Spectral Clustering

Part 3: The Normalized Laplacian

Ng Yen Kaow

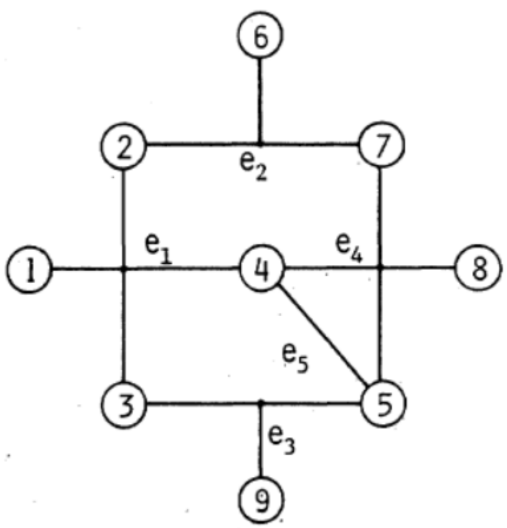
More constraint for balance

- Further constraints can be added to the eigenvalue system
- The solution to these problems will require the **generalized eigensystem** $Lx = \lambda Dx$

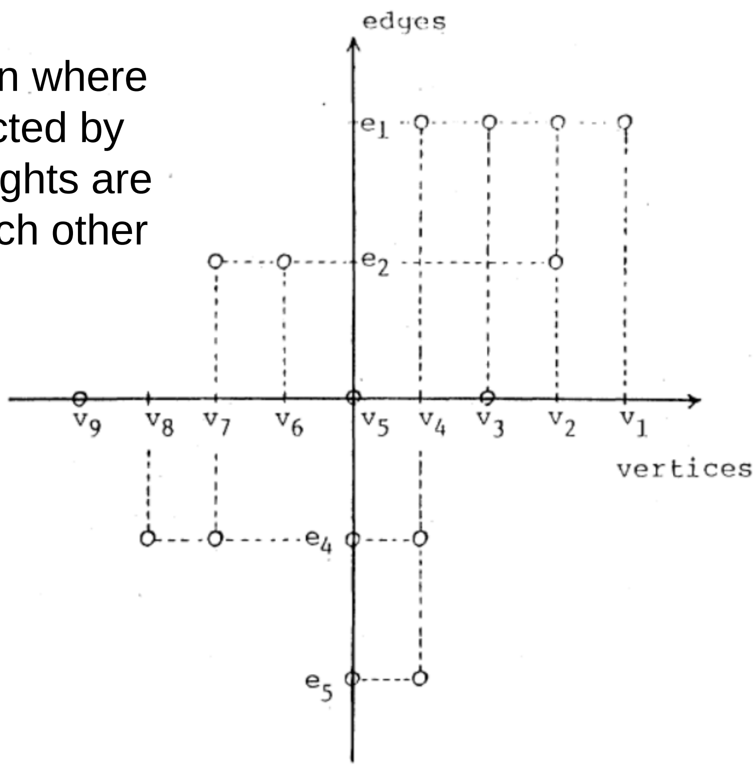
Generalized eigensystem $Lx = \lambda Dx$

- Proposed as a solution to the problem of representing hypergraphs in Euclidean space (Fukunaga *et al.*, 1984)

An edge in a hypergraph can be connected to multiple vertices



Find a representation where the vertices connected by edges with large weights are brought closer to each other



Generalized eigensystem $Lx = \lambda Dx$

- The problem is shown to be equivalent to that of solving $Lx = \lambda Dx$ (Van Driessche and Roose, 1995) which corresponds to the optimization problem
  - Minimize $x^\top Lx$
    subject to $x^\top Dx = 1$

Proof.

The Lagrangian $\mathcal{L}$ for the optimization problem is

$$\mathcal{L}(x, \lambda) = x^\top Lx + \lambda(x^\top Dx - 1)$$

Equating the derivative of $\mathcal{L}$ to zero,

$$\frac{\partial \mathcal{L}}{\partial x} = 2Lx - 2\lambda Dx = 0 \Rightarrow Lx = \lambda Dx$$

Generalized eigensystem $Lx=\lambda Dx$

- The problem is shown to be equivalent to that of solving $Lx = \lambda Dx$ (Van Driessche and Roose, 1995) which is from the optimization problem
  - Minimize $x^\top Lx$
    subject to $x^\top Dx = 1$
    - In this case, let $y = D^{1/2}x$ (i.e. $x = D^{-1/2}y$)
      Then $x^\top Lx \Rightarrow y^\top D^{-1/2}LD^{-1/2}y$, and
      $x^\top Dx = 1 \Rightarrow y^\top y = 1$

  $\Rightarrow$ Minimize $yD^{-1/2}LD^{-1/2}y$
  subject to $y^\top y = 1$

  which is a standard eigendecomposition problem of the matrix $D^{-1/2}LD^{-1/2}$

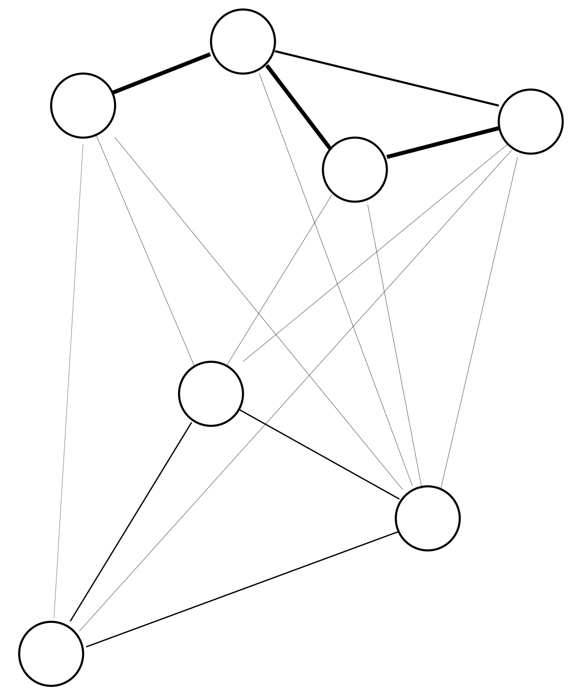
Normalized Laplacian $D^{-1/2}LD^{-1/2}$

- The matrix $D^{-1/2}LD^{-1/2}$ is now known as the **normalized Laplacian**
- It is shown to be positive semi-definite (Van Driessche and Roose, 1995)
  - $\Rightarrow$ Eigenvalues are all positive (does not matter for spectral clustering but still nice to have)
- However, $D^{-1/2}LD^{-1/2}$ have deviated very far from the incidence matrix

