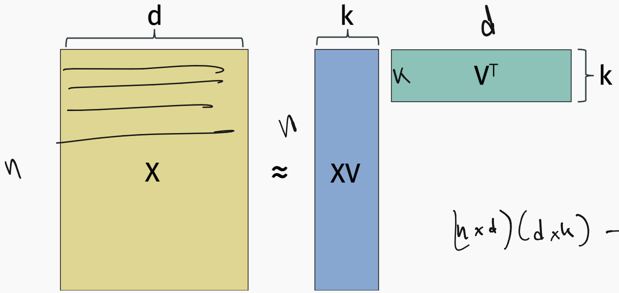
CS-GY 6763: Lecture 10
Spectral clustering, spectral graph theory.

NYU Tandon School of Engineering, Prof. Christopher Musco

LOW-RANK APPROXIMATION

Approximate ~~Write~~ X as a rank k factorization by projecting onto the subspace spanned by an orthonormal matrix $\underline{\mathbf{V}} \in \mathbb{R}^{d \times k}$



$(n \times d)(d \times k) \to n \times k$

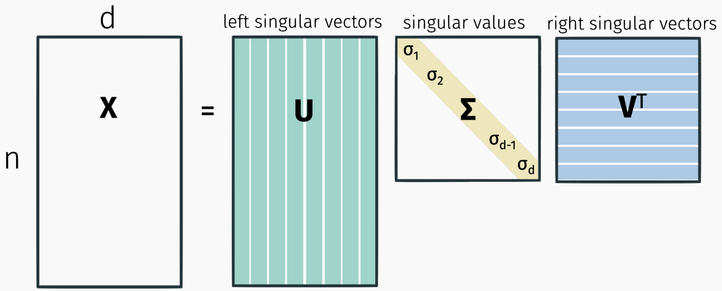
$k \times d$

$(XV)V^T$

SINGULAR VALUE DECOMPOSITION

One-stop shop for computing optimal low-rank approximations.

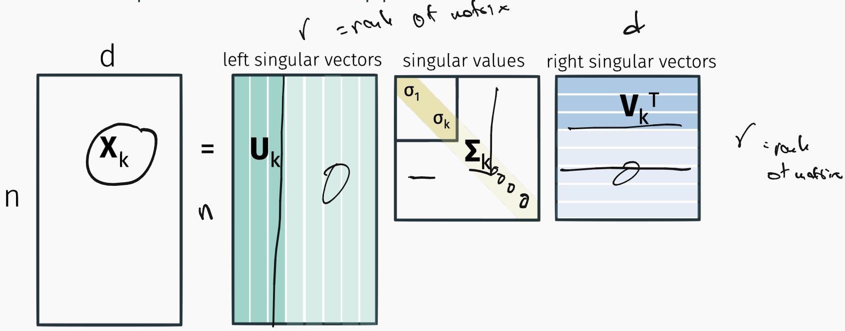
Any matrix **X** can be written:



Where $\mathbf{U}^T\mathbf{U} = \mathbf{I}$, $\mathbf{V}^T\mathbf{V} = \mathbf{I}$, and $\sigma_1 \geq \sigma_2 \geq \ldots \sigma_d \geq 0$.

SINGULAR VALUE DECOMPOSITION

Can read off optimal low-rank approximations from the SVD:

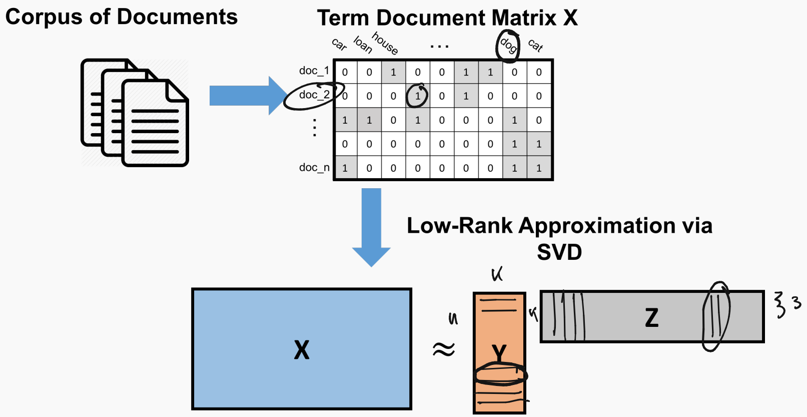


$$X_k = U_k \Sigma_k V_k^T = \boxed{U_k U_k^T X} = \boxed{X V_k V_k^T}.$$

$$X_k = \underset{\text{rank } k \text{ } B}{\arg\min} \|X - B\|_F^2.$$

Can be computed efficiently using power method or more advanced Krylov subspace methods.

APPLICATION OF PARTIAL SVD: ENTITY EMBEDDINGS

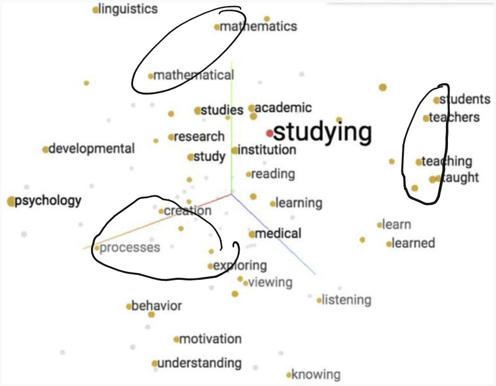


	car	loan	house	…			dog	cat	
doc_1	0	0	1	0	0	1	1	0	0
doc_2	0	0	0	1	0	1	0	0	0
⋮	1	1	0	1	0	0	0	1	0
	0	0	0	0	0	0	0	1	1
doc_n	1	0	0	0	0	0	0	1	1

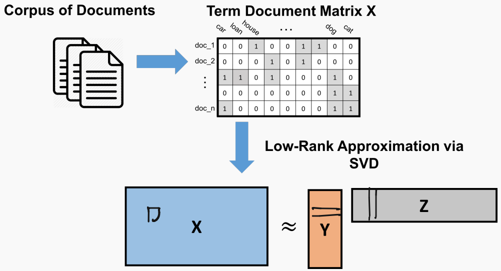
Words tend to map to “similar” columns in **z** if they have similar meaning.

APPLICATION OF PARTIAL SVD: ENTITY EMBEDDINGS

Words tend to map to similar columns in **z** (i.e. similar vectors) if they have similar meaning.

