WIKIPEDIA
The Free Encyclopedia

Laplacian matrix

In the mathematical field of graph theory, the **Laplacian matrix**, also called the **graph Laplacian**, **admittance matrix**, **Kirchhoff matrix** or **discrete Laplacian**, is a matrix representation of a graph. Named after Pierre-Simon Laplace, the graph Laplacian matrix can be viewed as a matrix form of the negative discrete Laplace operator on a graph approximating the negative continuous Laplacian obtained by the finite difference method.

The Laplacian matrix relates to many useful properties of a graph. Together with Kirchhoff's theorem, it can be used to calculate the number of spanning trees for a given graph. The sparsest cut of a graph can be approximated through the Fiedler vector — the eigenvector corresponding to the second smallest eigenvalue of the graph Laplacian — as established by Cheeger's inequality. The spectral decomposition of the Laplacian matrix allows constructing low dimensional embeddings that appear in many machine learning applications and determines a spectral layout in graph drawing. Graph-based signal processing is based on the graph Fourier transform that extends the traditional discrete Fourier transform by substituting the standard basis of complex sinusoids for eigenvectors of the Laplacian matrix of a graph corresponding to the signal.

The Laplacian matrix is the easiest to define for a simple graph, but more common in applications for an edge-weighted graph, i.e., with weights on its edges — the entries of the graph adjacency matrix. Spectral graph theory relates properties of a graph to a spectrum, i.e., eigenvalues, and eigenvectors of matrices associated with the graph, such as its adjacency matrix or Laplacian matrix. Imbalanced weights may undesirably affect the matrix spectrum, leading to the need of normalization — a column/row scaling of the matrix entries — resulting in normalized adjacency and Laplacian matrices.

Definitions for *simple graphs*

Laplacian matrix

Given a simple graph G with n vertices $v_1, \ldots, v_n$, its Laplacian matrix $L_{n\times n}$ is defined element-wise as[1]

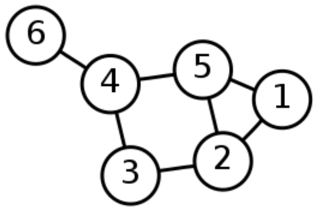
$$L_{i,j} := \begin{cases} \deg(v_i) & \text{if } i = j \\ -1 & \text{if } i \neq j \text{ and } v_i \text{ is adjacent to } v_j \\ 0 & \text{otherwise,} \end{cases}$$

or equivalently by the matrix

$$L = D - A,$$

where D is the degree matrix and A is the adjacency matrix of the graph. Since G is a simple graph, A only contains 1s or 0s and its diagonal elements are all 0s.

Here is a simple example of a labelled, undirected graph and its Laplacian matrix.

Labelled graph	Degree matrix	Adjacency matrix	Laplacian matrix
	$\begin{pmatrix} 2&0&0&0&0&0\\ 0&3&0&0&0&0\\ 0&0&2&0&0&0\\ 0&0&0&3&0&0\\ 0&0&0&0&3&0\\ 0&0&0&0&0&1 \end{pmatrix}$	$\begin{pmatrix} 0&1&0&0&1&0\\ 1&0&1&0&1&0\\ 0&1&0&1&0&0\\ 0&0&1&0&1&1\\ 1&1&0&1&0&0\\ 0&0&0&1&0&0 \end{pmatrix}$	$\begin{pmatrix} 2&-1&0&0&-1&0\\ -1&3&-1&0&-1&0\\ 0&-1&2&-1&0&0\\ 0&0&-1&3&-1&-1\\ -1&-1&0&-1&3&0\\ 0&0&0&-1&0&1 \end{pmatrix}$

We observe for the undirected graph that both the adjacency matrix and the Laplacian matrix are symmetric, and that row- and column-sums of the Laplacian matrix are all zeros.

For directed graphs, either the indegree or outdegree might be used, depending on the application, as in the following example:

Adjacency matrix	Out-Degree matrix	Out-Degree Laplacian	In-Degree matrix	In-Degree Laplacian
$\begin{pmatrix}0&1&1\\0&0&1\\1&0&0\end{pmatrix}$	$\begin{pmatrix}2&0&0\\0&1&0\\0&0&1\end{pmatrix}$	$\begin{pmatrix}2&-1&-1\\0&1&-1\\-1&0&1\end{pmatrix}$	$\begin{pmatrix}1&0&0\\0&1&0\\0&0&2\end{pmatrix}$	$\begin{pmatrix}1&-1&-1\\0&1&-1\\-1&0&2\end{pmatrix}$

In the directed graph, both the adjacency matrix and the Laplacian matrix are asymmetric. In its Laplacian matrix, column-sums or row-sums are zero, depending on whether the indegree or outdegree has been used.

