SJÄLVSTÄNDIGA ARBETEN I MATEMATIK

MATEMATISKA INSTITUTIONEN, STOCKHOLMS UNIVERSITET

Laplacian Matrices as a Tool for Spectral Clustering

av

Elin Tinnert

2025 - No K7

Laplacian Matrices as a Tool for Spectral Clustering

Elin Tinnert

Självständigt arbete i matematik, 15 högskolepoäng, grundnivå

Handledare: Gregory Arone

2025

Abstract

A graph can illustrate patterns in data. Clusters in a graph show which of its vertices are closely related. The Laplacian matrix is a powerful matrix that one can associate with a graph. Spectral analysis studies the relations between properties of a graph and eigenvalues of its Laplacian matrix. In this thesis, basic facts about the Laplacian matrix will be introduced and an investigation of its use in the study of clusters will be performed. We derive an approximation of a clustering method called RatioCut for finding two clusters in a graph, which uses the eigenvector of the second smallest eigenvalue as an indicator. We give examples that illustrate the method.

Sammanfattning

En graf kan åskådliggöra mönster i data, där kluster i grafen visar vilka noder som är nära relaterade. Laplacematrisen är en kraftfull matris som konstrueras utifrån grafens struktur. I spektralanalys studeras relationen mellan en grafs egenskaper och Laplacematrisens egenvärden. I uppsatsen redogör vi för grundläggande teori om Laplacematrisen och vi undersöker dess tillämpbarhet för att studera kluster. Vi utför en approximation av mtoden RatioCut för att identifiera två kluster i en graf, vilken använder egenvektorn tillhörande det näst minsta egenvärdet som indikator. Metoden illustreras med exempel.

Acknowledgements

I would like to express my gratitude to my supervisor, Professor Gregory Arone, who has been very supportive and provided valuable guidance and feedback. I would also like to thank my proofreader and friend, Maria Krajewska, for her kind support on language and citations.

Use of AI

When writing code in the programming language R for computations, I have used the AI tool *ChatGPT* for support. I had misunderstood the indices of the eigenvector matrix which led to eigenvectors not getting a matching index to their associating eigenvalue. So I asked how to index them correctly, which helped me get the code working.

Contents

1 Introduction

Thinking about a graph, different pictures may come to mind: a curve representing the trajectory of a discus throw, a family tree, or a social network. In either case, a graph is a visual representation of underlying information, which, when talking about the social network, is built of a set of vertices and edges. Each vertex represents an individual, and an edge between two individuals represents their social connection. In his article, "On the Magical Properties of the Laplacian Matrix", [Ita24], Uri Itai provides an introduction to the Laplacian matrix that generates curiosity and urges one to learn more about the topic. Among other interesting statements, Itai writes that a graph is a visualisation that reveals patterns in the underlying data, which is of great relevance when it comes to the analysis of information. One of the patterns that will be the focus of this thesis is clusters - and clustering - meaning dividing the vertex set of a graph into subsets based on the edges between them.

Spectral clustering, which is clustering based on the eigenvalues and eigenvectors of a graph, aims to cluster a graph of vertices and connecting edges into subsets of vertices where the intuition is that vertices within a subset are more similar to each other - a cluster, [vL07, p. 9]. In this thesis, we begin the discussion about clustering by showing that the connected components of a graph are determined by the eigenspace of zero. Further, the topic of identifying clusters will be limited to what is called the "RatioCut" method for spectral clustering, which is described in detail by Ulrike von Luxburg in her paper "A Tutorial on Spectral Clustering", [vL07]. The RatioCut method is a method with the goal of minimizing the number of edges between different clusters in a graph. This way, a problem of how to identify clusters in a graph is reformulated from focusing on similarities within a cluster, to focusing on minimizing the number of edges between clusters instead, [vL07, p. 9]. We will, based on Luxburg's paper, approximate the RatioCut method for clustering to the case of identifying two clusters in a graph, a subset of vertices, and its complement.

This thesis is divided into seven chapters, each containing sub-chapters to further divide and analyse the concepts in a coherent and clear manner. The first chapter offers an introduction to the topic of Laplacian matrices and their application as being a tool for spectral clustering of graphs. The second contains an overview of definitions and theorems that are crucial for understanding the concepts discussed in the thesis. In the third chapter, we introduce two graphs which serve as a pedagogical

function throughout the rest of the thesis - all mathematical theory is applied to these graphs. Chapter four defines the Laplacian Matrix and reviews some of its spectral properties that matter for the introduction of the RatioCut clustering method in Chapter five. In Chapter five, the method is described and analysed for a case of two clusters. Chapter six presents an overview of the limitations of the theory discussed in this thesis and gives an introduction into future expansions of the topic of clustering using the Laplacian matrix. Lastly, Chapter 7 wraps the thesis up with a concise and clear summary of the topic discussed.

By reading this thesis on the Laplacian matrix as a tool for spectral clustering, the reader will gain an understanding of how the spectral properties of a graph, visualised by the Laplacian matrix, can be used to identify clusters in underlying data. A detailed approximation of the problem of the RatioCut method for identifying two clusters is included, which gives an insight into how clustering problems of graphs relate to the analysis of the eigenvalues and eigenvectors of the Laplacian matrix.

2 Theoretical background

This chapter will present the theoretical background needed in this thesis, and mathematical terms will be defined. The definitions and theorems are divided into the following areas: 2.1 Concepts in graph theory, and 2.2 Spectral properties of matrices.

2.1 Concepts in graph theory

This introductory section uses books written by Norman L. Biggs *Discrete Mathematics*, [Big02], and Bogdan Nica's *A Brief Introduction to Spectral Graph Theory*, [Nic18], to provide useful definitions and theorems of concepts in graph theory needed to further understand the reasoning and derivations made in this thesis.

Definition 2.1. A finite graph G is a structure consisting of a finite set of vertices V, and a set of edges E, that consists of 2-subsets of V. V is called a vertex set, and E is called an edge set, [Big02, p. 178]. We write

$$G = (V, E).$$

For two vertices u, v in V such that they are connected, we use the notation $u \sim v$ to indicate that they are an edge $\{u, v\}$ in E.

Definition 2.2. An adjacency matrix A of a graph G is a square matrix indexed by the vertex set V, given by

$$A(u, v) = \begin{cases} 1, & \text{if } u \sim v \\ 0, & \text{otherwise,} \end{cases}$$

where u and v are vertices in V, [Nic18, p. 63].

Definition 2.3. The degree of a vertex v in V, denoted $deg(v)$, is the number of edges in E that contain v. The degree matrix D is given by

$$D(u, v) = \begin{cases} \deg(v), & \text{if } u = v \\ 0, & \text{otherwise,} \end{cases}$$

[Nic18, p. 63].

Definition 2.4. A partition of a vertex set V is a family of non-empty sets V_i, such that V is the union of all sets V_i and each pair V_i, V_j $(i \neq j)$ is disjoint, [Big02, p. 126].

