Spectral graph clustering

Introduction to Network Science
Carlos Castillo
Topic 23

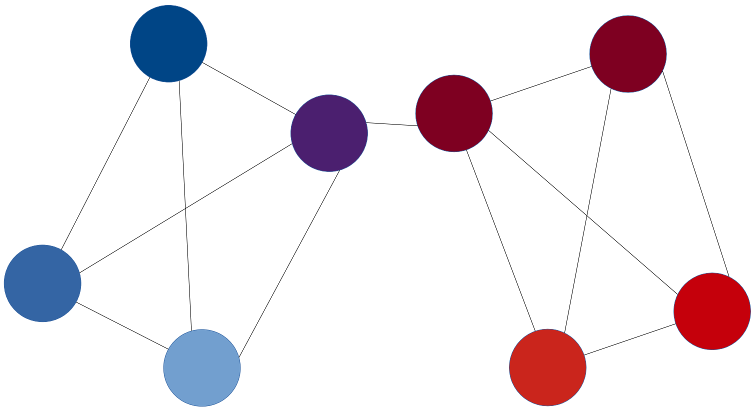
Universitat Pompeu Fabra Barcelona

Sources

- **Jure Leskovec (2016)**
  Defining the graph laplacian [video]
- Evimaria Terzi: Graph cuts / spectral graph partitioning
- Daniel A. Spielman (2009): The Laplacian
- C. Castillo (2017) Graph partitioning

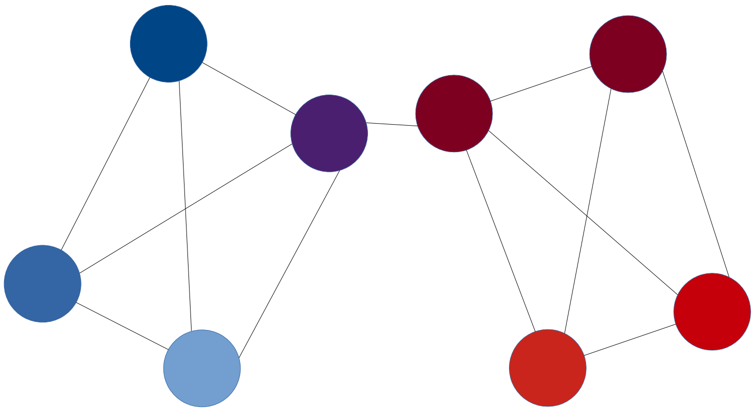
Graphs are nice, but ...

- They describe only local relationships
- We would like to understand a global structure
- Our objective is transforming a graph into a more familiar object: a cloud of points in R^k



Graphs are nice, but …

- They describe only local relationships
- We would like to understand a global structure
- Our objective is transforming a graph into a more familiar object: a cloud of points in R^k



Distances should be somehow preserved

What is a graph embedding?

- A graph embedding is a mapping from a graph to a vector space
- If the vector space is $\mathbb{R}^2$ you can think of an embedding as a way of drawing a graph

Try drawing this graph

V = {v1, v2, …, v8}

E = { (v1, v2), (v2, v3), (v3, v4), (v4, v1), (v5, v6), (v6, v7), (v7, v8), (v8, v5), (v1,v5), (v2, v6), (v3, v7), (v4, v8) }

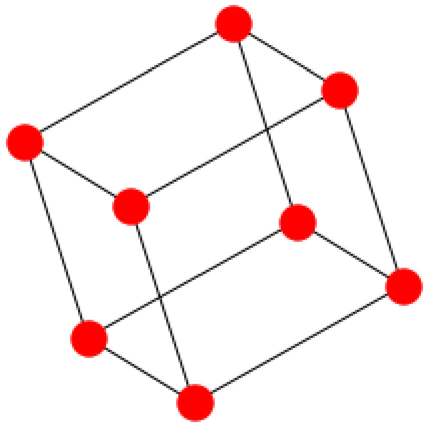
Draw this graph on paper

What constitutes a good drawing?

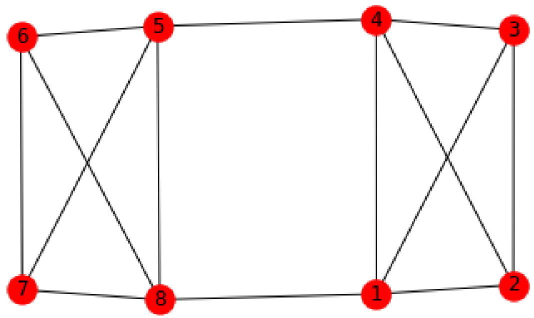
2D graph embeddings in NetworkX

python

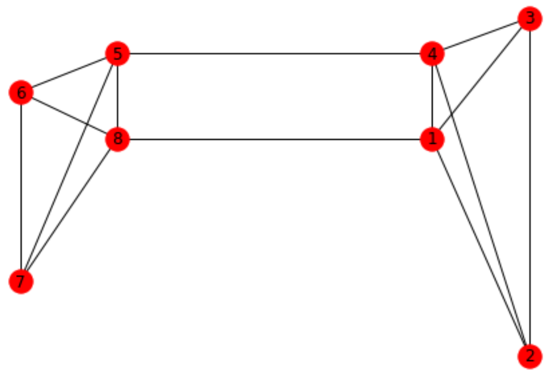
```
import matplotlib.pyplot as plt
import networkx as nx

plt.figure(figsize=(3,3))
G = nx.hypercube_graph(3)
nx.draw_spectral(G)
_ = plt.show()
```



nx.draw_networkx(g)



nx.draw_spectral(g)



In a good graph embedding …

- Pairs of nodes that are connected to each other should be close
- Pairs of nodes that are not connected should be far
- Compromises usually need to be made

Random 2D graph projection

- Start a BFS from a random node, that has $x=1$, and nodes visited have ascending x
- Start a BFS from another random node, which has $y=1$, and nodes visited have ascending y
- Project node i to position (x_i, y_i)

How do you think this works in practice?

Eigenvectors of adjacency matrix

Adjacency matrix of G=(V,E)

$$A_{ij} = \begin{cases} 1 & \text{if } (i,j) \in E \\ 0 & \text{otherwise} \end{cases}$$

- What is Ax? Think of x as a set of labels/values:

$$\begin{bmatrix} a_{11} & \cdots & a_{1n} \\ a_{21} & \cdots & a_{2n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

$$y_i = \sum_{j:(j,i)\in E} x_j$$

Ax is a vector whose i[th] coordinate contains the sum of the x_j who are in-neighbors of i

https://www.youtube.com/watch?v=AR7iFxM-NkA

Properties of adjacency matrix

$$A_{ij} = \begin{cases} 1 & \text{if } (i,j) \in E \\ 0 & \text{otherwise} \end{cases}$$

- How many non-zeros are in every row of A?

$$\begin{bmatrix} a_{11} & \cdots & a_{1n} \\ a_{21} & \cdots & a_{2n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{bmatrix}$$

https://www.youtube.com/watch?v=AR7iFxM-NkA

Understanding the adjacency matrix

$$A_{ij} = \begin{cases} 1 & \text{if } (i,j) \in E \\ 0 & \text{otherwise} \end{cases}$$

What is Ax? Think of x as a set of labels/values:

$$\begin{bmatrix} a_{11} & \cdots & a_{1n} \\ a_{21} & \cdots & a_{2n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

Ax is a vector whose i^{th} coordinate contains the sum of the x_j who are in-neighbors of i

$$y_i = \sum_{j:(j,i)\in E} x_j$$

https://www.youtube.com/watch?v=AR7iFxM-NkA

Spectral graph theory

- The study of the eigenvalues and eigenvectors of a graph matrix
  - Adjacency matrix $Ax = \lambda x$
  - Laplacian matrix (next)

- Suppose graph is d-regular, $k_i = d \; \forall i$ what is the value of:

$$\begin{bmatrix} a_{11} & \cdots & a_{1n} \\ a_{21} & \cdots & a_{2n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} = ?$$

- What does that imply?