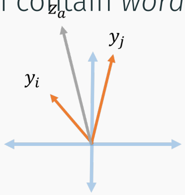


Kastrati, Kurti, Imran 2020

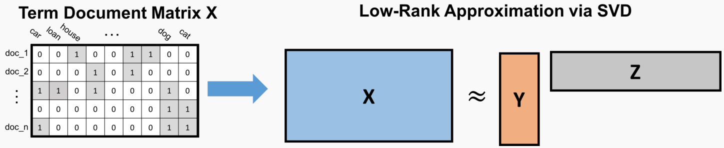


	car	loan	house		…			dog	cat
doc_1	0	0	1	0	0	1	1	0	0
doc_2	0	0	0	1	0	1	0	0	0
⋮	1	1	0	1	0	0	0	1	0
	0	0	0	0	0	0	0	1	1
doc_n	1	0	0	0	0	0	0	1	1

- $\langle \underline{\vec{y}_i}, \underline{\vec{z}_a} \rangle \approx 1$ when $\underline{doc_i}$ contains $\underline{word_a}$.
- If doc_i and doc_j both contain $word_a$, $\langle \vec{y}_i, \underline{\vec{z}_a} \rangle \approx \langle \vec{y}_j, \underline{\vec{z}_a} \rangle = 1$.

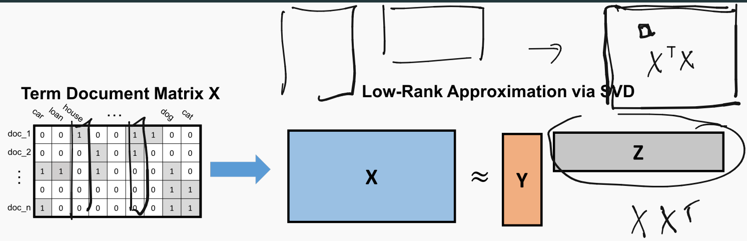


EXAMPLE: LATENT SEMANTIC ANALYSIS



- Similarly the rows $\vec{y}_1, \vec{y}_2, \ldots$ give representations of documents, with $\vec{y}_i$ and $\vec{y}_j$ tending to have high dot product if doc_i and doc_j share many words.
- Can also be used for document search (this was the original application).
- Can naturally be combined with search methods like LSH.

EXAMPLE: LATENT SEMANTIC ANALYSIS



	car	loan	house		…			dog	cat
doc_1	0	0	1	0	0	1	1	0	0
doc_2	0	0	0	1	0	1	0	0	0
⋮	1	1	0	1	0	0	0	1	0
	0	0	0	0	0	0	0	1	1
doc_n	1	0	0	0	0	0	0	1	1

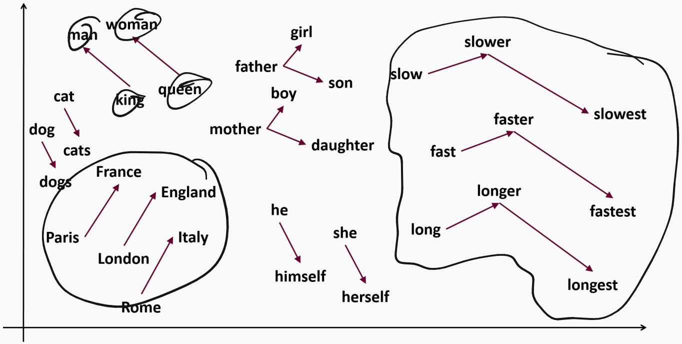
- The columns $\vec{z}_1, \vec{z}_2, \ldots$ give representations of words, with $\vec{z}_i$ and $\vec{z}_j$ tending to have high dot product if *word*$_i$ and *word*$_j$ appear in many of the same documents.
- $\mathbf{Z}$ corresponds to the top k right singular vectors: the eigenvectors of $\mathbf{X}^T\mathbf{X}$. In words, what are the entries in $\mathbf{X}^T\mathbf{X}$?
- $(\mathbf{X}^T\mathbf{X})_{i,j} =$ # of documents both word$_i$ and word$_j$ appear in.

EXAMPLE: WORD EMBEDDINGS

Not obvious how to convert a word into a feature vector that captures the meaning of that word. Approach suggested by LSA: build a $d \times d$ symmetric "similarity matrix" $\mathbf{M}$ between words, and factorize: $\mathbf{M} \approx \mathbf{F}^T\mathbf{F}$ for rank k $\mathbf{F}$.

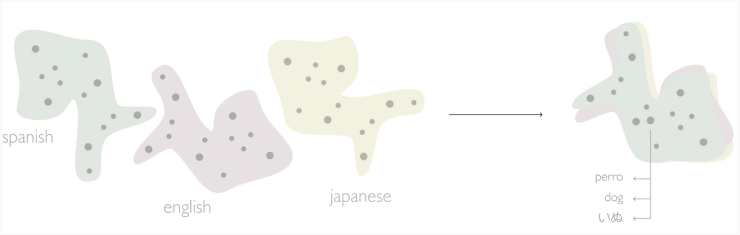
- **Similarity measures:** How often do $word_i, word_j$ appear in the same sentence, in the same window of w words, in similar positions of documents in different languages?
- Replacing $\mathbf{X}^T\mathbf{X}$ with these different metrics (sometimes appropriately transformed) leads to popular word embedding algorithms: `word2vec`, `GloVe`, etc.

EXAMPLE: WORD EMBEDDINGS

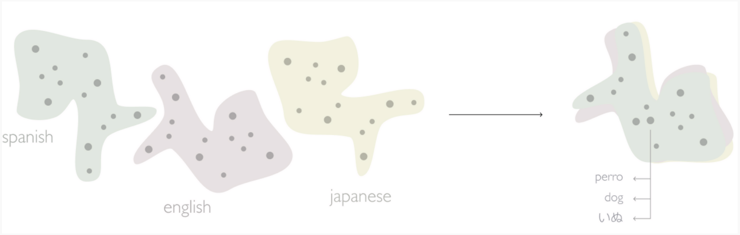


`word2vec` was originally described as a neural-network method, but Levy and Goldberg show that it is simply low-rank approximation of a specific similarity matrix. *Neural word embedding as implicit matrix factorization.*

APPLICATION: UNSUPERVISED TRANSLATION



APPLICATION: UNSUPERVISED TRANSLATION

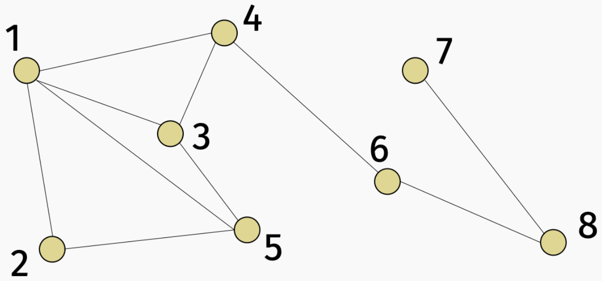


Why not monkey or whale language?

Earth Species Project (www.earthspecies.org), CETI Project (www.projectceti.org)

SPECTRAL GRAPH THEORY

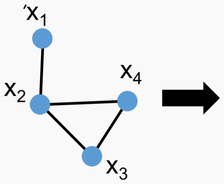
Main idea: Understand graph data by constructing natural matrix representations, and studying that matrix's spectrum (eigenvalues/eigenvectors).



For now assume $G = (V, E)$ is an undirected, unweighted graph with n nodes.