Definition 2.5. Let u, v be two vertices of a graph G. A path from u to v is a sequence of vertices $u = w_1, w_2, ..., w_n = v$, where for each i $(1 \leq i \leq n-1)$, either $w_i = w_{i+1}$ or $\{w_i, w_{i+1}\}$ is an edge, [Big02, p. 183].

Definition 2.6. We say that two vertices u and v in a graph are connected if there is a path from u to v. It is easy to check that being connected is an equivalence relation. The vertex set V is partitioned into k disjoint equivalence classes

$$V = V_1 \cup V_2 \cup \cdots \cup V_k,$$

where equivalence classes of this relation are called connected components of a graph. We say that a graph is connected if it has just one connected component. Equivalently, a graph is connected if for every two vertices u, v there is a path from u to v, [Big02, p. 183].

We use the word cluster to denote, informally, a subset of vertices of the vertex set V, that are more connected to each other than to vertices outside the subset. At this point, this is a heuristic rather than a rigorously defined notion. In Chapter 5, when we discuss the RatioCut method, we will see a way to formalize the notion of a cluster. Note that connected components are the most obvious types of clusters.

2.2 Spectral properties of matrices

This section uses books written by Friedberg et al. *Linear Algebra*, [FIS14], and Lütkepohl's *Handbook of Matrices*, [Lü96]. As the scope of this thesis concerns matrices, the definitions and theorems have been limited to the area relating to matrices, in order to present a concise and clear argument. This is recurring throughout the thesis.

Definition 2.7. Let M be a $n \times n$ matrix defined over a field F. A nonzero vector v in F is called an eigenvector of M if v satisfies $Mv = \lambda v$ for some scalar λ which is called the corresponding eigenvalue to the vector v, [FIS14, p. 246].

Theorem 2.8. *Let M be a $n \times n$ matrix defined over a field F. Then the following three statements are equivalent:*

1. *M is invertible,*
2. *If $Mv = 0$, then $v = 0$,*
3. *$det(M) \neq 0$,*

[FIS14, p. 71].

In this thesis we assume F to be the real numbers $\mathbb{R}$.

Definition 2.9. A real, symmetric $n \times n$ matrix M is positive semi-definite if $v^t M v \geq 0$ for all non-zero, real vectors v, [Lü96, p. 151].

Theorem 2.10. *A matrix M is positive semi-definite if it is a real, symmetric matrix and all its eigenvalues are non-negative, [Lü96, p. 267].*

3 Introductory example graphs

There will be two graphs used throughout the thesis as help when understanding the Laplacian matrix and its application to spectral clustering. Both graphs are similar, but when used together, offer a comprehensive way of explaining the definition of the Laplacian matrix. The inspiration for these graphs comes from a YouTube video titled "What is...the laplace matrix?", [Vis23, 1:10-4:57, 9:05-11:02].

Both graphs have seven nodes, with the only difference that Figure 1 has no edge between vertices 4 and 5, while Figure 2 has. By eye, we can see in Figure 1 that it consists of two connected components, while in Figure 2 there are two clusters connected by an edge.

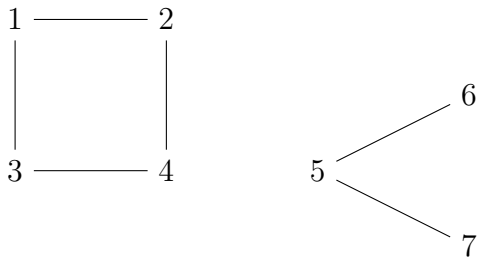

Figure 1: Graph with two connected components.

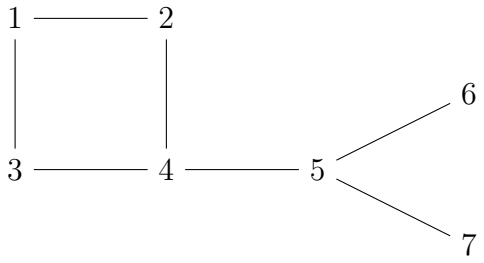

Figure 2: Connected graph.

4 The Laplacian matrix

4.1 Definition

We can now introduce the Laplacian matrix as an expression involving the adjacency matrix (Definition 2.2) and the degree matrix (Definition 2.3) defined above. There are several ways to define the Laplacian matrix; however, in this thesis, we will stick to Definition 4.1 below, which is used by Bogdan Nica in his book *A Brief Introduction to Spectral Graph Theory*, [Nic18, p. 63]. This definition is often called the "unnormalised" graph Laplacian.

Definition 4.1. The adjacency matrix A and the degree matrix D are related by the equation

$$L = D - A,$$

in other words, L is the Laplacian matrix defined as

$$L(u,v) = \begin{cases} \deg(v), & \text{u=v} \\ -1, & \text{if } u \sim v \\ 0, & \text{otherwise.} \end{cases}$$

It may be of relevance to be familiar with the existence of a normalised graph Laplacian as well. In her paper "A Tutorial on Spectral Clustering", [vL07, p. 5], Ulrike von Luxburg defines it in two versions as in Definition 4.2 below.

Definition 4.2. The symmetric matrix L_{sym} and the matrix L_{rw}, which is closely related to a "random walk" on a graph (stochastic process of random transitions from vertex to vertex, [vL07, p. 14]), are both called a normalised Laplacian matrix and are defined as

$$L_{\text{sym}} := D^{-1/2} L D^{-1/2} = I - D^{-1/2} A D^{-1/2},$$
$$L_{\text{rw}} := D^{-1} L = I - D^{-1} A.$$

We will not develop the reasoning about the normalised Laplacian matrix more than noting that it is useful for spectral clustering too, but with a slightly different method than the one that will be introduced for the unnormalised Laplacian matrix in Chapter 5. From now on, the unnormalised definition is meant when 'the Laplacian matrix' is mentioned.

4.1.1 Laplacian matrix for example graphs

Using Definition 4.1 we can construct the Laplacian matrix L as the difference $L = D - A$ and for the example graph in Figure 1 get

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

For the example graph in Figure 2 we get

$$L = \begin{bmatrix} 2 & -1 & -1 & 0 & 0 & 0 & 0 \\ -1 & 2 & 0 & -1 & 0 & 0 & 0 \\ -1 & 0 & 2 & -1 & 0 & 0 & 0 \\ 0 & -1 & -1 & 3 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 3 & -1 & -1 \\ 0 & 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 1 \end{bmatrix}.$$

4.2 Analogue of the second derivative

If we were to multiply the Laplacian matrix L, here using the matrix representing the graph in Figure 2, by a vector $\vec{u}$ representing every vertex in the graph, we would get a vector $L\vec{u}$ as follows:

$$L\vec{u} = \begin{bmatrix} 2 & -1 & -1 & 0 & 0 & 0 & 0 \\ -1 & 2 & 0 & -1 & 0 & 0 & 0 \\ -1 & 0 & 2 & -1 & 0 & 0 & 0 \\ 0 & -1 & -1 & 3 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 3 & -1 & -1 \\ 0 & 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \end{bmatrix} = \begin{bmatrix} 2u_1 - u_2 - u_3 \\ 2u_2 - u_1 - u_4 \\ 2u_3 - u_1 - u_4 \\ 3u_4 - u_2 - u_3 - u_5 \\ 3u_5 - u_4 - u_6 - u_7 \\ u_6 - u_5 \\ u_7 - u_5 \end{bmatrix}.$$

