universität wien

MASTERARBEIT | MASTER'S THESIS

Titel | Title

Analyzing the Spectral Properties of Biological Networks

verfasst von | submitted by

Edina Marica

angestrebter akademischer Grad | in partial fulfilment of the requirements for the degree of

Master of Science (MSc)

Wien | Vienna, 2026

Studienkennzahl lt. Studienblatt | Degree programme code as it appears on the student record sheet:	UA 066 645
Studienrichtung lt. Studienblatt | Degree programme as it appears on the student record sheet:	Masterstudium Data Science
Betreut von | Supervisor:	Univ.-Prof. Dipl.-Phys. Dr. Jörg Menche
Mitbetreut von | Co-Supervisor:	Dipl.-Phys. Dr. Felix Müller

Kurzfassung

Diese Arbeit untersucht die Anwendung der spektralen Graphentheorie auf die Analyse biologischer Netzwerke, mit einem besonderen Fokus auf Protein-Protein-Interaktionsnetzwerke. Wir analysieren, wie strukturelle Eigenschaften von Graphen in den Eigenwerten und Eigenvektoren der symmetrisch normalisierten Laplace-Matrixdarstellungen der Graphen zum Ausdruck kommen und wie diese Information zur Untersuchung von Proteinkomplexen und Krankheitsmodulen genutzt werden kann. Insbesondere betrachten wir die Eigenvektor-Lokalisierung von Proteinkomplexen, die Rekonstruktion von Krankheitsmodulen aus spektraler Information sowie die graphenbasierte Clusterung krebsbezogener Erkrankungen auf Basis von Laplace-Eigenwert-Einbettungen. Die spektralen Ansätze werden im Vergleich zu State-of-the-Art-Methoden evaluiert. Numerische Experimente zeigen, dass Laplace-basierte Verfahren biologisch relevante strukturelle Eigenschaften effektiv erfassen und ein leistungsfähiges Werkzeug für die Analyse biologischer Netzwerke darstellen.

Abstract

This thesis explores the application of spectral graph theory to biological network analysis, with a focus on the human protein-protein interaction network. We investigate how structural properties of graphs are reflected in the eigenvalues and eigenvectors of their symmetric normalized Laplacian matrix representation and how this information can be used to study protein complexes and disease modules. In particular, we examine eigenvector localization of protein complexes, the reconstruction of disease modules from spectral information, and the graph-level clustering of cancer diseases based on Laplacian eigenvalue embeddings. The spectral approaches are evaluated in comparison with state-of-the-art methods. The experimental results demonstrate that Laplacian-based approaches effectively capture biologically relevant structural properties and provide a powerful tool for biological network analysis.

Contents

List of Tables

List of Figures

1 Introduction

Human biological data is inherently complex and highly diverse. Understanding the human body and how its many processes interact is a question of high interest. Despite extensive research approaching the problem from different angles, many aspects of how the body functions are still not completely understood.

A way of viewing biological data is as networks, where nodes are genes, proteins or other biological entities, and edges correspond to relationships or interactions between them. One of the most important examples is the human protein-protein interaction network, in which nodes represent proteins in the human body and edges indicate that two proteins physically interact or participate together in specific biological processes. Viewing biological data through this network perspective allows us to analyze structure, detect patterns, and uncover relationships that may not be visible otherwise.

Understanding the structure, functionality and behavior of biological networks is important for applications such as disease similarity, drug discovery and repurposing, treatment plan development, gene prioritization, protein complex prediction, as well as generally gaining a deeper insight into the mechanisms of diseases and biological processes. Network science has therefore become a widely used tool in biology and medicine. One of the most commonly used structural descriptors are the shortest path distances. They are used for computing multiple centrality measures, such as betweenness or closeness centrality, as well as for determining diameter, robustness or community structure. However, most of the biological networks have small diameters making the shortest path lengths too coarse grained. This property is called the small-world property [Mil67], and therefore, shortest path metrics alone cannot fully capture the networks' complexity.

To address this limitation, we propose adopting a spectral perspective and utilizing the Laplacian matrix of a graph [BC13]. More specifically, the symmetric normalized Laplacian matrix [Chu97], which is defined as $L = D^{-\frac{1}{2}}(D - A)D^{-\frac{1}{2}}$, where A is the adjacency matrix of the graph and D is the degree matrix. This Laplacian matrix encodes how nodes are connected and how information diffuses across the network. Its eigenvalues and eigenvectors capture both global and local structural properties at a system level [PJV13, LRH14]. Hence, instead of relying solely on direct connections, spectral methods provide a richer representation of the network's organization.

