

On the Laplacian

Ali Fele-Paranj

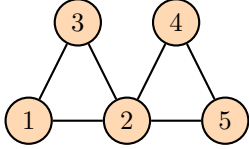
September 18, 2024

Abstract

I have gathered the sporadic contents on the notion of Laplacian in different settings (like defined for graphs, infinite graphs, real space (\mathbb{R}) as well as on the manifolds).

1 Laplacian Matrix of a Graph

For a graph, the Laplacian matrix contains the information on the in/out flow of stuff into the nodes. Then



the Laplacian matrix is given by

$$D = \begin{pmatrix} 2 & 0 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 2 \end{pmatrix},$$

and the adjacency matrix is given by

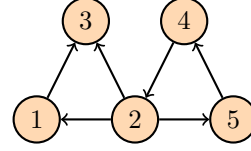
$$A = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \end{pmatrix},$$

and the Laplacian matrix is given by

$$L = D - A = \begin{pmatrix} 2 & -1 & -1 & 0 & 0 \\ -1 & 4 & -1 & -1 & -1 \\ -1 & -1 & 2 & 0 & 0 \\ 0 & -1 & 0 & 4 & -1 \\ 0 & -1 & 0 & -1 & 5 \end{pmatrix}.$$

It is straight forward to generalize the notion of Laplacian matrix to the weighed graphs, where the degree matrix D , the diagonal entries will be the sum of all weights of the edges connected to that node, and for the adjacency matrix, instead of zeros and ones, we will have the weights of the connections..

There is also another way of finding the Laplacian matrix by using the notion of incidence matrix. To do so, we first need to make our graph to be directed. Any combination of the direction on the edges will do the



job and will yield in a correct answer. For instance, consider the following directed graph

Its incidence matrix will be

$$M = \begin{pmatrix} -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & -1 & 0 & -1 \\ 1 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 \end{pmatrix}$$

The Laplacian matrix can be written as

$$\mathcal{L} = MM^T.$$

Note that in the case of the weighed graphs, we will have

$$\mathcal{L} = MWM^T \quad (1)$$

where W is a diagonal matrix containing the weights. These computations can be done easily on the NetworkX.

```
g = nx.Graph()
for i in range(1, 6):
    g.add_node(i)

    g.add_edge(1, 2, w=1)
    g.add_edge(1, 3, w=2)
    g.add_edge(2, 3, w=3)
    g.add_edge(2, 5, w=5)
    g.add_edge(4, 5, w=5)

W = np.diag([data['w'] for u, v, data
               in g.edges(data=True)])

% adj matrix
AD =
    nx.adjacency_matrix(g, weight="w")

% Laplacian matrix
L = nx.laplacian_matrix(g, weight="w")

% Incidence matrix
inc_matrix =
    nx.incidence_matrix(g, oriented=True)
```

The incidence matrix is also very useful in calculating the pressure difference between nodes of a particular edge. Let $\Delta = M^T$. Then given the vector P that contains the pressures on the vertices, then the pressure difference on the edges will be given by ΔP , where Δ is the transpose of the incidence matrix. This comes in handy when we want to calculate the flow of the edges which will be given by

$$\mathbf{Q} = \mathbf{C}L^{-1}\Delta\mathbf{P}, \quad (2)$$

where C is a diagonal matrix of the conductance of the edges, L is the diagonal matrix of the “length” of each edge, Δ is the transpose of the incidence matrix, and P is the pressure on the nodes. Q is the flow of the edges. In this particular example we are assuming that the relation between flow and the pressure difference is $Q_e = C_e(p_i - p_j)/L$. But we can have many other choices.

Knowing the sources and sinks on the nodes, the pressure can be determined by the Kirchhoff law

$$\mathcal{L}\mathbf{P} = \mathbf{q},$$

where the vector q is the sources and the sinks values for each node. This is the same as solving the **Poisson equation**. This can also be written in terms of the flow, i.e.

$$\Delta^T \mathbf{Q} = \mathbf{q}.$$

By (2) we can write

$$(\Delta^T \mathbf{C}L^{-1}\Delta)\mathbf{P} = \mathbf{q}.$$

Since $\Delta = M^T$, the expression inside the parentheses is clearly Equation (1).

Similar to the Poisson equation on the graph which is equivalent Kirchhoff’s law, we can solve other types of heat and wave equations on the graph as well. The Laplacian matrix play a key role.

$$\frac{\partial p}{\partial t} = -\mathcal{L}p + q,$$

for the heat equation, and

$$\frac{\partial^2 p}{\partial t^2} = -\mathcal{L}p + q,$$

for the wave equation.

2 Laplacian Operator of Infinite Graphs

In analogy with the Laplacian matrix for the graphs, we can develop a similar notion for the infinite graphs, in particular infinite Cayley graphs. In short, given a group G and a generating set S (where S is a subset of G not containing the identity and closed under taking inverses if the graph is to be undirected), the Cayley graph $\Gamma(G, S)$ is defined as follows

- **Edges:** There is an edge between g and gs for every $g \in G$ and $s \in S$.

In the case of the infinite graphs, we can not define an infinite dimensional Laplacian matrix, but instead we can define an operator. For an infinite Cayley graph $\Gamma(G, S)$, the **Laplacian** that acts on a function $f : G \rightarrow \mathbb{C}$ (note that the set of vertices is the group itself) can be defined as

$$(\Delta f)(g) = \sum_{s \in S} (f(g) - f(gs)) = |S|f(g) - \sum_{s \in S} f(gs).$$

There is a close parallel between the definition above and the definition of the Laplacian matrix of a graph $\mathcal{L} = D - A$. For instance, in the case of a Cayley graph of \mathbb{Z}^2 we have

$$(\Delta f)(i, j) = 4f(i, j) - f(i+1, j) - f(i-1, j) - f(i, j+1) - f(i, j-1).$$

This coincides with the notion of the discrete finite difference Laplacian operator.