MATRIX REPRESENTATIONS OF GRAPHS

Two most common representations: $n \times n$ adjacency matrix $\mathbf{A}$ and graph Laplacian $\mathbf{L} = \mathbf{D} - \mathbf{A}$ where $\mathbf{D}$ is the diagonal degree matrix.



$$\overset{\mathbf{D}}{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix}} - \overset{\mathbf{A}}{\begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}} = \overset{\mathbf{L}}{\begin{bmatrix} 1 & -1 & 0 & 0 \\ -1 & 3 & -1 & -1 \\ 0 & -1 & 2 & -1 \\ 0 & -1 & -1 & 2 \end{bmatrix}}$$

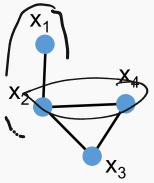
Also common to look at normalized versions of both of these: $\bar{\mathbf{A}} = \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$ and $\bar{\mathbf{L}} = \mathbf{I} - \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$.

SPECTRAL GRAPH THEORY TIDBITS

- If L have k eigenvalues equal to 0, then G has k connected components.
- Sum of cubes of A's eigenvalues is equal to number of triangles in the graph times 6.
- Sum of eigenvalues to the power q is proportional to the number of q cycles.

THE LAPLACIAN VIEW

n: # of nodes   m= # of edges   (n×m)(m×n)

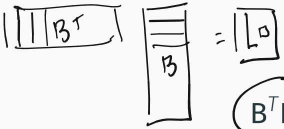


$$\underset{\mathbf{D}}{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix}} - \underset{\mathbf{A}}{\begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}} = n \underset{\boxed{\mathbf{L}}}{\begin{bmatrix} 1 & -1 & 0 & 0 \\ -1 & 3 & \boxed{-1} & -1 \\ 0 & \boxed{-1} & 2 & -1 \\ 0 & -1 & -1 & 2 \end{bmatrix}}_{n} \rightarrow \underline{n \times n}$$

$\boxed{L = B^T B}$ where $\underline{B}$ is the signed "edge-vertex incidence" matrix.

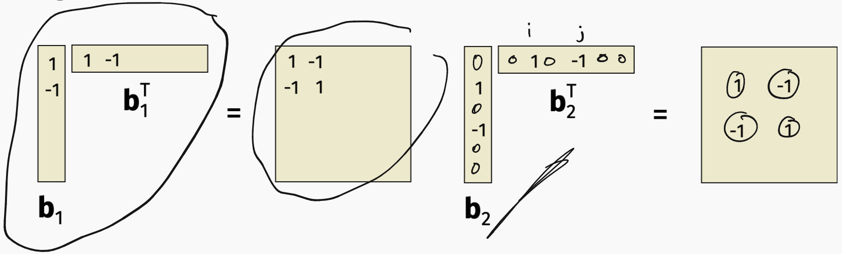
$$B = \begin{array}{cc} & n \\ m & \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & -1 \\ 0 & 1 & 0 & -1 \\ & & & \end{bmatrix} \end{array}$$

THE LAPLACIAN VIEW



$$\boxed{B^TB} = b_1b_1^T + b_2b_2^T + \ldots + b_mb_m^T,$$

where b_i is the i^{th} row of B (each row corresponds to a single edge).



THE LAPLACIAN VIEW

Conclusions from $L = \boxed{B^T B}$

$\underline{x^T L x = \|Bx\|_2^2} \geq 0$ for all x.

- $\mathbf{L}$ is positive semidefinite: $\underline{\mathbf{x}^T\mathbf{L}\mathbf{x} \geq 0}$ $\underline{\text{for all}}$ $\mathbf{x}$.
- $\mathbf{L} = \mathbf{V}\mathbf{\Sigma}^2\mathbf{V}^T$ where $\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ is $\mathbf{B}$'s SVD. Columns of $\mathbf{V}$ are $\underline{\text{eigenvectors}}$ of $\mathbf{L}$.
- For any vector $\underline{\mathbf{x} \in \mathbb{R}^n}$,

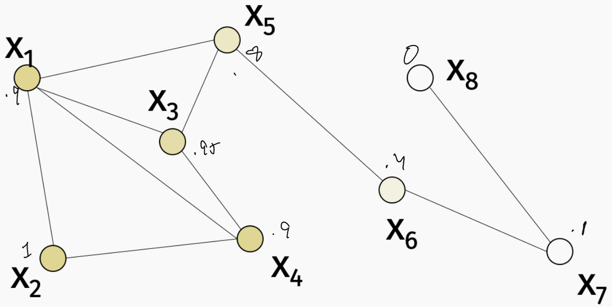
$$\boxed{\mathbf{x}^T L \mathbf{x}} = \boxed{\sum_{(i,j)\in E} (\mathbf{x}(i) - \mathbf{x}(j))^2.}$$

$$\|Bx\|_2^2 \qquad \begin{bmatrix} 1 & -1 & \\ & 1 & -1 \\ & & \\ 1 & -1 & \end{bmatrix} \begin{bmatrix} \\ \end{bmatrix} x = \begin{bmatrix} \leftarrow x(i) - x(j) \\ \\ \end{bmatrix}$$

THE LAPLACIAN VIEW

$x^T L x = \sum_{(i,j)\in E}(x(i) - x(j))^2$. So $x^T L x$ is small if x is a "smooth" function with respect to the graph.

$v^T L v$



Eigenvectors of the Laplacian with small eigenvalues correspond to smooth functions over the graph.

Courant–Fischer min-max principle

Let $V = [v_1, \ldots, v_n]$ be the eigenvectors of L.

$v_1^T L v_1 = \lambda_1 v_1$

$= \lambda_1$

$A^T A$

$A'^T A'$

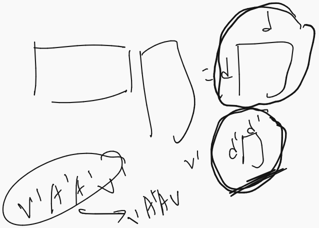
$$v_n = \arg\min_{\|v\|=1} v^T L v$$

$$v_{n-1} = \arg\min_{\|v\|=1, v \perp v_n} v^T L v$$

$$v_{n-2} = \arg\min_{\|v\|=1, v \perp v_n, v_{n-1}} v^T L v$$

$$\vdots$$

$$v_1 = \arg\min_{\|v\|=1, v \perp v_n, \ldots, v_2} v^T L v$$



LARGEST LAPLACIAN EIGENVECTOR

Courant–Fischer min-max principle

Let $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_n]$ be the eigenvectors of $\mathbf{L}$.

$$\mathbf{v}_1 = \underset{\|\mathbf{v}\|=1}{\arg\max}\, \mathbf{v}^T\mathbf{L}\mathbf{v}$$

$$\mathbf{v}_2 = \underset{\underline{\|\mathbf{v}\|=1}, \underline{\mathbf{v}\perp\mathbf{v}_1}}{\arg\max}\, \mathbf{v}^T\mathbf{L}\mathbf{v}$$

$$\mathbf{v}_3 = \underset{\|\mathbf{v}\|=1, \mathbf{v}\perp\mathbf{v}_1,\mathbf{v}_2}{\arg\max}\, \mathbf{v}^T\mathbf{L}\mathbf{v}$$

$$\vdots$$

$$\mathbf{v}_n = \underset{\|\mathbf{v}\|=1, \mathbf{v}\perp\mathbf{v}_1,\ldots,\mathbf{v}_{n-1}}{\arg\max}\, \mathbf{v}^T\mathbf{L}\mathbf{v}$$