However, biological networks present several challenges. The main challenge is the size and complexity of such networks. To handle this, we seek compact representations that preserve essential structural information. The spectrum of a graph, i.e., the set of its Laplacian eigenvalues, can be viewed as such a structural fingerprint. Even large and complex networks can be summarized through their eigenvalue distribution. The

spectrum reveals whether a network has strong community structure, repeating patterns, bottlenecks, or bipartite-like characteristics. Moreover, the corresponding eigenvectors further encode structural modes and node localization.

The overarching goal of this thesis is to understand how structural information encoded in the Laplacian relates to biological function in the human protein-protein interaction network. We aim to investigate whether structural patterns reflected in the Laplacian spectrum align with known biological knowledge, and whether they can reveal previously unknown relationships. To achieve this, in the Preliminaries 3, we begin by introducing the fundamental terminology, definitions, and mathematical concepts required to understand the thesis. Next, in the Theoretical foundations 4, we present the different forms of the Laplacian matrix of a graph, argue why we focus on the symmetric normalized version, and summarize its key theoretical properties, including those of its eigenvalues and eigenvectors.

Afterwards, we analyze the Laplacian spectra of graphs with well-understood structure, namely the Erdős-Rényi, Barabási-Albert, and Watts-Strogatz random graph models. These serve as a baseline for interpreting spectral behavior.

We then discuss in the Materials and methods chapter 6 three main Laplacian-based approaches for network analysis, i.e., spectral localization and prediction of subgraphs, reconstructing graphs from their Laplacian spectra, and clustering collections of graphs based on spectral embeddings.

Then, in the Results and discussion chapter 7, we turn to real biological data and investigate the described Laplacian-based methods on the human protein-protein interaction network, its protein complexes and disease modules.

First, we study spectral localization of subgraphs, analyzing how known protein complexes localize on the eigenvectors of the protein-protein interaction network. Based on these observations, we propose a novel approach for identifying new candidate protein complexes, and compare our method to existing, widely-used approaches.

Second, we turn to disease modules, which are sets of proteins associated with specific diseases. We begin by analyzing the Laplacian spectra of diseases of different types and sizes in order to understand how their structural characteristics are reflected in their eigenvalue distributions. Instead of visualizing entire (and potentially very large) networks, we can look at their eigenvalue distribution and already infer important structural properties.

We then investigate spectral reconstruction, examining how accurately disease modules can be reconstructed from subsets of their eigenvalues and eigenvectors, in particular, from the smallest eigenvalues and/or the largest eigenvalues together with their corresponding eigenvectors. This allows us to assess our theoretical insights regarding which eigenvalues carry relevant structural information and to further understand the characteristics of different diseases. Moreover, since computing the full set of eigenpairs is computationally expensive for large networks, reconstructing the graph from only a subset of them allows us to evaluate whether focusing on selected eigenvalues is in general sufficient.

Finally, we consider graph clustering, where the goal is to partition a collection of graphs into clusters, where graphs corresponding to the same cluster are structurally

similar. For this, we introduce two spectral graph embedding approaches that generate vector representations suitable as input for classical clustering algorithms (such as kmeans clustering). These methods are applied to cluster cancer-related diseases, and the results are compared with state-of-the-art techniques. Their performance is further assessed on a synthetic dataset consisting of graphs generated with the Erdős-Rényi, Barabási-Albert and Watts-Strogatz models, as well as on another biological network. The goal of the cancer clustering is to understand how common structural characteristics of the diseases, specifically the ones encoded by the Laplacian spectrum, align with the semantic similarity, while also potentially revealing connections and relationships between seemingly unrelated diseases, which can later prove valuable in drug repurposing or in understanding disease similarities.

By viewing biological networks purely as mathematical objects and applying tools from spectral graph theory, we gain an alternative and complementary perspective on biological networks. In this thesis, we show that the symmetric normalized Laplacian matrix, together with its eigenvalues and eigenvectors, is a powerful and valuable tool for encoding structural properties. Moreover, these also align well with the semantic and biological meaning of the underlying networks, often better than alternative methods for representing graph structure.

Overall, the thesis bridges Laplacian spectral theory and the study of the human protein-protein interaction network, with the aim of uncovering meaningful insights about protein complexes and disease modules from the underlying structural properties of the networks.