Normalized Cut Problem

- Given weight matrix $W = (w_{ij})$ and weighted degree matrix $D = (d_i)$
- Recall that a minimum **ratio cut** minimizes

$$\text{ratio}(S, \bar{S}) = \text{cut}(S, \bar{S}) \left(\frac{1}{|S|} + \frac{1}{|\bar{S}|} \right)$$



where $\text{cut}(S, \bar{S}) = \sum_{i \in S, j \in \bar{S}} w_{ij}$

  - Minimizes difference between the **number of vertices**

- A **normalized cut** attempts to minimize the difference between the **sum of the edge weights adjacent to each vertex**

Note: For convenience we use D to denote what we denoted as D' in earlier slides

Normalized Cut Problem

- Given weight matrix $W = (w_{ij})$ and weighted degree matrix $D = (d_i)$, the normalized cut of an undirected graph $G = (V, E)$ is a partition of V into two groups S and $\bar{S}$ such that

$$\mathrm{ncut}(S, \bar{S}) = \mathrm{cut}(S, \bar{S}) \left(\frac{1}{\mathrm{vol}(S)} + \frac{1}{\mathrm{vol}(\bar{S})} \right)$$

  is minimized, where $\mathrm{vol}(S) = \sum_{i \in S} d_i$, that is, sum of all the weights of the edges adjacent to vertices in S, and $\mathrm{cut}(S, \bar{S}) = \sum_{i \in S, j \in \bar{S}} w_{ij}$

Note: $\mathrm{vol}(S) + \mathrm{vol}(\bar{S}) = 2\sum w_{ij}$

Mathematical property

- Represent a partition S, $\bar{S}$ of V with $x \in \mathbb{R}^n$, where

$$x_i = \begin{cases} \dfrac{1}{\mathrm{vol}(S)} & \text{if } i \in S \\ -\dfrac{1}{\mathrm{vol}(\bar{S})} & \text{if } i \in \bar{S} \end{cases}$$

As in Ratio Cut, $|x_i|$ **changes according to the solution**

1. $x^\top L x = \sum_{ij} w_{ij}(x_i - x_j)^2 = \left(\frac{1}{\mathrm{vol}(S)} + \frac{1}{\mathrm{vol}(\bar{S})}\right)^2 \sum_{ij} w_{ij}$
$$= \left(\frac{1}{\mathrm{vol}(S)} + \frac{1}{\mathrm{vol}(\bar{S})}\right)^2 \mathrm{cut}(S, \bar{S})$$

2. $x^\top D x = \sum_i d_i (x_i)^2 = \sum_{i \in S} \frac{d_i}{\mathrm{vol}(S)^2} + \sum_{i \in \bar{S}} \frac{d_i}{\mathrm{vol}(\bar{S})^2} = \frac{1}{\mathrm{vol}(S)} + \frac{1}{\mathrm{vol}(\bar{S})}$

$$1 + 2 \Rightarrow \frac{x^\top L x}{x^\top D x} = \mathrm{cut}(S, \bar{S}) \left(\frac{1}{\mathrm{vol}(S)} + \frac{1}{\mathrm{vol}(\bar{S})}\right) = \mathrm{ncut}(S, \bar{S})$$

Constrained optimization problem

- Minimize $x^\top L x$ where $L = D - W$

  subject to $x_i \in \left\{\frac{1}{\mathrm{vol}(S)}, -\frac{1}{\mathrm{vol}(\bar{S})}\right\}$,

  $x^\top D x = 1$, and

  $\mathbf{1}^\top D x = 0$
- Problem is NP-hard
- Note:
  - $\mathbf{1}^\top D x = \sum_{i \in S} \frac{d_i}{\mathrm{vol}(S)} - \sum_{i \in \bar{S}} \frac{d_i}{\mathrm{vol}(\bar{S})} = 1 - 1 = 0$
  - $\frac{1}{\mathrm{vol}(S)}, -\frac{1}{\mathrm{vol}(\bar{S})}$ are not the only possible choices
    - See https://arxiv.org/abs/1311.2492

Relaxed Rayleigh quotient version

- Minimize $x^\top Lx$ where $L = D - W$ subject to $x^\top Dx = 1$ and $\mathbf{1}^\top Dx = 0$
- This is equivalent to the earlier **generalized eigensystem** $Lx = \lambda Dx$ except for the additional requirement of $\mathbf{1}^\top Dx = 0$

Generalized eigensystem

- Minimize $x^\top L x$ where $L = D - W$
  subject to $x^\top D x = 1$ and $\mathbf{1}^\top D x = 0$
- Let $y = D^{1/2} x$, that is, $x = D^{-1/2} y$

$$x^\top L x \Rightarrow y^\top D^{-1/2} L D^{-1/2} y$$

$$x^\top D x = 1 \Rightarrow y^\top y = 1$$

$$\mathbf{1}^\top D x = 0 \Rightarrow \mathbf{1}^\top D^{1/2} y = 0$$

Hence equivalently

- Minimize $y D^{-1/2} L D^{-1/2} y$
  subject to $y^\top y = 1$ and $\mathbf{1}^\top D^{1/2} y = 0$

Generalized eigensystem

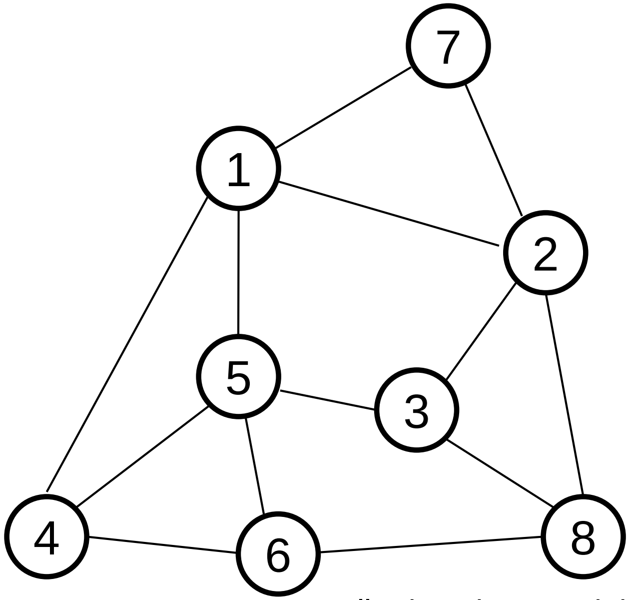
- Minimize $yD^{-1/2}LD^{-1/2}y$ where $L = D - W$ subject to $y^\top y = 1$ and $\mathbf{1}^\top D^{1/2}y = 0$
- All eigenvectors of $D^{-1/2}LD^{-1/2}$ fulfill $\mathbf{1}^\top D^{1/2}y = 0$
  - As $\mathbf{1}$ is a eigenvector for $Lx = \lambda Dx$ with eigenvalue 0, $D^{1/2}\mathbf{1}$ is a eigenvector for this system with eigenvalue 0 (smallest)
  - Since eigenvectors of this system are orthogonal, $\left(D^{1/2}\mathbf{1}\right)\mu_{k-1} = 0$ $\Rightarrow \mathbf{1}^\top D^{1/2}y = 0$ fulfilled

