

Lecture 1: From Data to Graphs, Weighted Graphs and Graph Laplacian

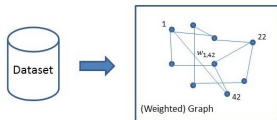
Radu Balan

University of Maryland

March 26, 2024

From Data to Graphs

Datasets Diversity



- Social Networks: Set of individuals ("agents", "actors") interacting with each other (e.g., Facebook, Twitter, joint paper authorship, etc.)
- Communication Networks: Devices (phones, laptops, cars) communicating with each other (emails, spectrum occupancy)
- Transportation networks: Road networks, Air/Sea travel network
- Biological Networks: Macroscale: How animals interact with each other; Microscale: How proteins interact with each other.
- Databases of signals: speech, images, movies; graph relationship tends to reflect signal similarity: the higher the similarity, the larger the weight.
- Chemical networks: chemical compounds; each node is an atom.
- Point Clouds: scanners of 3D objects; aerial LIDAR images.

Databases of Graphs

Public Datasets

Here are several public databases:

- 1 Duke: <https://dnac.ssri.duke.edu/datasets.php>
- 2 Stanford: <https://snap.stanford.edu/data/>
- 3 Uni. Koblenz: <http://konect.uni-koblenz.de/>
- 4 M. Newman (U. Michigan): <http://www-personal.umich.edu/~mejn/netdata/>
- 5 A.L. Barabasi (U. Notre Dame): <http://www3.nd.edu/~networks/resources.htm>
- 6 UCI: <https://networkdata.ics.uci.edu/resources.php>
- 7 Google/YouTube: <https://research.google.com/youtube8m/>
- 8 Chemical Compounds: <http://quantum-machine.org/datasets/>
- 9 Princeton Shape Benchmark: <https://shape.cs.princeton.edu/benchmark/>
- 10 Road networks: <http://networkrepository.com/road.php>
- 11 USGS: <https://catalog.data.gov/dataset/usgs-national-transportation-dataset-ntd-downloadable-data-collectionde7d2>
- 12 FSU: <https://people.sc.fsu.edu/~jburkardt/datasets/cities/cities.html>

Graphs

Typology

Two classes of graphs:

- 1 Geometric graphs: vertices are points in a vector space, e.g. \mathbb{R}^3 ;
- 2 Relational (or abstract) graphs: edges represents similarity or dependency between vertices;

Graphs

Typology

Two classes of graphs:

- 1 Geometric graphs: vertices are points in a vector space, e.g. \mathbb{R}^3 ;
- 2 Relational (or abstract) graphs: edges represents similarity or dependency between vertices;
- 1 Unweighted graphs: edges carry same weight (1).
- 2 Weighted undirected graphs: edges may have different weights; the weight matrix is symmetric.
- 3 Directed graphs: edges are oriented and the weight matrix is not symmetric.

Weighted Graphs

W

The main goal of this lecture is to introduce basic concepts of weighted and undirected graphs, its associated graph Laplacian, and geometric graphs.

Graphs (and weights) reflect either similarity between nodes, or functional dependency, or interaction strength.

- SIMILARITY: Distance, similarity between nodes \Rightarrow weight $w_{i,j}$
- PREDICTIVE/DEPENDENCY: How node i is predicted by its neighbor node $j \Rightarrow$ weight $w_{i,j}$
- INTERACTION: Force, or interaction energy between atom i and atom j

Definitions

$$G = (\mathcal{V}, \mathcal{E}) \text{ and } G = (\mathcal{V}, \mathcal{E}, w)$$

An *undirected graph* G is given by two pieces of information: a set of *vertices* \mathcal{V} and a set of *edges* \mathcal{E} , $G = (\mathcal{V}, \mathcal{E})$.

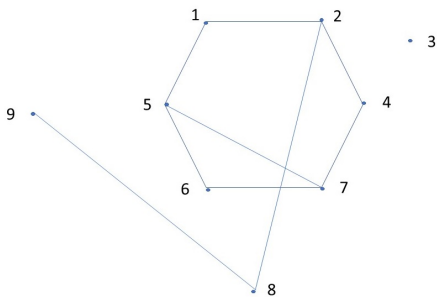
A *weighted graph* has three pieces of information: $G = (\mathcal{V}, \mathcal{E}, w)$, the set of vertices \mathcal{V} , the set of edges \mathcal{E} , and a *weight function* $w : \mathcal{E} \rightarrow \mathbb{R}$.

Definitions

$$G = (\mathcal{V}, \mathcal{E}) \text{ and } G = (\mathcal{V}, \mathcal{E}, w)$$

An *undirected graph* G is given by two pieces of information: a set of *vertices* \mathcal{V} and a set of *edges* \mathcal{E} , $G = (\mathcal{V}, \mathcal{E})$.

A *weighted graph* has three pieces of information: $G = (\mathcal{V}, \mathcal{E}, w)$, the set of vertices \mathcal{V} , the set of edges \mathcal{E} , and a *weight function* $w : \mathcal{E} \rightarrow \mathbb{R}$.



$$\mathcal{V} = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$$

$$\mathcal{E} = \{(1, 2), (2, 4), (4, 7), (6, 7), (1, 5), (5, 6), (5, 7), (2, 8), (8, 9)\}$$

9 vertices, 9 edges

Undirected graph, edges are not oriented. Thus $(1, 2) \sim (2, 1)$.