2 Related literature

The study of spectral graph theory has gained significant attention due to its ability to reveal structural properties of networks through eigenvalues and eigenvectors of graph matrices. In this thesis, we are interested in the spectral theory of the normalized Laplacian matrix, which will serve as the fundamental method used throughout our work. F. R. K. Chung defined the symmetric normalized Laplacian matrix in 1997 [Chu97], and has made significant contributions to the study of the normalized Laplacian spectra and its properties, conducting most of the foundational work in this area [CLV03, CL04]. Some key properties shown by Chung include the fact that all eigenvalues of any graph lie between 0 and 2, and that normalization yields to better scalability and stability. These properties also constitute the primary reason why we chose to work with the normalized Laplacian instead of its standard version.

Building on Chung's foundational work, the normalized Laplacian has since been extensively studied due to its effectiveness in capturing global structural properties of the underlying graphs, such as community structure, clustering or connectivity.

A. Coja-Oghlan (2007) [CO07] developed properties of the normalized Laplacian spectrum of Erdős-Rényi random graphs, while R. O. Braga et al. (2015) [BDVRT15] explored the eigenvalues of tree graphs. S. Butler (2016) [But16] and R. P. Varghese et al. (2020) [VS20] analyzed the normalized Laplacian eigenvalues of specific graph structures, such as duplication or splitting of graphs, and S. Sun (2020) [SD20] studied the normalized Laplacian spectrum of complete multipartite graphs. P. Xie (2023) [XZC16] determined the effects of triangularization on the normalized Laplacian spectrum, while T. Chen (2024) [CYP24] generalized it, by exploring how the spectrum changes if each edge of a graph is replaced with an n-polygon.

All these studies highlight the growing interest in the normalized Laplacian spectrum in recent years. The findings of these papers provide valuable insights for our work, as many biological networks have similar structural characteristics as the ones studied in the mentioned theoretical papers. However, despite the extensive study of the Laplacian spectrum in mathematics and its ability to reveal global structural properties, its practical applications haven't yet been that extensively explored.

In 2008, A. Banerjee [Ban08], described "the spectrum of the Laplacian as a tool for analyzing the structure and evolution of networks". This dissertation is close in scope to our work. It analyzes the Laplacian spectra of known graphs, describes some observed properties and tries to reconstruct protein-protein interaction graphs from their spectra. A. Banerjee also attempts to classify networks across different fields based on the Laplacian eigenvalues. Moreover, a coarsening scheme is introduced as well, to reduce the size of graphs, to overcome the difficulty of computing the spectrum of large networks. While the main idea of this PhD dissertation is similar to our goals for this thesis, our work

focuses more specifically on the human protein-protein interaction network, and discusses different methods based on the normalized Laplacian.

In a follow-up work in 2009 [BJ09], A. Banerjee proposes the use of the normalized Laplacian spectrum (with slightly different definition) to uncover structural properties of biological networks. However, the paper only contains plots of random graph models and real biological networks, along with a brief description of the key properties of the Laplacian eigenvalues. Hence, it serves more as a first attempt at exploring the potential of the Laplacian spectrum for biological networks. It does not provide in-depth analysis or significant results. Later, in another paper [Ban12], A. Banerjee introduces a novel method to compute the topological distance between graphs based on the Laplacian spectrum (with a different definition of the normalized Laplacian then ours) and applies it to biological networks to identify those that are evolutionarily close.

In recent years, research on the Laplacian spectra of graphs has gained some more attention in the study of biological systems. However, these papers have a different goals than this thesis. For example, S. C. de Lange et al. (2014) [LRH14] analyzes the Laplacian spectra of neural networks, more concretely the brain of the macaque, the cat and the C. elegans. While another, slightly different definition of the normalized Laplacian is used, the paper compares the spectra of the three brain networks to the Erdős-Rényi, Barabási-Albert and Watts-Strogatz random graph models. It also explores the effect of noise by rewiring different percentages of the edges.

Another example of the study of the Laplacian spectrum in biological systems is the paper of E. Lewitus et al. (2015) [LM15], where the Laplacian spectrum is used to interpret patterns in Phylogenetic trees. U. Shashankaditya et al. (2018) [UB18] compare the weighted spectral distributions [FHU$^+$08] of the normalized Laplacian of three ecological graphs. One, more extensive study is made by S. Windels et al. (2019) [WMDP19], where the authors define a new type of Laplacian, the Graphlet Laplacian and use it to show that it captures at least as much of the biological function as other types of Laplacians. However, they do not compare to the normalized Laplacian.

This thesis focuses on three main approaches based on the normalized Laplacian matrix and its spectrum, namely eigenvector localization of protein complexes, disease module reconstruction and clustering collections of graphs, such as cancer diseases. We now mention the related papers for each approach separately.