    In fact the eigenvalues for this system are the same as those for $Lx = \lambda Dx$, even though the eigenvectors are different (related by $y = M^{1/2}x$)

$\Rightarrow$ Eigendecomposition of $D^{-1/2}LD^{-1/2}$ suffices

Exercise

- Find normalized Laplacian $D^{-1/2}LD^{-1/2}$ for graph and eigendecompose it



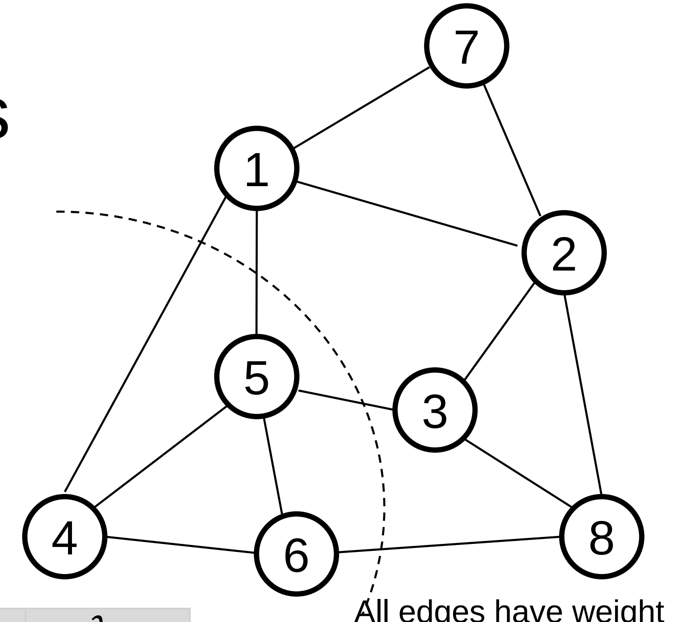
All edges have weight 1

To find $D^{-1/2}$ in Python, use SciPy

```
scipy.linalg.sqrtm(scipy.linalg.inv(D))
```

Eigendecomposition

- Eigenvalues and eigenvectors



λ_1	λ_2	λ_3	λ_4	λ_5	λ_6	λ_7	λ_8
1.6760	1.5100	1.42700	1.3100	0.9900	0.5880	0.4990	0.0

μ_1	μ_2	μ_3	μ_4	μ_5	μ_6	μ_7	μ_8
0.3485	0.0034	0.6240	-0.2451	-0.0704	-0.5023	0.1342	0.3922
-0.0304	0.6546	-0.3393	-0.2014	0.0768	0.0885	0.4973	0.3922
0.4129	-0.3896	-0.1906	-0.0484	-0.5545	0.4474	0.1265	0.3397
-0.2148	-0.2574	-0.4363	-0.5537	0.0989	-0.2859	-0.4286	0.3397
-0.4292	0.2801	0.1122	0.4236	-0.5021	-0.0836	-0.3638	0.3922
0.5058	0.1486	-0.0793	0.3598	0.4989	0.1541	-0.4454	0.3397
-0.1662	-0.4557	-0.2360	0.5096	0.2180	-0.3552	0.4457	0.2774
-0.4397	-0.2128	0.4406	-0.1475	0.3513	0.5487	0.0744	0.3397

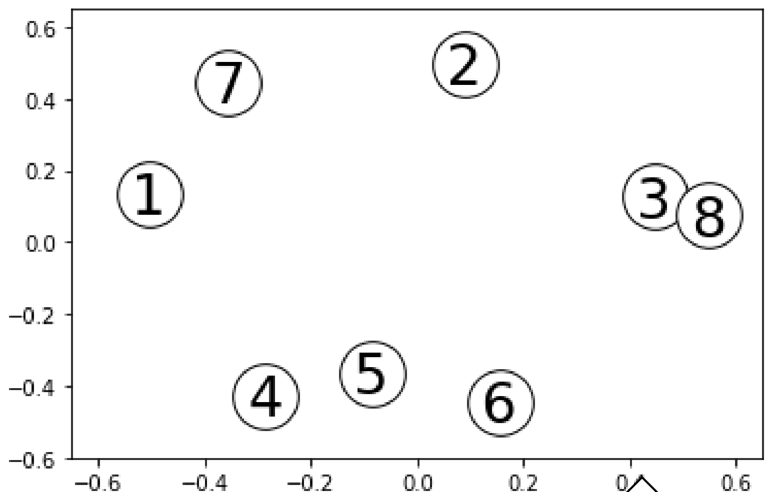
The limiting distribution of the normalized Laplacian is not $f(v) = \text{const}$ since the normalized Laplacian is unrelated to the incidence matrix

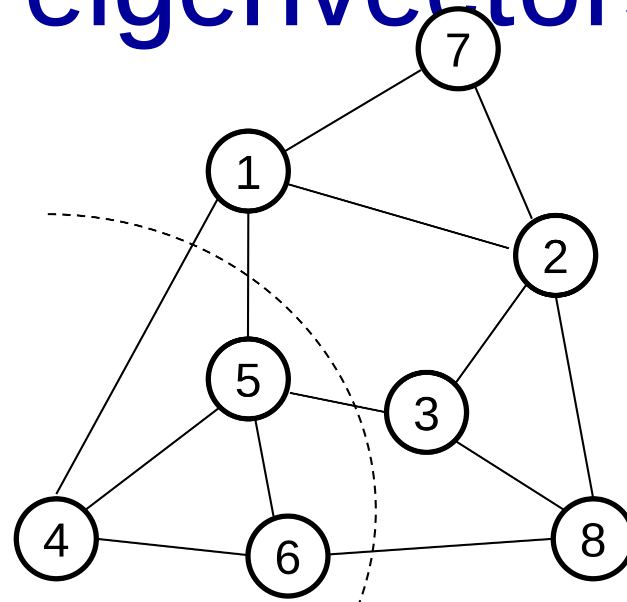
Shi and Malik (1997, 2000)

- Proposed the NP-hard ncut problem
- Related ncut to generalized eigenvalue system, resulting in the now ubiquitous **normalized Laplacian**
- Use Gaussian function $e^{-d^2/2\sigma^2}$ for weights
  - Previously used for min-cut (Wu and Leahy, 1993)
  - Used for RatioCut later (Wang and Siskin, 2003)
- Clustering with multiple eigenvectors (Van Driessche and Roose, 1995; Shi and Malik, 2000)