An easy eigenvector of A

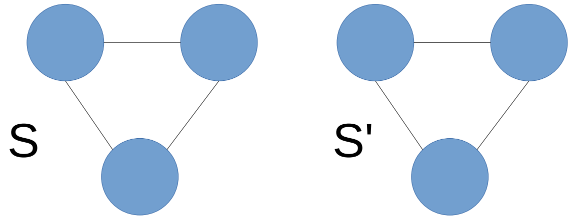
- Suppose graph is d-regular, i.e. all nodes have degree d:

$$\begin{bmatrix} a_{11} & \cdots & a_{1n} \\ a_{21} & \cdots & a_{2n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} = \begin{bmatrix} d \\ d \\ \vdots \\ d \end{bmatrix} = d \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}$$

- **So $[1, 1, \ldots, 1]^T$ is an eigenvector of eigenvalue d**

Disconnected graphs

- Suppose graph is disconnected (still regular)

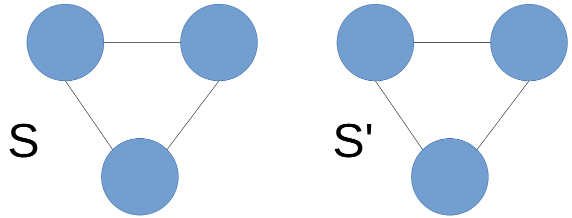


- Then its adjacency matrix has block structure:

$$A = \begin{bmatrix} S & 0 \\ 0 & S' \end{bmatrix}$$

Disconnected graphs

- Suppose graph is disconnected (still regular)

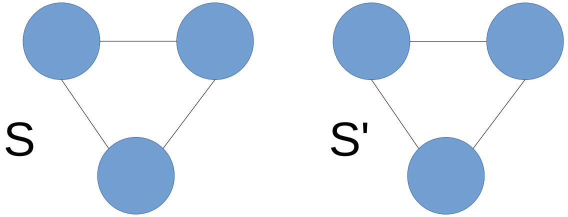


$$\text{Let } x_i^S = \begin{cases} 1 & \text{if } i \in S \\ 0 & \text{if } i \in S' \end{cases}$$

$$\boxed{\begin{bmatrix} S & 0 \\ 0 & S' \end{bmatrix} \begin{bmatrix} 1 \\ \vdots \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} = ?}$$

Disconnected graphs

- Suppose graph is disconnected (still regular)



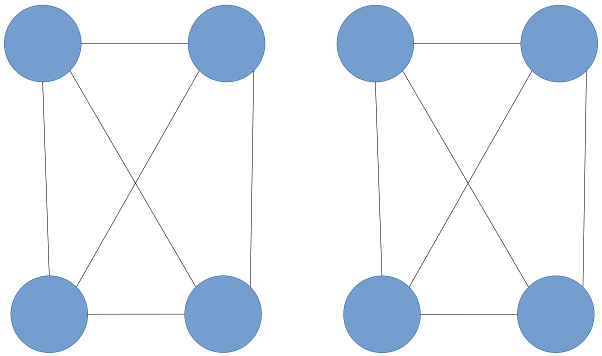
$$Ax^S = dx^S$$

$$Ax^{S'} = dx^{S'}$$

What happens if there are more than 2 connected components?

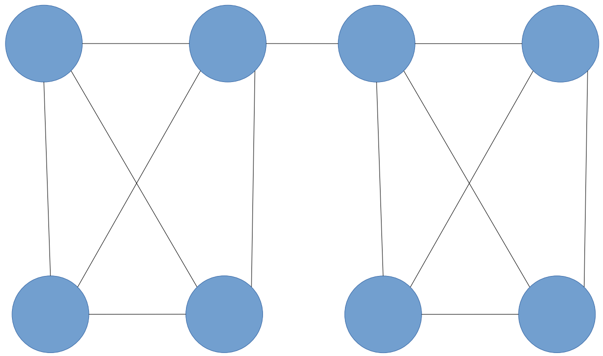
In general

Disconnected graph



$$\lambda_1 = \lambda_2$$

Almost disconnected graph



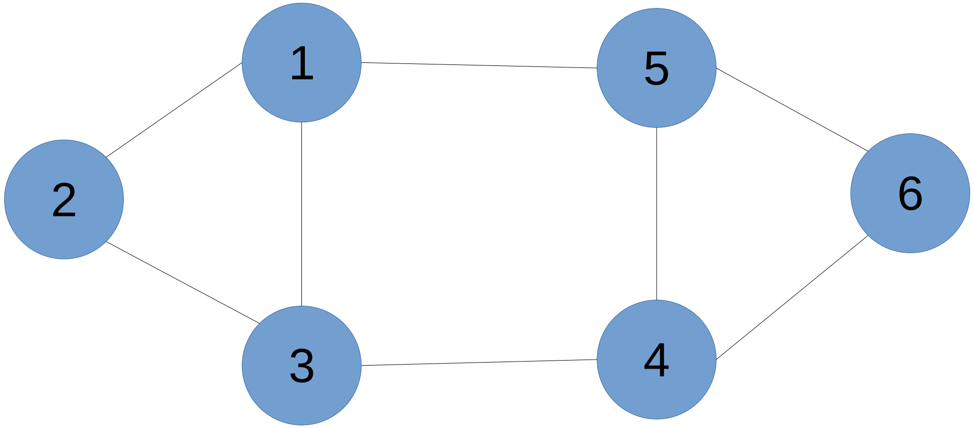
$$\lambda_1 \approx \lambda_2$$

Small "**eigengap**"

Graph Laplacian

Adjacency matrix

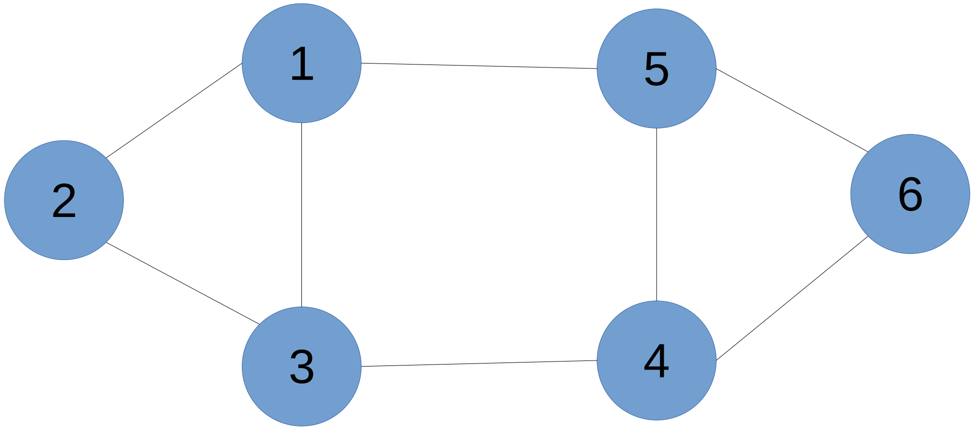
$$A_{ij} = \begin{cases} 1 & \text{if } (i,j) \in E \\ 0 & \text{otherwise} \end{cases}$$



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

Degree matrix

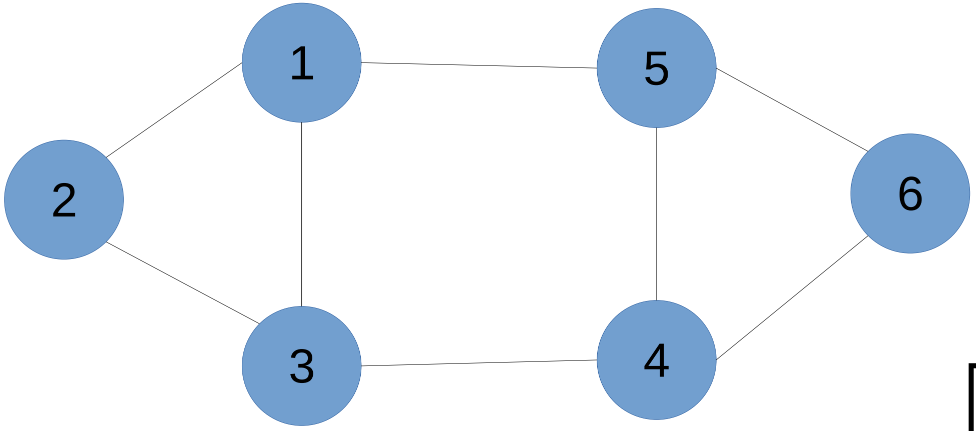
$$D_{ij} = \begin{cases} k_i & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$$



$$D = \begin{bmatrix} 3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 \end{bmatrix}$$

Laplacian matrix

$$L = D - A$$



$$L = \begin{bmatrix} 3 & -1 & -1 & 0 & -1 & 0 \\ -1 & 2 & -1 & 0 & 0 & 0 \\ -1 & -1 & 3 & -1 & 0 & 0 \\ 0 & 0 & -1 & 3 & -1 & -1 \\ -1 & 0 & 0 & -1 & 3 & -1 \\ 0 & 0 & 0 & -1 & -1 & 2 \end{bmatrix}$$

Laplacian matrix L = D - A

- Symmetric
- Eigenvalues non-negative and real
- Eigenvectors real and orthogonal

$$L\vec{1} = \begin{bmatrix} 3 & -1 & -1 & 0 & -1 & 0 \\ -1 & 2 & -1 & 0 & 0 & 0 \\ -1 & -1 & 3 & -1 & 0 & 0 \\ 0 & 0 & -1 & 3 & -1 & -1 \\ -1 & 0 & 0 & -1 & 3 & -1 \\ 0 & 0 & 0 & -1 & -1 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} =?$$

Constant vector is eigenvector of L

- The constant vector x=[1,1,...,1]^T is an eigenvector, and has eigenvalue 0

$$Lx = \begin{bmatrix} 3 & -1 & -1 & 0 & -1 & 0 \\ -1 & 2 & -1 & 0 & 0 & 0 \\ -1 & -1 & 3 & -1 & 0 & 0 \\ 0 & 0 & -1 & 3 & -1 & -1 \\ -1 & 0 & 0 & -1 & 3 & -1 \\ 0 & 0 & 0 & -1 & -1 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = 0 \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

- Is this true **for this graph** or **for any graph**?