Symmetric Laplacian via the incidence matrix

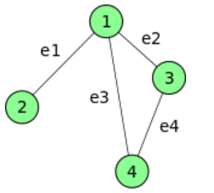
The $|v| \times |e|$ incidence matrix B with element B_{ve} for the vertex v and the edge e (connecting vertices v_i and v_j, with $i > j$) is defined by

$$B_{ve} = \begin{cases} 1, & \text{if } v = v_i \\ -1, & \text{if } v = v_j \\ 0, & \text{otherwise.} \end{cases}$$

Even though the edges in this definition are technically directed, their directions can be arbitrary, still resulting in the same symmetric Laplacian $|v| \times |v|$ matrix L defined as

$$L = BB^{\mathsf{T}}$$

where B^{T} is the matrix transpose of B.

Undirected graph	Incidence matrix	Laplacian matrix
	$\begin{pmatrix}1&1&1&0\\-1&0&0&0\\0&-1&0&1\\0&0&-1&-1\end{pmatrix}$	$\begin{pmatrix}3&-1&-1&-1\\-1&1&0&0\\-1&0&2&-1\\-1&0&-1&2\end{pmatrix}$

An alternative product $B^{\mathsf{T}}B$ defines the so-called $|e| \times |e|$ *edge-based Laplacian,* as opposed to the original commonly used *vertex-based Laplacian* matrix L.

Symmetric Laplacian for a directed graph

The Laplacian matrix of a directed graph is by definition generally non-symmetric, while, e.g., traditional spectral clustering is primarily developed for undirected graphs with symmetric adjacency and Laplacian matrices. A trivial approach to apply techniques requiring the symmetry is to turn the original directed graph into an undirected graph and build the Laplacian matrix for the latter.

In the matrix notation, the adjacency matrix of the undirected graph could, e.g., be defined as a Boolean sum of the adjacency matrix A of the original directed graph and its matrix transpose A^T, where the zero and one entries of A are treated as logical, rather than numerical, values, as in the following example:

Adjacency matrix	Symmetrized adjacency	Symmetric Laplacian matrix
$\begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}$	$\begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}$	$\begin{pmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{pmatrix}$

Laplacian matrix normalization

A vertex with a large degree, also called a *heavy node,* results in a large diagonal entry in the Laplacian matrix dominating the matrix properties. Normalization is aimed to make the influence of such vertices more equal to that of other vertices, by dividing the entries of the Laplacian matrix by the vertex degrees. To avoid division by zero, isolated vertices with zero degrees are excluded from the process of the normalization.

Symmetrically normalized Laplacian

The symmetrically normalized Laplacian matrix is defined as:[1]

$$L^{\text{sym}} := (D^+)^{1/2} L (D^+)^{1/2} = I - (D^+)^{1/2} A (D^+)^{1/2},$$

where D^+ is the Moore–Penrose inverse.

The elements of L^{sym} are thus given by

$$L^{\text{sym}}_{i,j} := \begin{cases} 1 & \text{if } i = j \text{ and } \deg(v_i) \neq 0 \\ -\frac{1}{\sqrt{\deg(v_i)\deg(v_j)}} & \text{if } i \neq j \text{ and } v_i \text{ is adjacent to } v_j \\ 0 & \text{otherwise.} \end{cases}$$

The symmetrically normalized Laplacian matrix is symmetric if and only if the adjacency matrix is symmetric.

Adjacency matrix	Degree matrix	Normalized Laplacian
$\begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$	$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & -\sqrt{1/2} & 0 \\ -\sqrt{1/2} & 1 & -\sqrt{1/2} \\ 0 & -\sqrt{1/2} & 1 \end{pmatrix}$

For a non-symmetric adjacency matrix of a directed graph, either of indegree and outdegree can be used for normalization:

Adjacency matrix	Out-Degree matrix	Out-Degree normalized Laplacian	In-Degree matrix	In-Degree normalized Laplacian
$\begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}$	$\begin{pmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & -\sqrt{1/2} & -\sqrt{1/2} \\ 0 & 1 & -1 \\ -\sqrt{1/2} & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix}$	$\begin{pmatrix} 1 & -1 & -\sqrt{1/2} \\ 0 & 1 & -\sqrt{1/2} \\ -\sqrt{1/2} & 0 & 1 \end{pmatrix}$