Clustering w/ multiple eigenvectors

- With normalized Laplacian





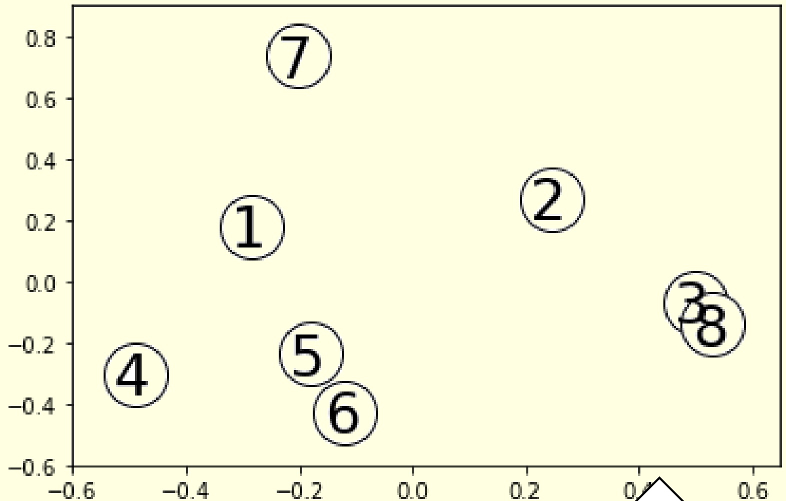
All edges have weight 1

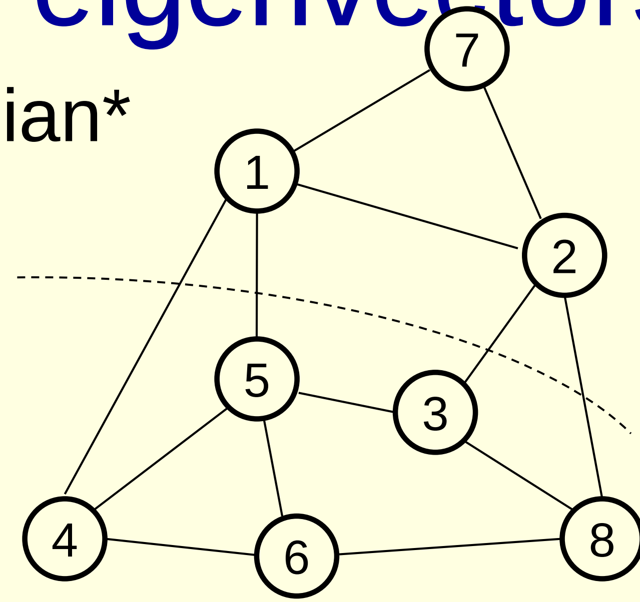
μ_1	μ_2	μ_3	μ_4	μ_5	μ_6	μ_7	μ_8
0.3485	0.0034	0.6240	-0.2451	-0.0704	-0.5023	0.1342	0.3922
-0.0304	0.6546	-0.3393	-0.2014	0.0768	0.0885	0.4973	0.3922
0.4129	-0.3896	-0.1906	-0.0484	-0.5545	0.4474	0.1265	0.3397
-0.2148	-0.2574	-0.4363	-0.5537	0.0989	-0.2859	-0.4286	0.3397
-0.4292	0.2801	0.1122	0.4236	-0.5021	-0.0836	-0.3638	0.3922
0.5058	0.1486	-0.0793	0.3598	0.4989	0.1541	-0.4454	0.3397
-0.1662	-0.4557	-0.2360	0.5096	0.2180	-0.3552	0.4457	0.2774
-0.4397	-0.2128	0.4406	-0.1475	0.3513	0.5487	0.0744	0.3397

Use the values from the top few eigenvectors for clustering (with, for example, k-means)

Clustering w/ multiple eigenvectors

- With graph partitioning Laplacian*





All edges and vertices have weight 1

μ_1	μ_2	μ_3	μ_4	μ_5	μ_6	μ_7	μ_8
0.5677	-0.1583	-0.4862	0.3536	0.2315	-0.2855	0.1766	0.3536
-0.4281	0.6222	-0.2059	0.3536	0.0622	0.2469	0.2690	0.3536
0.3517	0.1203	0.2984	-0.3536	0.5170	0.5007	-0.0694	0.3536
-0.0855	0.0612	0.6267	0.3536	0.1159	-0.4899	-0.3044	0.3536
-0.5514	-0.3549	-0.3566	-0.3536	0.3216	-0.1795	-0.2392	0.3536
0.2351	0.3822	-0.2014	-0.3536	-0.5589	-0.1183	-0.4263	0.3536
-0.0354	-0.1476	0.2596	-0.3536	-0.2798	-0.2029	0.7349	0.3536
-0.0540	-0.5251	0.0654	0.3536	-0.4096	0.5286	-0.1411	0.3536

The resultant eigenvectors are less suitable for clustering

*see Appendix

Single/multiple eigenvectors use

- Historical use based on Fiedler vector
  - Sign cut or zero threshold cut
  - Median cut (ensures balance)
  - Sweep/criterion cut
    - Sort vertices by Fiedler vector values and cut at the lowest value of some cost function
  - Jump/gap cut
    - Sort vertices by Fiedler vector values and cut at the point of largest gap
- After Shi and Malik, multiple eigenvectors
  - Simultaneous k-way (Shi and Malik, 2000)
  - k-means (Ng, Jordan and Weiss, 2001)

Theoretical justification

- How should we view the normalized Laplacian
  - Since normalized Laplacian cannot be related to the incidence matrix, it requires a new characterization

  ⇒ Random walk characterization (Meilă and Shi, 2000)

- Arguments based on minimizing divergence and objective functions justify only the use of only one eigenvector (not multiple eigenvectors)
  - Furthermore, both arguments are no longer valid for the normalized Laplacian

  ⇒ (Weiss, 1999; Meilă and Shi, 2000; Ng, Jordan and Weiss, 2001) successively give justification for the use of the eigenvectors

Random walk characterization

- Let $P = D^{-1}W$ (where $L = D - W$)
  - A solution x for $Px = \lambda x$ is a solution for the generalized eigensystem $Lx = \lambda Dx$ (with eigenvalues $1 - \lambda$), and vice versa

    Proof.

$$
\begin{aligned}
Lx = \lambda Dx \Rightarrow D^{-1}(D - W)x &= D^{-1}\lambda Dx \\
(I - P)x &= \lambda x \\
Px &= (I - \lambda)x \\
Lx &= \lambda Dx
\end{aligned}
$$

$$
\begin{aligned}
Px = (I - \lambda)x \Rightarrow D^{-1}Wx &= (I - \lambda)x \\
(I - D^{-1}W)x &= \lambda x \\
(D - W)x &= D\lambda x \\
Lx &= D\lambda x
\end{aligned}
$$

Random walk characterization

- Let $P = D^{-1}W$ (where $L = D - W$)
  - A solution x for $Px = \lambda x$ is a solution for the generalized eigensystem $Lx = \lambda Dx$ (with eigenvalues $1 - \lambda$), and vice versa
    - **The normalized Laplacian $D^{-1/2}LD^{-1/2}$ computes the solutions to $Px = \lambda x$ for the normalized matrix P**
  - However, P is not symmetric
    - Doesn't decompose to orthogonal eigenbasis
  - On the other hand $D^{-1/2}LD^{-1/2}$ is symmetric
    - Chosen over P for spectral clustering

