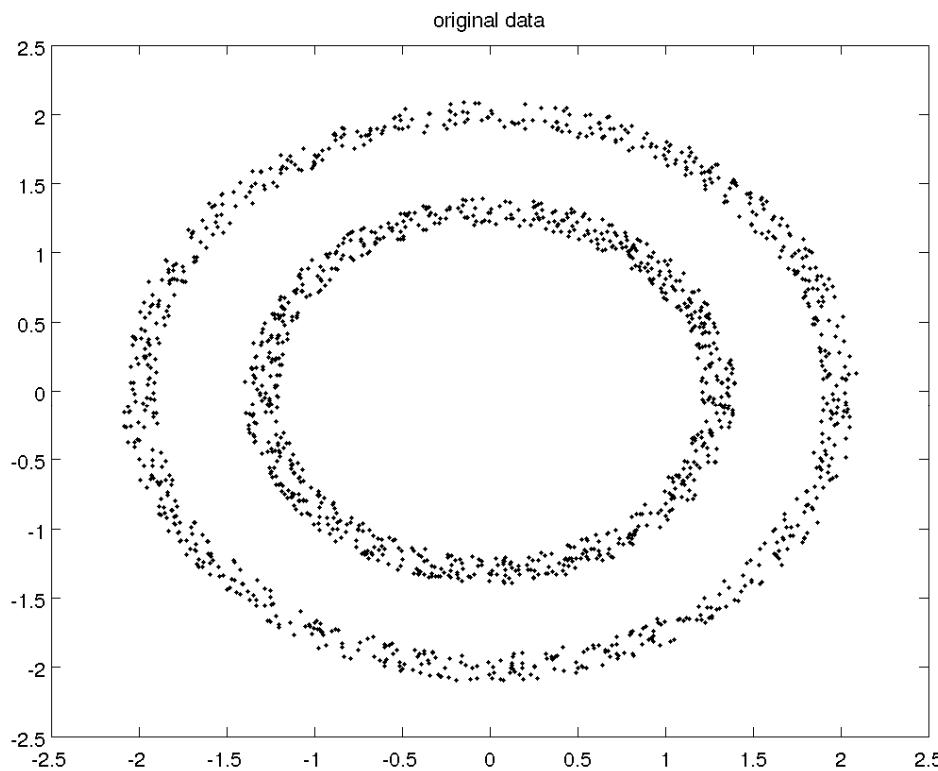
$$\min_{c,\pi} \sum_{i=1}^m d(x^i, c^{\pi(i)})^2$$



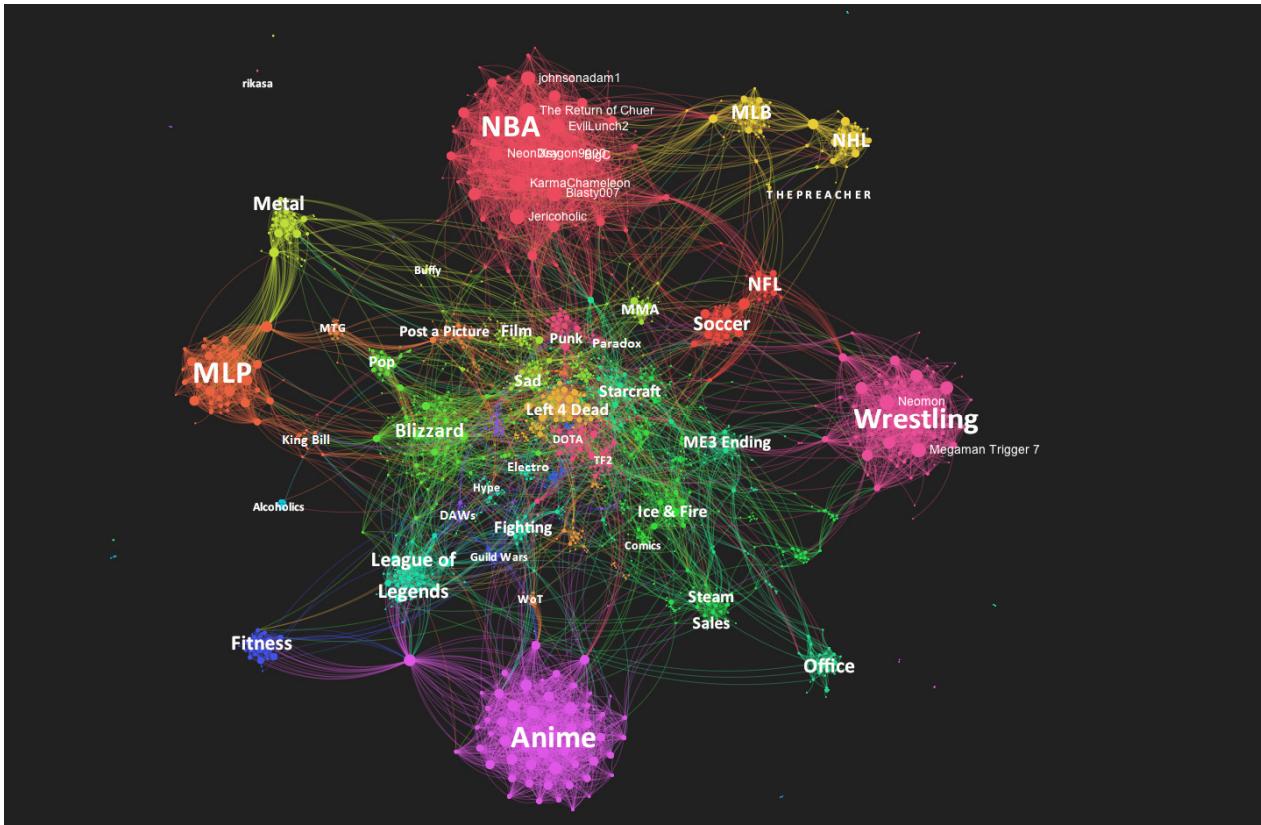
K-means algorithm

- 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**)
$$\pi(i) = \operatorname{argmin}_{j=1,\dots,k} \|x^i - c^j\|^2$$
 - Adjust the cluster centers (**center adjustment**)
$$c^j = \frac{1}{|\{i: \pi(i) = j\}|} \sum_{i: \pi(i)=j} x^i$$
- While any cluster center has been changed