If the graph is disconnected

- If there are two connected components, the same argument as for the adjacency matrix applies, and $\lambda_1 = \lambda_2 = 0$
- The multiplicity of eigenvalue 0 is equal to the number of connected components

$$x^T L x$$

Prove this!

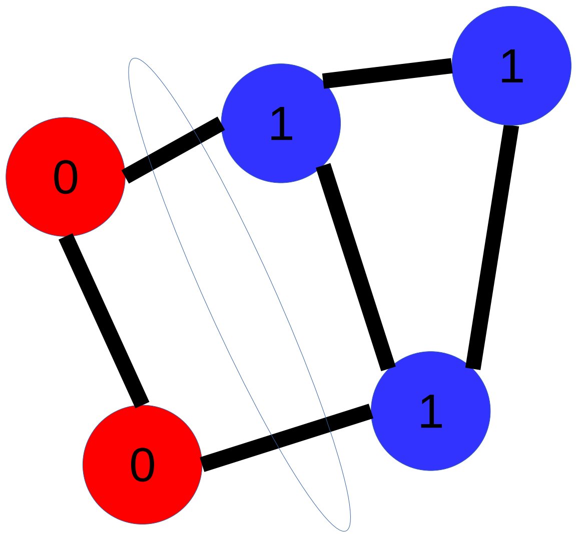
- Prove that $\sum_{(i,j)\in E}(x_i - x_j)^2 = x^T L x$

$$L = D - A$$

$$D_{ij} = \begin{cases} k_i & \text{if } i = j \\ 0 & \text{otherwise} \end{cases} \qquad A_{ij} = \begin{cases} 1 & \text{if } (i,j) \in E \\ 0 & \text{otherwise} \end{cases}$$

$x^T Lx$ and graph cuts

- Suppose (S, S') is a cut of graph G
- Set $x_i = \begin{cases} 1 & \text{if } i \in S \\ 0 & \text{if } i \in S' \end{cases}$



$$|c(S, S')| = 2$$

$$x^T Lx = \sum_{(i,j)\in E} (x_i - x_j)^2 = \sum_{(i,j)\in c(S,S')} 1^2 = |c(S, S')|$$

Important fact

- For symmetric matrices

$$\lambda_2 = \min_x \frac{x^T M x}{x^T x}$$

https://en.wikipedia.org/wiki/Rayleigh_quotient

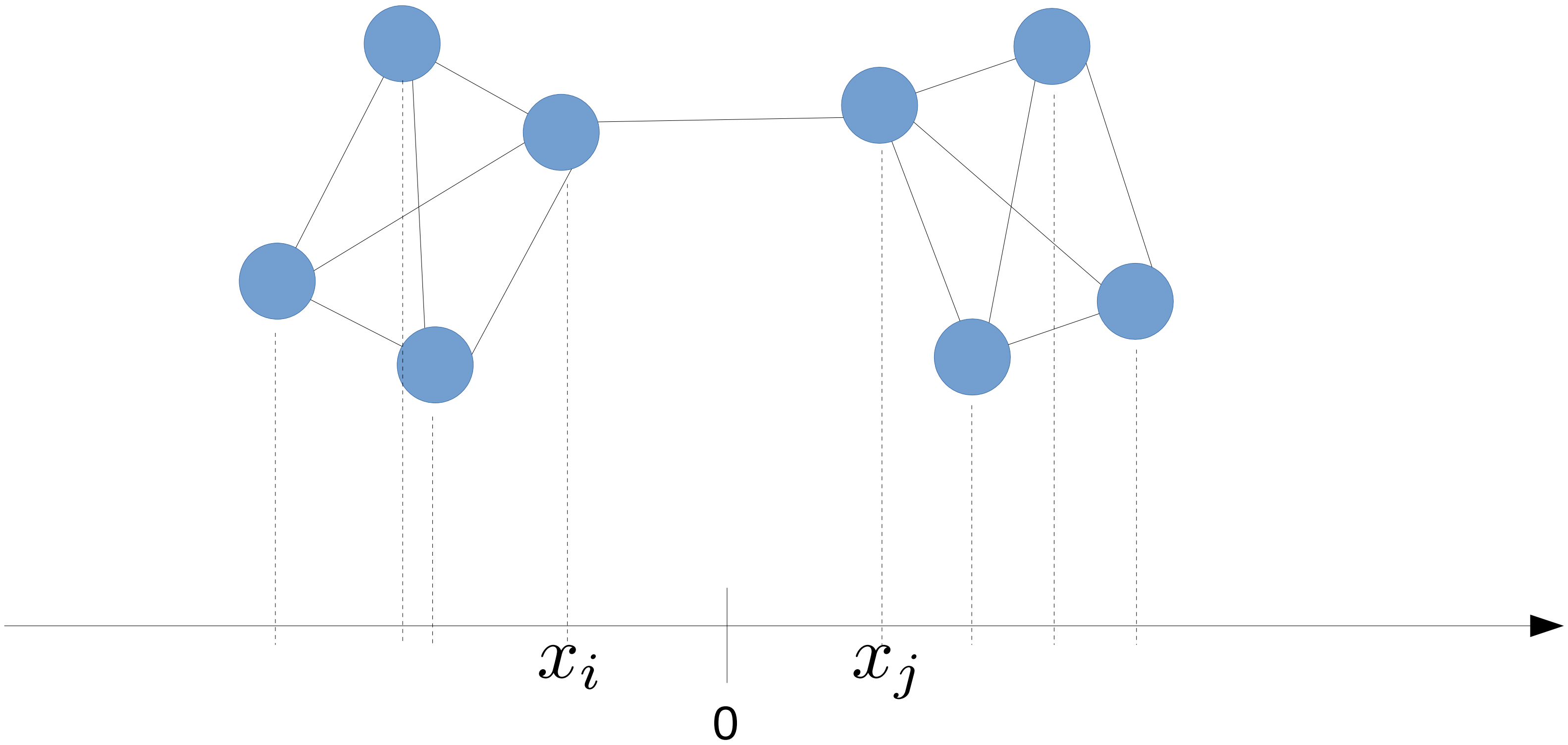
Second eigenvector

- Orthogonal to the first one: $x \cdot \vec{1} = 0 \Rightarrow \sum_i x_i = 0$
- Normal: $\sum_i x_i^2 = 1$

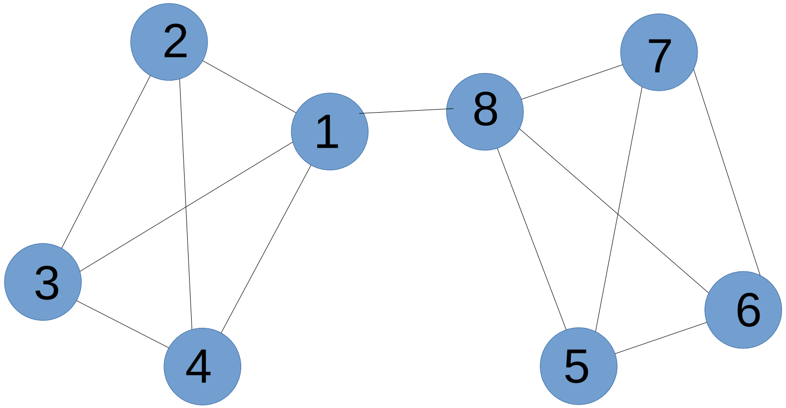
$$\lambda_2 = \min_x \frac{x^T L x}{x^T x} = \min_{x : \sum x_i = 0} \frac{x^T L x}{\sum x_i^2} = \min_{x : \sum x_i = 0} \sum_{(i,j) \in E} (x_i - x_j)^2$$

What does this mean?

$$\lambda_2 = \min_{x:\sum x_i=0} \sum_{(i,j)\in E} (x_i - x_j)^2$$

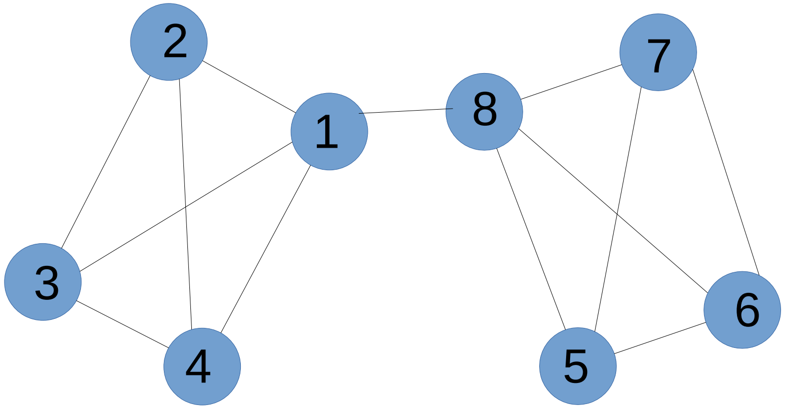


Example Graph 1



$$L = \begin{bmatrix} 4 & -1 & -1 & -1 & 0 & 0 & 0 & -1 \\ -1 & 3 & -1 & -1 & 0 & 0 & 0 & 0 \\ -1 & -1 & 3 & -1 & 0 & 0 & 0 & 0 \\ -1 & -1 & -1 & 3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3 & -1 & -1 & -1 \\ 0 & 0 & 0 & 0 & -1 & 3 & -1 & -1 \\ 0 & 0 & 0 & 0 & -1 & -1 & 3 & -1 \\ -1 & 0 & 0 & 0 & -1 & -1 & -1 & 4 \end{bmatrix}$$