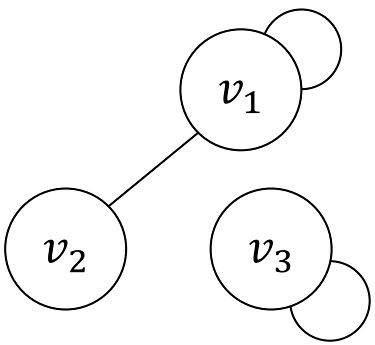
Random walk characterization

- Each row in P sums to 1 (normalized)
  - P **is a Markovian transition matrix**
- To start a walk from v_1, let $x = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \end{bmatrix}$, then $P^l x$ is the probability distribution after l steps from v_1
- x_i for neighboring vertices will become more similar $\Rightarrow$ gradients decrease
- Parts of the graph will even out more quickly

Random walk characterization

- Example: Let P be a 3×3 matrix

$$Px = \begin{pmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ p_{31} & p_{32} & p_{33} \end{pmatrix} \begin{pmatrix} f(v_1) \\ f(v_2) \\ f(v_3) \end{pmatrix} = \begin{pmatrix} p_{11} f(v_1) + p_{12} f(v_2) + p_{13} f(v_3) \\ p_{21} f(v_1) + p_{22} f(v_2) + p_{23} f(v_3) \\ p_{31} f(v_1) + p_{32} f(v_2) + p_{33} f(v_3) \end{pmatrix}$$



$$\Rightarrow \; Px = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} f(v_1) \\ f(v_2) \\ f(v_3) \end{pmatrix} = \begin{pmatrix} f(v_1) + f(v_2) \\ f(v_1) \\ f(v_3) \end{pmatrix}$$

- x_i for neighboring vertices will become more similar ⇒ gradients decrease
- Parts of the graph will even out more quickly

Random walk characterization

- A limiting/stable/stationary state for a random walk P is a distribution x^* where $Px^* = x^*$
  - By definition x^* is a eigenvector of P with $\lambda = 1$

Furthermore, x^* is everywhere constant if P is

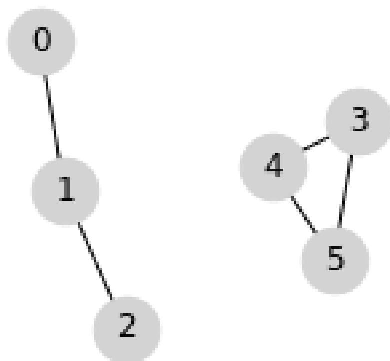
- A transition matrix for a regular graph
  By symmetry of the graph, a random walk from any vertex is equally likely to be at any other vertex in the limit
- A Laplacian $L = MM^\top$ for incidence matrix M
  First note that x^* minimizes $x^\top Lx$. On the other hand we know that $x^\top Lx = \sum_v f(v)\Delta f(v)$. Since $\Delta f(v) = 0$ for the everywhere constant x', we have $x'^\top Lx' = 0$, its minimum. Hence $x^* = x'$

Why use multiple eigenvectors

- For illustrative convenience use (an adjacency matrix) $L' = D^{-1/2}(W)D^{-1/2}$ instead of the normalized Laplacian L
  - $L' = I - L$ ($L =$ normalized Laplacian)

    Proof. $L = D^{-1/2}(D - W)D^{-1/2}$
    $$= D^{-1/2}(D)D^{-1/2} - D^{-1/2}(W)D^{-1/2}$$
    $$= I - D^{-1/2}(W)D^{-1/2} = I - L'$$
  - Results in the same eigenvectors but eigenvalues become $1 - \lambda_1, \ldots, 1 - \lambda_k$
    - Since eigenvalues of L has range in [0,2], eigenvalues of L' has range in [-1,1]

Why use multiple eigenvectors



$$L' = \begin{array}{c} \begin{array}{cccccc} \quad L'_u \quad & & & & & \end{array} \\ \begin{bmatrix} 0 & .7 & 0 & 0 & 0 & 0 \\ .7 & 0 & .7 & 0 & \text{zeroes!} & 0 \\ 0 & .7 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & .5 & .5 \\ 0 & \text{zeroes!} & 0 & .5 & 0 & .5 \\ 0 & 0 & 0 & .5 & .5 & 0 \end{bmatrix} \\ \begin{array}{cccccc} & & & & \quad L'_l \quad & \end{array} \end{array}$$

Matrix	Eigenvalues/vectors (decreasing order)	
L'_u	$\lambda_1^u = 1$ $\lambda_2^u = 0$ $\lambda_3^u = -1$	$v_1^u = [.5 \quad .7 \quad .5]$ $v_2^u = [.7 \quad 0 \quad -.7]$ $v_3^u = [.5 \quad -.7 \quad .5]$
L'_l	$\lambda_1^l = 1$ $\lambda_2^l = -.5$ $\lambda_3^l = -.5$	$v_1^l = [.6 \quad .6 \quad .6]$ $v_2^l = [0 \quad -.7 \quad -.7]$ $v_3^l = [-.8 \quad .4 \quad .4]$
L'	$\lambda_1 = 1$ $\lambda_2 = 1$ $\lambda_3 = 0$ $\lambda_4 = -.5$ $\lambda_5 = -.5$ $\lambda_6 = -1$	$v_1 = [0 \quad 0 \quad 0 \quad .6 \quad .6 \quad .6]$ $v_2 = [.5 \quad .7 \quad .5 \quad 0 \quad 0 \quad 0]$ $v_3 = [.7 \quad 0 \quad -.7 \quad 0 \quad 0 \quad 0]$ $v_4 = [0 \quad 0 \quad 0 \quad 0 \quad -.7 \quad .7]$ $v_5 = [0 \quad 0 \quad 0 \quad -.8 \quad .4 \quad .4]$ $v_6 = [.5 \quad -.7 \quad .5 \quad 0 \quad 0 \quad 0]$

- The eigenvalues/vectors of L' compose of the eigenvalues/vectors of the submatrices L'_u and L'_l, with unconnected vertices set to 0
- The largest eigenvalue of L'_u and L'_l are both 1 for the ideal case

Why use multiple eigenvectors

- The largest eigenvalue of L_u' and L_l' is 1 for the ideal (disconnected) case

$$\lambda_1 = \lambda_2 = 1 \Rightarrow |\lambda_1 - \lambda_2| = 0$$

  - In non-ideal case, $\lambda_2 < \lambda_1$
  - The larger the eigenvalue (for L'), the more cohesive the cluster (this is opposite for L)
- $|\lambda_k - \lambda_{k+1}|$ is called **eigengap** or **spectral gap**
  - Large $|\lambda_k - \lambda_{k+1}|$ implies higher cohesion in the clusters given by μ_k than those by μ_{k+1}
  - Evaluate whether to use a eigenvector in clustering by its eigengap from the previous