How about this dataset? (Run `test_tworings.m`)



How about clustering nodes in social networks

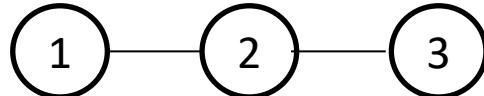


Representing graph using matrices

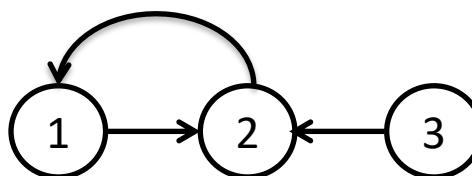
- Adjacency matrix for unweighted graph

$$A_{ij} = \begin{cases} 1, & \text{if there is an edge from } i \text{ to } j \\ 0, & \text{otherwise.} \end{cases}$$

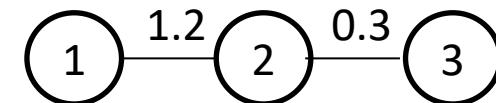
- Directed vs. undirected graph
- Weighted vs. unweighted graph



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



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



$$A = \begin{bmatrix} 0 & 1.2 & 0 \\ 1.2 & 0 & 0.3 \\ 0 & 0.3 & 0 \end{bmatrix}$$

Graph Laplacian (undirected graph)

- For a graph with m nodes, adjacency matrix $A \in R^{m \times m}$
- Vertex degree $d_i = \sum A_{ij}$ (for unweighted graph: the number of neighboring nodes)
- Degree matrix

$$D = \text{diag}\{d_1, d_2, \dots, d_m\}$$

- Graph Laplacian L is positive semi-definite (meaning all the eigenvalues are non-negative)

$$L = D - A$$

- measuring to what extent a graph differs at one vertex from its values at nearby vertices.

Graph Laplacian Example



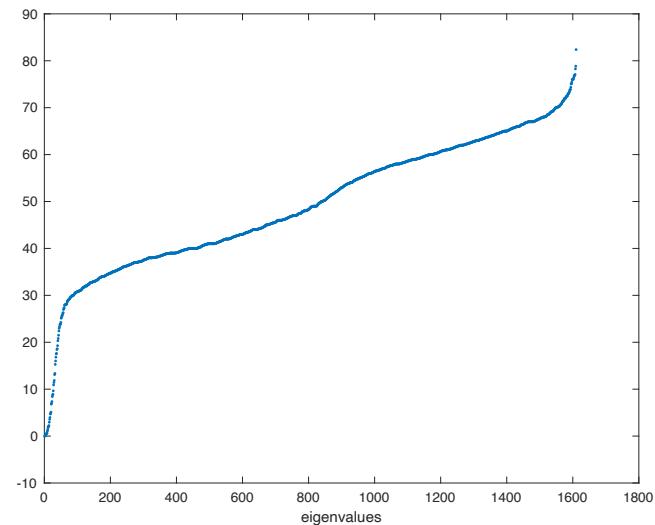
$$A = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

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

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

Eigenvalue problem

- Given a symmetric matrix $C \in R^{n \times n}$
 - Find a vector $u \in R^n$ and $\|u\| = 1$
 - Such that
$$Cu = \lambda u$$
- There will be multiple solution: $u^1, u^2, \dots u^n$ (called the **eigenvectors**) with different $\lambda_1, \lambda_2, \dots \lambda_n$ (called the **eigenvalues**.)
 - Eigenvectors are ortho-normal:
 $u^i{}^\top u^i = 1, u^i{}^\top u^j = 0$
 - Eigenvalues are called spectrum

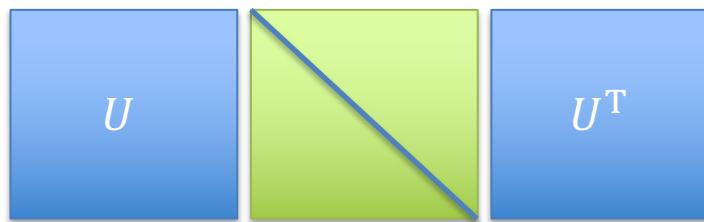


Eigenvalues of graph Laplacian
of the "two-ring" data

Eigendecomposition

- Given a symmetric matrix $C \in R^{n \times n}$
- Eigendecomposition

$$C = U\Lambda U^T$$



- Example: $C = \begin{bmatrix} 1 & 1 & 1 \\ 2 & 3 & 2 \\ 3 & 6 & 3 \end{bmatrix}$
- $U = \begin{bmatrix} -0.2673 & -0.0455 & -0.3135 \\ -0.5345 & 0.7288 & -0.4961 \\ -0.8018 & -0.6832 & 0.8097 \end{bmatrix}, \Lambda = \begin{bmatrix} 7.2749 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -0.2749 \end{bmatrix}$