$Lv_2 = \lambda v_2$

for some

λ

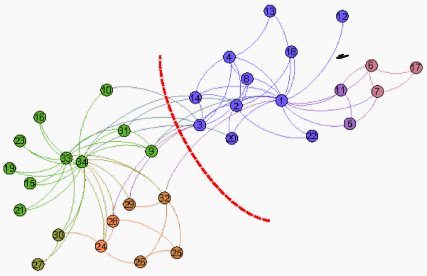
EXAMPLE APPLICATION OF SPECTRAL GRAPH THEORY

- Study graph partitioning problem important in 1) understanding social networks 2) nonlinear clustering in unsupervised machine learning (spectral clustering).
- See how this problem can be solved approximately using Laplacian eigenvectors.
- Give a full analysis of the method for a common random graph model.
- Use two tools: matrix concentration and eigenvector perturbation bounds.

BALANCED CUT

Common goal: Given a graph $G = (V, E)$, partition nodes along a cut that:

- Has few crossing edges: $|\{(u, v) \in E : u \in S, v \in T\}|$ is small.
- Separates large partitions: $|S|, |T|$ are not too small.



(a) Zachary Karate Club Graph

Important in understanding community structure in social networks.

SOCIAL NETWORKS IN THE 1970S

Wayne W. Zachary (1977). An Information Flow Model for Conflict and Fission in Small Groups.

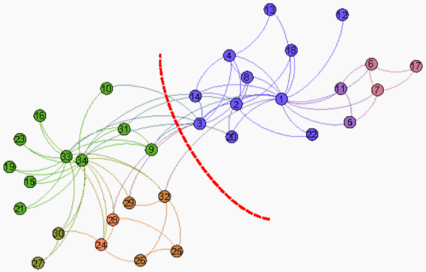
“The network captures 34 members of a karate club, documenting links between pairs of members who interacted outside the club. During the study a conflict arose between the administrator ”John A” and instructor ”Mr. Hi” (pseudonyms), which led to the split of the club into two. Half of the members formed a new club around Mr. Hi; members from the other part found a new instructor or gave up karate. Based on collected data Zachary correctly assigned all but one member of the club to the groups they actually joined after the split.” – Wikipedia

Beautiful paper – definitely worth checking out!

BALANCED CUT

Common goal: Given a graph $G = (V, E)$, partition nodes along a cut that:

- Has few crossing edges: $|\{(u, v) \in E : u \in S, v \in T\}|$ is small.
- Separates large partitions: $|S|, |T|$ are not too small.

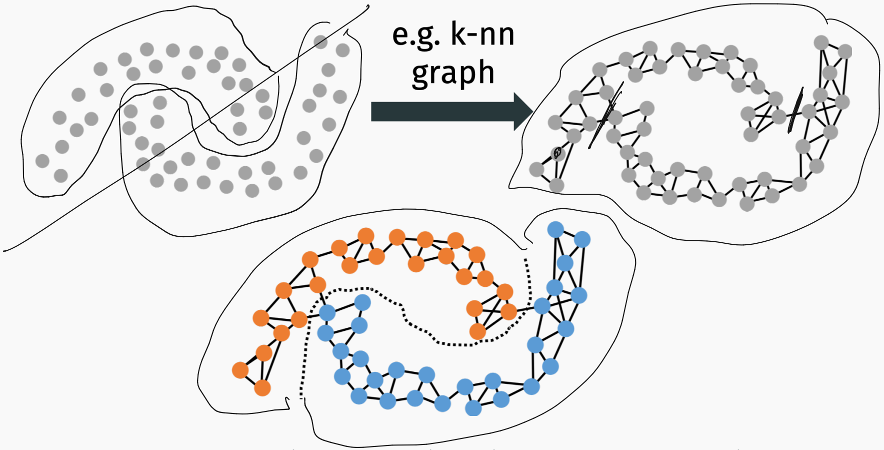


(a) Zachary Karate Club Graph

Important in understanding community structure in social networks.

SPECTRAL CLUSTERING

Idea: Construct synthetic graph for data that is hard to cluster.



Spectral Clustering, Laplacian Eigenmaps, Locally linear embedding, Isomap, etc.

There are many way's to formalize Zachary's problem:

β-Balanced Cut:

$$\min_S \text{cut}(S, V \setminus S) \quad \text{such that} \quad \min(|S|, |V \setminus S|) \geq \beta n \text{ for } \beta \leq .5$$

Sparsest Cut:

$$\min_S \frac{\text{cut}(S, V \setminus S)}{\min(|S|, |V \setminus S|)}$$

Most formalizations lead to computationally hard problems. Lots of interest in designing polynomial time approximation algorithms, but tend to be slow. In practice, much simpler methods based on the graph spectrum are used.

THE LAPLACIAN VIEW

$[-1\ -1\ -1\ 1\ -1\ 1\ 1\ 1]$

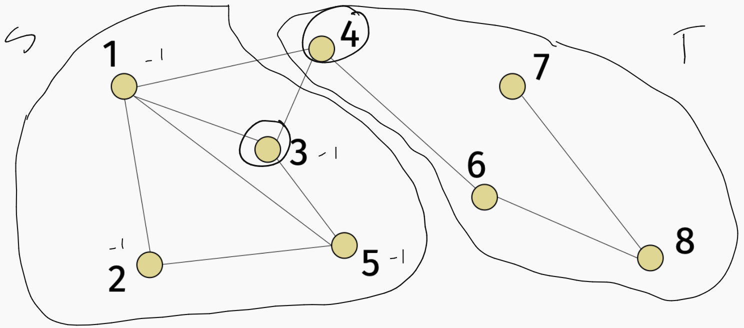
Another conclusion from $\underline{\mathbf{L}} = \underline{\mathbf{B}^T\mathbf{B}}$:

For a cut indicator vector $\mathbf{c} \in \{-1, 1\}^n$ with $\mathbf{c}(i) = -1$ for $i \in S$ and $\mathbf{c}(i) = 1$ for $i \in T = V \setminus S$:

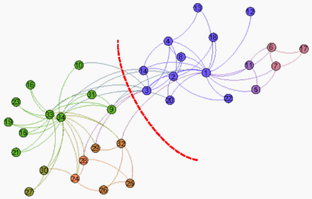
$$\boxed{\mathbf{c}^T L \mathbf{c}} = \sum_{\substack{(i,j)\in E \\ 12}} (\overset{c(3)}{\underset{-1}{\mathbf{c}(i)}} - \overset{-c(4)}{\underset{-\ 1}{\mathbf{c}(j)}})^2 = \boxed{4 \cdot \boxed{\text{cut}(S, T).}} \qquad (1)$$

$= (-2)^2$

c_i



THE LAPLACIAN VIEW



(a) Zachary Karate Club Graph

For a cut indicator vector $\mathbf{c} \in \{-1, 1\}^n$ with $\mathbf{c}(i) = -1$ for $i \in S$ and $\mathbf{c}(i) = 1$ for $i \in T$:

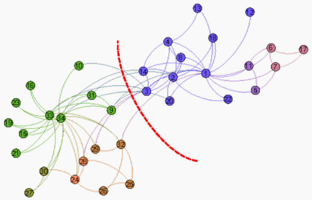
- $\mathbf{c}^T L \mathbf{c} = 4 \cdot cut(S, T)$.
- $\mathbf{c}^T \mathbf{1} = |T| - |S|$.

$$\sum_{i=1}^{n} c(i) = \sum_{i \in T} 1 + \sum_{j \in S} -1$$

$$= |T| - |S|$$

Want to minimize both $\mathbf{c}^T L \mathbf{c}$ (cut size) and $\mathbf{c}^T \mathbf{1}$ (imbalance).