Reconciliation with divergence

- No direct relation between the normalized L' (or L) with divergence

  $\Rightarrow$ As we have seen values in the eigenvector of largest eigenvalue μ_1 for L' is not constant

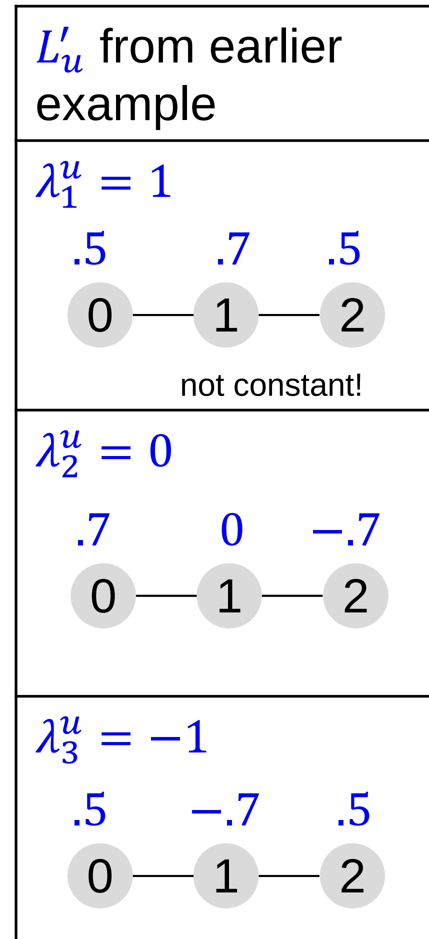
- To see a relationship requires new insights from graph signal processing
  - Values in eigenvectors of smaller eigenvalues for L' vary more rapidly across the graph

Reconciliation with divergence

- Values in eigenvectors of smaller eigenvalues for L' vary more rapidly across the graph

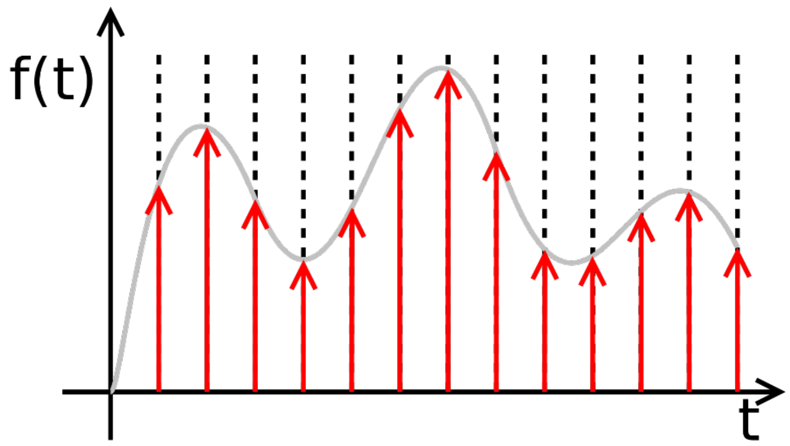
Example:

- At the largest eigenvalue (for L')
  - Not exactly but still, almost constant everywhere
  - Coincides with the lowest divergence case
- At larger eigenvalues (for L')
  - Smaller variation across connected vertices
  - Coincides with lower divergence case
- At small eigenvalues (for L')
  - Large variation across connected vertices
  - Coincides with higher divergence case



Signal processing

- A discrete-time signal is a sequence of (sampled) values $f(0), \ldots, f(N-1)$ of some variable



- Signal processing transforms the signal from one domain to another to detect possible properties
  - Fourier transform converts signals from the time domain into the frequency domain $U(0), \ldots, U(N-1)$

$$U(k) = \sum_{t=0}^{N-1} f(t) \cdot e^{-\frac{i2\pi}{N}kt}$$

- A signal in the time domain is a 1-D vector
  - More flexible if consider as a graph
    - Use **eigenbasis** as transformed domain

Graph Signal Processing

1970 Hall An r-dimensional quadratic placement algorithm

1972 Donath and Hoffman Algorithms for partitioning of graphs and computer logic based on eigenvectors of connected matrices

1973 Fiedler Algebraic connectivity of graphs

Donath and Hoffman Lower bounds for the partitioning of graphs

1975 Fiedler Eigenvectors of acyclic matrices

Fiedler A property of eigenvectors of nonnegative symmetric matrices & its applications to graph theory

1982 Barnes An algorithm for partitioning of nodes of a graph

1984 Barnes and Hoffman Partitioning, spectra and linear programming

1989 Pothen *et al.* Partitioning sparse matrices with eigenvalues of graph

1991 Wei and Cheng Ratio cut partitioning for hierarchical designs

1992 Hagen and Kahng New spectral methods for ratio cut partitioning and clustering

1993 Wu and Leahy An optimal graph theoretic approach to data clustering

1997 Shi and Malik Normalized cuts and image segmentation

2001 Ng *et al.* On spectral clustering: Analysis and an algorithm

2003 Belkin and Niyogi Laplacian eigenmaps for dimensionality reduction and data representation

2009 Hammond *et al.* Wavelets on graph via spectral graph theory

2013 Shuman *et al.* The emerging field of signal processing on graphs

2019 Stanković and Sejdić (Ed) **Vertex-frequency analysis of graph signals**

Interpreting the eigenbasis

- A eigenvector x of the (non-normalized) graph Laplacian L fulfills $Lx = \lambda x$
- Since $Lx = \begin{bmatrix} \Delta f(v_1) \\ \vdots \end{bmatrix}$ *(recall Part 1)*, $\lambda x = \begin{bmatrix} \Delta f(v_1) \\ \vdots \end{bmatrix}$
- The eigenvector x corresponds to the values $f(v)$ where $\lambda f(v) \approx \Delta f(v)$
  - **A small λ indicates that $f(v)$ does not vary much from $f(v')$ of its neighbors v'**
- The smallest λ (for a connected graph) is 0, indicating that $\forall v \Delta f(v) = 0$
  - In which case $f(v) = \text{const}$ (stationary state)

Interpreting the eigenbasis

- A eigenvector $x = [f(v_1) \quad f(v_2) \quad ...]$ of L furthermore minimizes $\frac{x^\top Lx}{x^\top x}$ (Rayleigh quotient)
- Since $Lx = \begin{bmatrix} \Delta f(v_1) \\ \vdots \end{bmatrix}$, we have

$$x^\top Lx = [f(v_1) \quad ...] \begin{bmatrix} \Delta f(v_1) \\ \vdots \end{bmatrix} = \sum_v f(v) \Delta f(v)$$