M. Cucuringu and M. W. Mahoney (2011) [CM11] plot how the Laplacian eigenvectors of a senate and a migration graph localize, but do not show any methods that could predict denser subgraphs based on that. F. Slanina and Z. Konopasek (2010) [SK10] present a technique for finding communities based on eigenvector localization of the adjacency matrix, and test it on an Amazon network. Their approach, however, also requires randomization and the selection of an appropriate rescaling factor, which introduces additional complexity into the problem. Both of the papers use the inverse participation ratio to find the most localized eigenvectors, an idea that we also apply in this thesis.

Regarding the reconstruction of graphs, as already mentioned, A. Banerjee [Ban08] shows an approach for reconstruction of protein-protein interaction networks. Additionally, E. Maiorino et al. (2017) [MRSG17] analyzed the normalized Laplacian spectra of protein

contact networks. Their present an approach to construct graphs with similar Laplacian spectrum as a given graph. Both papers, focus on constructing graphs with a similar Laplacian spectrum, in contrast to this work, where we use Laplacian eigenvalues directly for graph reconstruction. Therefore, their objectives differ from ours.

It is important to note, however, that while the Laplacian matrix uniquely defines a graph up to isomorphism, a given Laplacian spectrum may correspond to multiple non-isomorphic graphs. Two graphs that have the same normalized Laplacian spectrum are called *cospectral graphs*. S. Butler (2012) [BG12] showed the existence of cospectral graphs and proposed a method to construct different graphs with the same normalized Laplacian spectrum.

Finally, we turn to the literature on clustering. Among the three main topics considered in this thesis, clustering has been the most extensively studied. However, this has been done in a different context, as the Laplacian matrix has been primarily used for clustering d-dimensional data points rather than sets of graphs. One of the most important papers in this context is by A. I. Ng et al. (2001) [NJW01], who proposed a spectral clustering algorithm based on the normalized Laplacian. In their approach, a similarity matrix between data points is first computed. From this matrix, the Laplacian is constructed, and the first k eigenvectors are extracted. The data points are then embedded into the vector space spanned by these eigenvectors, and k-means clustering is applied to obtain k clusters. In our work, we also use this clustering approach among others, after embedding the graphs into vector representations. There are also other spectral clustering methods based on different Laplacians, U. Luxburg (2004) [Lux04] provided an overview of the most well-known ones, including clustering based on the normalized Laplacian spectrum [NJW01].

In [NJW01], the authors mention that the same idea can be extended to cluster the nodes of a graph instead of data points. In this setting, the (normalized) Laplacian of the graph is computed, the nodes are embedded into k-dimensional vectors based on the first k eigenvectors of the Laplacian matrix, and k-means clustering is applied to partition the nodes into k clusters.

In contrast, our objective is graph-level clustering. In this context, E. Pineau (2019) [Pin19] introduced a graph classification approach based on embedding graphs into k-dimensional vectors using their Laplacian eigenvalues. The main differences between their work and ours are that we aim to solve an unsupervised clustering problem rather than a supervised classification task. Moreover, to handle embeddings of inconsistent lengths, Pineau proposes padding the embeddings of smaller graphs with zeros. Instead, we bin the spectrum and use the counts within each bin as features, to avoid the misleading structural information introduced by the zeros. Although this binning strategy is mentioned in [Pin19] as well, it is not described in detail, nor included in their experiments. Additionally, we also experiment with using only the smallest or largest eigenvalues, rather than the full spectrum or only the largest eigenvalues as considered in their work.

3 Preliminaries

In this chapter, we cover some basic definitions, theorems, and terminology, essential for the introduction and understanding of the work done in the thesis.

3.1 Graph theory

Definition 3.1.1 (Undirected simple graph)**.** *An* undirected simple graph *is a pair* $G = (V, E)$*, where* V *is a finite set of* vertices *and* E *is a set of unordered pairs of the vertices (*edges*).*

Unless stated otherwise, the term *graph* refers to an undirected graph. We use the terms *node* and *vertex* interchangeably. Throughout this thesis, we also use the term *network* to refer to real-world systems that can be represented by a graph structure.

By definition, self-loops, i.e., edges that connect a vertex to itself are not allowed in simple graphs. We call graphs that contain such edges *simple graphs with self-loops*.

If the edges of the graph are ordered pairs, we call the graph *directed*. If a weight function $w : E \to \mathbb{R}$ is assigned to the edges, we call the graph *weighted*.

Given a graph G with n vertices and m edges, the degree d_v of a vertex v is the number of vertices it is connected to. It is trivial to see that for any graph, the sum of the degrees equals twice the number of edges, i.e., $\sum_{i=1}^{n} d_i = 2m$.