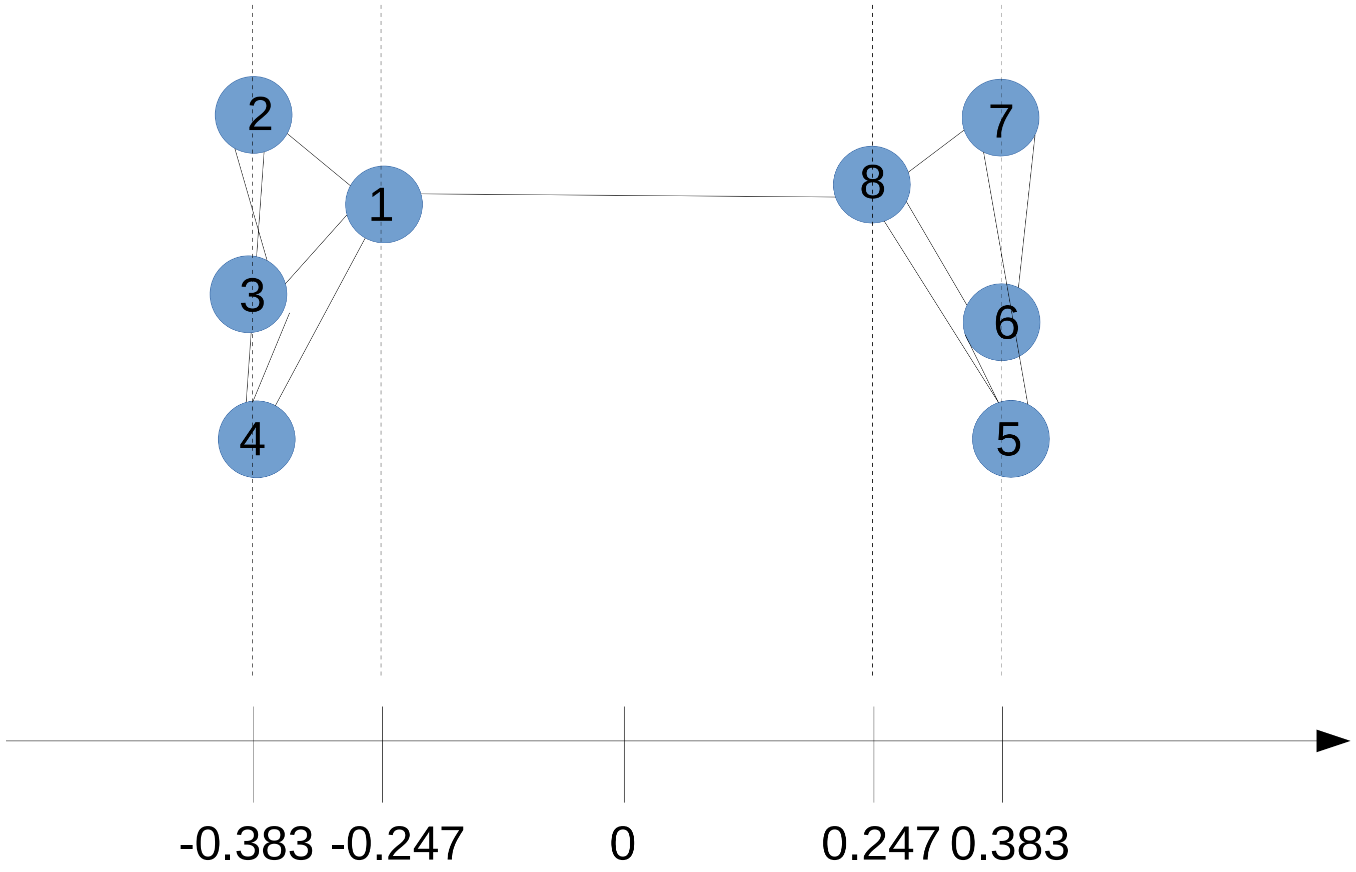
Example Graph 1 (second eigenvalue)



$$\lambda_1 = 0$$
$$\lambda_2 = 0.354$$

$$L = \begin{bmatrix} 4 & -1 & -1 & -1 & 0 & 0 & 0 & -1 \\ -1 & 3 & -1 & -1 & 0 & 0 & 0 & 0 \\ -1 & -1 & 3 & -1 & 0 & 0 & 0 & 0 \\ -1 & -1 & -1 & 3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3 & -1 & -1 & -1 \\ 0 & 0 & 0 & 0 & -1 & 3 & -1 & -1 \\ 0 & 0 & 0 & 0 & -1 & -1 & 3 & -1 \\ -1 & 0 & 0 & 0 & -1 & -1 & -1 & 4 \end{bmatrix} \qquad v_2 = \begin{bmatrix} 0.247 \\ 0.383 \\ 0.383 \\ 0.383 \\ -0.383 \\ -0.383 \\ -0.383 \\ -0.247 \end{bmatrix}$$

Example Graph 1, projected

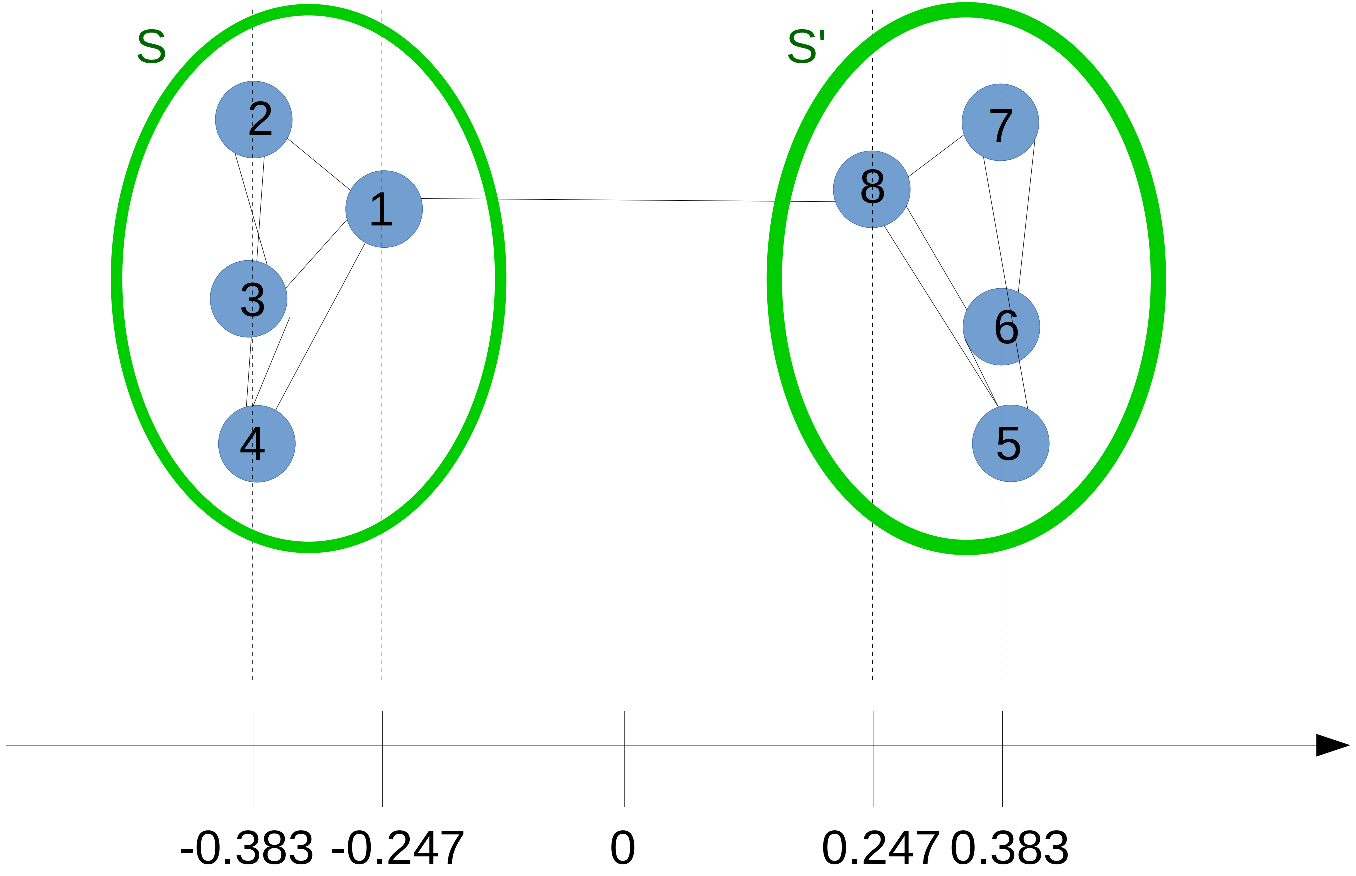


$$\lambda_1 = 0$$

$$\lambda_2 = 0.354$$

$$v_2 = \begin{bmatrix} 0.247 \\ 0.383 \\ 0.383 \\ 0.383 \\ -0.383 \\ -0.383 \\ -0.383 \\ -0.247 \end{bmatrix}$$