We can rewrite this as follows below, where we divide each row $i = 1, ..., 7$ in $L\vec{u}$ by the degree of u_i. This is the same operation as multiplying $L\vec{u}$ by the inverted diagonal matrix D^{-1}, which diagonal is given by $1/\deg(u_i)$. We denote the new matrix L'.

$$L' = D^{-1} \cdot L\vec{u} = \begin{bmatrix} u_1 - \frac{u_2+u_3}{2} \\ u_2 - \frac{u_1+u_4}{2} \\ u_3 - \frac{u_1+u_4}{2} \\ u_4 - \frac{u_2+u_3+u_5}{3} \\ u_5 - \frac{u_4+u_6+u_7}{3} \\ u_6 - \frac{u_5}{1} \\ u_7 - \frac{u_5}{1} \end{bmatrix}.$$

We can identify that every row i in L' shows the vertex u_i minus the average of its adjacent vertices, and can be represented by the expression

$$u_i \ - \frac{1}{\deg(u_i)} \sum_{u_j \sim u_i} u_j.$$

Intuitively, we can say that every row i gives a measure of how much u_i deviates from its adjacent vertices. In his article *On the Magical Properties of the Laplacian Matrix*, [Ita24], Itai describes this as an analogue of the second derivative as it strongly reminds one of the definition of the second derivative of functions, which is given in Definition 4.3 below.

Definition 4.3. The second derivative of a function $f(x)$ is given by

$$f''(x) = \lim_{h \to 0} \frac{f(x+h) - 2f(x) + f(x-h)}{h^2}.$$

Itai continues by approximating the second derivative to

$$f''(x) \approx \frac{f(x+h) - 2f(x) + f(x-h)}{h^2},$$

which can be rewritten as

$$\begin{aligned} h^2 \cdot f''(x) &\approx f(x+h) - 2f(x) + f(x-h) \\ &= -2f(x) + f(x+h) + f(x-h) \\ &= -2\left(f(x) - \frac{f(x+h) + f(x-h)}{2}\right). \end{aligned}$$

If we divide both sides by -2 we obtain

$$-\frac{1}{2}h^2 \cdot f''(x) \approx f(x) - \frac{f(x+h) + f(x-h)}{2},$$

where the right-hand side of the approximation side looks similar to the expression we obtained earlier by looking at the example graph in Figure 2. If $f(x)$ represents the vertex u_i, then $(f(x+h) + f(x-h))/2$ represents the average of its adjacent vertices. We get a margin of error of $-h^2/2$ when deriving an approximation of the second derivative. This is why we say the Laplacian can be viewed as an *analogue* of the second derivative, even though it does not satisfy an equality. This is interesting as spectral graph theory often aims to find patterns in a graph, which is stated by Itai, [Ita24].

4.3 Computing eigenvalues and eigenvectors

To be able to compute eigenvectors and eigenvalues, we need to be certain that such exist. The Laplacian matrix is a real, symmetric matrix, which entails that we can apply the spectral theorem. For the case of real, symmetric matrices, Nica formulates it as in Theorem 4.4 below, [Nic18, p. 64].

Theorem 4.4. *If M is a real, symmetric $n \times n$ matrix, then there exists an orthogonal basis of eigenvectors of M, and M has n real eigenvalues counted with multiplicities.*

The Laplacian matrix L satisfies the criteria for Theorem 4.4 since it is a real, symmetric $n \times n$ matrix. Here n is the number of vertices in the underlying graph with vertex set V, so the Laplacian matrix L has $n = |V|$ real eigenvalues, [Nic18, p. 64]. In other words, the number of real eigenvalues equals the number of vertices in the graph. Each eigenvalue λ has an associated eigenspace, whose non-zero elements are the eigenvectors of λ in the vector space $\mathbb{R}^n$.

Note that elements of $\mathbb{R}^n$ can be interpreted as functions $f : V \to \mathbb{R}$. In particular, eigenvectors of L can be interpreted as a function of such form that assigns a real number to each vertex in the vertex set V. This is why eigenvectors are often called eigenfunctions. To compute the eigenvalues, we can use Theorem 4.5 below.

Theorem 4.5. *Let M be a $n \times n$ matrix defined over the field F. A scalar λ is an eigenvalue of M if and only if $\det(M - \lambda I_n) = 0$, [FIS14, p. 248].*

Proof. We prove both ways of the equivalence:

1. By Definition 2.7, if λ is an eigenvalue of M, then $Mv = \lambda v$ for a nonzero vector v in F. By factorisation, we can write $(M - \lambda I_n)v = 0$. Because $v \neq 0$, the matrix $M - \lambda I_n$ is not invertible, that is equivalent to $\det(M - \lambda I_n) = 0$, according to Theorem 2.8.

2. By Theorem 2.8, if $\det(M-\lambda I_n) = 0$, then the matrix $M-\lambda I_n$ is not invertible. Equally, there exists a nonzero vector v in F such that $(M - \lambda I_n)v = 0$. Restructuring the equation, we get $Mv = \lambda v$, so λ is an eigenvalue of M by Definition 2.7.

□

Computing the eigenvalues from the equivalence $\det(M - \lambda I_n) = 0$, is called 'solving the characteristic equation'.

Definition 4.6. The function $f(t) = \det(M - tI_n)$ is called the characteristic polynomial of the matrix M, [FIS14, p. 248].

Once the eigenvalues are computed, the associated non-zero eigenvectors v can be found by solving the equation

$$(M - \lambda I_n)v = 0$$

for each eigenvalue λ, [FIS14, p. 250].