Property I of Graph Laplacian

- $L = D - A$

- The multiplicity of the eigenvalue 0 corresponds to the number of connected components in the graph

- Example



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

$$Lv_1 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & -1 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$Lv_2 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & -1 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Property II of Graph Laplacian

- $L = D - A$
- The eigenvectors with eigenvalue 0 contains cluster assignment information

- Example

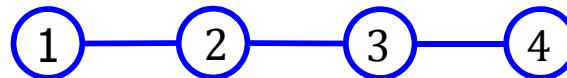


$$L = \begin{pmatrix} 1 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & -1 & 1 \end{pmatrix}$$

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

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

What if the graph has only 1 component



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

$$D = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$L = \begin{pmatrix} 1 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 1 \end{pmatrix}$$

Special eigenvector with all 1's

- $L = D - A$
- The smallest eigenvalue of L is 0, corresponding a constant eigenvector $\frac{1}{\sqrt{m}} \mathbf{1}$
- Example



$$L = \begin{pmatrix} 1 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 1 \end{pmatrix}$$

$$\frac{1}{\sqrt{4}} L \mathbf{1} = \frac{1}{\sqrt{m}} \begin{pmatrix} 1 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} = \frac{1}{\sqrt{m}} \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} = 0 \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

What if the graph has k components

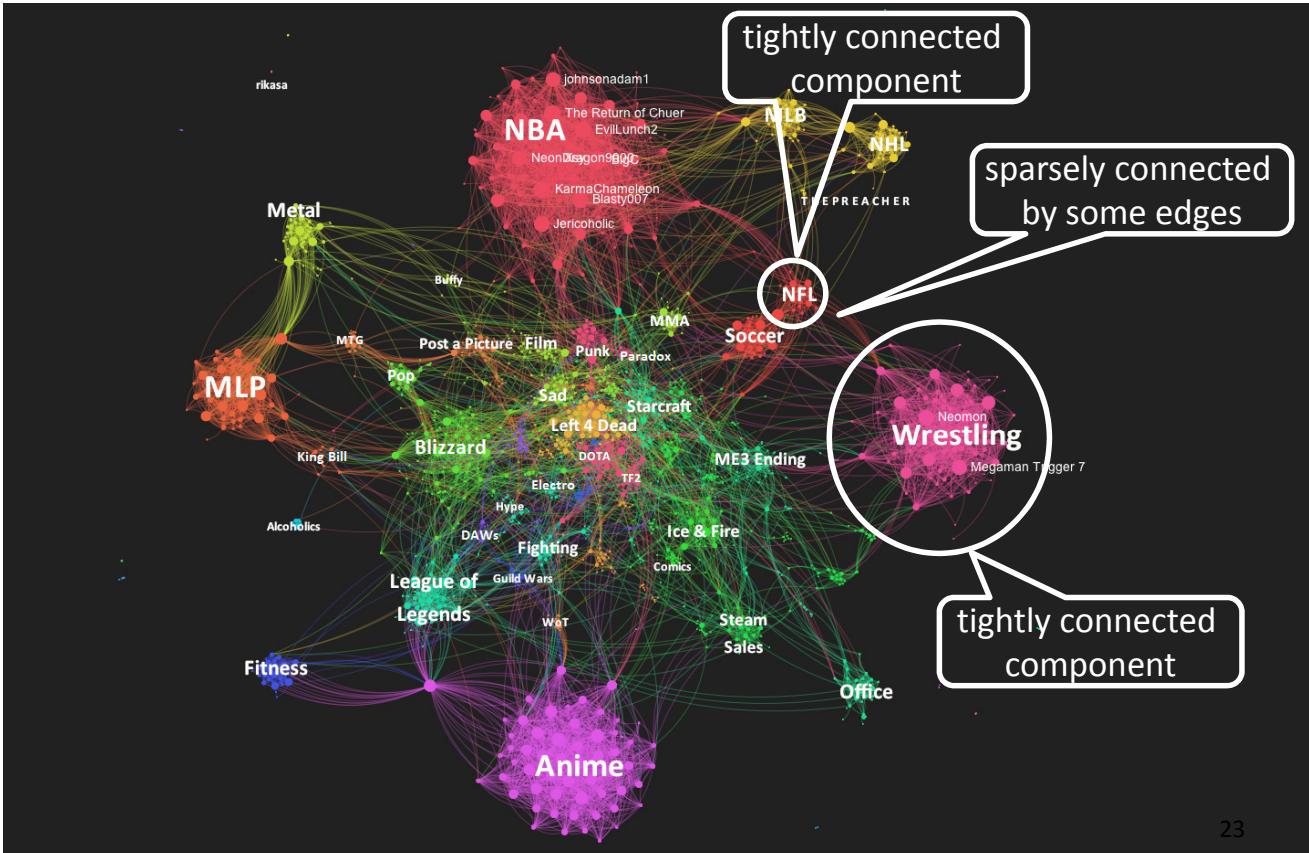
- If a graph has k connected components (or k clusters)
- The graph Laplacian has k blocks

$$L = \begin{pmatrix} L_1 & 0 & 0 & 0 \\ 0 & L_2 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & L_k \end{pmatrix}$$

- The graph Laplacian has k eigenvectors with zero eigenvalues
- Eigenvector 1 is constant in block 1, but 0 in other blocks;
eigenvector 2 is constant in block 2, but 0 in other blocks;

...