Left (random-walk) and right normalized Laplacians

The left (random-walk) normalized Laplacian matrix is defined as:

$$L^{\text{rw}} := D^+ L = I - D^+ A,$$

where D^+ is the Moore–Penrose inverse. The elements of L^{rw} are given by

$$L^{\text{rw}}_{i,j} := \begin{cases} 1 & \text{if } i = j \text{ and } \deg(v_i) \neq 0 \\ -\frac{1}{\deg(v_i)} & \text{if } i \neq j \text{ and } v_i \text{ is adjacent to } v_j \\ 0 & \text{otherwise.} \end{cases}$$

Similarly, the right normalized Laplacian matrix is defined as

$$LD^+ = I - AD^+.$$

The left or right normalized Laplacian matrix is not symmetric if the adjacency matrix is symmetric, except for the trivial case of all isolated vertices. For example,

Adjacency matrix	Degree matrix	Left normalized Laplacian	Right normalized Laplacian
$\begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$	$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & -1 & 0 \\ -1/2 & 1 & -1/2 \\ 0 & -1 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & -1/2 & 0 \\ -1 & 1 & -1 \\ 0 & -1/2 & 1 \end{pmatrix}$

The example also demonstrates that if G has no isolated vertices, then D^+A right stochastic and hence is the matrix of a random walk, so that the left normalized Laplacian $L^{\text{rw}} := D^+L = I - D^+A$ has each row summing to zero. Thus we sometimes alternatively call L^{rw} the random-walk normalized Laplacian. In the less uncommonly used right normalized Laplacian $LD^+ = I - AD^+$ each column sums to zero since AD^+ is left stochastic.

For a non-symmetric adjacency matrix of a directed graph, one also needs to choose indegree or outdegree for normalization:

Adjacency matrix	Out-Degree matrix	Out-Degree left normalized Laplacian	In-Degree matrix	In-Degree right normalized Laplacian
$\begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}$	$\begin{pmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & -1/2 & -1/2 \\ 0 & 1 & -1 \\ -1 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix}$	$\begin{pmatrix} 1 & -1 & -1/2 \\ 0 & 1 & -1/2 \\ -1 & 0 & 1 \end{pmatrix}$

The left out-degree normalized Laplacian with row-sums all 0 relates to right stochastic $D_{\text{out}}^+ A$, while the right in-degree normalized Laplacian with column-sums all 0 contains left stochastic AD_{in}^+.

Definitions for graphs with weighted edges

Common in applications graphs with weighted edges are conveniently defined by their adjacency matrices where values of the entries are numeric and no longer limited to zeros and ones. In spectral clustering and graph-based signal processing, where graph vertices represent data points, the edge weights can be computed, e.g., as inversely proportional to the distances between pairs of data points, leading to all weights being non-negative with larger values informally corresponding to more similar pairs of data points. Using correlation and anti-correlation between the data points naturally leads to both positive and negative weights. Most definitions for simple graphs are trivially extended to the standard case of non-negative weights, while negative weights require more attention, especially in normalization.

Laplacian matrix

The Laplacian matrix is defined by

$$L = D - A,$$

where *D* is the degree matrix and *A* is the adjacency matrix of the graph.

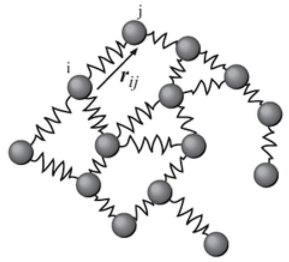
For directed graphs, either the indegree or outdegree might be used, depending on the application, as in the following example:

Adjacency matrix	In-Degree matrix	In-Degree Laplacian	Out-Degree matrix	Out-Degree Laplacian
$\begin{pmatrix} 0 & 1 & 2 \\ 3 & 0 & 5 \\ 6 & 7 & 0 \end{pmatrix}$	$\begin{pmatrix} 9 & 0 & 0 \\ 0 & 8 & 0 \\ 0 & 0 & 7 \end{pmatrix}$	$\begin{pmatrix} 9 & -1 & -2 \\ -3 & 8 & -5 \\ -6 & -7 & 7 \end{pmatrix}$	$\begin{pmatrix} 3 & 0 & 0 \\ 0 & 8 & 0 \\ 0 & 0 & 13 \end{pmatrix}$	$\begin{pmatrix} 3 & -1 & -2 \\ -3 & 8 & -5 \\ -6 & -7 & 13 \end{pmatrix}$

Graph self-loops, manifesting themselves by non-zero entries on the main diagonal of the adjacency matrix, are allowed but do not affect the graph Laplacian values.