THE LAPLACIAN VIEW



(a) Zachary Karate Club Graph

Equivalent formulation if we divide everything by $\underline{\sqrt{n}}$ so that **c** has norm 1. Then $\mathbf{c} \in \underline{\{-\frac{1}{\sqrt{n}}, \frac{1}{\sqrt{n}}\}}^n$ and:

$\|c\|_2^2 = 1$

- $\underline{\mathbf{c}^T L \mathbf{c} = \frac{4}{n} \cdot cut(S, T)}$.
- $\underline{\mathbf{c}^T \mathbf{1}} = \frac{1}{\sqrt{n}}(|T| - |S|)$.

Want to minimize both $\mathbf{c}^T L \mathbf{c}$ (cut size) and $\mathbf{c}^T \mathbf{1}$ (imbalance).

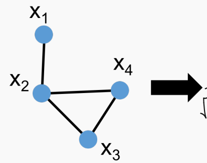
SMALLEST LAPLACIAN EIGENVECTOR

The smallest eigenvector/singular vector $\mathbf{v}_n$ satisfies:

$$\mathbf{v}_n = \frac{1}{\sqrt{n}} \cdot \mathbf{1} = \underset{v\in\mathbb{R}^n \text{ with } \|\mathbf{v}\|=1}{\arg\min} \mathbf{v}^T L \mathbf{v}$$

with $\mathbf{v}_n^T L \mathbf{v}_n = 0$.

$$L v_n = 0 \cdot v_n$$



$$\frac{1}{\sqrt{n}} \overset{\mathbf{L}}{\begin{bmatrix} 1 & -1 & 0 & 0 \\ -1 & 3 & -1 & -1 \\ 0 & -1 & 2 & -1 \\ 0 & -1 & -1 & 2 \end{bmatrix}} \mathbf{1} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

SECOND SMALLEST LAPLACIAN EIGENVECTOR

By Courant-Fischer, $\mathbf{v}_{n-1}$ is given by:

$$\mathbf{v}_{n-1} = \underset{\|\mathbf{v}\|=1,\ \mathbf{v}_n^T\mathbf{v}=0}{\arg\min} \mathbf{v}^T L \mathbf{v}$$

$V_{n-1} \in \{-\frac{1}{\sqrt{n}}, \frac{1}{\sqrt{n}}\}$

If $\mathbf{v}_{n-1}$ were binary $\{-\frac{1}{\sqrt{n}}, \frac{1}{\sqrt{n}}\}^n$ it would have:

- $\mathbf{v}_{n-1}^T L \mathbf{v}_{n-1} = \frac{1}{n}\mathbf{cut}(S,T)$ as small as possible given that $\mathbf{v}_{n-1}^T \mathbf{1} = |T| - |S| = 0$.
- $\mathbf{v}_{n-1}$ would indicate the smallest perfectly balanced cut.

$\mathbf{v}_{n-1} \in \mathbb{R}^n$ is not generally binary, but a natural approach is to 'round' the vector to obtain a cut.

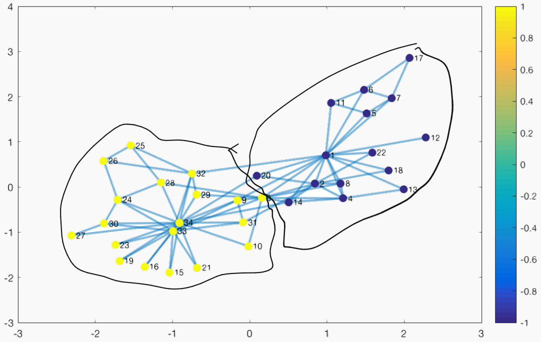
Same idea as in the LP relaxation lecture!

CUTTING WITH THE SECOND LAPLACIAN EIGENVECTOR

Find a good partition of the graph by computing

$$\mathbf{v}_{n-1} = \underset{v\in\mathbb{R}^n \text{ with } \|\mathbf{v}\|=1,\ \mathbf{v}^T\mathbf{1}=0}{\arg\min} \mathbf{v}^T L \mathbf{v}$$

Set S to be all nodes with $\underline{\mathbf{v}_{n-1}(i) < 0}$, and T to be all with $\mathbf{v}_{n-1}(i) \geq 0$.

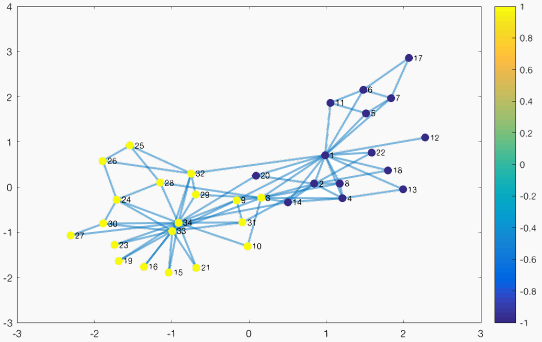


CUTTING WITH THE SECOND LAPLACIAN EIGENVECTOR

Find a good partition of the graph by computing

$$\mathbf{v}_{n-1} = \underset{v\in\mathbb{R}^n \text{ with } \|\mathbf{v}\|=1,\ \mathbf{v}^T\mathbf{1}=0}{\arg\min} \mathbf{v}^T L \mathbf{v}$$

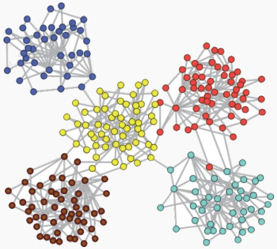
Set S to be all nodes with $\mathbf{v}_{n-1}(i) < 0$, and T to be all with $\mathbf{v}_{n-1}(i) \geq 0$.



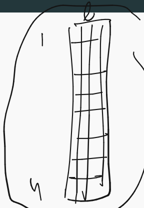
SPECTRAL PARTITIONING IN PRACTICE

The Shi-Malik normalized cuts algorithm is one of the most commonly used variants of this approach, using the normalized Laplacian $\bar{L} = D^{-1/2} L D^{-1/2}$.