Real world not perfectly block



High level idea of spectral clustering

- Examine the properties of graph Laplacian for the perfect cases
 - The number of 0 eigenvalues corresponds to the number of connected components
 - Eigenvectors correspond to cluster assignment
- Then use the intuition from perfect cases to design algorithms for the imperfect case.
 - Eigenvectors no longer correspond exactly cluster indicator
 - Perform post processing to obtain cluster assignment

In general (imperfect case)

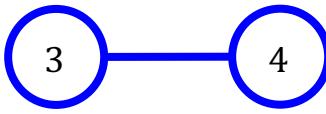
- If a graph has k **tightly** connected components (or k clusters) with **sparsely** connected edges
- The graph Laplacian has **approximately** k blocks
- The graph Laplacian has k eigenvectors with **small** eigenvalues
- Eigenvector 1 is **approximately** constant in block 1, but 0 in other blocks; eigenvector 2 ...

Ideas of spectral clustering

- Step 1: represent graph as adjacency matrix $A \in R^{m \times m}$
- Step 2: form a special matrix $L = D - A$, the graph Laplacian
- Step 3: compute k eigenvectors, v^1, v^2, \dots, v^k , of L corresponding to the k **smallest** eigenvalues ($k \ll m$)
- Step 4: run kmeans algorithm on $Z = (v^1, v^2, \dots, v^k)$ by treating each row as a new data point

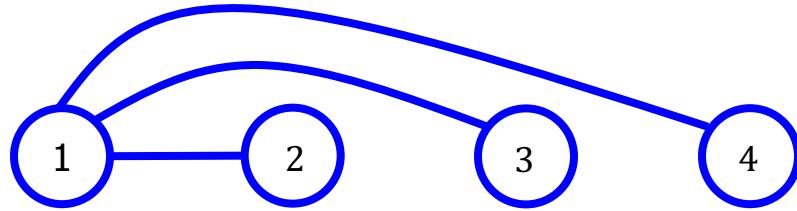
Spectral clustering algorithm

Step 1: represent graph as adjacency matrix $A \in R^{m \times m}$, m : number of nodes



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

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



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

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

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

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

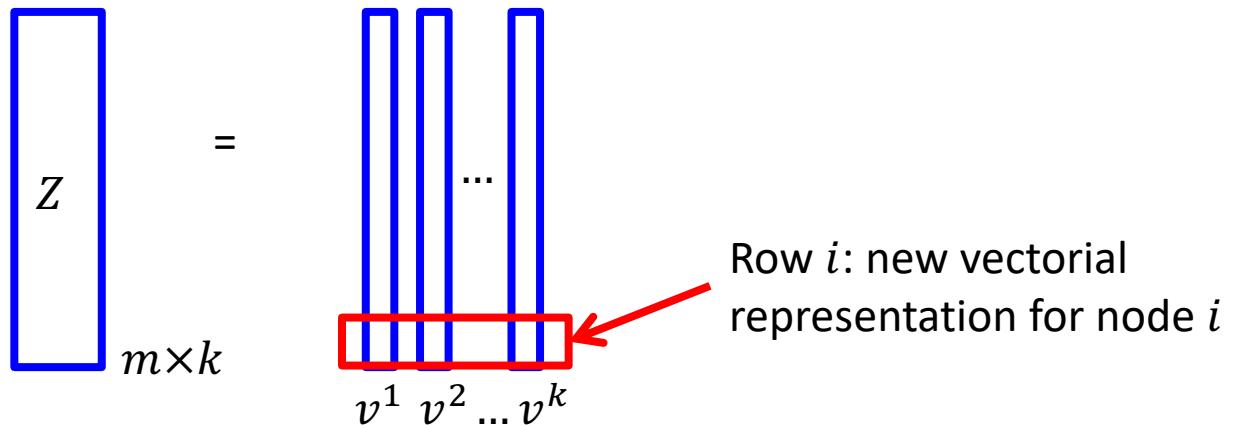
$$L = \begin{pmatrix} 3 & -1 & -1 & -1 \\ -1 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{pmatrix}$$

Spectral clustering algorithm (cont.)

Step 3: Perform eigendecomposition of **graph Laplacian** L , compute k eigenvectors, v^1, v^2, \dots, v^k , corresponding to the k **smallest** eigenvalues ($k \ll m$)

$$L v^1 = \begin{pmatrix} 1 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & -1 & 1 \end{pmatrix} v^1 = \lambda_1 v^1$$

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



Questions

- Similarity in spectral clustering is based on
 - Euclidean distance
 - Connectivity
- How to pick the number eigenvectors?
 - Random
 - Look at the eigengap

Run demo test_football.m

PLAY FANTASY The Most Award Winning Fantasy game with real time scoring, top expert analysis, custom settings, and more. **PLAY NOW**

NCAA FB Home Scores Standings Schedules Stats Teams Players Rankings Picks Recruiting Signing Day

NCAA FB SCORES

24 MIZZOU TOLEDO	Sat 12:00 pm	FAU 2 BAMA	Sat 12:00 pm	20 KSTATE IOWAST	Sat 12:00 pm	MCNST 19 NEB	Sat 12:00 pm	4 OKLA TULSA	Sat 12:00 pm	V 18	FULL NCAA FB SCOREBOARD
------------------	--------------	------------	--------------	------------------	--------------	--------------	--------------	--------------	--------------	------	-------------------------

MAYWEATHER VS MADAINA 2 SAT SEPT 13 8PM/5PM LIVE ON PAY-PER-VIEW FROM MGM GRAND CLICK TO ORDER xfinity ROLLOVER FOR MORE INFO

COLLEGE FOOTBALL SCHEDULES

FBS FCS

By Week 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9 - 10 - 11 - 12 - 13 - 14 - 15 - 16