Example Graph 1, communities

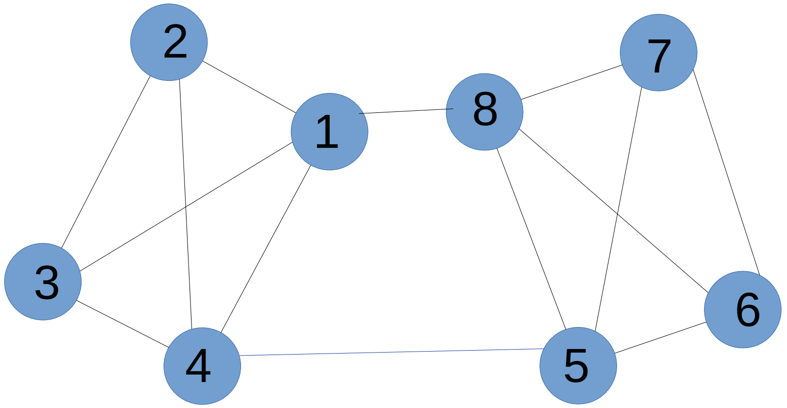


$$\lambda_1 = 0$$

$$\lambda_2 = 0.354$$

$$v_2 = \begin{bmatrix} 0.247 \\ 0.383 \\ 0.383 \\ 0.383 \\ -0.383 \\ -0.383 \\ -0.383 \\ -0.247 \end{bmatrix}$$

Example Graph 2



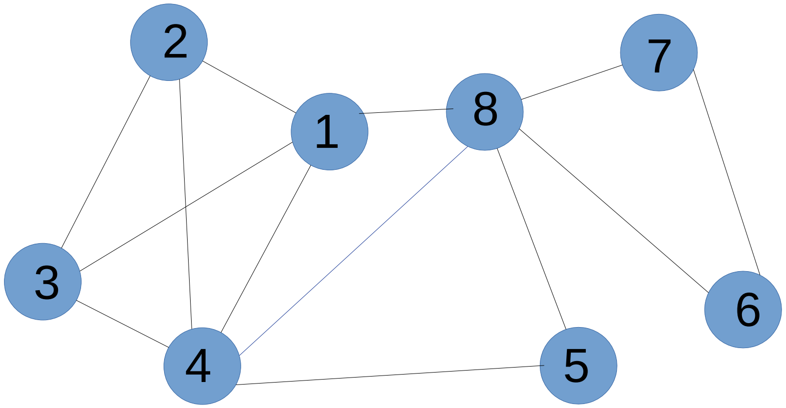
$$\lambda_1 = 0$$

$$\lambda_2 = 0.764$$

$$L = \begin{bmatrix} 4 & -1 & -1 & -1 & 0 & 0 & 0 & -1 \\ -1 & 3 & -1 & -1 & 0 & 0 & 0 & 0 \\ -1 & -1 & 3 & -1 & 0 & 0 & 0 & 0 \\ -1 & -1 & -1 & 4 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 4 & -1 & -1 & -1 \\ 0 & 0 & 0 & 0 & -1 & 3 & -1 & -1 \\ 0 & 0 & 0 & 0 & -1 & -1 & 3 & -1 \\ -1 & 0 & 0 & 0 & -1 & -1 & -1 & 4 \end{bmatrix}$$

$$v_2 = \begin{bmatrix} 0.263 \\ 0.425 \\ 0.425 \\ 0.263 \\ -0.263 \\ -0.425 \\ -0.425 \\ -0.263 \end{bmatrix}$$

Example Graph 3



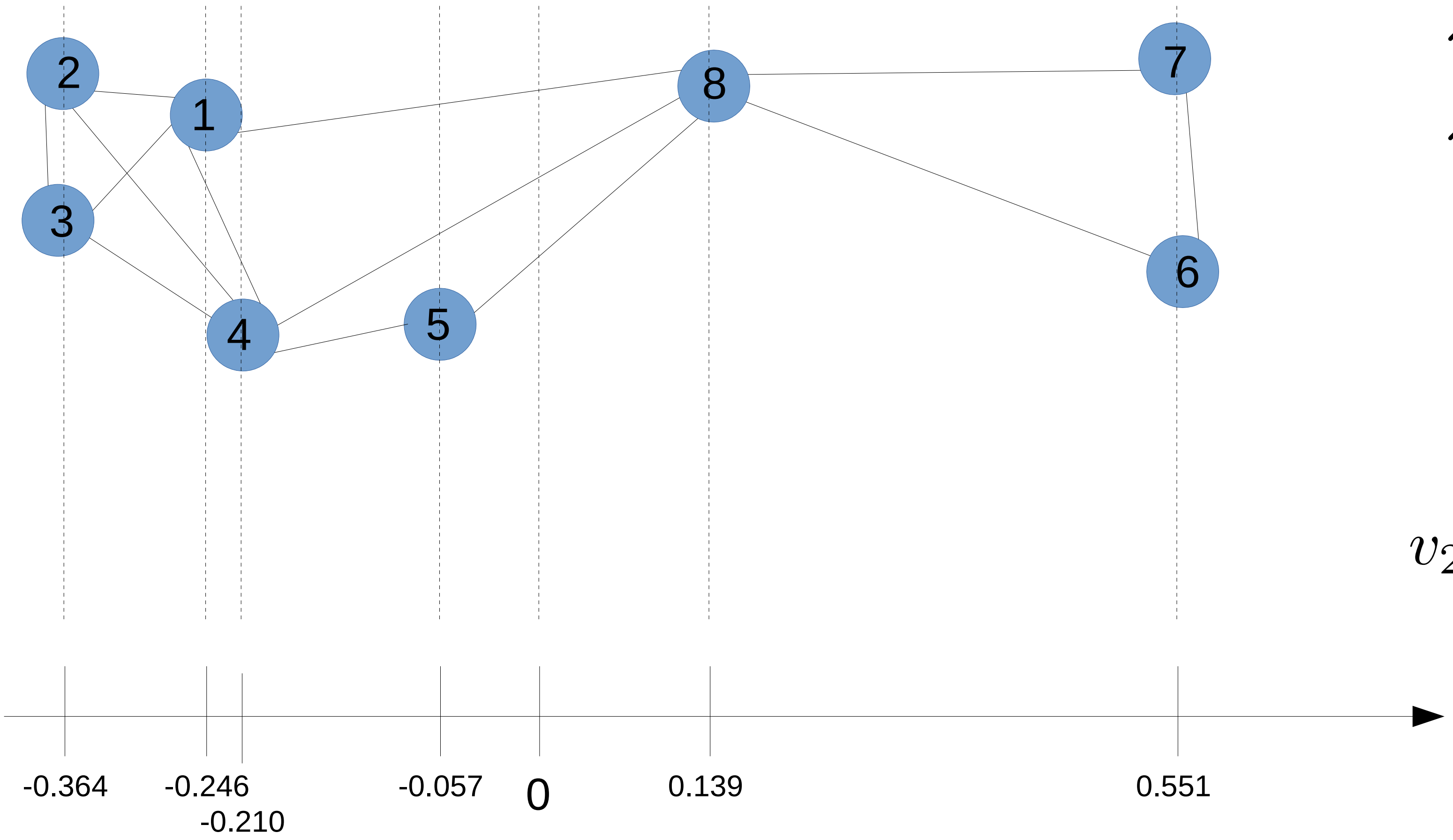
$$\lambda_1 = 0$$

$$\lambda_2 = 0.748$$

$$L = \begin{bmatrix} 4 & -1 & -1 & -1 & 0 & 0 & 0 & -1 \\ -1 & 3 & -1 & -1 & 0 & 0 & 0 & 0 \\ -1 & -1 & 3 & -1 & 0 & 0 & 0 & 0 \\ -1 & -1 & -1 & 5 & -1 & 0 & 0 & -1 \\ 0 & 0 & 0 & -1 & 2 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 2 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 2 & -1 \\ -1 & 0 & 0 & -1 & -1 & -1 & -1 & 5 \end{bmatrix}$$

$$v_2 = \begin{bmatrix} -0.246 \\ -0.364 \\ -0.364 \\ -0.210 \\ -0.057 \\ 0.551 \\ 0.551 \\ 0.139 \end{bmatrix}$$

Example Graph 3, projected (where to cut?)