4.3.1 Computing eigenvalues for example graphs

By applying Theorem 4.5, the following characteristic equation is produced for Figure 1:

$$\det(L - \lambda I_7) = \det \begin{bmatrix} 2-\lambda & -1 & -1 & 0 & 0 & 0 & 0 \\ -1 & 2-\lambda & 0 & -1 & 0 & 0 & 0 \\ -1 & 0 & 2-\lambda & -1 & 0 & 0 & 0 \\ 0 & -1 & -1 & 2-\lambda & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2-\lambda & -1 & -1 \\ 0 & 0 & 0 & 0 & -1 & 1-\lambda & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 1-\lambda \end{bmatrix} = 0.$$

This is a large matrix to compute the determinant by hand, so we use the programming language R to compute eigenvalues (code is included in Appendix A.1). This gives us:

$$\lambda_1 = 0, \lambda_2 = 0, \lambda_3 = 1, \lambda_4 = 2, \lambda_5 = 2, \lambda_6 = 3, \lambda_7 = 4$$

For Figure 2 we get

$$\det(L - \lambda I_7) = \det \begin{bmatrix} 2-\lambda & -1 & -1 & 0 & 0 & 0 & 0 \\ -1 & 2-\lambda & 0 & -1 & 0 & 0 & 0 \\ -1 & 0 & 2-\lambda & -1 & 0 & 0 & 0 \\ 0 & -1 & -1 & 3-\lambda & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 3-\lambda & -1 & -1 \\ 0 & 0 & 0 & 0 & -1 & 1-\lambda & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 1-\lambda \end{bmatrix} = 0.$$

Using the programming language R (code is included in Appendix A.1), approximated to 3 decimals, we get the eigenvalues

$$\lambda_1 = 0.000, \lambda_2 = 0.359, \lambda_3 = 1.000, \lambda_4 = 2.000, \lambda_5 = 2.276, \lambda_6 = 3.589, \lambda_7 = 4.776.$$

4.4 0 is a Laplacian eigenvalue

As already done with the eigenvalues for the two example graphs, Figure 1 and Figure 2, we can order the Laplacian eigenvalues as follows:

$$\lambda_{\min} = \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n = \lambda_{\max},$$

[Nic18, p. 64]. For both Figure 1 and Figure 2, we can note that $\lambda_1 = 0$. That zero is a Laplacian eigenvalue is an useful property, which is stated in Theorem 4.9. The theorem is proved by Nica, [Nic18, p. 65-66], under the assumption that a graph is connected, and therefore, he is using the condition that an eigenfunction of zero is constant. However, we will prove the theorem without this assumption, which is why we instead have the condition that the function is locally constant, as defined in Definition 4.7 below.

Definition 4.7. If $f : V \to \mathbb{R}$ is a function that satisfies that $f(u) = f(v)$ when u and v are in the same connected component A in a graph G, then f is called locally constant. If G is connected, then f is constant.

To prove Theorem 4.9, we also need Lemma 4.8, which is a rephrasing of the scalar product $Lf \cdot f$. The rephrasing uses the fact that the Laplacian matrix L is related to the degree matrix D and the adjacency matrix A by the equation $L = D - A$.

Lemma 4.8. *If L is a Laplacian matrix and f is a function $f : V \to \mathbb{R}$, then L satisfies the equality*

$$Lf \cdot f = \sum_{\{u,v\} \in E} (f(u) - f(v))^2,$$

[Nic18, p. 64].

Proof. The scalar product is given by

$$Lf \cdot f = \sum_{v \in V} (Lf)(v) f(v),$$

where $Lf(v)$ can be written as

$$Lf(v) = \deg(v) f(v) - \sum_{u : u \sim v} f(u)$$

since $L = D - A$, [Nic18, p. 63-64]. We get

$$\begin{aligned} Lf \cdot f &= \sum_{v \in V} \left(\deg(v) f(v) - \sum_{u : u \sim v} f(u) \right) f(v) \\ &= \sum_{v \in V} \deg(v) f(v)^2 - \sum_{v \in V} \sum_{u : u \sim v} f(u) f(v). \end{aligned}$$

We will rewrite the sums so they range over edges $\{u, v\} \in E$ instead. The first term sums over all vertices $v \in V$ where each $f(v)^2$ is counted $\deg(v)$ times, in other words, once for every edge $\{u, v\} \in E$ it is part of. This includes both $f(u)^2$ and $f(v)^2$. In the second sum, the outermost summation ranges over all vertices $v \in V$, so, for every vertex v, we count all its adjacent vertices u. Therefore, when we sum over edges $\{u, v\} \in E$ instead, we count every $f(u)f(v)$ twice. We get

$$\begin{aligned} Lf \cdot f &= \sum_{\{u,v\}\in E} (f(u)^2 + f(v)^2) - \sum_{\{u,v\}\in E} 2f(u)f(v) \\ &= \sum_{\{u,v\}\in E} (f(u)^2 - 2f(u)f(v) + f(v)^2) \\ &= \sum_{\{u,v\}\in E} (f(u) - f(v))^2. \end{aligned}$$

□

We now have all the pieces needed to be able to prove that zero is a Laplacian eigenvalue. The proof of Theorem 4.9 is highly inspired by Nica, [Nic18, p. 65-66].

Theorem 4.9. *A function $f : V \to \mathbb{R}$ is an eigenfunction with eigenvalue 0 of the Laplacian matrix L if and only if it is locally constant.*

Proof. We prove the theorem in two parts:

1. Let $f : V \to \mathbb{R}$ be a locally constant function in $\mathbb{R}$. Then,

$$Lf(v) = \deg(v)f(v) - \sum_{u:u\sim v} f(u) = f(v)(\deg(v) - \deg(v)) = 0,$$

   for all $v \in V$. So, f is an eigenfunction with associated eigenvalue 0.

2. Let $f : V \to \mathbb{R}$ be an eigenfunction with eigenvalue 0. By Definition 2.7 this means $Lf = 0$. We have the scalar product

$$Lf \cdot f = \sum_{v\in V} (Lf)(v)f(v).$$

   By Lemma 4.8 it holds that

$$Lf \cdot f = \sum_{\{u,v\}\in E} (f(u) - f(v))^2.$$

   Because $Lf = 0$ we then have

$$0 = \sum_{\{u,v\}\in E} (f(u) - f(v))^2,$$

which leads to $f(u) = f(v)$ for all edges $\{u, v\} \in E$. So, f is locally constant.

□

This way, we can refer to zero as the trivial Laplacian eigenvalue. Its multiplicity is the number of connected components of a graph, [Nic18, p. 66], which we will learn more about in the chapter on clustering.

4.5 All Laplacian eigenvalues are non-negative

We have shown that the eigenvalue zero is a Laplacian eigenvalue, but not that it is the smallest one. This follows trivially if we can show that all Laplacian eigenvalues are non-negative. We prove this in Theorem 4.10 below, [vL07, p. 4].

Theorem 4.10. *The Laplacian matrix L has n non-negative eigenvalues.*

Proof. In Lemma 4.8 we proved the following equality for locally constant eigenfunctions $f : V \to \mathbb{R}$:

$$Lf \cdot f = \sum_{\{u,v\} \in E} (f(u) - f(v))^2.$$

The right side of the equality is a sum of squares and therefore non-negative, which implies that $Lf \cdot f \geq 0$ for all non-zero eigenfunctions $f : V \to \mathbb{R}$. Therefore, L is positive semi-definite by Definition 2.9. By Theorem 2.10 this means the Laplacian matrix has n non-negative eigenvalues counted with multiplicities. □

Earlier we ordered the Laplacian eigenvalues as

$$\lambda_{\text{min}} = \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_{\text{n}} = \lambda_{\text{max}}.$$

Now we can conclude that the smallest eigenvalue $\lambda_{\text{min}} = 0$ for all Laplacian matrices.