BUY COLLEGE FOOTBALL TICKETS

WEEK 1

SATURDAY, AUG. 23

GAME	TIME/SCORE	TV	LOCATION/TICKETS
Sam Houston St. at E. Washington	Eastern Washington 56-35	ESPN	Woodward Stadium

WEDNESDAY, AUG. 27

GAME	TIME/SCORE	TV	LOCATION/TICKETS
Aubit Chr. at Georgia State	Georgia State 38-37	ESPNNU	Georgia Dome

THURSDAY, AUG. 28

GAME	TIME/SCORE	TV	LOCATION/TICKETS
Texas A&M at South Carolina	Texas A&M 52-28	SEC Network	Williams-Brice Stadium
E. Illinois at Minnesota	Minnesota 42-20	Big Ten Network	TCF Bank Stadium
Presbyterian at Northern Illinois	Northern Illinois 55-3		Huskie Stadium
Missouri St. at Northwestern St.	Missouri State 34-27		Turpin Stadium
Bryant at Stony Brook	Bryant 13-7		
Wake Forest at La-Monroe	Louisiana-Monroe 17-10	ESPNNU	Malone Stadium
Chattanooga at C. Michigan	Central Michigan 20-16		Kelly Shorts Stadium
Howard at Akron	Akron 41-0		InfoCision Stadium - Summa Field
Charlotte at Campbell	Charlotte 33-9		Barker-Lane Stadium
Reinhardt at Mercer	Merger 45-42		Moye Complex
E. Kentucky at Robert Morris	Eastern Kentucky 29-10		Joe Walton Stadium
Point U at Charleston So.	Charleston Southern 61-9		CSU Field
Missouri Baptist at SE Missouri St.	Southeast Missouri State 77-0		Houck Stadium
Idaho State at Utah	Utah 56-14	PAC-12 Network	Rice Eccles Stadium
Valparaiso at W. Illinois	Western Illinois 45-6		Hanson Field
Boise St. at Ole Miss	Ole Miss 35-13	ESPN	Georgia Dome
Kentucky Chr. at Tenn. Tech	Tennessee Tech 33-7		Tucker Stadium