Definitions

$$G = (\mathcal{V}, \mathcal{E})$$

A weighted graph $G = (\mathcal{V}, \mathcal{E}, w)$ can be directed or undirected depending whether $w(i, j) = w(j, i)$.

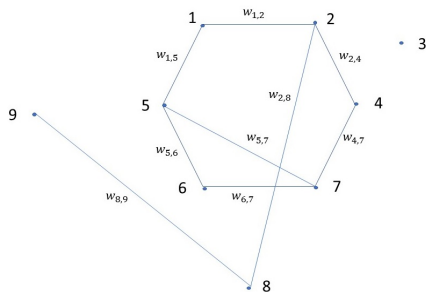
Symmetric weights \implies Undirected graphs

Definitions

$$G = (\mathcal{V}, \mathcal{E})$$

A weighted graph $G = (\mathcal{V}, \mathcal{E}, w)$ can be directed or undirected depending whether $w(i, j) = w(j, i)$.

Symmetric weights \implies Undirected graphs



Example of a Weighted Graph

UCINET IV Datasets: Bernard & Killworth Office

Available online at:

<http://vlado.fmf.uni-lj.si/pub/networks/data/ucinet/ucidata.htm>

Content: Two 40×40 matrices: symmetric (B) and non-symmetric (C) Bernard & Killworth, later with the help of Sailer, collected five sets of data on human interactions in bounded groups and on the actors' ability to recall those interactions. In each study they obtained measures of social interaction among all actors, and ranking data based on the subjects' memory of those interactions. The names of all cognitive (recall) matrices end in C, those of the behavioral measures in B.

These data concern interactions in a small business office, again recorded by an "unobtrusive" observer. Observations were made as the observer patrolled a fixed route through the office every fifteen minutes during two four-day periods. BKOFFB contains the observed frequency of interactions; BKOFFC contains rankings of interaction frequency as recalled by the employees over the two week period.

Example of a Weighted Graph

UCINET IV Datasets: Bernard & Killworth Office

bkoff.dat (similar for *bkfrat.dat*)

```
DL
N=40 NM=2
FORMAT = FULLMATRIX DIAGONAL PRESENT
LEVEL LABELS:
BKOFFB
BKOFFC
DATA:
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
0 0 0 0 0 1 0 0 0 0 0 2 1 0
0 0 4 8 0 3 3 0 1 1 0 0 3 0 0 0 2 1 1 0 0 2 2 0 0 0
0 1 0 0 2 9 0 1 1 0 0 3 0 0
0 4 0 2 1 0 14 0 0 0 1 0 0 1 1 1 0 0 2 0 0 10 0 1 1 0
0 0 0 4 0 1 0 1 8 1 0 0 2 1
```

Example of a Weighted Graph

UCINET IV Datasets: Bernard & Killworth Office

...
 0 0 1 3 0 0 2 0 0 0 0 1 0 5 0 0 0 0 0 0 0 5 0 0 0 0

0 1 1 0 0 0 2 4 5 0 0 0 0 0

0 27 3 36 23 34 14 19 13 9 3 26 21 25 1 8 22 12 11 4 2 37 35 17 5 20
 7 33 32 39 38 16 28 30 29 24 6 10 18 31

...
 29 38 17 4 31 37 6 35 36 22 17 24 39 20 19 26 12 30 32 28 25 1 18 14 33
 34
 27 8 9 21 11 10 5 3 2 15 23 16 13 0

For bkfrat.dat, a similar structure: Lines 8 to 181 contain the symmetric weight matrix. Note, that each row in W takes three lines in the .dat file (due to file formatting).

Example of a Geometric Graph

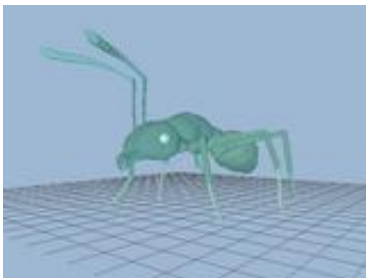


Figure: m2

Movie: See the link

Dataset: Point Clouds from Princeton Shape Benchmark (PSB)

<https://shape.cs.princeton.edu/benchmark/>
File: 2; Ant

$n=3495$ vertices (6667 faces; 61 connected components)

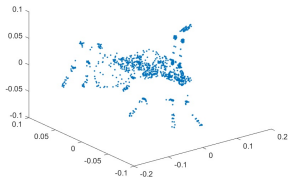


Figure: Graph Embedding

Project 2.1: Euclidian Embeddings and Image Registration

Project on Geometric Graphs

Projects deal with optimization problems related to geometric graphs.

Input Data: (i) a $n \times n$ symmetric matrix of (noisy) pairwise distances between n vertices; (ii) Data set of n -node graphs.