5 Clustering using the Laplacian matrix

5.1 Number of connected components

In Theorem 4.9, we proved that zero is a Laplacian eigenvalue. We will now learn to use the eigenvalue 0 as an indicator of the number of connected components $A_1, ..., A_k$ in a graph. Luxburg formulates a theorem, [vL07, p. 4], here stated in Theorem 5.1, whose proof is highly inspired by her.

Theorem 5.1. *Let G be the underlying graph to a Laplacian matrix L. The multiplicity k of the eigenvalue zero of L equals the number of connected components $A_1, ..., A_k$ in the graph. Indicator vectors $\mathbb{1}_{A_1}, ..., \mathbb{1}_{A_k}$ span the eigenspace of the eigenvalue 0.*

Proof. We prove the theorem by dividing it into two cases: $k = 1$ and $k \geq 1$.

1. Let $k = 1$, meaning the graph is connected, equally, contains only one connected component A_1. From Theorem 4.9, we know that 0 is a Laplacian eigenvalue whose eigenvectors are locally constant functions. Because G is connected, the only locally constant functions are the constant ones, and the space of constant functions has dimension one. It follows that in this case, the eigenvalue zero has multiplicity one with corresponding eigenvector $\mathbb{1}_{A_1}$.

2. Let $k \geq 1$, so that the graph contains $k \geq 1$ connected components $A_1, ..., A_k$. For the same reason as above, 0 is a Laplacian eigenvalue with the locally constant eigenfunction f. Assuming the vertices are ordered according to the connected components to which they belong to, we can for each connected component $A_1, ..., A_k$ construct a Laplacian matrix $L_1, ..., L_k$. The Laplacian matrix L for the whole graph G will then be

$$L = \begin{pmatrix} L_1 & & & \\ & L_2 & & \\ & & \ddots & \\ & & & L_k \end{pmatrix}.$$

   L is a block sum matrix of the union $L_1 \cup L_2 \cup ... \cup L_k$. From the case $k = 1$ we know that every Laplacian matrix L_i has the eigenvalue 0 with multiplicity 1, with corresponding eigenvector $\mathbb{1}_{A_i}$. Since L is the block sum of $L_1, ..., L_k$ it

then has k eigenvalues that equals 0.

□

5.1.1 Number of connected components in example graphs

In Theorem 5.1 above, we proved that the multiplicity of the eigenvalue zero equals the number of connected components in a graph. Figure 1 consists of two connected components which we can now conclude directly from the eigenvalues of its associated Laplacian matrix. We can see that it has two connected components as it has $k = 2$ eigenvalues that equals zero; namely $\lambda_1 = \lambda_2 = 0$. Its eigenvalues, which we computed earlier, are

$$\lambda_1 = 0, \lambda_2 = 0, \lambda_3 = 1, \lambda_4 = 2, \lambda_5 = 2, \lambda_6 = 3, \lambda_7 = 4.$$

For Figure 2, which is a connected graph, and therefore is a case of part one of the proof of Theorem 5.1 above, has $k = 1$ eigenvalues that equal zero; $\lambda_1 = 0$. The eigenvalues of its Laplacian matrix are

$$\lambda_1 = 0, \lambda_2 = 0.359, \lambda_3 = 1.000, \lambda_4 = 2.000, \lambda_5 = 2.276, \lambda_6 = 3.589, \lambda_7 = 4.776.$$

5.2 RatioCut clustering method

We will now assume that a graph G is connected, meaning it consists of only one connected component. An example of such a graph is Figure 2, as we have seen before. The RatioCut clustering method is a method for partitioning the graph into clusters that are not completely disconnected from each other. The method aims to minimize the number of edges between the two clusters, which we want to construct, while maximizing the number of edges within a cluster, [vL07, p. 9]. This way, the RatioCut clustering method helps us identify clusters in a graph. We will focus on the case of partitioning G into two clusters, and based on Ulrike von Luxburg's paper "A Tutorial on Spectral Clustering, [vL07, p. 10-11], derive an approximation of the RatioCut method for this specific case.

Before introducing the RatioCut definition, which is given for the general case of k clusters in Definition 5.2, we need to start referring to the number of edges within a set as the 'weight', w. An edge between two vertices u_i and u_j is undirected, which is why $w_{ij} = w_{ji}$, [vL07, p. 2]. This equals 1 if $\{u_i, u_j\} \in E$ is an edge and zero

otherwise. You could say the adjacency matrix $A(u_i, u_j)$ defined as in Definition 2.2 is a matrix that displays the weights between vertices in a graph G. For two subsets A and its complement $\overline{A}$ of V, we use the notation

$$W(A, \overline{A}) := \sum_{i \in A, j \in \overline{A}} w_{ij},$$

[vL07, p. 2]. In other words, $W(A, \overline{A})$ is the number of all edges connecting a vertex in A to a vertex that is not in A. Now we have a clear understanding of all components needed in Definition 5.2 below.

Definition 5.2. Given k connected components $A_1, ..., A_k$, the objective function for RatioCut is defined as

$$\text{RatioCut}(A_1, ..., A_k) := \sum_{i=1, j \neq i}^{k} \frac{W(A_i, \overline{A_j})}{|A_i|},$$

[vL07, p. 9].

As mentioned, we will focus on the case of $k = 2$ clusters where we have the subsets A and $\overline{A}$. Then Definition 5.2 can be simplified accordingly:

$$\begin{aligned} \text{RatioCut}(A, \overline{A}) &= \frac{W(A, \overline{A})}{|A|} + \frac{W(A, \overline{A})}{|\overline{A}|} \\ &= W(A_i, \overline{A_i}) \left(\frac{1}{|A|} + \frac{1}{|\overline{A}|} \right). \end{aligned}$$

The sum $1/|A| + 1/|\overline{A}|$ is minimized if the size of the subsets satisfies $|A| = |\overline{A}|$, in other words, our goal is to identify a partition where $|A|$ and $|\overline{A}|$ have as similar size as possible, and the number of edges between A and $\overline{A}$ is as small as possible. This is an optimization problem of minimizing the RatioCut objective function where A and $\overline{A}$ are subsets of the vertex set V, [vL07, p. 10], namely

$$\min_{A \subset V} \text{RatioCut}(A, \overline{A}).$$

To achieve a solution to this problem, we will express it in terms of the quadratic form $f^t L f$, which is formulated in Lemma 5.4 below. The vector f is defined in a specific way according to Luxburg in Definition 5.3, [vL07, p. 10].

Definition 5.3. $f = (f_1, ..., f_n)^t \in \mathbb{R}^n$ is a vector given that all u_i $(1 \leq i \leq n)$ are vertices in V, and that A and $\overline{A}$ are subsets of V. Each coordinate f_i in f is given by

$$f_i = \begin{cases} \sqrt{\frac{|\overline{A}|}{|A|}}, & \text{if } u_i \in A, \\ -\sqrt{\frac{|A|}{|\overline{A}|}}, & \text{if } u_i \in \overline{A}. \end{cases}$$

The rephrasing of $f^t L f$ follows in Lemma 5.4 below.