Symmetric Laplacian via the incidence matrix

For graphs with weighted edges one can define a weighted incidence matrix B and use it to construct the corresponding symmetric Laplacian as $L = BB^{\mathsf{T}}$. An alternative cleaner approach, described here, is to separate the weights from the connectivity: continue using the incidence matrix as for regular graphs and introduce a matrix just holding the values of the weights. A spring system is an example of this model used in mechanics to describe a system of springs of given stiffnesses and unit length, where the values of the stiffnesses play the role of the weights of the graph edges.



A 2-dimensional spring system.

We thus reuse the definition of the weightless $|v| \times |e|$ incidence matrix B with element B_{ve} for the vertex v and the edge e (connecting vertexes v_i and v_j, with $i > j$) defined by

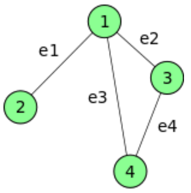
$$B_{ve} = \begin{cases} 1, & \text{if } v = v_i \\ -1, & \text{if } v = v_j \\ 0, & \text{otherwise.} \end{cases}$$

We now also define a diagonal $|e| \times |e|$ matrix W containing the edge weights. Even though the edges in the definition of B are technically directed, their directions can be arbitrary, still resulting in the same symmetric Laplacian $|v| \times |v|$ matrix L defined as

$$L = BWB^{\mathsf{T}}$$

where B^{T} is the matrix transpose of B.

The construction is illustrated in the following example, where every edge e_i is assigned the weight value i, with $i = 1, 2, 3, 4$.

Undirected graph	Incidence matrix	Edge weights	Laplacian matrix
	$\begin{pmatrix} 1 & 1 & 1 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 \\ 0 & 0 & -1 & -1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 4 \end{pmatrix}$	$\begin{pmatrix} 6 & -1 & -2 & -3 \\ -1 & 1 & 0 & 0 \\ -2 & 0 & 6 & -4 \\ -3 & 0 & -4 & 7 \end{pmatrix}$

Symmetric Laplacian for a directed graph

Just like for simple graphs, the Laplacian matrix of a directed weighted graph is by definition generally non-symmetric. The symmetry can be enforced by turning the original directed graph into an undirected graph first before constructing the Laplacian. The adjacency matrix of the undirected graph could, e.g., be defined as a sum of the adjacency matrix A of the original directed graph and its matrix transpose A^T as in the following example:

Adjacency matrix	Symmetrized adjacency matrix	Symmetric Laplacian matrix
$\begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}$	$\begin{pmatrix} 0 & 1 & 2 \\ 1 & 0 & 1 \\ 2 & 1 & 0 \end{pmatrix}$	$\begin{pmatrix} 3 & -1 & -2 \\ -1 & 2 & -1 \\ -2 & -1 & 3 \end{pmatrix}$

where the zero and one entries of A are treated as numerical, rather than logical as for simple graphs, values, explaining the difference in the results - for simple graphs, the symmetrized graph still needs to be simple with its symmetrized adjacency matrix having only logical, not numerical values, e.g., the logical sum is 1 ∨ 1 = 1, while the numeric sum is 1 + 1 = 2.

Alternatively, the symmetric Laplacian matrix can be calculated from the two Laplacians using the indegree and outdegree, as in the following example:

Adjacency matrix	Out-Degree matrix	Out-Degree Laplacian	In-Degree matrix	In-Degree Laplacian
$\begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}$	$\begin{pmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 2 & -1 & -1 \\ 0 & 1 & -1 \\ -1 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix}$	$\begin{pmatrix} 1 & -1 & -1 \\ 0 & 1 & -1 \\ -1 & 0 & 2 \end{pmatrix}$

The sum of the out-degree Laplacian transposed and the in-degree Laplacian equals to the symmetric Laplacian matrix.

Laplacian matrix normalization

The goal of normalization is, like for simple graphs, to make the diagonal entries of the Laplacian matrix to be all unit, also scaling off-diagonal entries correspondingly. In a weighted graph, a vertex may have a large degree because of a small number of connected edges but with large weights just as well as due to a large number of connected edges with unit weights.

Graph self-loops, i.e., non-zero entries on the main diagonal of the adjacency matrix, do not affect the graph Laplacian values, but may need to be counted for calculation of the normalization factors.