We call a graph $G = (V, E)$ *connected* if for each pair of vertices $\{u, v\}$ with $u, v \in V$, there exists a path from u to v.

A graph $G = (V, E)$ is *complete* if, for every pair of vertices $\{u, v\}$ with $u, v \in V$, there is an edge $e \in E$ connecting them.

Definition 3.1.2 (Path)**.** *A* path $P = (v_1, v_2, \ldots, v_k)$ *of graph* $G = (V, E)$ *is a finite sequence of vertices with* $v_i \in V, \forall i = \{1, \ldots, k\}$ *and the property that each pair of consecutive vertices is joined by an edge* $e \in E$*.*

A path $P = (r_1, r_2, \ldots, r_k)$ of graph G is *simple* if all vertices of the path $v_i \in P, i = [1, \ldots k]$ are distinct. A *cycle* $C = (c_1, c_2, \ldots, c_k)$ of graph G is a path, where $c_1 = c_k$. A cycle $C = (c_1, c_2, \ldots, c_k)$ of graph G is *simple* if its vertices are distinct, except c_1 and c_k. A graph is *acyclic* if it does not contain any cycles.

Now, we define some graphs with special structure.

- A *tree* is a connected acyclic graph.
- A *forest* is a disjoint union of trees.

- A *star graph* with n vertices is a tree with one non-leaf center vertex and $n-1$ leaves, where a vertex is called *leaf* if its degree is 1.
- A graph $G = (V, E)$ is *bipartite* if its vertex set V can be partitioned into two subsets V_1 and V_2 with $V_1 \cup V_2 = V$,$V_1 \cap V_2 = \emptyset$ and no edges OF G connect vertices from the same subset. We denote by $G_{k,p}$ a bipartite graph with $|V_1| = k$ and $|V_2| = p$ nodes.

We call two graphs *isomorphic* if there is a one-to-one correspondence between their nodes such that adjacency is preserved, i.e., if one graph can be obtained from the other one by relabeling its nodes, without changing which pairs of nodes are connected by edges.

3.2 Linear algebra

In order to work on graphs, it is convenient to represent them in a matrix form.

Definition 3.2.1 (Degree matrix)**.** *The degree matrix representation of a graph with n vertices is a matrix $D \in \mathbb{N}_{\geq 0}^{n \times n}$, where*

$$D_{i,j} = \begin{cases} d_{v_i}, & \text{if } i = j \\ 0, & \text{otherwise.} \end{cases} \tag{3.1}$$

A matrix is called a *square matrix* if the number of its rows is the same as the number of its columns. The degree matrix is square and is *diagonal*, as all of its non-diagonal values are zero. Another example of a diagonal matrix is the *identity matrix* I, with 1 as its diagonal elements, and 0 for every other entry.

Definition 3.2.2 (Adjacency matrix)**.** *The adjacency matrix representation of a graph with n vertices is a matrix $A \in \{0,1\}^{n \times n}$, where*

$$A_{i,j} = \begin{cases} 1, & \text{if there is an edge between vertex } v_i \text{ and vertex } v_j \\ 0, & \text{otherwise.} \end{cases} \tag{3.2}$$

We now introduce some fundamental concepts from linear algebra, starting with eigenvalues and eigenvectors, which are the central components of this thesis.

Definition 3.2.3 (Eigenvalue, eigenvector)**.** *The number λ is an* eigenvalue *of a matrix M if there exists a nonzero vector v for which $Mv = \lambda v$. v is an* eigenvector *of the matrix M.*

The eigenvalues of M are the roots of the characteristic polynomial

$$p(\lambda) = det(M - \lambda I),$$

obtained by solving the characteristic equation

$$p(\lambda) = 0.$$

Here, I denotes the identity matrix of same size as M. Once the eigenvalues are calculated, the eigenvectors can be computed by solving the

$$Mv = \lambda v$$

equation for each eigenvalue. The set of all vectors that are solutions to this equation forms the eigenspace associated with eigenvalue λ.

If the matrix M is the representation of a graph G, then we refer to the eigenvalues and eigenvectors as the eigenvalues and eigenvectors of G. Note that, for any graph, its adjacency matrix representation uniquely determines the graph up to isomorphism, and hence, also uniquely determines its set of eigenpairs. However, two non-isomorphic graphs may have the same set of eigenvalues (i.e., they may be cospectral).