Objectives:

- 1 Find a 3D embedding of the graph (i.e., find coordinates of these points);
- 2 Find the optimal rigid transformation to each of shapes in the data set, compute the residual errors and select the "closest" shape from dataset;
- 3 Create a movie of the continuous transformation.
- 4 Repeat first two questions when only a subset of the $n(n - 1)/2$ pairwise distances are known.

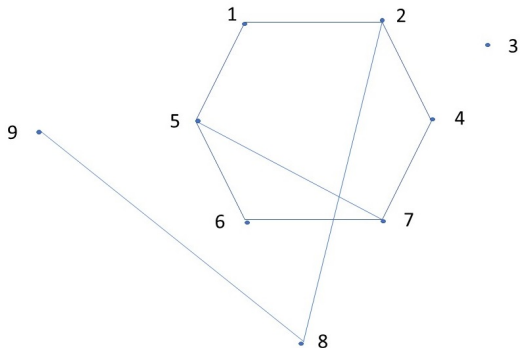
(see movie)

Definitions

Paths

Concept: A **path** is a sequence of edges where the right vertex of one edge coincides with the left vertex of the following edge.

Example:

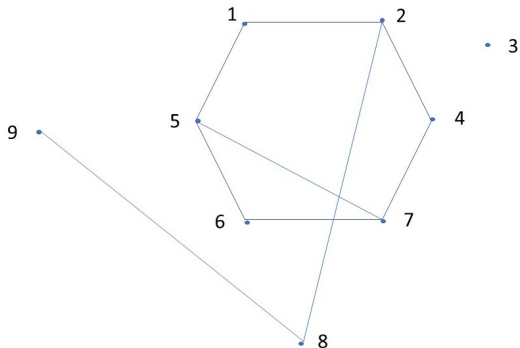


Definitions

Paths

Concept: A **path** is a sequence of edges where the right vertex of one edge coincides with the left vertex of the following edge.

Example:



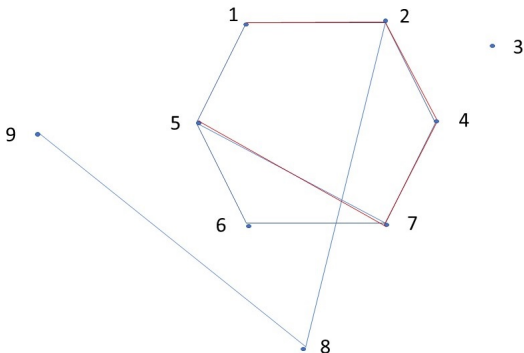
$$\begin{aligned} \{(1, 2), (2, 4), (4, 7), (7, 5)\} &= \\ &= \{(1, 2), (2, 4), (4, 7), (5, 7)\} \end{aligned}$$

Definitions

Paths

Concept: A **path** is a sequence of edges where the right vertex of one edge coincides with the left vertex of the following edge.

Example:



$$\begin{aligned} & \{(1, 2), (2, 4), (4, 7), (7, 5)\} = \\ & = \{(1, 2), (2, 4), (4, 7), (5, 7)\} \end{aligned}$$

Definitions

Graph Attributes

Graph Attributes (Properties):

- **Connected Graphs:** Graphs where any two distinct vertices can be connected through a path.

Definitions

Graph Attributes

Graph Attributes (Properties):

- **Connected Graphs:** Graphs where any two distinct vertices can be connected through a path.
- **Complete (or Totally Connected) Graphs:** Graphs where any two distinct vertices are connected by an edge.

Definitions

Graph Attributes

Graph Attributes (Properties):

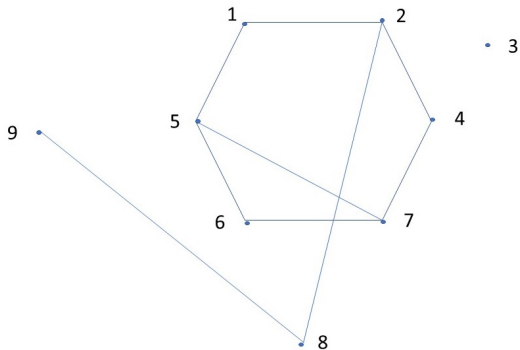
- **Connected Graphs:** Graphs where any two distinct vertices can be connected through a path.
- **Complete (or Totally Connected) Graphs:** Graphs where any two distinct vertices are connected by an edge.

A complete graph with n vertices has $m = \binom{n}{2} = \frac{n(n-1)}{2}$ edges.

Definitions

Graph Attributes

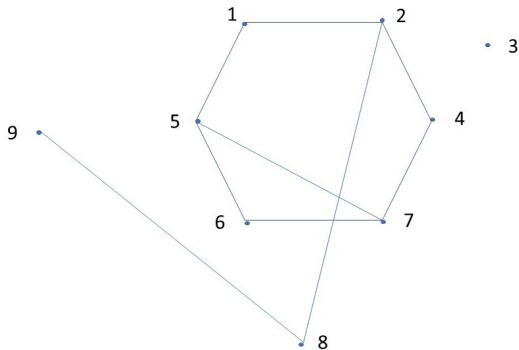
Example:



Definitions

Graph Attributes

Example:



- 3
- This graph is not connected.
- It is not complete.
- It is the union of two connected graphs.