Symmetrically normalized Laplacian

The **symmetrically normalized Laplacian** is defined as

$$L^{\text{sym}} := (D^+)^{1/2} L (D^+)^{1/2} = I - (D^+)^{1/2} A (D^+)^{1/2},$$

where L is the unnormalized Laplacian, A is the adjacency matrix, D is the degree matrix, and D^+ is the Moore–Penrose inverse. Since the degree matrix D is diagonal, its reciprocal square root $(D^+)^{1/2}$ is just the diagonal matrix whose diagonal entries are the reciprocals of the square roots of the diagonal entries of D. If all the edge weights are nonnegative then all the degree values are automatically also nonnegative and so every degree value has a unique positive square root. To avoid the division by zero, vertices with zero degrees are excluded from the process of the normalization, as in the following example:

Adjacency matrix	In-Degree matrix	In-Degree normalized Laplacian	Out-Degree matrix	Out-Degree normalized Laplacian
$\begin{pmatrix} 0 & 1 & 0 \\ 4 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$	$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 0 \end{pmatrix}$	$\begin{pmatrix} 1 & -1/2 & 0 \\ -2 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$	$\begin{pmatrix} 4 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$	$\begin{pmatrix} 1 & -1/2 & 0 \\ -2 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$

The symmetrically normalized Laplacian is a symmetric matrix if and only if the adjacency matrix A is symmetric and the diagonal entries of D are nonnegative, in which case we can use the term the ***symmetric normalized Laplacian***.

The symmetric normalized Laplacian matrix can be also written as

$$L^{\text{sym}} := (D^+)^{1/2} L (D^+)^{1/2} = (D^+)^{1/2} B W B^{\textsf{T}} (D^+)^{1/2} = S S^T$$

using the weightless $|v| \times |e|$ incidence matrix B and the diagonal $|e| \times |e|$ matrix W containing the edge weights and defining the new $|v| \times |e|$ weighted incidence matrix $S = (D^+)^{1/2} B W^{1/2}$ whose rows are indexed by the vertices and whose columns are indexed by the edges of G such that each column corresponding to an edge *e* = *{u, v}* has an entry $\frac{1}{\sqrt{d_u}}$ in the row corresponding to u, an entry $-\frac{1}{\sqrt{d_v}}$ in the row corresponding to v, and has 0 entries elsewhere.

Random walk normalized Laplacian

The **random walk normalized Laplacian** is defined as

$$L^{\text{rw}} := D^{+}L = I - D^{+}A$$

where *D* is the degree matrix. Since the degree matrix *D* is diagonal, its inverse D^{+} is simply defined as a diagonal matrix, having diagonal entries which are the reciprocals of the corresponding diagonal entries of *D*. For the isolated vertices (those with degree 0), a common choice is to set the corresponding element $L^{\text{rw}}_{i,i}$ to 0. The matrix elements of L^{rw} are given by

$$L^{\text{rw}}_{i,j} := \begin{cases} 1 & \text{if } i = j \text{ and } \deg(v_i) \neq 0 \\ -\frac{1}{\deg(v_i)} & \text{if } i \neq j \text{ and } v_i \text{ is adjacent to } v_j \\ 0 & \text{otherwise.} \end{cases}$$

The name of the random-walk normalized Laplacian comes from the fact that this matrix is $L^{\text{rw}} = I - P$, where $P = D^{+}A$ is simply the transition matrix of a random walker on the graph, assuming non-negative weights. For example, let e_i denote the i-th standard basis vector. Then $x = e_i P$ is a probability vector representing the distribution of a random walker's locations after taking a single step from vertex i; i.e., $x_j = \mathbb{P}\left(v_i \to v_j\right)$. More generally, if the vector x is a probability distribution of the location of a random walker on the vertices of the graph, then $x' = xP^t$ is the probability distribution of the walker after t steps.

The random walk normalized Laplacian can also be called the left normalized Laplacian $L^{\text{rw}} := D^{+}L$ since the normalization is performed by multiplying the Laplacian by the normalization matrix D^{+} on the left. It has each row summing to zero since $P = D^{+}A$ is right stochastic, assuming all the weights are non-negative.

In the less uncommonly used right normalized Laplacian $LD^{+} = I - AD^{+}$ each column sums to zero since AD^{+} is left stochastic.

For a non-symmetric adjacency matrix of a directed graph, one also needs to choose indegree or outdegree for normalization:

Adjacency matrix	Out-Degree matrix	Out-Degree left normalized Laplacian	In-Degree matrix	In-Degree right normalized Laplacian
$\begin{pmatrix}0&1&0\\0&0&2\\1&0&0\end{pmatrix}$	$\begin{pmatrix}1&0&0\\0&2&0\\0&0&1\end{pmatrix}$	$\begin{pmatrix}1&-1&0\\0&1&-1\\-1&0&1\end{pmatrix}$	$\begin{pmatrix}1&0&0\\0&1&0\\0&0&2\end{pmatrix}$	$\begin{pmatrix}1&-1&0\\0&1&-1\\-1&0&1\end{pmatrix}$

The left out-degree normalized Laplacian with row-sums all 0 relates to right stochastic $D_{\text{out}}^{+}A$, while the right in-degree normalized Laplacian with column-sums all 0 contains left stochastic AD_{in}^{+}.