A vector set is *linearly independent* if no vector of the set can be written as the linear combination of the other vectors of the set. A *vector basis* of a vector space V is a set of vectors $(v_1, \ldots, v_n)$ that are linearly independent and span V, i.e. every vector $v \in V$ can be uniquely written as a linear combination of the vectros $v_1, \ldots, v_n$. An *orthonormal basis* is a vector basis with orthonormal vectors, i.e. a basis with unit vectors, orthogonal to each other. Here, we call a vector *unit vector*, if its length is equal to 1.

Going back to the eigenvalues and eigenvectors, since the same λ can be an eigenvalue of M corresponding to more than one linearly independent eigenvector v, the eigenvalues of a matrix form a multiset. Moreover, we call the number of times λ appears as a root of the characteristic polynomial of M the *algebraic multiplicity* of λ. On the other hand, the number of linearly independent eigenvectors associated with eigenvalue λ is called the *geometric multiplicity* of λ.

The *spectrum* of a matrix is the multiset of its eigenvalues, while the *spectral gap* of a matrix is the absolute difference of its first two smallest eigenvalues.

Definition 3.2.4 (Spectral decomposition)**.** *The* spectral decomposition *of a matrix $M \in \mathbb{R}^{n \times n}$ is given by $M = Q\Lambda Q^{-1}$, where $Q \in \mathbb{R}^{n \times n}$ is the matrix whose i-th column is M's i-th eigenvector and $\Lambda \in \mathbb{R}^{n \times n}$ is a diagonal matrix D with $D_{i,i} = \lambda_i$.*

The spectral decomposition of a matrix can be derived from the formula of calculating the eigenvalues and eigenvectors introduced in Definition 3.2.3.

Now, we define some special matrices.

Definition 3.2.5.

- *The* transpose *of a matrix $M \in \mathbb{R}^{n \times m}$ is the matrix M^T, which is obtained by flipping the matrix over its diagonal. $M^T_{ij} = M_{ji}, \forall i \in [m], \forall j \in [n]$.*
- *A matrix M is* symmetric *if it is a square matrix, and it is equal to its transpose, i.e. $M = M^T$.*
- *A matrix $M \in \mathbb{R}^{n \times n}$ is* positive semidefinite (PSD) *if $z^T M z \geq 0, \forall z \in \mathbb{R}^n$.*

Definition 3.2.6 (Rank of matrix)**.** *The rank $r(M)$ of a matrix A is the maximum number of linearly independent columns of M.*

Next, we prove a lemma needed for an important theorem's proof in the next chapter.

Lemma 3.2.1. *The multiset of eigenvalues of a block-diagonal matrix is equal to the union of the eigenvalues of the diagonal blocks.*

A matrix is called *block-diagonal* if it is a square matrix and it contains square blocks along its diagonal, and zeros everywhere else.

Proof. Let the block-diagonal matrix M have k blocks of sizes $n_1, n_2, , n_k$. Then,

$$M = \begin{pmatrix} M_1 & 0 & \dots & 0 \\ 0 & M_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & M_k \end{pmatrix} = \tag{3.3}$$

$$= \begin{pmatrix} M_1 & 0 & \dots & 0 \\ 0 & I_{n_2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & I_{n_k} \end{pmatrix} \cdot \begin{pmatrix} I_{n_1} & 0 & \dots & 0 \\ 0 & M_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & I_{n_k} \end{pmatrix} \cdot \dots \cdot \begin{pmatrix} I_{n_1} & 0 & \dots & 0 \\ 0 & I_{n_2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & M_n \end{pmatrix}.$$

Since

$$\det \begin{pmatrix} I_{n_1} & 0 & \dots & & 0 \\ 0 & \ddots & & & 0 \\ \vdots & & M_j & & \vdots \\ & & & \ddots & 0 \\ 0 & 0 & \dots & 0 & I_{n_k} \end{pmatrix} = \det(M_j),$$

and $\det(AB) = \det(A) \cdot \det(B)$ for any two matrices, we can write

$$\det(M) = \det(M_1) \cdot \det(M_2) \cdot \dots \cdot \det(M_k).$$

Hence, the characteristic polynomial $\det(M - \lambda I)$ can be written as

$$\det(M - \lambda I) = \det(M_1 - \lambda I) \cdot \det(M_2 - \lambda I) \cdot \dots \cdot \det(M_k - \lambda I),$$

and the lemma directly follows. □

3.3 Random graph models

A random graph is a graph that is generated randomly according to some probability distribution or random process. There are several models according to which random graphs can be generated, the most popular being the Erdős-Rényi model. In this section, we present three different random graph models. Since these random graph models and their properties have been widely studied, we use them later in the thesis as a baseline for understanding the examined problems better before turning to real-world networks.