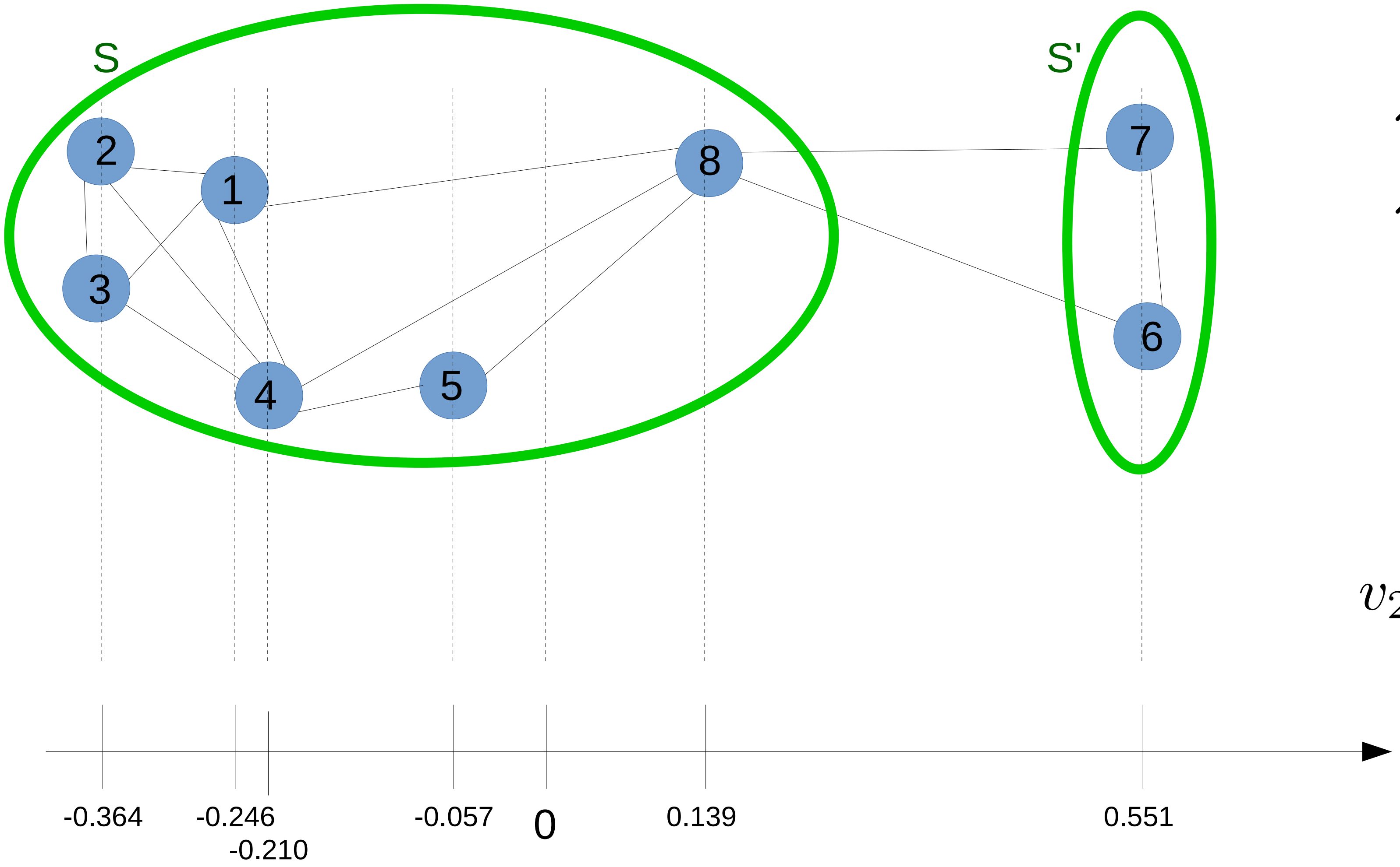


$$\lambda_1 = 0$$

$$\lambda_2 = 0.748$$

$$v_2 = \begin{bmatrix} -0.246 \\ -0.364 \\ -0.364 \\ -0.210 \\ -0.057 \\ 0.551 \\ 0.551 \\ 0.139 \end{bmatrix}$$

Example Graph 3, projected (where to cut?)

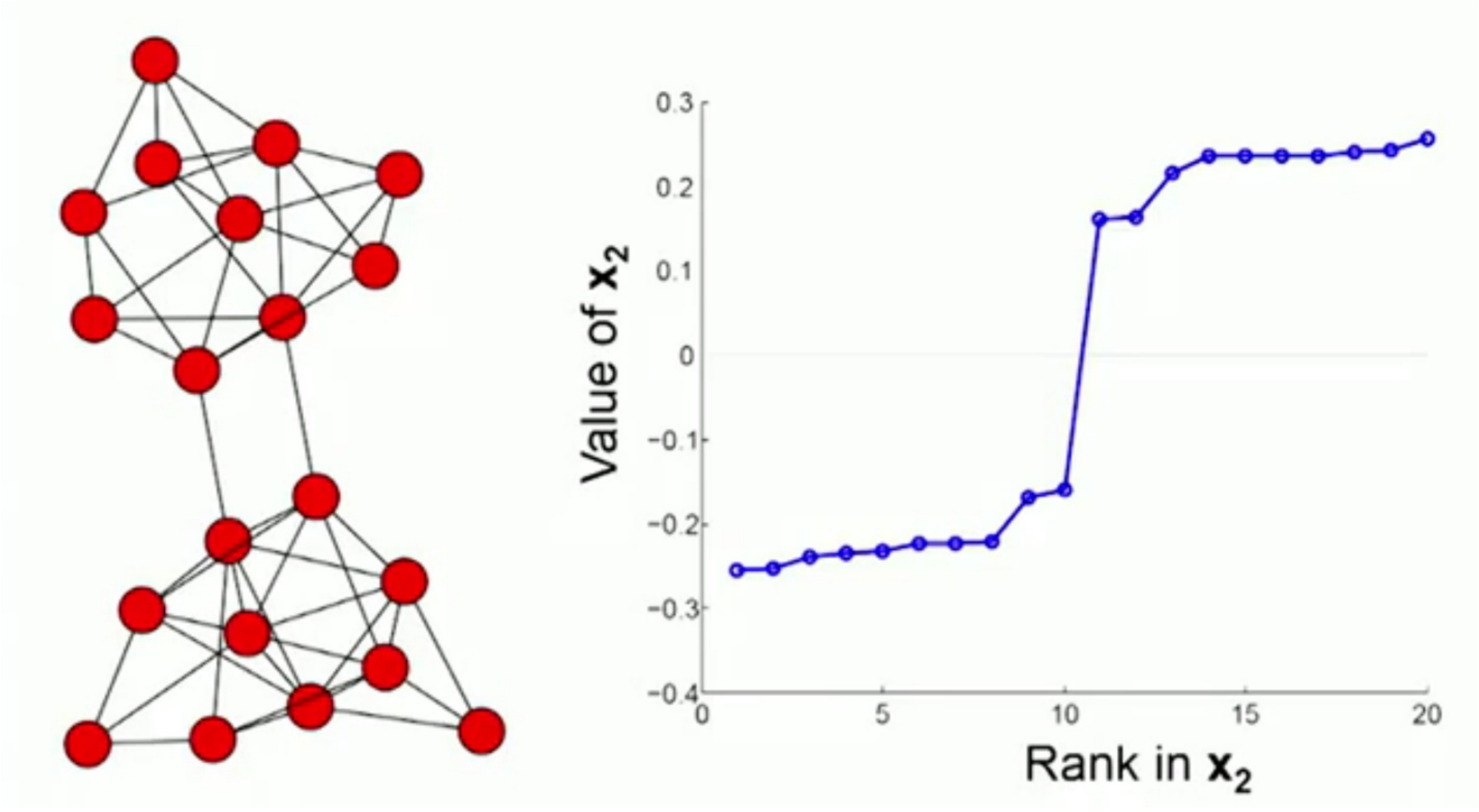


$$\lambda_1 = 0$$

$$\lambda_2 = 0.748$$

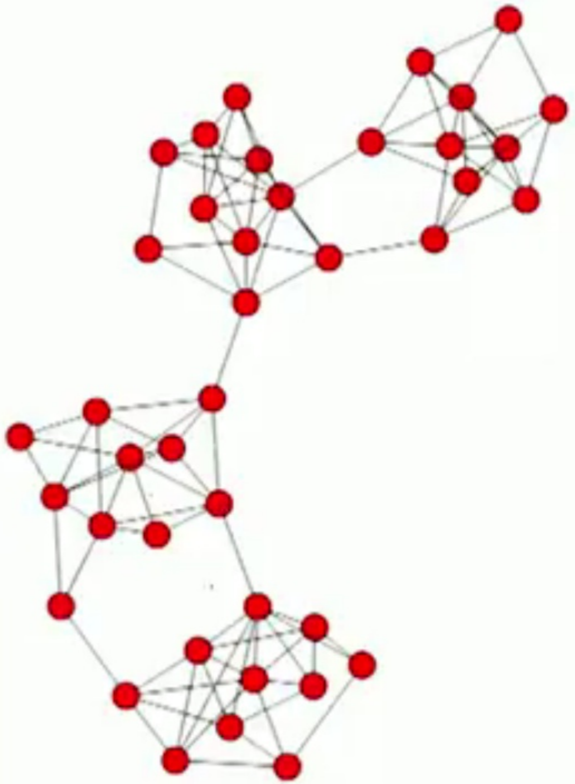
$$v_2 = \begin{bmatrix} -0.246 \\ -0.364 \\ -0.364 \\ -0.210 \\ -0.057 \\ 0.551 \\ 0.551 \\ 0.139 \end{bmatrix}$$