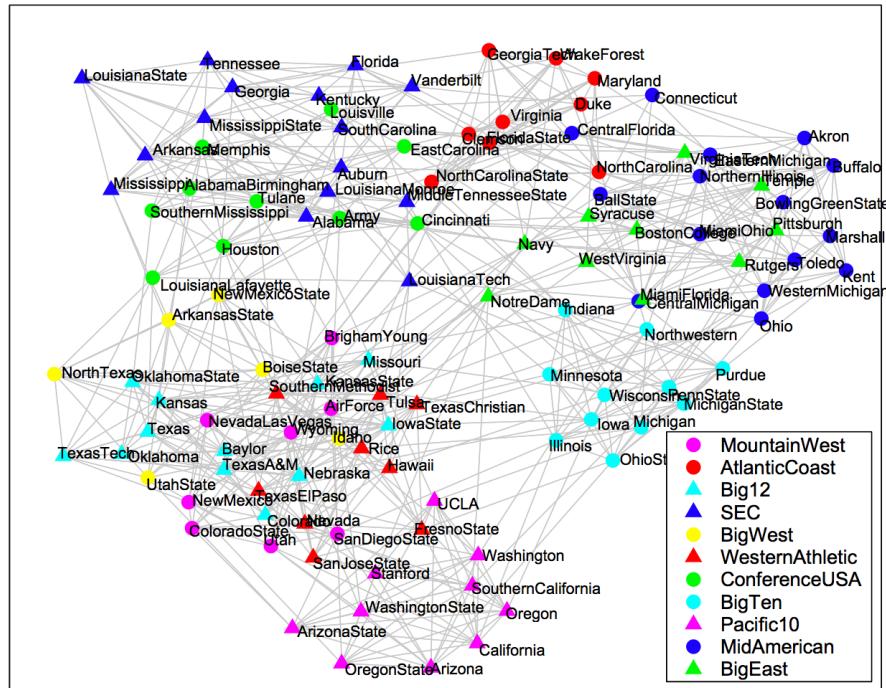
T-Mobile

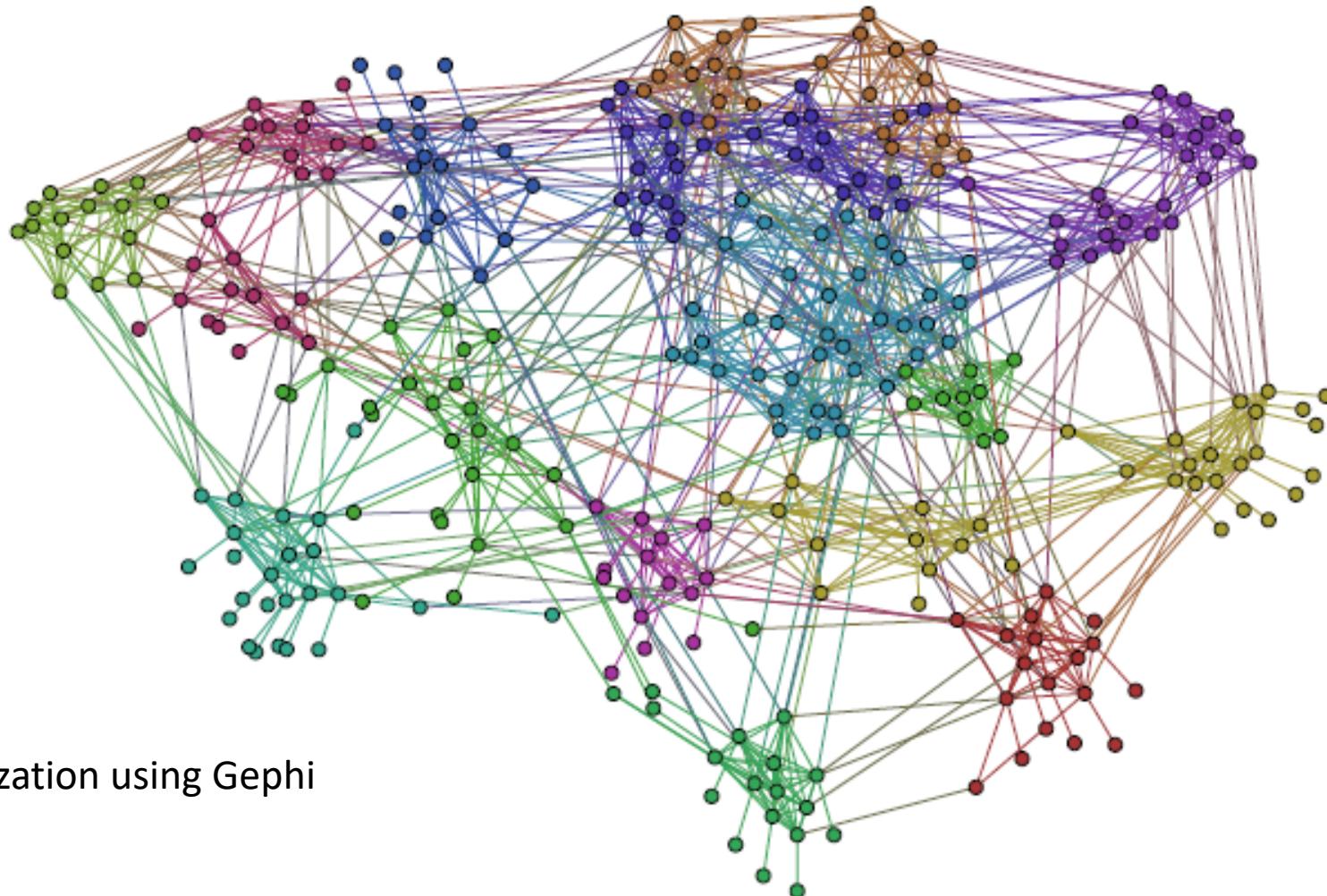
BRING YOUR OWN PHONE TO T-MOBILE

SWITCH NOW

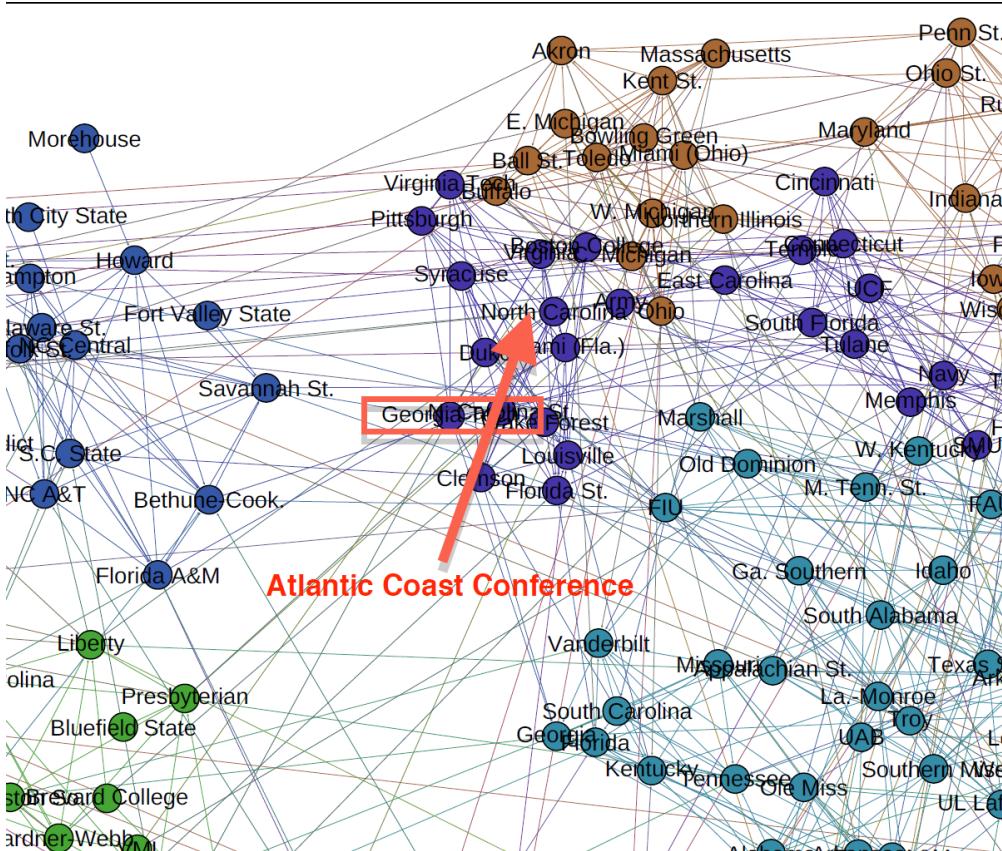
Capable device required. Qualifying service plan required.