Negative weights

Negative weights present several challenges for normalisation:

- The presence of negative weights may naturally result in zero row- and/or column-sums for non-isolated vertices. A vertex with a large row-sum of positive weights and equally negatively large row-sum of negative weights, together summing up to zero, could be considered a heavy node and both large values scaled, while the diagonal entry remains zero, like for a isolated vertex.
- Negative weights may also give negative row- and/or column-sums, so that the corresponding diagonal entry in the non-normalized Laplacian matrix would be negative and a positive square root needed for the symmetric normalization would not exist.
- Arguments can be made to take the absolute value of the row- and/or column-sums for the purpose of normalization, thus treating a possible value -1 as a legitimate unit entry of the main diagonal of the

normalized Laplacian matrix.

Properties

For an (undirected) graph G and its Laplacian matrix L with eigenvalues $\lambda_0 \le \lambda_1 \le \cdots \le \lambda_{n-1}$:

- *L* is symmetric.
- *L* is positive-semidefinite (that is $\lambda_i \ge 0$ for all i). This can be seen from the fact that the Laplacian is symmetric and diagonally dominant.
- *L* is an M-matrix (its off-diagonal entries are nonpositive, yet the real parts of its eigenvalues are nonnegative).
- Every row sum and column sum of *L* is zero. Indeed, in the sum, the degree of the vertex is summed with a "−1" for each neighbor.
- In consequence, $\lambda_0 = 0$, because the vector $\mathbf{v}_0 = (1, 1, \dots, 1)$ satisfies $L\mathbf{v}_0 = \mathbf{0}.$ This also implies that the Laplacian matrix is singular.
- The number of connected components in the graph is the dimension of the nullspace of the Laplacian and the algebraic multiplicity of the 0 eigenvalue.
- The smallest non-zero eigenvalue of *L* is called the spectral gap.
- The second smallest eigenvalue of *L* (could be zero) is the algebraic connectivity (or Fiedler value) of *G* and approximates the sparsest cut of a graph.
- The Laplacian is an operator on the n-dimensional vector space of functions $f : V \to \mathbb{R}$, where V is the vertex set of G, and $n = |V|$.
- When G is k-regular, the normalized Laplacian is: $\mathcal{L} = \tfrac{1}{k} L = I - \tfrac{1}{k} A$, where A is the adjacency matrix and I is an identity matrix.
- For a graph with multiple connected components, *L* is a block diagonal matrix, where each block is the respective Laplacian matrix for each component, possibly after reordering the vertices (i.e. *L* is permutation-similar to a block diagonal matrix).
- The trace of the Laplacian matrix *L* is equal to $2m$ where m is the number of edges of the considered graph.
- Now consider an eigendecomposition of L, with unit-norm eigenvectors $\mathbf{v}_i$ and corresponding eigenvalues λ_i:

$$\begin{aligned}\lambda_i &= \mathbf{v}_i^\textsf{T} L \mathbf{v}_i \\ &= \mathbf{v}_i^\textsf{T} M^\textsf{T} M \mathbf{v}_i \\ &= \left(M\mathbf{v}_i\right)^\textsf{T} \left(M\mathbf{v}_i\right).\end{aligned}$$

Because λ_i can be written as the inner product of the vector $M\mathbf{v}_i$ with itself, this shows that $\lambda_i \ge 0$ and so the eigenvalues of L are all non-negative.

- All eigenvalues of the normalized symmetric Laplacian satisfy $0 = \mu_0 \le \ldots \le \mu_{n-1} \le 2$. These eigenvalues (known as the spectrum of the normalized Laplacian) relate well to other graph invariants for general graphs.[1]
- One can check that:

$$L^\text{rw} = I - D^{-\frac{1}{2}} \left(I - L^\text{sym}\right) D^{\frac{1}{2}},$$

i.e., L^rw is similar to the normalized Laplacian L^sym. For this reason, even if L^rw is in general not symmetric, it has real eigenvalues — exactly the same as the eigenvalues of the normalized symmetric Laplacian L^sym.