Erdős-Rényi (ER) model

The Erdős-Rényi (ER) model [ER59] takes two parameters: n, the number of vertices and $0 \leq p \leq 1$, the probability of including each edge to the graph. Based on these parameters, a random n-node graph is constructed, and for each pair $\{u, v\}$ of vertices, an edge connecting the them is added with probability p, independently of other edges. The average number of edges of an ER graph is $\binom{n}{2}p$.

Barabási-Albert (BA) model

Barabási and Albert [BA99] observed that many real-world networks are different from the ER random graphs in the way nodes connect to each other. In the ER model, one node connects to each of the other ones with the same probability, while in real life, there tends to be a rule of *preferential attachment*, meaning that a node is more likely to connect to a node which already has many connections (higher degree). This is true for biological networks, as well as internet or social networks [DM03]. Moreover, the ER model requires a fixed number n of nodes, while in real-world applications, the number of nodes increases over time.
In order to capture the growth of a network and the preferential attachment, Barabási and Albert came up with a novel scale-free model, meaning that the degree distribution follows a *power-law*. There are some high degree nodes called hubs and the rest of the nodes have low degree, connecting mostly to these hubs. Nodes are added iteratively to the graph and whenever a new node is added, it connects to an existing node with a probability proportional to the existing node's degree. Formally, let $G = (V, E)$ be a graph with $|E| = m_0$ edges and $d_v \geq 1, \forall v \in V$. Let $m \geq 1$ be a fixed parameter. In every step a new vertex with m edges is added to the graph, connecting the new node to m existing ones by the following preferential attachment rule: $\forall v \in V$, the probability $p(v)$ that the new node connects to node v is equal to

$$p(v) = \frac{d_v}{\sum_{u \in V} d_u}. \tag{3.4}$$

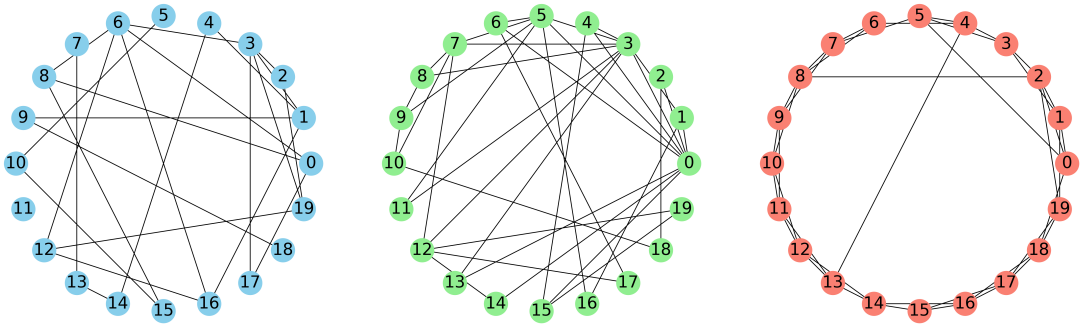
This means that the new node will more likely connect to an existing node with an already higher degree than to an existing node with lower degree. Due to the hubs, the resulting graphs have *small-world property* [Mil67], meaning that the average shortest path distance between the node pairs is low, while the clustering coefficient is relatively large. The clustering coefficient of an undirected graph G is a number between 0 and 1 and is defined as three times the number of triangles in the graph divided by the number of connected node triplets. It measures how well the nodes of a graph cluster together.

Watts–Strogatz (WS) model

A third, very popular random graph model is the Watts-Strogatz (WS) model [WS98]. Similarly to BA graphs, these graphs also exhibit small-world property, however, for a different reason. Given the number of nodes $|V| = n$, a degree $\ln(n) \leq k \leq n$ and a

rewiring probability $0 \leq p \leq 1$, the model constructs an undirected graph by ordering the nodes in a ring, connecting each of the nodes to $k/2$ neighbors on both sides of the node (node v_i is connected to nodes v_j for $i - k/2 \pmod{n} < j < i + k/2 \pmod{n}, \quad j \neq i$), and rewiring each of the edges with a probability p by choosing another endpoint uniformly at random from the nodes, except itself.