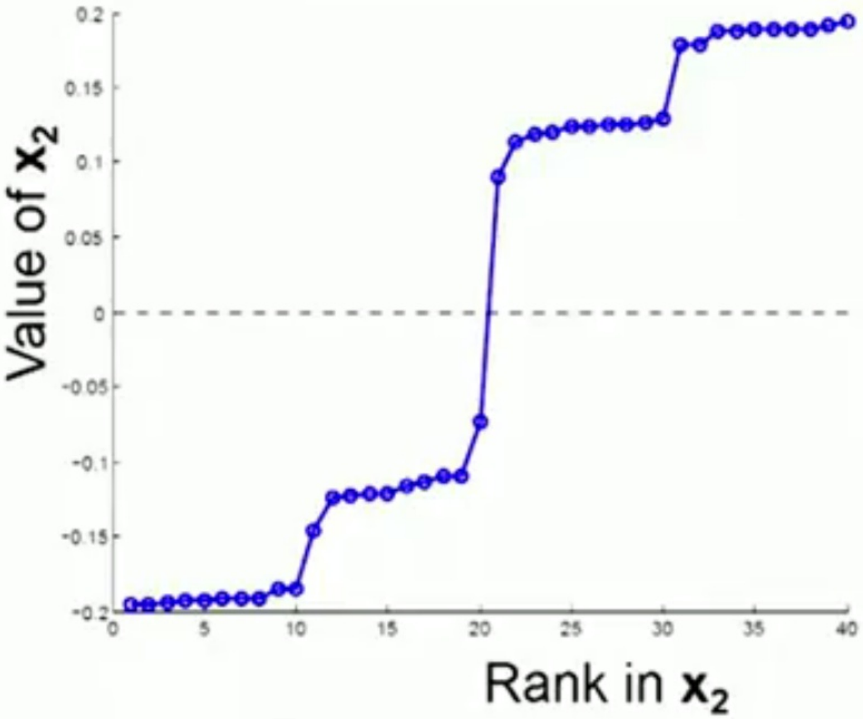
A more complex graph



https://www.youtube.com/watch?v=jpTjj5PmcMM

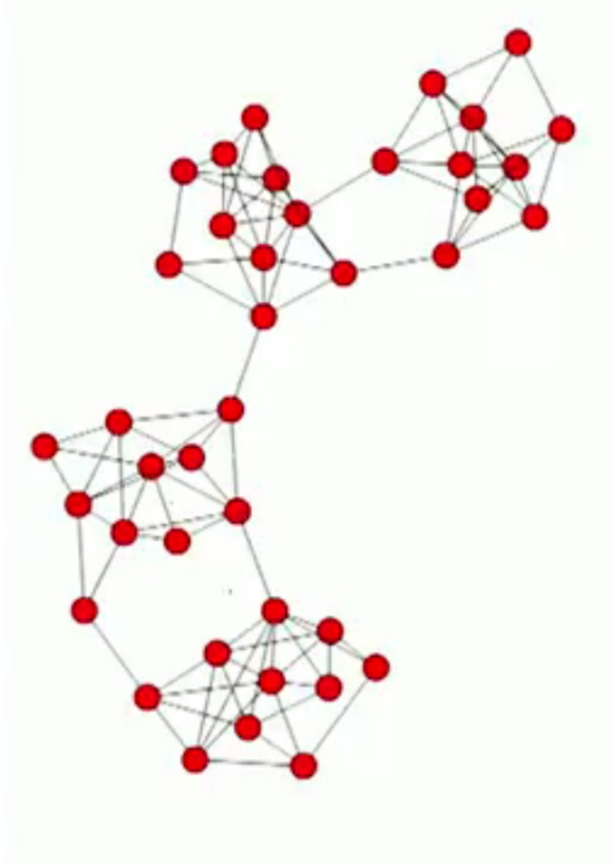
A graph with 4 “communities”

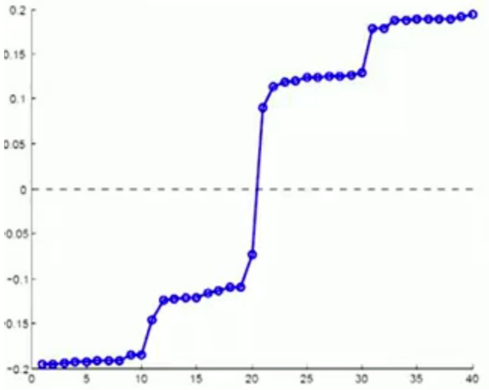




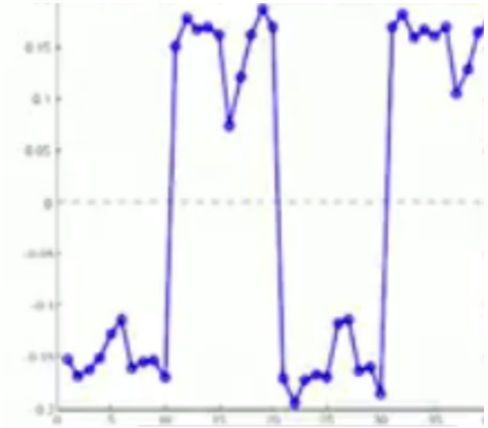
https://www.youtube.com/watch?v=jpTjj5PmcMM

Other eigenvectors

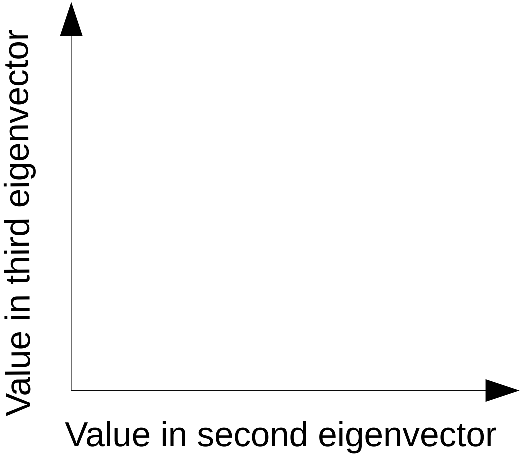




Second eigenvector

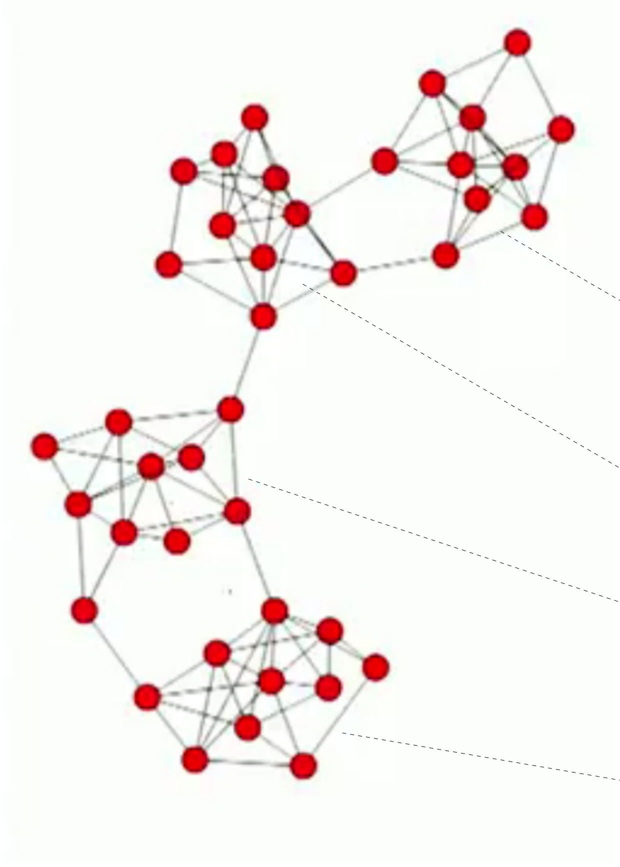


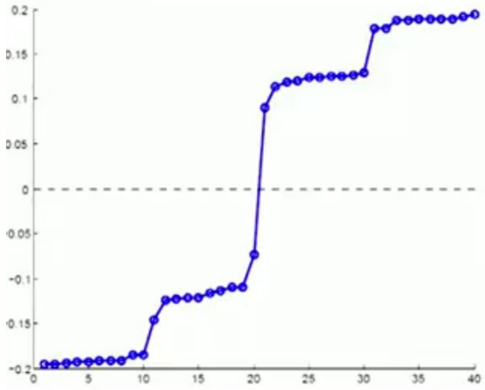
Third eigenvector



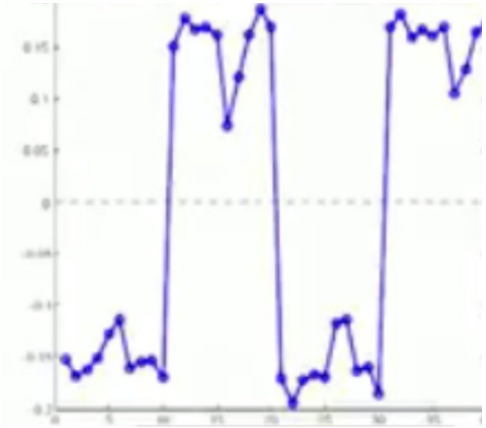
https://www.youtube.com/watch?v=jpTjj5PmcMM