Definitions

Metric

Distance between vertices: For two vertices x, y , the distance $d(x, y)$ is the length of the shortest path connecting x and y . If $x = y$ then $d(x, x) = 0$.

Definitions

Metric

Distance between vertices: For two vertices x, y , the distance $d(x, y)$ is the length of the shortest path connecting x and y . If $x = y$ then $d(x, x) = 0$. In a connected graph the distance between any two vertices is finite. In a complete graph the distance between any two distinct vertices is 1.

Definitions

Metric

Distance between vertices: For two vertices x, y , the distance $d(x, y)$ is the length of the shortest path connecting x and y . If $x = y$ then $d(x, x) = 0$. In a connected graph the distance between any two vertices is finite. In a complete graph the distance between any two distinct vertices is 1.

The converses are also true:

- 1 If $\forall x, y \in \mathcal{V}, d(x, y) < \infty$ then $(\mathcal{V}, \mathcal{E})$ is connected.
- 2 If $\forall x \neq y \in \mathcal{V}, d(x, y) = 1$ then $(\mathcal{V}, \mathcal{E})$ is complete.

Definitions

Metric

Graph diameter: The diameter of a graph $G = (\mathcal{V}, \mathcal{E})$ is the largest distance between two vertices of the graph:

$$D(G) = \max_{x, y \in \mathcal{V}} d(x, y)$$

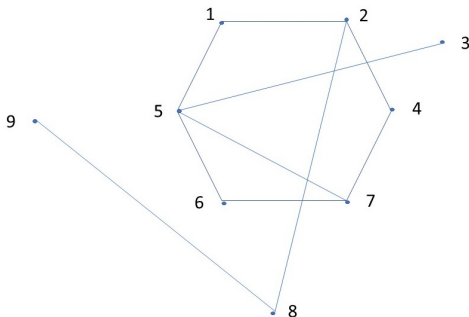
Definitions

Metric

Graph diameter: The diameter of a graph $G = (\mathcal{V}, \mathcal{E})$ is the largest distance between two vertices of the graph:

$$D(G) = \max_{x, y \in \mathcal{V}} d(x, y)$$

Example:



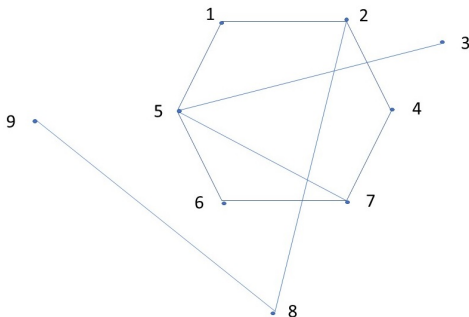
Definitions

Metric

Graph diameter: The diameter of a graph $G = (\mathcal{V}, \mathcal{E})$ is the largest distance between two vertices of the graph:

$$D(G) = \max_{x, y \in \mathcal{V}} d(x, y)$$

Example:



$$D = 5 = d(6, 9) = d(3, 9)$$

Definitions

The Adjacency Matrix

For a graph $G = (\mathcal{V}, \mathcal{E})$ the **adjacency** matrix is the $n \times n$ matrix A defined by:

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

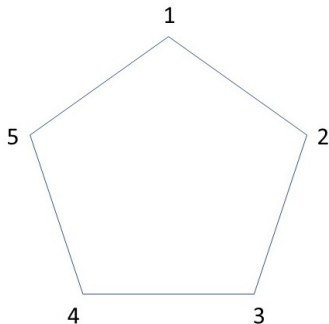
Definitions

The Adjacency Matrix

For a graph $G = (\mathcal{V}, \mathcal{E})$ the **adjacency** matrix is the $n \times n$ matrix A defined by:

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

Example:



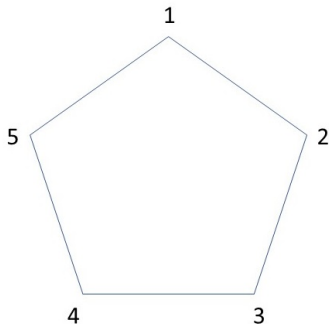
Definitions

The Adjacency Matrix

For a graph $G = (\mathcal{V}, \mathcal{E})$ the **adjacency** matrix is the $n \times n$ matrix A defined by:

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

Example:



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

Definitions

The Adjacency Matrix

For undirected graphs the adjacency matrix is always symmetric:

$$A^T = A$$

For directed graphs the adjacency matrix may not be symmetric.

Definitions

The Adjacency Matrix

For undirected graphs the adjacency matrix is always symmetric:

$$A^T = A$$

For directed graphs the adjacency matrix may not be symmetric.

For weighted graphs $G = (\mathcal{V}, \mathcal{E}, W)$, the **weight** matrix W is simply given by

$$W_{i,j} = \begin{cases} w_{i,j} & \text{if } (i,j) \in \mathcal{E} \\ 0 & \text{otherwise} \end{cases}$$

Degree Matrix

$d(v)$ and D

For an undirected graph $G = (\mathcal{V}, \mathcal{E})$, let $d_v = d(v)$ denote the number of edges at vertex $v \in \mathcal{V}$. The number $d(v)$ is called the **degree** (or valency) of vertex v . The n -vector $d = (d_1, \dots, d_n)^T$ can be computed by

$$d = A \cdot \mathbf{1}$$

where A denotes the adjacency matrix, and $\mathbf{1}$ is the vector of 1's, $\text{ones}(n,1)$.