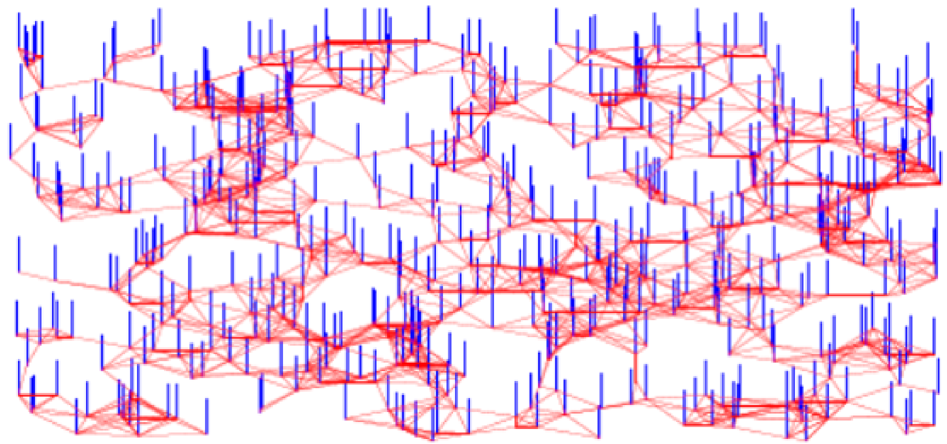
$\Rightarrow x^\top Lx$ = projection of Δf on eigenvector x

$\Rightarrow \frac{x^\top Lx}{x^\top x}$ = projection of Δf on unit eigenvector x

  - Furthermore the projection $\frac{x^\top Lx}{x^\top x} = \lambda$ (eigenvalue of x)
- A eigenvector is **a distribution f that minimizes the total differences between neighboring $f(v)$ values**

Interpreting the eigenbasis

- A eigenvector = a distribution f that minimizes the total differences between neighboring $f(v)$ values
- $f(v)$ values from eigenvector of $\lambda = 0$
  - $f(v) = \text{const}$
    $\Rightarrow$ zero differences

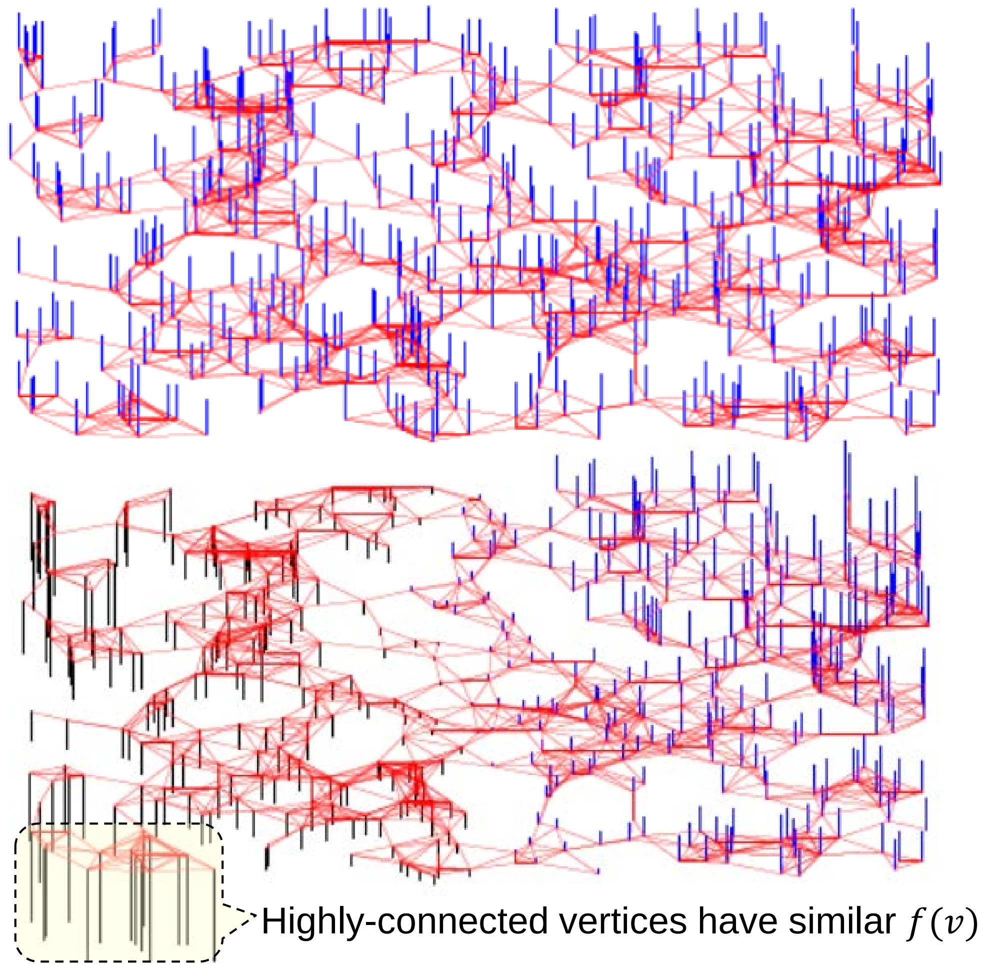


From Shuman *et al. The emerging field of signal processing on graphs*, 2013

- If the graph consists of two disconnected components, the $f(v)$ values of the individual components can have different constant values