Important consideration: What to do when we want to split the graph into more than two parts?



SPECTRAL PARTITIONING IN PRACTICE



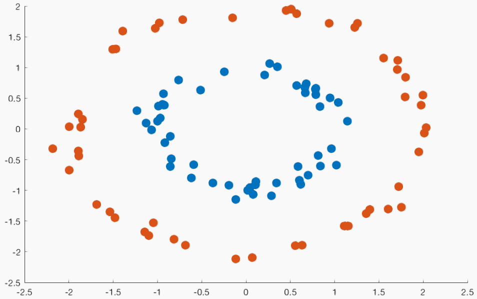
$\ell = 2$

Spectral Clustering:

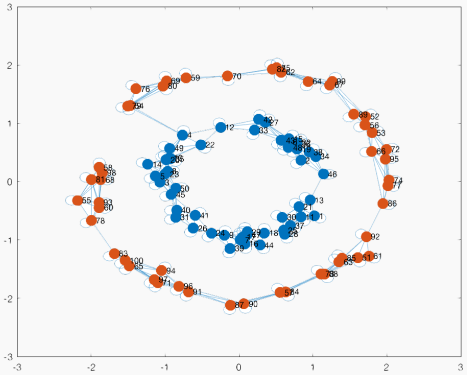
- Compute smallest $\overset{\ell}{\cancel{k}}$ eigenvectors $\underline{\mathbf{v}_{n-1}}, \ldots, \underline{\mathbf{v}_{n-\ell}}$ of $\mathbf{L}$.
- Represent each node by its corresponding row in $\mathbf{V} \in \mathbb{R}^{n \times \ell}$ whose rows are $\mathbf{v}_{n-1}, \ldots \mathbf{v}_{n-\ell}$.
- Cluster these rows using k-means clustering (or really any clustering method).
- Often we choose $\ell = k$, but not necessarily.

LAPLACIAN EMBEDDING

Original Data: (not linearly separable)

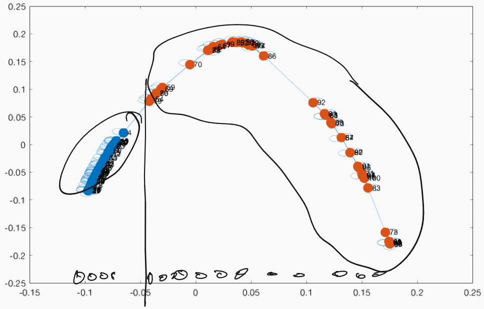


k-Nearest Neighbors Graph:



LAPLACIAN EMBEDDING

Embedding with eigenvectors $\mathbf{v}_{n-1}, \mathbf{v}_{n-2}$: (linearly separable)

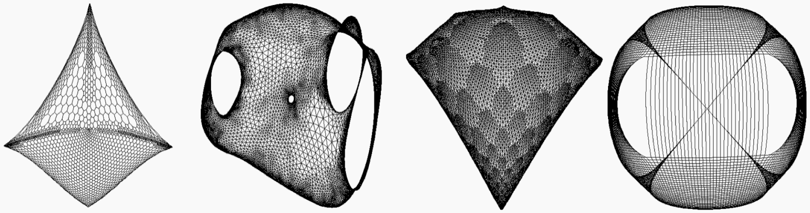


WHY DOES THIS WORK?

Intuitively, since $\mathbf{v} \in \mathbf{v}_1, \ldots \mathbf{v}_k$ are smooth over the graph,

$$\sum_{i,j \in E} (\mathbf{v}[i] - \mathbf{v}[j])^2$$

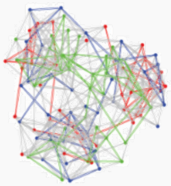
is small for each coordinate. I.e. this embedding explicitly encourages nodes connected by an edge to be placed in nearby locations in the embedding.

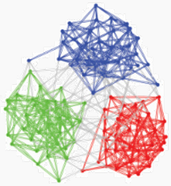


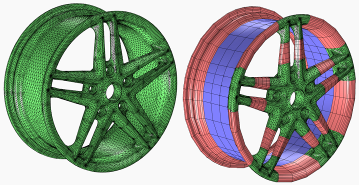
Also useful e.g., in graph drawing.

TONS OF OTHER APPLICATIONS!

Fast balanced partitioning algorithms are also use in distributing data in graph databases, for partitioning finite element meshes in scientific computing (e.g., that arise when solving differential equations), and more.







Lots of good software packages (e.g. METIS).

GENERATIVE MODELS

So far: Showed that spectral clustering partitions a graph along a small cut between large pieces.

- No formal guarantee on the 'quality' of the partitioning.
- Would be difficult to analyze for general input graphs.

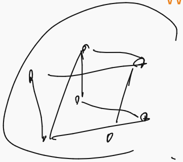
Common approach: Design a natural generative model that produces random but realistic inputs and analyze how the algorithm performs on inputs drawn from this model.

- Very common in algorithm design and analysis. Great way to start approaching a problem.
- This is also the whole idea behind Bayesian Machine Learning (can be used to justify ℓ_2 linear regression, k-means clustering, PCA, etc.)

STOCHASTIC BLOCK MODEL

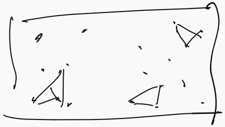
Scale Free

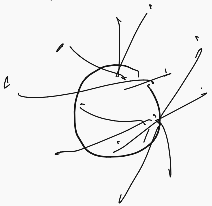
Ideas for a generative model for **social network graphs** that would allow us to understand partitioning?



erdos - renyi

Geometric Random Graphs





Preferential Attachment

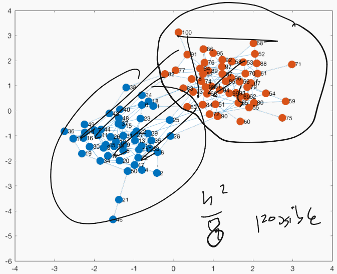
STOCHASTIC BLOCK MODEL

Stochastic Block Model (Planted Partition Model):

Let $G_n(p,q)$ be a distribution over graphs on n nodes, split equally into two groups $\underline{B}$ and $\underline{C}$, each with $n/2$ nodes.

- Any two nodes in the same group are connected with probability p (including self-loops).
- Any two nodes in different groups are connected with prob. $\boxed{q < p}$.

$\binom{n/2}{2}$ $\quad \frac{n^2}{8}$ possible edges



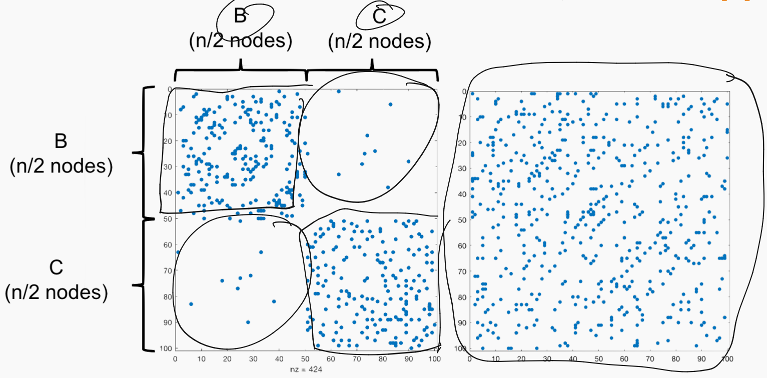
$\frac{n^2}{8}$ possible edges

$\frac{n^2}{4}$ possible edges

LINEAR ALGEBRAIC VIEW

Let G be a stochastic block model graph drawn from $G_n(p, q)$.