Lemma 5.4. *The quadratic form of the Laplacian matrix, $f^t L f$, can be expressed in terms of the RatioCut for $k = 2$ clusters as*

$$f^t L f = |V| \cdot \mathit{RatioCut}(A, \overline{A}),$$

[vL07, p. 10].

Proof. The quadratic form $f^t L f$ equals the scalar product $Lf \cdot f$, so by Lemma 4.8 it holds that

$$f^t L f = \sum_{\{u_i, u_j\} \in E} (f_i - f_j)^2.$$

Inserting the values of f_i and f_j according to the definition of f in Definition 5.3, the sum now ranges over edges $\{u_i, u_j\} \in E$ where u_i and u_j belong to different subsets. By simplification, we get

$$\begin{aligned} f^t L f &= \sum_{\{i,j\} \in E, i \in A, j \in \overline{A}} \left(\sqrt{\frac{|\overline{A}|}{|A|}} + \sqrt{\frac{|A|}{|\overline{A}|}} \right)^2 \\ &= \sum_{\{i,j\} \in E, i \in A, j \in \overline{A}} \left(\frac{|\overline{A}|}{|A|} + 2 + \frac{|A|}{|\overline{A}|} \right) \\ &= \sum_{\{i,j\} \in E, i \in A, j \in \overline{A}} \left(\frac{|A| + |\overline{A}|}{|A|} + \frac{|A| + |\overline{A}|}{|\overline{A}|} \right). \end{aligned}$$

The summation range over all edges between A and $\overline{A}$, which is the sum of all weights, $W(A, \overline{A})$. We can also note that $|A| + |\overline{A}| = |V|$ since A is a subset of V

and $\overline{A}$ is its complement. Therefore, it follows that

$$\begin{aligned} f^t L f &= W(A, \overline{A}) \left(\frac{|V|}{|A|} + \frac{|V|}{|\overline{A}|} \right) \\ &= |V| \cdot W(A, \overline{A}) \left(\frac{1}{|A|} + \frac{1}{|\overline{A}|} \right). \end{aligned}$$

We can now recognise the RatioCut definition for the case $k = 2$ which follows from Definition 5.2, and get

$$f^t L f = |V| \cdot \text{RatioCut}(A, \overline{A})$$

□

To continue the approximation of minimizing the RatioCut objective function for two clusters A and $\overline{A}$, we will look at some properties of the vector f, defined as in Definition 5.3. As Luxburg notes, the sum of all coordinates f_i equals zero according to the calculation below, [vL07, p. 10]. By this, it follows that the vector f is orthogonal to the constant vector $\mathbb{1}$ as the inner product $\langle f, \mathbb{1} \rangle$ then equals zero.

$$\begin{aligned} \sum_{i=1}^{n} f_i &= \sum_{i \in A} \sqrt{\frac{|\overline{A}|}{|A|}} + \sum_{i \in \overline{A}} \left(-\sqrt{\frac{|A|}{|\overline{A}|}} \right) \\ &= \sum_{i \in A} \sqrt{\frac{|\overline{A}|}{|A|}} - \sum_{i \in \overline{A}} \sqrt{\frac{|A|}{|\overline{A}|}} \\ &= |A| \sqrt{\frac{|\overline{A}|}{|A|}} - |\overline{A}| \sqrt{\frac{|A|}{|\overline{A}|}} \\ &= \sqrt{|A|}\sqrt{|\overline{A}|} - \sqrt{|\overline{A}|}\sqrt{|A|} \\ &= 0. \end{aligned}$$

Luxburg continues by noting that the vector f satisfies that the length of the vector squared equals n according to

$$
\begin{aligned}
||f||^2 &= \sum_{i=1}^{n} f_i^2 \\
&= \sum_{i \in A} f_i^2 + \sum_{i \in \overline{A}} f_i^2 \\
&= \sum_{i \in A} \frac{|\overline{A}|}{|A|} + \sum_{i \in \overline{A}} \frac{|A|}{|\overline{A}|} \\
&= |A| \frac{|\overline{A}|}{|A|} + |\overline{A}| \frac{|A|}{|\overline{A}|} \\
&= |\overline{A}| + |A| \\
&= |V| \\
&= n,
\end{aligned}
$$

[vL07, p. 10].

Since $f^t L f = |V| \cdot \text{RatioCut}(A, \overline{A})$ according to Lemma 5.4, the optimization problem of minimizing the $\text{RatioCut}(A, \overline{A})$ is equivalent to minimizing the quadratic form under the conditions that f, given in Definition 5.3, is orthogonal to the constant vector $\mathbb{1}$, and that the length of f is $\sqrt{n}$, [vL07, p. 10].

$$
\min_{A \subset V} f^t L f \quad \text{subject to} \quad f \perp \mathbb{1}, \quad ||f|| = \sqrt{n}.
$$

As Luxburg states, this is a discrete optimization problem because of the condition that f is defined as in Definition 5.3, which is only allowed to take two values, [vL07, p. 10]. This problem is known to be NP-hard, which means that there is no polynomial-time algorithm known for it. This is why we make a relaxation by dropping the condition of f being defined as in Definition 5.3, and letting f_i be arbitrary real numbers:

$$
\min_{f \in \mathbb{R}} f^t L f \quad \text{subject to} \quad f \perp \mathbb{1}, \quad ||f|| = \sqrt{n},
$$

[vL07, p. 11].

We will now introduce the Rayleigh-Ritz theorem for real, symmetric matrices that is mentioned by Helmut Lütkepohl in his book "Handbook of Matrices", [Lü96, p. 67]. We formulate it in Theorem 5.5 below.

Theorem 5.5. *If M is a real, symmetric $n \times n$ matrix, then*

$$\lambda_{\min}(M) = \min\left\{\frac{x^t M x}{x^t x} : x \in \mathbb{R}^{n\times 1}, x \neq 0\right\},$$
$$\lambda_{\max}(M) = \max\left\{\frac{x^t M x}{x^t x} : x \in \mathbb{R}^{n\times 1}, x \neq 0\right\}.$$

In our case, M in Theorem 5.5 is the Laplacian matrix L, and the vector x is f. It says that the smallest eigenvalue $\lambda_{\min}$ of L is the eigenvalue that is minimizing the Rayleigh quotient, which is written below in the form that Aaron Sidford uses in his lecture notes on spectral graph theory, [Sid18, p. 8]:

$$R(f) = \frac{f^t L f}{f^t f}.$$

Luxburg states that it from Theorem 5.5 can be seen that the solution to the relaxation of the optimization problem is given by the vector f, which is the eigenvector corresponding to the second smallest eigenvalue of L, [vL07, p. 11]. We will now motivate this statement.