Let D denote the diagonal matrix formed from the degree vector d : $D_{k,k} = d_k$, $k = 1, 2, \dots, n$. D is called the *degree matrix*.

Degree Matrix

$d(v)$ and D

For an undirected graph $G = (\mathcal{V}, \mathcal{E})$, let $d_v = d(v)$ denote the number of edges at vertex $v \in \mathcal{V}$. The number $d(v)$ is called the **degree** (or valency) of vertex v . The n -vector $d = (d_1, \dots, d_n)^T$ can be computed by

$$d = A \cdot \mathbf{1}$$

where A denotes the adjacency matrix, and $\mathbf{1}$ is the vector of 1's, ones($n,1$).

Let D denote the diagonal matrix formed from the degree vector d : $D_{k,k} = d_k$, $k = 1, 2, \dots, n$. D is called the *degree matrix*.

Key observation: $(D - A) \cdot \mathbf{1} = 0$ always holds. This means the matrix $D - A$ has a non-zero null-space (kernel), hence it is rank deficient.

Degree Matrix

$d(v)$ and D

For an undirected graph $G = (\mathcal{V}, \mathcal{E})$, let $d_v = d(v)$ denote the number of edges at vertex $v \in \mathcal{V}$. The number $d(v)$ is called the **degree** (or valency) of vertex v . The n -vector $d = (d_1, \dots, d_n)^T$ can be computed by

$$d = A \cdot \mathbf{1}$$

where A denotes the adjacency matrix, and $\mathbf{1}$ is the vector of 1's, $\text{ones}(n,1)$.

Let D denote the diagonal matrix formed from the degree vector d : $D_{k,k} = d_k$, $k = 1, 2, \dots, n$. D is called the *degree matrix*.

Key observation: $(D - A) \cdot \mathbf{1} = 0$ always holds. This means the matrix $D - A$ has a non-zero null-space (kernel), hence it is rank deficient.

Second observation: The dimension of the null-space of $D - A$ equals the number of connected components in the graph.

Vertex Degree

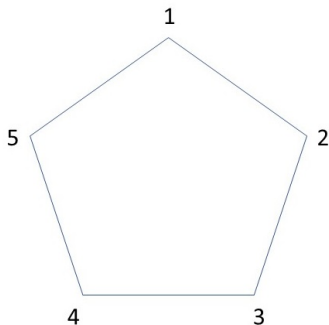
Matrix D

For an undirected graph $G = (\mathcal{V}, \mathcal{E})$ of n vertices, we denote by D the $n \times n$ diagonal matrix of degrees: $D_{i,i} = d(i)$.

Vertex Degree

Matrix D

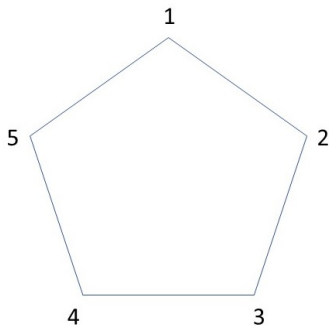
For an undirected graph $G = (\mathcal{V}, \mathcal{E})$ of n vertices, we denote by D the $n \times n$ diagonal matrix of degrees: $D_{i,i} = d(i)$. Example:



Vertex Degree

Matrix D

For an undirected graph $G = (\mathcal{V}, \mathcal{E})$ of n vertices, we denote by D the $n \times n$ diagonal matrix of degrees: $D_{i,i} = d(i)$. Example:



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

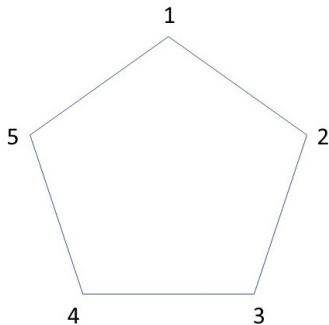
Graph Laplacian

 Δ

For a graph $G = (\mathcal{V}, \mathcal{E})$ the **graph Laplacian** is the $n \times n$ symmetric matrix Δ defined by:

$$\Delta = D - A$$

Example:



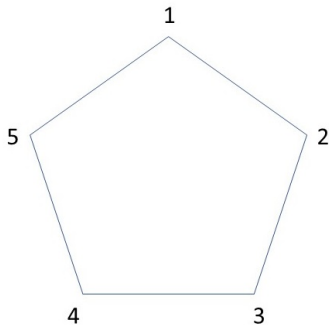
Graph Laplacian

 Δ

For a graph $G = (\mathcal{V}, \mathcal{E})$ the **graph Laplacian** is the $n \times n$ symmetric matrix Δ defined by:

$$\Delta = D - A$$

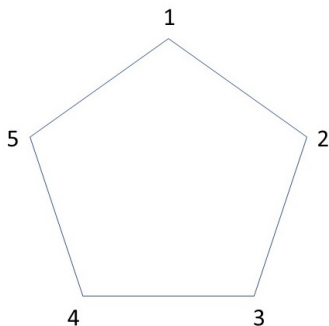
Example:



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

Graph Laplacian

Intuition

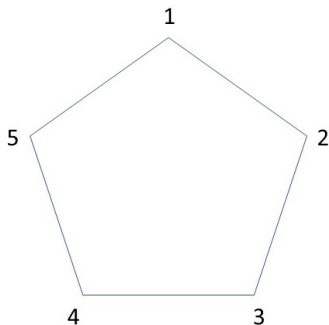


Assume $x = [x_1, x_2, x_3, x_4, x_5]^T$ is a signal of five components defined over the graph. The *Dirichlet* energy E , is defined as

$$E = \sum_{\substack{(i,j) \in \mathcal{E} \\ i < j}} (x_i - x_j)^2 =$$

Graph Laplacian

Intuition



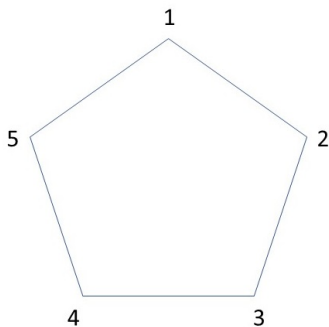
Assume $x = [x_1, x_2, x_3, x_4, x_5]^T$ is a signal of five components defined over the graph. The *Dirichlet* energy E , is defined as

$$E = \sum_{\substack{(i,j) \in \mathcal{E} \\ i < j}} (x_i - x_j)^2 = (x_2 - x_1)^2 + (x_3 - x_2)^2$$

$$+ (x_4 - x_3)^2 + (x_5 - x_4)^2 + (x_1 - x_5)^2.$$

Graph Laplacian

Intuition



Assume $x = [x_1, x_2, x_3, x_4, x_5]^T$ is a signal of five components defined over the graph. The *Dirichlet* energy E , is defined as

$$E = \sum_{\substack{(i,j) \in \mathcal{E} \\ i < j}} (x_i - x_j)^2 = (x_2 - x_1)^2 + (x_3 - x_2)^2$$

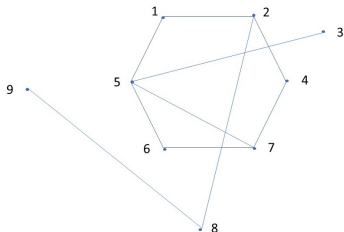
$$+ (x_4 - x_3)^2 + (x_5 - x_4)^2 + (x_1 - x_5)^2.$$

By regrouping the terms we obtain:

$$E = \langle \Delta x, x \rangle = x^T \Delta x = x^T (D - A)x$$

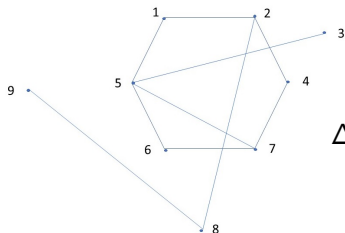
Graph Laplacian

Example



Graph Laplacian

Example



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

Normalized Laplacians

 $\tilde{\Delta}$

Normalized Laplacian: (using pseudo-inverses)

$$\tilde{\Delta} = D^{\dagger/2} \Delta D^{\dagger/2} = \tilde{I} - D^{\dagger/2} A D^{\dagger/2}$$

$$\tilde{\Delta}_{i,j} = \begin{cases} 1 & \text{if } i = j \text{ and } d_i > 0 \text{ (non - isolated vertex)} \\ -\frac{1}{\sqrt{d(i)d(j)}} & \text{if } (i,j) \in \mathcal{E} \\ 0 & \text{otherwise} \end{cases}$$

where \tilde{I} denotes the diagonal matrix: $\tilde{I}_{k,k} = 1$ if $d(k) > 0$ and $\tilde{I}_{k,k} = 0$ otherwise. Thus \tilde{I} is the identity matrix if and only if the graph has no isolated vertices.

D^{\dagger} denotes the pseudo-inverse, which is the diagonal matrix with elements:

$$D^{\dagger}_{k,k} = \begin{cases} \frac{1}{d(k)} & \text{if } d(k) > 0 \\ 0 & \text{if } d(k) = 0 \end{cases}$$

Normalized Laplacians

L

Normalized Asymmetric Laplacian:

$$L = D^\dagger \Delta = \tilde{I} - D^\dagger A$$

$$L_{i,j} = \begin{cases} 1 & \text{if } i = j \text{ and } d_i > 0 \text{ (non - isolated vertex)} \\ -\frac{1}{d(i)} & \text{if } (i,j) \in \mathcal{E} \\ 0 & \text{otherwise} \end{cases}$$