Other eigenvectors

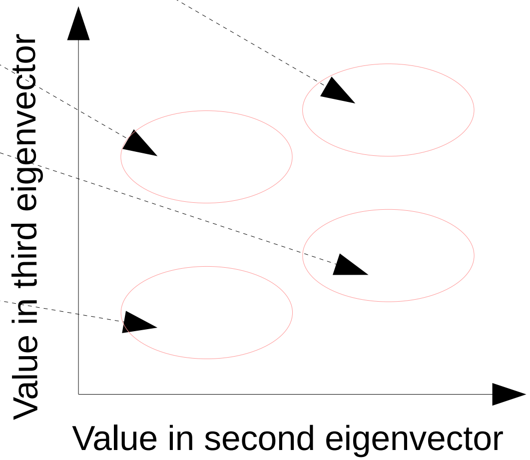




Second eigenvector



Third eigenvector



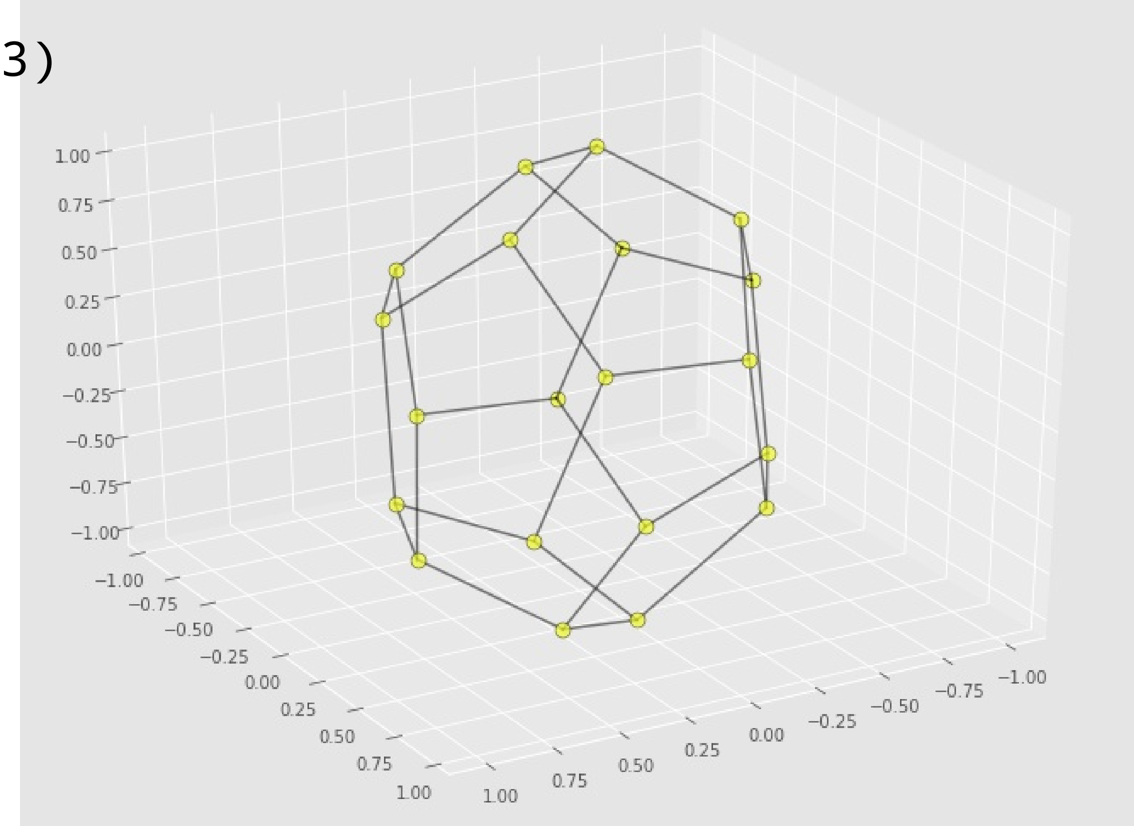
Clustering can be done using Euclidean distance

https://www.youtube.com/watch?v=jpTjj5PmcMM

Dodecahedral graph in 3D

python

```
g = nx.dodecahedral_graph()
pos = nx.spectral_layout(g, dim=3)
network_plot_3D_alt(g, 60, pos)
```

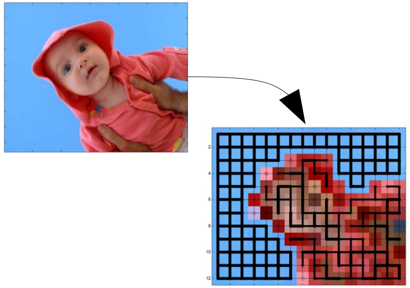


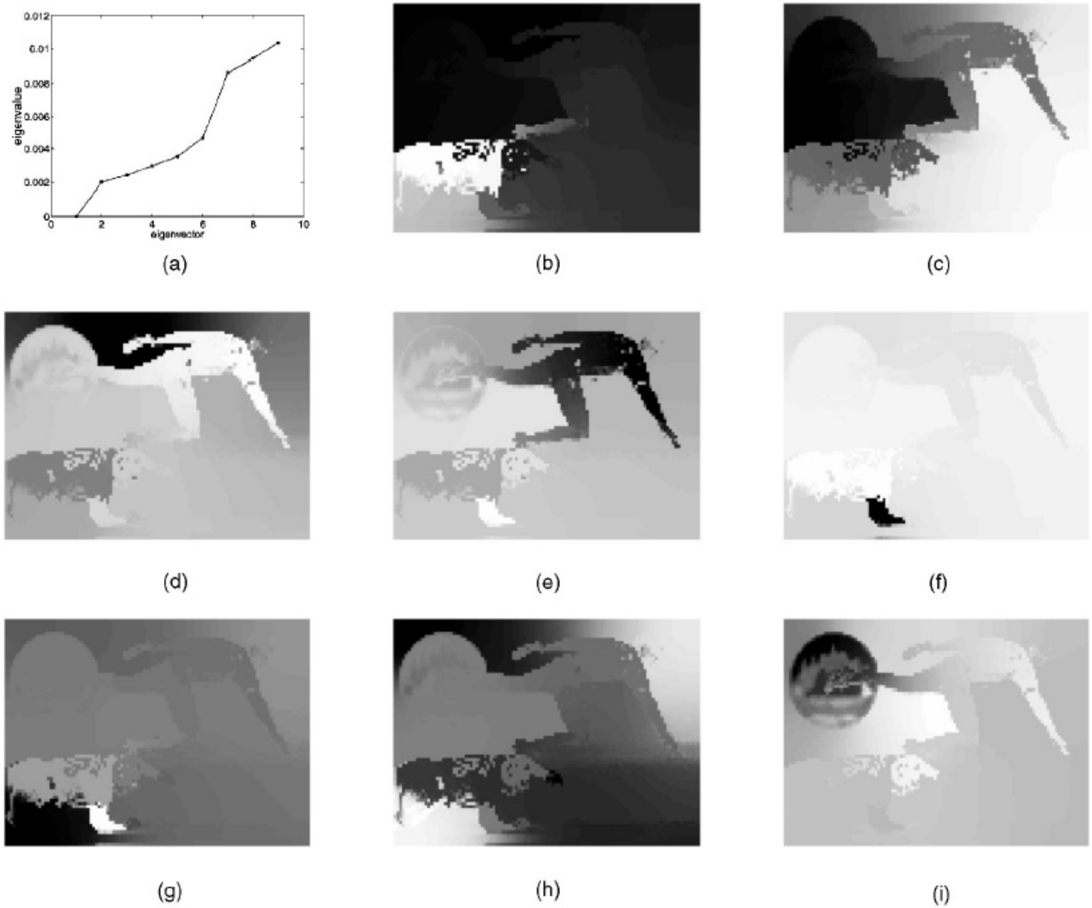
https://www.idtools.com.au/3d-network-graphs-python-mplot3d-toolkit/

Application: image segmentation

[Shi & Malik 2000]

Transform into grid graph with edge weights proportional to pixel similarity





Summary

Things to remember

- Graph Laplacian
- Laplacian and graph components
- Spectral graph embedding

Exercises for this topic

- Mining of Massive Datasets (2014) by Leskovec et al.
  - Exercises 10.4.6