Let us note that the spectral theorem (Theorem 4.4) implies that the symmetric matrix L can be written in the form $L = Q^t D Q$ where Q is an orthogonal $n\times n$ matrix whose columns are eigenvectors of L, and D is a diagonal matrix of eigenvalues of L such that

$$L = Q^t D Q = Q^t \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix} Q.$$

Then the Rayleigh quotient can be rewritten using the equality $L = Q^t D Q$ as below. By assigning a vector $Y = (y_1, ..., y_n)^t$ such that $Qf = Y$ and $||Y|| = 1$, we get

$$R(f) = \frac{f^t Q^t D Q f}{f^t Q^t Q f} = \frac{Y^t D Y}{Y^t Y} = \frac{\lambda_1 y_1^2 + ... + \lambda_n y_n^2}{y_1^2 + ... + y_n^2},$$

[Sid18, p. 8].

In the optimization problem, we have the condition that f is orthogonal to the

constant vector $\mathbb{1}$, and as mentioned, Q is an orthogonal matrix. This implies that

$$f \perp \mathbb{1} \Rightarrow Y = Qf \perp Q\mathbb{1} = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix},$$

meaning Y is orthogonal to the vector $(1, 0, ..., 0)^t$, which leads to $y_1 = 0$. Therefore, we can rewrite $R(f)$ as

$$R(f) = \frac{\lambda_2 y_2^2 + ... + \lambda_n y_n^2}{y_2^2 + ... + y_n^2}.$$

This can be seen as the weighted average of the eigenvalues $\lambda_2 \leq ... \leq \lambda_n$, which is minimized when the coefficients of $\lambda_3, ..., \lambda_n$ are all zero, and the coefficient of λ_2 is one. Therefore, the Rayleigh quotient is minimized by the eigenvector f associated with the eigenvalue λ_2, and just like Sidford, [Sid18, p. 9], we get

$$\min_{f \in \mathbb{R}} R(f) = \min_{f \in \mathbb{R}} \frac{f^t L f}{f^t f} = \lambda_2.$$

So, an approximate minimization of RatioCut is given by the eigenvector of the second smallest eigenvalue of L; λ_2, [vL07, p. 11]. This means that the relaxed optimization problem

$$\min_{f \in \mathbb{R}} f^t L f \quad \text{subject to} \quad f \perp \mathbb{1}, \quad ||f|| = \sqrt{n}.$$

is minimized when we use the eigenvector f, which corresponds with λ_2.

5.3 RatioCut clustering method in practice

Now that we have approximated the RatioCut clustering method for two clusters, we will see how the eigenvector of the second smallest eigenvalue can be used as an indicator for how to cluster a graph.

The vector f was earlier defined as $f = (f_1, ..., f_n)^t \in \mathbb{R}^n$ in Definition 5.3, where f_i is given by

$$f_i = \begin{cases} \sqrt{\frac{|\overline{A}|}{|A|}}, & \text{if } u_i \in A, \\ -\sqrt{\frac{|A|}{|\overline{A}|}}, & \text{if } u_i \in \overline{A}. \end{cases}$$

To obtain a partition of the graph where each of A and $\overline{A}$ are clusters, we have to rewrite the real-valued solution vector f of the relaxed problem into a discrete indicator vector, [vL07, p. 11]. This is done in terms of the vertex u_i, where the sign of f_i becomes an indicator of what subset u_i belongs to, such that

$$\begin{cases} u_i \in A, & \text{if } f_i \geq 0, \\ u_i \in \overline{A}, & \text{if } f_i < 0. \end{cases}.$$

Since f is proved to be the eigenvector of the second smallest eigenvalue, vertices associated with a coordinate f_i that is greater or equal to zero belong to one subset A of the vertices - a cluster. Vertices associated with a coordinate f_i that is smaller than zero belong to the complement $\overline{A}$ - another cluster. This is only one of many indicators that can be derived from the Laplacian matrix, and be used to divide a graph into two clusters, which will be discussed in Chapter 6. However, the sign of f_i is what will be used in this thesis when applying the RatioCut method to graphs.

To summarize how the approximated RatioCut clustering method for two clusters A and $\overline{A}$ can be used; we formulate a short step-by-step instruction:

1. Identify the Laplacian matrix L according to Definition 4.1.

2. Identify the eigenvector f associated with the second smallest eigenvalue λ_2.

3. Divide the vertex set V into two clusters A and $\overline{A}$ based on the value of f_i where
$$\begin{cases} u_i \in A, & \text{if } f_i \geq 0, \\ u_i \in \overline{A}, & \text{if } f_i < 0. \end{cases}.$$

5.3.1 Applying RatioCut clustering method to example graph

We will now apply the RatioCut method approximated to two clusters to the example graph in Figure 2, which is a connected graph. As earlier calculated in Chapter 4.1.1,

its associated Laplacian matrix L equals

$$L = \begin{bmatrix} 2 & -1 & -1 & 0 & 0 & 0 & 0 \\ -1 & 2 & 0 & -1 & 0 & 0 & 0 \\ -1 & 0 & 2 & -1 & 0 & 0 & 0 \\ 0 & -1 & -1 & 3 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 3 & -1 & -1 \\ 0 & 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 1 \end{bmatrix}.$$

According to calculations in Chapter 4.3.1, the eigenvalues are computed to

$$\lambda_1 = 0, \lambda_2 = 0.359, \lambda_3 = 1.000, \lambda_4 = 2.000, \lambda_5 = 2.276, \lambda_6 = 3.589, \lambda_7 = 4.776.$$

The second smallest eigenvalue is noted to be $\lambda_2 = 0.359$. Using the programming language R (code is provided in Appendix A.1), the eigenvector f associated to λ_2, with values approximated to three decimals, is computed to

$$f = \begin{bmatrix} -0.424 \\ -0.348 \\ -0.348 \\ -0.147 \\ 0.308 \\ 0.480 \\ 0.480 \end{bmatrix}.$$

We can see that those f_i such that $i = 1, 2, 3, 4$ are negative values, while those f_i such that $i = 5, 6, 7$ are positive values. Therefore, we can conclude that vertices $u_1, u_2, u_3, u_4 \in V$ belong to one cluster A, and that $u_5, u_6, u_7 \in V$ belong to another cluster $\overline{A}$. This corresponds to how we described Figure 2 in Chapter 3, where we noted that the only difference to Figure 1, is that Figure 2 includes an edge between vertex u_4 and u_5, while Figure 1 does not. Between vertices u_4 and u_5 is a natural spot to partition the graph into two clusters, as this is the partition that minimizes the number of edges between two clusters at the same time as A and $\overline{A}$ are as similar in size as possible.

6 Discussion

6.1 Limitations and critique

This thesis has mainly focused on the unnormalised version of the Laplacian matrix, its spectral properties, and its application to spectral clustering using the RatioCut clustering method. However, it must be emphasized that it has certain limitations.