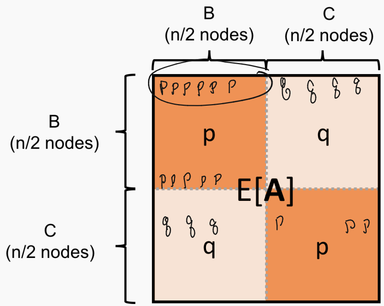
- Let $\mathbf{A} \in \mathbb{R}^{n \times n}$ be the adjacency matrix of G. What is $\mathbb{E}[\mathbf{A}]$?



Note that we are arbitrarily ordering the nodes in A by group. In reality A would look “scrambled” as on the right.

EXPECTED ADJACENCY SPECTRUM

Letting G be a stochastic block model graph drawn from $G_n(p,q)$ and $\mathbf{A} \in \mathbb{R}^{n\times n}$ be its adjacency matrix. $(\mathbb{E}[\mathbf{A}])_{i,j} = p$ for i,j in same group, $(\mathbb{E}[\mathbf{A}])_{i,j} = q$ otherwise.



What are the eigenvectors and eigenvalues of $\mathbb{E}[\mathbf{A}]$?

EXPECTED LAPLACIAN

What is the expected Laplacian $G_n(p, q)$?

$$\mathbb{E}[L] = \mathbb{E}(D - A) = \mathbb{E}[D] - \mathbb{E}[A]$$

$$\downarrow$$

$$\underline{\left(p \cdot \frac{n}{2} + q \cdot n/2\right) \cdot I} = \mathbb{E}(D)$$

$$-A = \mathbb{E}[L]$$

$$\mathbb{E}[A] = \begin{bmatrix} p & q \\ q & p \end{bmatrix}$$

A and **L** have the same eigenvectors and eigenvalue are equal up to a shift.

EXPECTED ADJACENCY SPECTRUM

Letting G be a stochastic block model graph drawn from $G_n(p,q)$ and $\mathbf{A} \in \mathbb{R}^{n\times n}$ be its adjacency matrix, what are the eigenvectors and eigenvalues of $\mathbb{E}[\mathbf{A}]$?

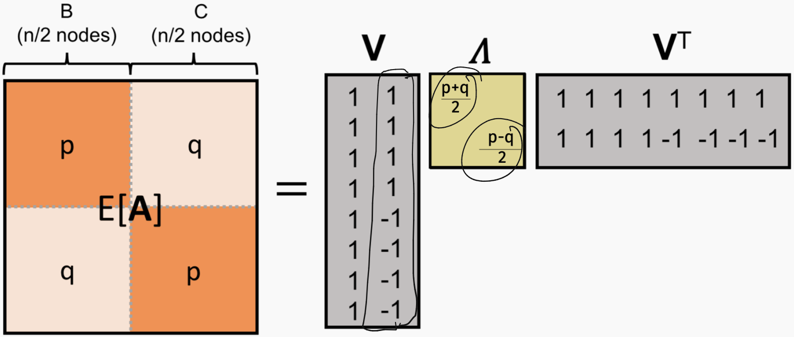
$$\begin{bmatrix} p & q \\ q & p \end{bmatrix} \underbrace{\begin{bmatrix} +1 \\ +1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}}_{x} = \lambda \underbrace{\begin{bmatrix} \\ \\ \\ \end{bmatrix}}_{x}$$

$\rightarrow p \cdot \frac{n}{2} + q \cdot \frac{n}{2}$

$\rightarrow q \cdot \frac{n}{2} + p \cdot \frac{n}{2}$

$$Ax = \left(p \cdot \frac{n}{2} - q \cdot \frac{n}{2}\right) \cdot x \qquad A\mathbb{1} = \left(p \cdot \frac{n}{2} + q \cdot \frac{n}{2}\right) \cdot \mathbb{1}$$

EXPECTED ADJACENCY SPECTRUM



$$\mathbf{E}[\mathbf{A}] = \mathbf{V}\Lambda\mathbf{V}^T,\quad \mathbf{V} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & -1 \\ 1 & -1 \\ 1 & -1 \\ 1 & -1 \end{bmatrix},\quad \Lambda = \begin{bmatrix} \frac{p+q}{2} & \\ & \frac{p-q}{2} \end{bmatrix},\quad \mathbf{V}^T = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \end{bmatrix}$$

- $\mathbf{v}_1 \sim \mathbf{1}$ with eigenvalue $\lambda_1 = \frac{(p+q)n}{2}$.
- $\mathbf{v}_2 \sim \boldsymbol{\chi}_{B,C}$ with eigenvalue $\lambda_2 = \frac{(p-q)n}{2}$.
- $\chi_{B,C}(i) = 1$ if $i \in B$ and $\chi_{B,C}(i) = -1$ for $i \in C$.

If we compute $\mathbf{v}_2$ then we recover the communities B and C!

EXPECTED LAPLACIAN SPECTRUM

Upshot: The second smallest eigenvector of $\mathbb{E}[\mathbf{L}]$, equivalently the second largest of $\mathbb{E}[\mathbf{A}]$, is $\chi_{B,C}$ – the indicator vector for the cut between the communities.

- If the random graph G (equivilantly $\mathbf{A}$ and $\mathbf{L}$) were exactly equal to its expectation, partitioning using this eigenvector would exactly recover communities B and C.

How do we show that a matrix (e.g., $\mathbf{A}$) is close to its expectation? **Matrix concentration inequalities.**

- Analogous to scalar concentration inequalities like Markovs, Chebyshevs, Bernsteins.

MATRIX CONCENTRATION

$G_n(p,q)$

Matrix Concentration Inequality: If $p \geq \underline{O\left(\frac{\log^4 n}{n}\right)}$, then with high probability

$$\|A - \mathbb{E}[A]\|_2 \leq O(\sqrt{pn}).$$

where $\|\cdot\|_2$ is the matrix spectral norm (operator norm).

For $X \in \mathbb{R}^{n\times d}$, $\underline{\|X\|_2} = \max_{z\in\mathbb{R}^d : \underline{\|z\|_2=1}} \|Xz\|_2$.

For the stochastic block model application, we want to show that the second eigenvectors of A and $\mathbb{E}[A]$ are close. How does this relate to their difference in spectral norm?