Normalized Laplacians

L

Normalized Asymmetric Laplacian:

$$L = D^\dagger \Delta = \tilde{I} - D^\dagger A$$

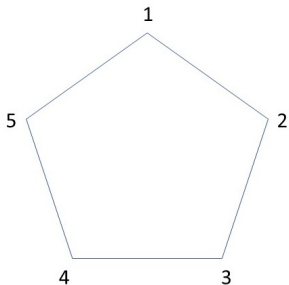
$$L_{i,j} = \begin{cases} 1 & \text{if } i = j \text{ and } d_i > 0 \text{ (non - isolated vertex)} \\ -\frac{1}{d(i)} & \text{if } (i,j) \in \mathcal{E} \\ 0 & \text{otherwise} \end{cases}$$

$$\Delta D^\dagger = \tilde{I} - A D^\dagger = L^T$$

Normalized Laplacians

Example

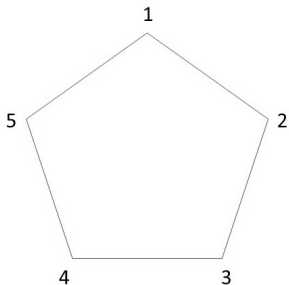
Example:



Normalized Laplacians

Example

Example:



$$\tilde{\Delta} = \begin{bmatrix} 1 & -0.5 & 0 & 0 & -0.5 \\ -0.5 & 1 & -0.5 & 0 & 0 \\ 0 & -0.5 & 1 & -0.5 & 0 \\ 0 & 0 & -0.5 & 1 & -0.5 \\ -0.5 & 0 & 0 & -0.5 & 1 \end{bmatrix}$$

Laplacian and Normalized Laplacian for Weighted Graphs

In the case of a weighted graph, $G = (\mathcal{V}, \mathcal{E}, w)$, the weight matrix W replaces the adjacency matrix A .

Laplacian and Normalized Laplacian for Weighted Graphs

In the case of a weighted graph, $G = (\mathcal{V}, \mathcal{E}, w)$, the weight matrix W replaces the adjacency matrix A .

The other matrices:

$$D = \text{diag}(W \cdot \mathbf{1}) \quad , \quad D_{k,k} = \sum_{j \in \mathcal{V}} W_{k,j}$$

$$\Delta = D - W \quad , \quad \dim \ker(D - W) = \text{number connected components}$$

$$\tilde{\Delta} = D^{\dagger/2} \Delta D^{\dagger/2}$$

$$L = D^{\dagger} \Delta$$

where $D^{\dagger/2}$ and D^{\dagger} denote the diagonal matrices:

$$(D^{\dagger/2})_{k,k} = \begin{cases} \frac{1}{\sqrt{D_{k,k}}} & \text{if } D_{k,k} > 0 \\ 0 & \text{if } D_{k,k} = 0 \end{cases} \quad , \quad (D^{\dagger})_{k,k} = \begin{cases} \frac{1}{D_{k,k}} & \text{if } D_{k,k} > 0 \\ 0 & \text{if } D_{k,k} = 0 \end{cases} .$$

Laplacian and Normalized Laplacian for Weighted Graphs

Dirichlet Energy

For symmetric (i.e., undirected) weighted graphs, the Dirichlet energy is defined as (note edges contribute two terms in the sum)

$$E = \frac{1}{2} \sum_{i,j \in \mathcal{V}} w_{i,j} |x_i - x_j|^2$$

Expanding the square and grouping the terms together, the expression simplifies to

$$\sum_{i \in \mathcal{V}} |x_i|^2 \sum_j w_{ij} - \sum_{i,j \in \mathcal{V}} w_{i,j} x_i x_j = \langle Dx, x \rangle - \langle Wx, x \rangle = x^T (D - W)x.$$

Hence:

$$E = \frac{1}{2} \sum_{i,j \in \mathcal{V}} w_{i,j} |x_i - x_j|^2 = x^T \Delta x$$

where $\Delta = D - W$ is the weighted graph Laplacian.

Spectral Analysis

Eigenvalues and Eigenvectors

Recall the **eigenvalues** of a matrix T are the zeros of the characteristic polynomial:

$$p_T(z) = \det(zI - T) = 0.$$

There are exactly n eigenvalues (including multiplicities) for a $n \times n$ matrix T . The set of eigenvalues is called its *spectrum*.

Spectral Analysis

Eigenvalues and Eigenvectors

Recall the **eigenvalues** of a matrix T are the zeros of the characteristic polynomial:

$$p_T(z) = \det(zI - T) = 0.$$

There are exactly n eigenvalues (including multiplicities) for a $n \times n$ matrix T . The set of eigenvalues is called its *spectrum*.

If λ is an eigenvalue of T , then an associated eigenvector is the non-zero n -vector x such that $Tx = \lambda x$.

Spectral Analysis

Eigenvalues and Eigenvectors

Recall the **eigenvalues** of a matrix T are the zeros of the characteristic polynomial:

$$p_T(z) = \det(zI - T) = 0.$$

There are exactly n eigenvalues (including multiplicities) for a $n \times n$ matrix T . The set of eigenvalues is called its *spectrum*.

If λ is an eigenvalue of T , then an associated eigenvector is the non-zero n -vector x such that $Tx = \lambda x$.

Recall: If $T = T^T$ then T is called a *symmetric matrix*. Furthermore:

- Every eigenvalue of T is real.
- There is a set of n eigenvectors $\{e_1, \dots, e_n\}$ normalized so that the matrix $U = [e_1 | \dots | e_n]$ is orthogonal ($UU^T = U^T U = I_n$) and $T = U\Lambda U^T$, where Λ is the diagonal matrix of eigenvalues.