Interpretation as the discrete Laplace operator approximating the continuous Laplacian

The graph Laplacian matrix can be further viewed as a matrix form of the negative discrete Laplace operator on a graph approximating the negative continuous Laplacian operator obtained by the finite difference method. (See Discrete Poisson equation)[2] In this interpretation, every graph vertex is treated as a grid point; the local connectivity of the vertex determines the finite difference approximation stencil at this grid point, the grid size is always one for every edge, and there are no constraints on any grid points, which corresponds to the case of the homogeneous Neumann boundary condition, i.e., free boundary. Such an interpretation allows one, e.g., generalizing the Laplacian matrix to the case of graphs with an infinite number of vertices and edges, leading to a Laplacian matrix of an infinite size.

Generalizations and extensions of the Laplacian matrix

Generalized Laplacian

The generalized Laplacian Q is defined as:[3]

$$\begin{cases} Q_{i,j} < 0 & \text{if } i \neq j \text{ and } v_i \text{ is adjacent to } v_j \\ Q_{i,j} = 0 & \text{if } i \neq j \text{ and } v_i \text{ is not adjacent to } v_j \\ \text{any number} & \text{otherwise.} \end{cases}$$

Notice the ordinary Laplacian is a generalized Laplacian.

Magnetic Laplacian

Entries of the adjacency matrix can be complex-valued, in which case the notion of the matrix symmetry needs to be replaced with a Hermitian matrix. Magnetic Laplacian for a directed graph with real weights w_{ij} is constructed as the Hadamard product of the real symmetric matrix of the symmetrized Laplacian and the Hermitian phase matrix with the complex entries

$$\gamma_q(i,j) = e^{i2\pi q(w_{ij} - w_{ji})}$$

which encode the edge direction into the phase in the complex plane. In the context of quantum physics, the magnetic Laplacian can be interpreted as the operator that describes the phenomenology of a free charged particle on a graph, which is subject to the action of a magnetic field and the parameter q is called electric charge.[4] In the following example $q = 1/4$:

Adjacency matrix	Symmetrized Laplacian	Phase matrix	Magnetic Laplacian
$\begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$	$\begin{pmatrix} 2 & -2 & 0 & 0 \\ -2 & 3 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & i & 1 \\ 1 & -i & 1 & -i \\ 1 & 1 & i & 1 \end{pmatrix}$	$\begin{pmatrix} 2 & -2 & 0 & 0 \\ -2 & 3 & -i & 0 \\ 0 & i & 2 & i \\ 0 & 0 & -i & 1 \end{pmatrix}$

Deformed Laplacian

The **deformed Laplacian** is commonly defined as

$$\Delta(s) = I - sA + s^2(D - I)$$

where I is the identity matrix, A is the adjacency matrix, D is the degree matrix, and s is a (complex-valued) number. [5]
The standard Laplacian is just $\Delta(1)$ and $\Delta(-1) = D + A$ is the signless Laplacian.

Signless Laplacian

The **signless Laplacian** is defined as

$$Q = D + A$$

where D is the degree matrix, and A is the adjacency matrix.[6] Like the signed Laplacian L, the signless Laplacian Q also is positive semi-definite as it can be factored as

$$Q = RR^\textsf{T}$$

where R is the incidence matrix. Q has a 0-eigenvector if and only if it has a bipartite connected component other than isolated vertices. This can be shown as

$$\mathbf{x}^\textsf{T} Q \mathbf{x} = \mathbf{x}^\textsf{T} R R^\textsf{T} \mathbf{x} \implies R^\textsf{T} \mathbf{x} = \mathbf{0}.$$

This has a solution where $\mathbf{x} \neq \mathbf{0}$ if and only if the graph has a bipartite connected component.

Directed multigraphs

An analogue of the Laplacian matrix can be defined for directed multigraphs.[7] In this case the Laplacian matrix L is defined as

$$L = D - A$$

where D is a diagonal matrix with $D_{i,i}$ equal to the outdegree of vertex i and A is a matrix with $A_{i,j}$ equal to the number of edges from i to j (including loops).

Open source software implementations

- SciPy[8]
- NetworkX[9]

Application software

- scikit-learn Spectral Clustering[10]
- PyGSP: Graph Signal Processing in Python[11]
- megaman: Manifold Learning for Millions of Points[12]
- smoothG[13]
- Laplacian Change Point Detection for Dynamic Graphs (KDD 2020)[14]
- LaplacianOpt (A Julia Package for Maximizing Laplacian's Second Eigenvalue of Weighted Graphs) [15]
- LigMG (Large Irregular Graph MultiGrid)[16]
- Laplacians.jl[17]

See also

- Stiffness matrix
- Resistance distance
- Transition rate matrix
- Calculus on finite weighted graphs
- Graph Fourier transform

References

1. Chung, Fan (1997) [1992]. *Spectral Graph Theory* (http://www.math.ucsd.edu/~fan/research/revised.html). American Mathematical Society. ISBN 978-0821803158.