Interpreting the eigenbasis

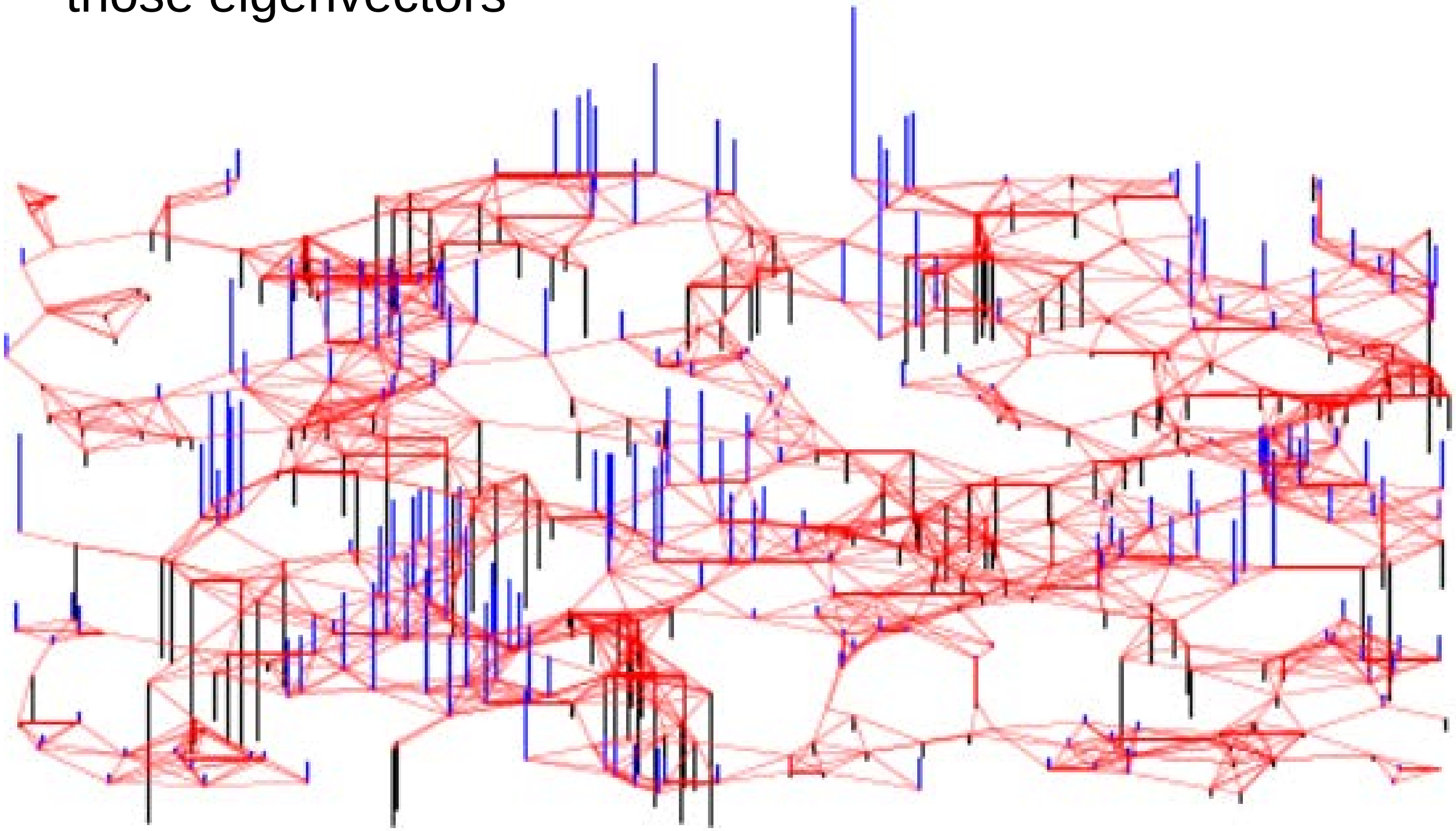
- A eigenvector = a distribution f that minimizes the total differences between neighboring $f(v)$ values
- $f(v)$ values from eigenvector of $\lambda = 0$
  - $f(v) = \text{const}$
    $\Rightarrow$ zero differences
- $f(v)$ values for eigenvector of 2nd smallest λ
  - Orthogonality with eigenvector of $\lambda = 0$ forces large variations in $f(v)$



From Shuman *et al. The emerging field of signal processing on graphs*, 2013

Interpreting the eigenbasis

- $f(v)$ values from eigenvector of 50th smallest λ
  - Orthogonality of this eigenvector with the 1st~49th smallest eigenvectors forces distinctly different variations in $f(v)$ from those eigenvectors



From Shuman *et al. The emerging field of signal processing on graphs*, 2013

Interpreting the eigenbasis

- Further developments on graph Fourier transform leads to the introduction of the Graph Neural Networks

Appendix

Other generalized eigensystem

- A partitioning problem called graph partitioning problem was proposed in (Hendrickson *et al.*, 1996)
- The problem gives rise to an interesting eigensystem $Lx = \lambda Mx$, as pointed out in (Shewchuk, 2011)
- For completeness we discuss this problem here

Graph Partitioning Problem

- Given edge weight matrix $W = (w_{ij})$ and vertex mass matrix M with diagonal elements (m_i), a 2-partitioning of an undirected graph $G = (V, E)$ is a partition of V into two groups S and $\bar{S}$ such that $\mathrm{cut}(S, \bar{S}) = \sum_{i \in S, j \in \bar{S}} w_{ij}$ is minimized under the constraint that $\sum_{i \in S} m_i = \sum_{i \in \bar{S}} m_i$, or $\mathbf{1}^\top M x = 0$
  - Observe that if $m_i = 1$ for all i, then the condition $\sum_{i \in S} m_i = \sum_{i \in \bar{S}} m_i$ is the same as $|S| = |\bar{S}|$

Constrained optimization problem

- Minimize $x^\top Lx$ where $L = D' - W$
  subject to $x^\top M \in \{1, -1\}$ and $\mathbf{1}^\top Mx = 0$
  - $x_i \in \{1, -1\}$ and $\mathbf{1}^\top Mx = 0$ together enforce balance in the solution
  - However, problem is NP-hard
    - Recall that even the minimum bisection problem, where all edges and vertices have the same weight, is NP-hard

Relaxed Rayleigh quotient version

- Minimize $x^\top L x$ where $L = D' - W$ subject to $x^\top M x = \sum_i m_i$ and $\mathbf{1}^\top M x = 0$
  - $x_i \in \{1, -1\} \Rightarrow x^\top M x = \sum_i m_i$ but not the other way around
  - Balance no longer enforced but that's the least of our worry for now because instead of the standard eigensystem
- Optimization must now be achieved through solving the generalized eigensystem

$$Lx = \lambda M x$$

Relaxed Rayleigh quotient version

- Minimize $x^\top L x$ where $L = D' - W$ subject to $x^\top M x = \sum_i m_i$ and $\mathbf{1}^\top M x = 0$
- Optimize through $Lx = \lambda M x$
- Since $\mathbf{1}$ fulfills condition for L and M, $\mu_k = \mathbf{1}$
  - However, eigenvectors in the solutions are not orthogonal but rather, M-orthogonal ($\mu_i M \mu_j = 0$ for $i \neq j$)
    - $\mathbf{1}^\top M \mu_{k-1} = 0$ is fulfilled
- Convert to a standard eigenvalue system $M^{-1/2} L M^{-1/2} x = \lambda x$ to compute

Generalized eigensystem

- Minimize $x^\top L x$ where $L = D' - W$
  subject to $x^\top M x = \sum_i m_i$ and $\mathbf{1}^\top M x = 0$
- Let $y = M^{1/2} x$, that is, $x = M^{-1/2} y$

$$x^\top L x \Rightarrow y^\top M^{-1/2} L M^{-1/2} y$$

$$x^\top M x = \sum_i m_i \Rightarrow y^\top y = \sum_i m_i$$

$$\mathbf{1}^\top M x = 0 \Rightarrow \mathbf{1}^\top M^{1/2} y = 0$$

Hence equivalently

- Minimize $y M^{-1/2} L M^{-1/2} y$
  subject to $y^\top y = \sum_i m_i$ and $\mathbf{1}^\top M^{1/2} y = 0$

Generalized eigensystem

- Minimize $yM^{-1/2}LM^{-1/2}y$
  subject to $y^\top y = \sum_i m_i$ and $\mathbf{1}^\top M^{1/2}y = 0$
- By similar arguments as those for the Normalized Cut problem, it suffices that we eigendecompose $M^{-1/2}LM^{-1/2}$