Eigenvector Perturbation

$\mathbb{E}[A]$

Davis-Kahan Eigenvector Perturbation Theorem: Suppose $\mathbf{A}, \overline{\mathbf{A}} \in \mathbb{R}^{d\times d}$ are symmetric with $\underline{\|\mathbf{A} - \overline{\mathbf{A}}\|_2} \leq \underline{\epsilon}$ and eigenvectors $\underline{v_1, v_2, \ldots, v_d}$ and $\underline{\bar{v}_1, \bar{v}_2, \ldots, \bar{v}_d}$. Letting $\theta(v_i, \bar{v}_i)$ denote the angle between v_i and $\bar{v}_i$, for all i:

want small

$$\sin[\theta(v_i, \bar{v}_i)] \leq \frac{\epsilon}{\min_{j\neq i} |\bar{\lambda}_i - \bar{\lambda}_j|}$$

where $\bar{\lambda}_1, \ldots, \bar{\lambda}_d$ are the eigenvalues of $\overline{\mathbf{A}}$.

The error gets larger if there are eigenvalues with similar magnitudes.

EIGENVECTOR PERTURBATION

$$\underbrace{\begin{bmatrix} 1+\varepsilon & 0 \\ 0 & 1 \end{bmatrix}}_{\mathbf{A}} - \underbrace{\begin{bmatrix} 1 & 0 \\ 0 & 1+\varepsilon \end{bmatrix}}_{\bar{\mathbf{A}}} = \underbrace{\begin{bmatrix} \varepsilon & 0 \\ 0 & \varepsilon \end{bmatrix}}_{\mathbf{A}-\bar{\mathbf{A}}}$$

$$\underset{v_1}{\begin{bmatrix} 1 \\ 0 \end{bmatrix}} \quad \underset{v_2}{\begin{bmatrix} 0 \\ 1 \end{bmatrix}} \qquad \underset{\bar{v}_1}{\begin{bmatrix} 0 \\ 1 \end{bmatrix}} \quad \underset{\bar{v}_2}{\begin{bmatrix} 1 \\ 0 \end{bmatrix}}$$

$$\|A - \bar{A}\|_2 = \varepsilon$$

APPLICATION TO STOCHASTIC BLOCK MODEL

Claim 1 (Matrix Concentration): For $p \geq O\left(\frac{\log^4 n}{n}\right)$,

$$\|\mathbf{A} - \mathbb{E}[\mathbf{A}]\|_2 \leq O(\sqrt{pn}).$$

Claim 2 (Davis-Kahan): For $p \geq O\left(\frac{\log^4 n}{n}\right)$,

$$\sin\theta(v_2, \bar{v}_2) \leq \frac{O(\sqrt{pn})}{\min_{j\neq i}|\lambda_i - \lambda_j|} \leq \frac{O(\sqrt{pn})}{(p-q)n/2} = O\left(\frac{\sqrt{p}}{(p-q)\sqrt{n}}\right)$$

Recall: $\mathbb{E}[\mathbf{A}]$, has eigenvalues $\lambda_1 = \frac{(p+q)n}{2}$, $\lambda_2 = \frac{(p-q)n}{2}$, $\lambda_i = 0$ for $i \geq 3$.

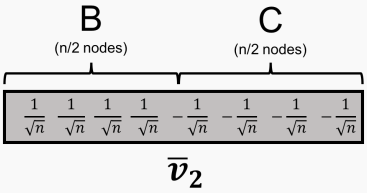
$$\min_{j\neq i}|\lambda_i - \lambda_j| = \min\left(qn, \frac{(p-q)n}{2}\right).$$

Assume $\left|\frac{(p-q)n}{2} - 0\right|$ will be the minimum of the two gaps. I.e. smaller than $\left|\frac{(p+q)n}{2} - \frac{(p-q)n}{2}\right| = qn$.

APPLICATION TO STOCHASTIC BLOCK MODEL

So Far: $\sin\theta(v_2, \bar{v}_2) \leq O\left(\frac{\sqrt{p}}{(p-q)\sqrt{n}}\right)$. What does this give us?

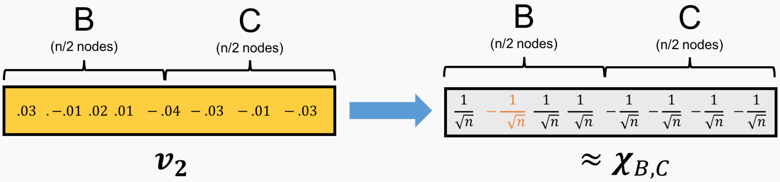
- Can show that this implies $\|v_2 - \bar{v}_2\|_2^2 \leq O\left(\frac{p}{(p-q)^2 n}\right)$ (exercise).
- $\bar{v}_2$ is $\frac{1}{\sqrt{n}}\chi_{B,C}$: the community indicator vector.


B
(n/2 nodes)
C
(n/2 nodes)
$\frac{1}{\sqrt{n}}$ $\frac{1}{\sqrt{n}}$ $\frac{1}{\sqrt{n}}$ $\frac{1}{\sqrt{n}}$ $-\frac{1}{\sqrt{n}}$ $-\frac{1}{\sqrt{n}}$ $-\frac{1}{\sqrt{n}}$ $-\frac{1}{\sqrt{n}}$
$\bar{v}_2$

- Every i where $v_2(i), \bar{v}_2(i)$ differ in sign contributes $\geq \frac{1}{n}$ to $\|v_2 - \bar{v}_2\|_2^2$.
- So they differ in sign in at most $O\left(\frac{p}{(p-q)^2}\right)$ positions.

APPLICATION TO STOCHASTIC BLOCK MODEL

Upshot: If G is a stochastic block model graph with adjacency matrix $\mathbf{A}$, if we compute its second large eigenvector v_2 and assign nodes to communities according to the sign pattern of this vector, we will correctly assign all but $O\left(\frac{p}{(p-q)^2}\right)$ nodes.


B
(n/2 nodes)
C
(n/2 nodes)
.03 .−.01 .02 .01 −.04 −.03 −.01 −.03
v_2
B
(n/2 nodes)
C
(n/2 nodes)
$\frac{1}{\sqrt{n}}$ $-\frac{1}{\sqrt{n}}$ $\frac{1}{\sqrt{n}}$ $\frac{1}{\sqrt{n}}$ $-\frac{1}{\sqrt{n}}$ $-\frac{1}{\sqrt{n}}$ $-\frac{1}{\sqrt{n}}$ $-\frac{1}{\sqrt{n}}$
$\approx \chi_{B,C}$

- Why does the error increase as q gets close to p?
- Even when $p - q = O(1/\sqrt{n})$, assign all but an $O(n)$ fraction of nodes correctly. E.g., assign 99% of nodes correctly.

RANDOMIZED NUMERICAL LINEAR ALGEBRA

Forget about the previous problem, but still consider the matrix $\mathbf{M} = \mathbb{E}[\mathbf{A}]$.

- Dense $n \times n$ matrix.
- Computing top eigenvectors takes $\approx O(n^2/\sqrt{\epsilon})$ time.

If someone asked you to speed this up and return approximate top eigenvectors, what could you do?

We will discuss this more next class!