Spectral Analysis

Eigenvalues and Eigenvectors

Recall the **eigenvalues** of a matrix T are the zeros of the characteristic polynomial:

$$p_T(z) = \det(zI - T) = 0.$$

There are exactly n eigenvalues (including multiplicities) for a $n \times n$ matrix T . The set of eigenvalues is called its *spectrum*.

If λ is an eigenvalue of T , then an associated eigenvector is the non-zero n -vector x such that $Tx = \lambda x$.

Recall: If $T = T^T$ then T is called a *symmetric matrix*. Furthermore:

- Every eigenvalue of T is real.
- There is a set of n eigenvectors $\{e_1, \dots, e_n\}$ normalized so that the matrix $U = [e_1 | \dots | e_n]$ is orthogonal ($UU^T = U^T U = I_n$) and $T = U\Lambda U^T$, where Λ is the diagonal matrix of eigenvalues.

Remark. Since $\det(A_1 A_2) = \det(A_1)\det(A_2)$ and $L = D^{-1/2} \tilde{\Delta} D^{1/2}$ it follows that $\text{eigs}(\tilde{\Delta}) = \text{eigs}(L) = \text{eigs}(L^T)$.

Spectral Analysis

UCINET IV Database: Bernard & Killworth Office Dataset

For the Bernard & Killworth Office dataset (bkoff.dat) dataset we obtained the following results:

The graph is connected. $\text{rank}(\Delta) = \text{rank}(\tilde{\Delta}) = \text{rank}(L) = 39$.

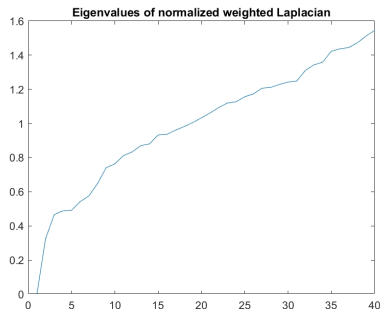
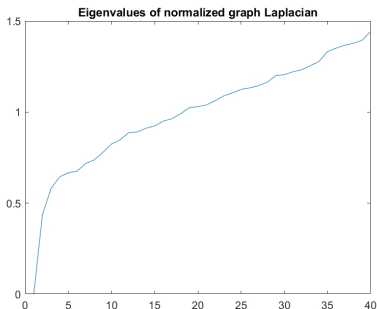


Figure: Adjacency Matrix based Graph Laplacian

Figure: Weight Matrix based Graph Laplacian

Spectral Analysis

Symmetric Matrices

Recall the following result:

Theorem

Assume T is a real symmetric $n \times n$ matrix. Then:

- 1 All eigenvalues of T are real numbers.
- 2 There are n eigenvectors that can be normalized to form an orthonormal basis for \mathbb{R}^n .
- 3 The largest eigenvalue λ_{\max} and the smallest eigenvalue λ_{\min} satisfy

$$\lambda_{\max} = \max_{x \neq 0} \frac{\langle Tx, x \rangle}{\langle x, x \rangle}, \quad \lambda_{\min} = \min_{x \neq 0} \frac{\langle Tx, x \rangle}{\langle x, x \rangle}$$

Spectral Analysis

Rayleigh Quotient

For two symmetric matrices T, S we say $T \leq S$ if $\langle Tx, x \rangle \leq \langle Sx, x \rangle$ for all $x \in \mathbb{R}^n$.

Spectral Analysis

Rayleigh Quotient

For two symmetric matrices T, S we say $T \leq S$ if $\langle Tx, x \rangle \leq \langle Sx, x \rangle$ for all $x \in \mathbb{R}^n$.

Consequence 3 can be rewritten:

$$\lambda_{\min} I \leq T \leq \lambda_{\max} I$$

Spectral Analysis

Rayleigh Quotient

For two symmetric matrices T, S we say $T \leq S$ if $\langle Tx, x \rangle \leq \langle Sx, x \rangle$ for all $x \in \mathbb{R}^n$.







Consequence 3 can be rewritten:

$$\lambda_{\min} I \leq T \leq \lambda_{\max} I$$

In particular we say T is positive semidefinite $T \geq 0$ if $\langle Tx, x \rangle \geq 0$ for every x .

It follows that T is positive semidefinite if and only if every eigenvalue of T is positive semidefinite (i.e. non-negative).

References

-  B. Bollobás, **Graph Theory. An Introductory Course**, Springer-Verlag 1979. **99**(25), 15879–15882 (2002).
-  F. Chung, L. Lu, The average distances in random graphs with given expected degrees, Proc. Nat.Acad.Sci.
-  R. Diestel, **Graph Theory**, 3rd Edition, Springer-Verlag 2005.
-  P. Erdős, A. Rényi, On The Evolution of Random Graphs
-  G. Grimmett, **Probability on Graphs. Random Processes on Graphs and Lattices**, Cambridge Press 2010.
-  J. Leskovec, J. Kleinberg, C. Faloutsos, Graph Evolution: Densification and Shrinking Diameters, ACM Trans. on Knowl.Disc.Data, **1**(1) 2007.