CBSSPORTS.COM SHOP

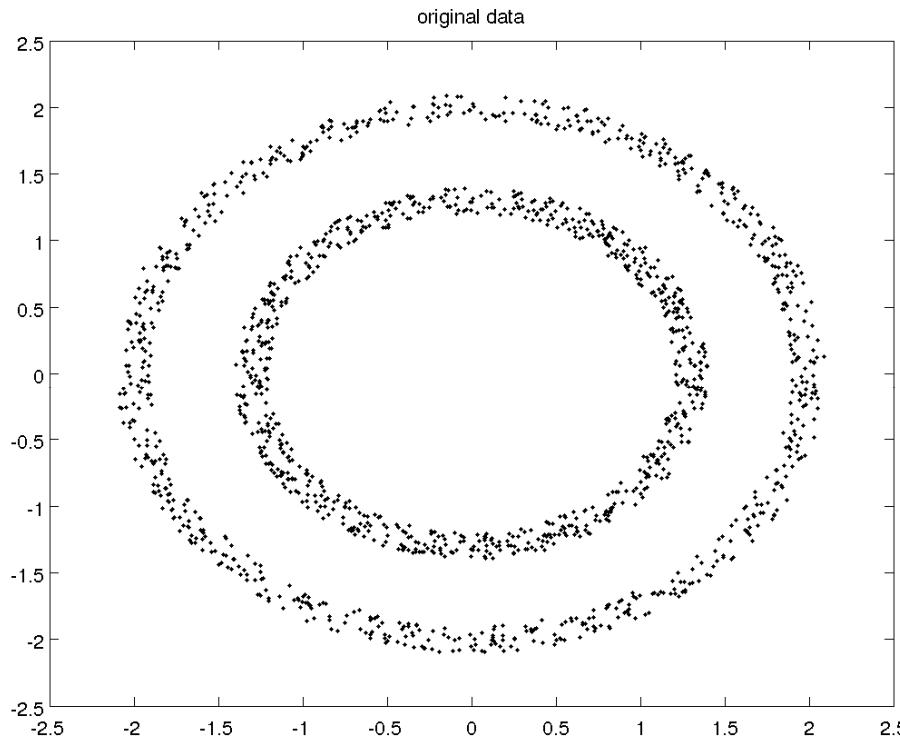





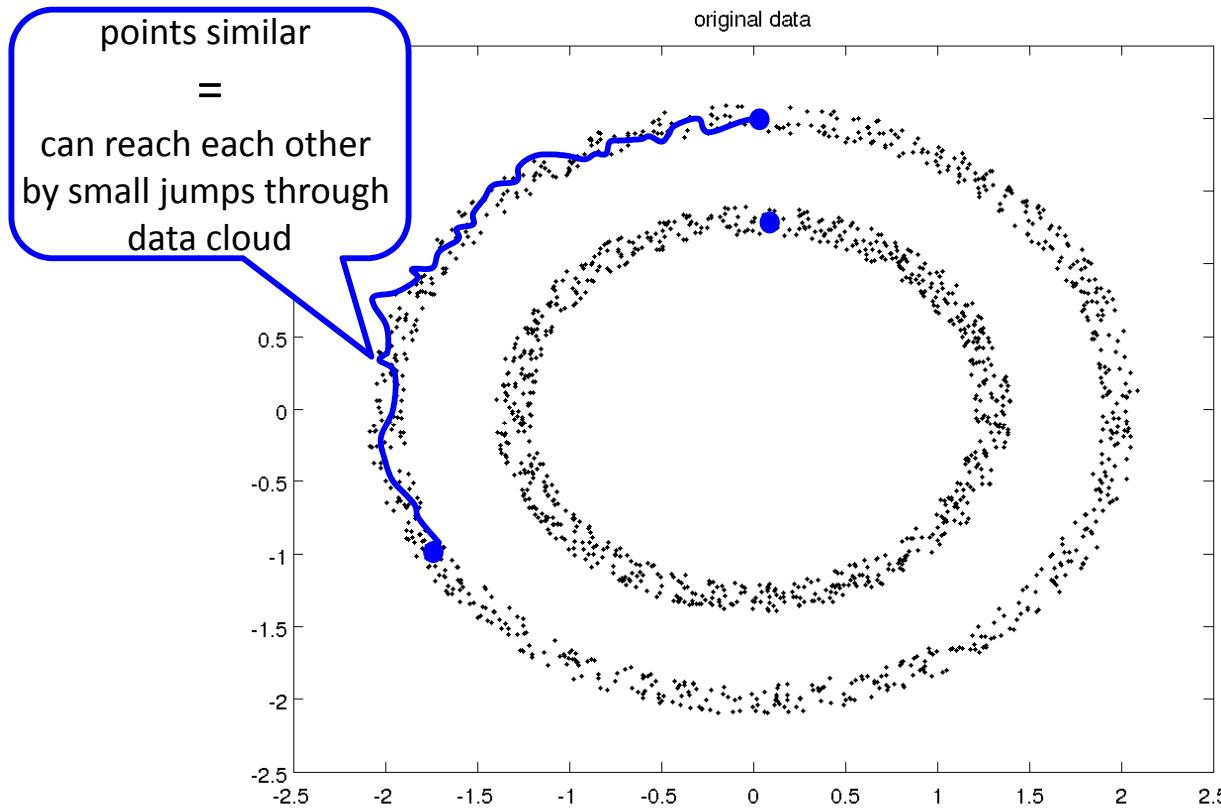
Visualization using Gephi



How about this dataset?