2. Smola, Alexander J.; Kondor, Risi (2003), "Kernels and regularization on graphs", *Learning Theory and Kernel Machines: 16th Annual Conference on Learning Theory and 7th Kernel Workshop, COLT/Kernel 2003, Washington, DC, USA, August 24–27, 2003, Proceedings*, Lecture Notes in Computer Science, vol. 2777, Springer, pp. 144–158, CiteSeerX 10.1.1.3.7020 (https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.3.7020), doi:10.1007/978-3-540-45167-9_12 (https://doi.org/10.1007%2F978-3-540-45167-9_12), ISBN 978-3-540-40720-1.
3. Godsil, C.; Royle, G. (2001). *Algebraic Graph Theory, Graduate Texts in Mathematics*. Springer-Verlag.
4. Satoshi Furutani; Toshiki Shibahara; Mitsuaki Akiyama; Kunio Hato; Masaki Aida (2020). *Graph Signal Processing for Directed Graphs based on the Hermitian Laplacian* (https://ecmlpkdd2019.org/downloads/paper/499.pdf) (PDF). ECML PKDD 2019: Machine Learning and Knowledge Discovery in Databases. pp. 447–463. doi:10.1007/978-3-030-46150-8_27 (https://doi.org/10.1007%2F978-3-030-46150-8_27).
5. Morbidi, F. (2013). "The Deformed Consensus Protocol" (http://hal.archives-ouvertes.fr/docs/00/96/14/91/PDF/Morbidi_AUTO13_ExtVer.pdf) (PDF). *Automatica*. **49** (10): 3049–3055. doi:10.1016/j.automatica.2013.07.006 (https://doi.org/10.1016%2Fj.automatica.2013.07.006). S2CID 205767404 (https://api.semanticscholar.org/CorpusID:205767404).
6. Cvetković, Dragoš; Simić, Slobodan K. (2010). "Towards a Spectral Theory of Graphs Based on the Signless Laplacian, III". *Applicable Analysis and Discrete Mathematics*. **4** (1): 156–166. doi:10.2298/AADM1000001C (https://doi.org/10.2298%2FAADM1000001C). ISSN 1452-8630 (https://www.worldcat.org/issn/1452-8630). JSTOR 43671298 (https://www.jstor.org/stable/43671298).
7. Chaiken, S.; Kleitman, D. (1978). "Matrix Tree Theorems" (https://doi.org/10.1016%2F0097-3165%2878%2990067-5). *Journal of Combinatorial Theory, Series A*. **24** (3): 377–381. doi:10.1016/0097-3165(78)90067-5 (https://doi.org/10.1016%2F0097-3165%2878%2990067-5). ISSN 0097-3165 (https://www.worldcat.org/issn/0097-3165).
8. "SciPy" (https://github.com/scipy/scipy/blob/main/scipy/sparse/csgraph/_laplacian.py). *GitHub*. 24 March 2022.
9. "NetworkX" (https://github.com/networkx/networkx/blob/main/networkx/linalg/laplacianmatrix.py). *GitHub*. 24 March 2022.
10. "2.3. Clustering" (https://scikit-learn.org/stable/modules/clustering.html#spectral-clustering).
11. "PyGSP: Graph Signal Processing in Python" (https://github.com/epfl-lts2/pygsp). *GitHub*. 23 March 2022.
12. "Megaman: Manifold Learning for Millions of Points" (https://github.com/mmp2/megaman). *GitHub*. 14 March 2022.
13. "SmoothG" (https://github.com/LLNL/smoothG). *GitHub*. 17 September 2020.
14. "Announcing Our Paper at KDD 2020" (https://complexdatalabmcgill.github.io/papers/post-andy-kdd2020paper/).
15. "Harshangrjn/LaplacianOpt.jl" (https://github.com/harshangrjn/LaplacianOpt.jl). *GitHub*. 2 February 2022.
16. "LigMG (Large Irregular Graph MultiGrid)-- A distributed memory graph Laplacian solver for large irregular graphs" (https://github.com/ligmg/ligmg). *GitHub*. 5 January 2022.
17. "Laplacians.jl" (https://github.com/danspielman/Laplacians.jl). *GitHub*. 11 March 2022.