Firstly, we focused on the unnormalised Laplacian matrix, while, as earlier mentioned, there are normalised definitions of the Laplacian matrix too. Luxburg, [vL07, p. 24], writes that for graphs that are very regular and most vertices have approximately the same degree, it does not matter which definition is chosen. But for the case of spectral clustering, the unnormalised Laplacian matrix is sensitive to variations of the degree of vertices in the graph, [vL07, p. 22], as A and $\overline{A}$ are not necessarily similar within each cluster even if they are similar in size. Although the RatioCut clustering method works well for smaller graphs, it may not do so for larger, more irregular graphs. This is due to the RatioCut method not partitioning the graph in order to maximise the similarity of vertices within a cluster, but rather to minimise the similarities of different clusters, [vL07, p. 25]. Spectral clustering methods using the normalised Laplacian matrix, eg. NCut (also covered by Luxburg), take both objectives into account. No comparisons between the unnormalised and normalised Laplacian matrices, and not between the RatioCut and NCut clustering methods, were included in this thesis.

Secondly, the relaxation of the RatioCut clustering method - an approximation of the RatioCut minimization problem to the case of two clusters - can not be guaranteed to be a good solution compared to the exact one, [vL07, p. 13]. The problem of minimizing the RatioCut by an exact solution is, as earlier mentioned, known to be NP-hard, and nothing that was covered in this thesis.

Finally, this thesis is primarily theoretical. While several spectral properties of the unnormalised Laplacian matrix were discussed, and an approximation of the RatioCut clustering method was derived, no empirical investigation on any real-world clustering problem was performed. This limits the possibility of drawing any general conclusions about the effectiveness of the topics discussed in this thesis.

However, having identified limitations of this thesis, it has to be stressed that (apart from the uncertainty of the quality of the approximation of the RatioCut method for two clusters) this does not have an influence on the calculations that are presented. The aim of this thesis was to present the Laplacian matrix and give an

introduction to its applicability as a tool for spectral clustering in a clear and pedagogical way. By including example graphs, this has been achieved through deriving one clustering method, the RatioCut method, and applying it to the mentioned graphs.

Keeping the limitations in mind, this thesis still presents valuable information that can be of interest in further studies on the topic and be expanded in the future.

6.2 Further outlook

This thesis can be used as a basis for further expansions of the topic of the Laplacian matrices as a tool for spectral clustering, as it provides a detailed starting-point for further research. The natural next step for the reader who want to learn more, is to read about the normalised Laplacian matrix and the NCut clustering method to gain more knowledge and a broader understanding of the subject. Deriving the spectral properties of the normalised definition in a similar way to what was provided for the unnormalised Laplacian in this thesis, would supply a foundation to continue the reasoning on spectral clustering in terms of NCut. It is recommended to read the paper "A Tutorial on Spectral Clustering" by Ulrike von Luxburg, [vL07], as she reasons about both kinds of Laplacian matrices and different clustering methods.

Another interesting expansion of research would be to apply these clustering methods to real data and try to validate their correctness for large data sets. Real-world data often tends to take more parameters into account, and as the graphs considered in this thesis are undirected, it would be of relevance to examine the applicability of Laplacian spectral clustering for directed graphs.

7 Summary

This thesis focused on the Laplacian matrix as a tool for spectral clustering. It provided an introduction to the concept of the Laplacian matrix and its spectral properties, presenting important definitions and theorems relating to the topic. Through the introduction of example graphs, the reader got the opportunity to practically see how the mathematical theory introduced can be applied to undirected graphs.

The RatioCut method was also introduced, which is a clustering method using Laplacian, which aims to minimize the number of edges between clusters in a graph. We derived an approximation of this method to the case of identifying two clusters. It was derived that the eigenvector associated with the second smallest eigenvalue achieves this minimization, and that the signs of each coordinate in the eigenvector can be used to indicate which vertices should be divided into which cluster.

The limitations of the thesis were acknowledged, where the main takeaway is the fact that we in this thesis use the unnormalised Laplacian matrix while it might be of greater relevance for applications to get familiar with the normalised Laplacian matrix. Researching this definition as well, together with its spectral properties, would provide a foundational basis to further gain knowledge on spectral clustering using Laplacian matrices.

To conclude, the mathematical calculations and theorems presented in this thesis offer a valuable look into the Laplacian Matrix and its spectral properties. It also provides the reader with a starting-point to further learn about spectral clustering.

References

[Big02] Norman L. Biggs. *Discrete Mathematics*. Oxford University Press, second edition, 2002.

[FIS14] Stephen Friedberg, Arnold Insel, and Lawrence Spence. *Linear Algebra*. Pearson, fourth edition, 2014.

[Ita24] Uri Itai. *On the Magical Properties of the Laplacian Matrix*, 2024. Medium, June 10. Accessed: May 18, 2025. URL: https://medium.com/coinmonks/on-the-magical-properties-of-the-laplacian-matrix-2a38e2817dd1.

[Lü96] Helmut Lütkepohl. *Handbook of matrices*. John Wiley and Sons, first edition, 1996.

[Nic18] Bogdan Nica. *A Brief Introduction to Spectral Graph Theory*. European Mathematical Society, first edition, 2018. doi:10.4171/188.

[Sid18] Aaron Sidford. *Lecture 10 - Spectral Graph Theory*, 2018. [Lecture notes for CME 305: Discrete Mathematics and Algorithms, Stanford University]. Accessed: May 18, 2025. URL: https://web.stanford.edu/class/cme305/Files/l10.pdf.

[Vis23] VisualMath. *What is...the Laplace matrix?*, 2023. [Video]. YouTube, April 23. Accessed: May 18, 2025. URL: https://www.youtube.com/watch?v=-mPkP3qh-DI&t=70s.

[vL07] Ulrike von Luxburg. A Tutorial on Spectral Clustering. *Statistics and Computing*, 17(4), August 2007. Accessed: May 19, 2025. URL: https://arxiv.org/pdf/0711.0189.

A Appendix

A.1 R-code for computations

R-code for computations on a Laplacian matrix L. Here the Laplacian matrix for the graph in Figure 2 is used, but if wanted, one can insert the matrix of another graph where L is defined.

```
#Figure 2: Connected graph
L <- matrix(c(2, -1, -1, 0, 0, 0, 0,
              -1, 2, 0, -1, 0, 0, 0,
              -1, 0, 2, -1, 0, 0, 0,
              0, -1, -1, 3, -1, 0, 0,
              0, 0, 0, -1, 3, -1, -1,
              0, 0, 0, 0, -1, 1, 0,
              0, 0, 0, 0, -1, 0, 1), nrow = 7, byrow = TRUE)

# Compute eigenvalues and eigenvectors of L
eigenvalues <- eigen(L)$values
eigenvectors <- eigen(L)$vectors

# Eigenvalues
print("Eigenvalues:")
print(sort(round(eigenvalues, 3)))

# Eigenvector of second smallest eigenvalue
print("Eigenvector of the second smallest eigenvalue:")
index <- length(eigenvalues) - 1
f <- eigenvectors[, index]
print(round(f, 3))
```