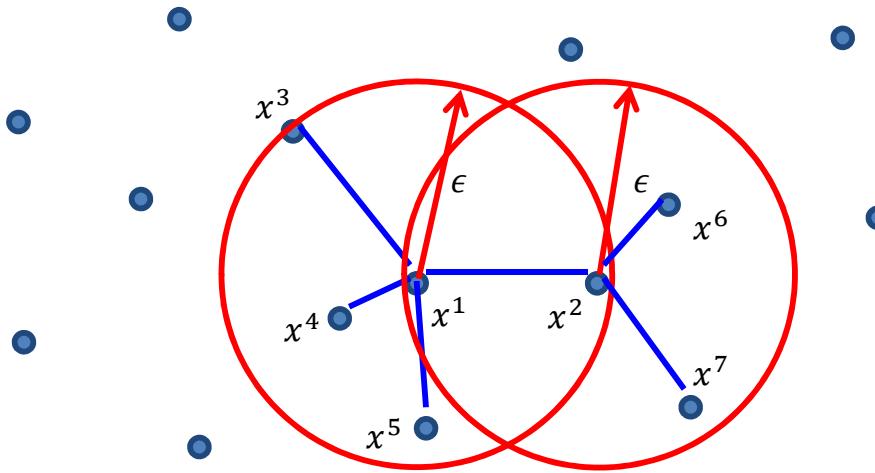
What's a reasonable similarity measure?



Nearest neighbor graph

- Given m data points, threshold ϵ , construct matrix $A \in R^{m \times m}$

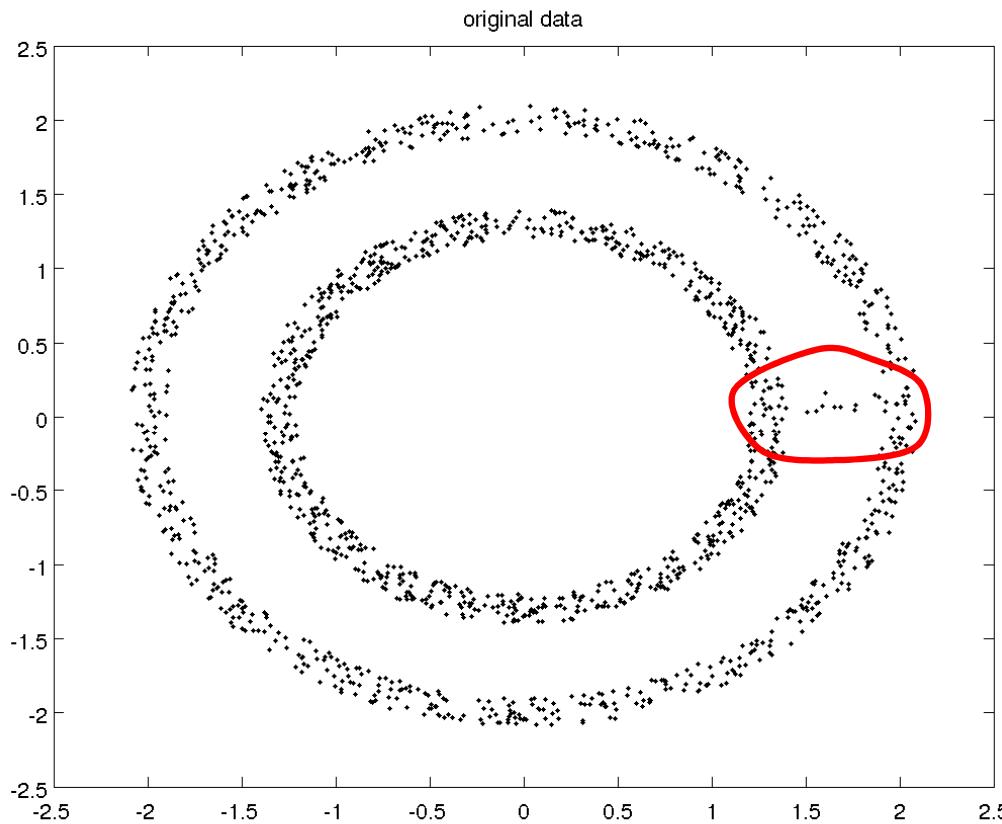
$$A^{ij} = \begin{cases} 1, & \text{if } \|x^i - x^j\| \leq \epsilon \\ 0, & \text{otherwise} \end{cases}$$



Spectral clustering for vector data

- Given m nodes, $\{x^1, x^2, \dots, x^m\} \in R^n$
- Step 1: build an adjacency matrix A using nearest neighbors
- Step 2: represent graph as adjacency matrix $A \in R^{m \times m}$
- Step 3: form a special matrix $L = D - A$, the graph Laplacian
- Step 4: compute k eigenvectors, v^1, v^2, \dots, v^k , of L corresponding to the k **smallest** eigenvalues ($k \ll m$)
- Step 5: run kmeans algorithm on $Z = (v^1, v^2, \dots, v^k)$ by treating each row as a new data point

What happens by adding more data points?



Variants of spectral clustering

- Given m data points (nodes), $\{x^1, x^2, \dots, x^m\} \in R^n$
- Build an adjacency matrix A using **kernel functions** (if the input is already a graph, skip this step)
- Compute $B = D^{-1/2}AD^{-1/2}$, where D is the degree matrix
- Compute k eigenvectors, v^1, v^2, \dots, v^k , of B corresponding to the k **largest** eigenvalues
- Use $z^1 = (v_1^1, v_1^2, \dots, v_1^k), z^2 = (v_2^1, v_2^2, \dots, v_2^k) \dots$ as the new coordinates for data point 1, 2, ..., and then run kmeans on these new coordinates