While the constructed graphs would be highly regular, the rewired edges act as shortcuts, shortening the average path length significantly. While increasing p, the graph becomes more random, the structure still remains more regular than an ER graph, since all the nodes have at least $k/2$ edges. Although this model is not as realistic as the BA model for instance, we still include it in our analysis with the goal to uncover how its structural characteristics are shown in the spectrum. Figure 3.1 shows an Erdős-Rényi, a Barabási-Albert and a Watts-Strogatz graph, each with 20-20 nodes ordered in a ring for better comparability. It is clearly visible that the ER model produces a more homogeneous graph, while the BA graph contains some hub nodes, namely node 0 and 3, which are indeed nodes added among the earliest ones. The WS graph shows the highly regular structure described previously, with only a few rewired edges.



(a) ER graph (b) BA graph (c) WS graph

Figure 3.1: Example of an Erdős-Rényi graph with $p = 0.2$, a Barabási-Albert graph with $m = 2$, and a Watts-Strogatz graph with $k = 4$ and $p = 0.1$

4 Theoretical foundations

After introducing the basic definitions and terminology in the previous chapter, we now establish the mathematical framework that the thesis builds upon.

4.1 The Laplacian matrix

In Chapter 3, we have already introduced the notion of adjacency matrices, which are among the most well-known tools for representing graph connectivity, carrying information on how vertices are connected to each other. Although widely used to understand the underlying graph's properties, they are not the only type of connectivity matrices that carry meaningful information about the structure of graphs. Another important matrix that plays a central role in graph theory is the Laplacian matrix, which will be the key focus of this thesis.

The Laplacian matrix of a simple undirected graph $G = (V, E)$ with $|V| = n$ is a matrix $L \in \mathbb{Z}^{n \times n}$, with

$$L_{i,j} = \begin{cases} d_{v_i}, & \text{if } i = j \\ -1, & \text{if } i \neq j \text{ and } \{v_i, v_j\} \in E \\ 0, & \text{otherwise} \end{cases} \tag{4.1}$$

Equivalently,

$$L = D - A, \tag{4.2}$$

where D is the degree matrix and A is the adjacency matrix of G.
By definition, the Laplacian matrix is symmetric.

Lemma 4.1.1. *L is a positive semidefinite (PSD) matrix.*

Proof. We need to prove that $x^T L x \geq 0, \forall x \in \mathbb{R}^n$.
$L = D - A$, therefore we need $x^T(D - A)x \geq 0, \forall x \in \mathbb{R}^n$.

$$x^T D x = \sum_{i=1}^{n} D_{i,i} x_i^2 = \sum_{i=1}^{n} d_{v_i} x_i^2. \tag{4.3}$$

$$x^T A x = \sum_{i,j=1}^{n} A_{i,j} x_i x_j = \sum_{\{v_i, v_j\} \in E} x_i x_j. \tag{4.4}$$

Summing up the two terms, we get

$$\begin{aligned} x^T L x = x^T (D - A)x = \sum_{i=1}^{n} d_{v_i} x_i^2 - \sum_{i,j:\{v_i,v_j\}\in E} x_i x_j = \sum_{i=1}^{n} (d_{v_i} x_i^2 - \sum_{j:\{v_i,v_j\}\in E} x_i x_j) \\ = \frac{1}{2} \sum_{i=1}^{n} \sum_{j:\{v_i,v_j\}\in E} (x_i - x_j)^2 = \sum_{i,j:\{v_i,v_j\}\in E} (x_i - x_j)^2 \geq 0. \end{aligned} \tag{4.5}$$

□

4.1.1 Spectral decomposition of the Laplacian matrix

To extract meaningful information about the underlying graph from its Laplacian matrix, we need to analyze the matrix's spectrum. The distribution of the eigenvalues encodes insights about the graph's structure. However, the spectrum of the Laplacian defined above in Equation 4.1 is highly sensitive to the size of the graph. High degree nodes show in the Laplacian matrix as big diagonal entries, yielding the matrix properties being dominated by these high values. This makes it difficult to interpret and compare spectra across graphs of different sizes or with uneven degree distributions. To address this issue, in the following subsection we define the symmetric normalized Laplacian matrix of a graph, and argue why its spectra is more useful in uncovering structural properties.

4.2 The symmetric normalized Laplacian matrix

Normalization of the Laplacian matrix ensures that the influence of high degree nodes is reduced, and that the spectrum becomes more predictable and less sensitive to the degree distribution, yielding better scalability and stability. We obtain the normalized Laplacian matrix by by dividing the entries of the standard Laplacian matrix by the square root of the node degrees, leading to the following definition of the symmetric normalized Laplacian matrix of a simple undirected graph G:

$$L_{i,j}^{symm} = \begin{cases} 1, & \text{if } i = j \text{ and } d_{v_i} \neq 0 \\ -\frac{1}{\sqrt{d_{v_i} d_{v_j}}}, & \text{if } i \neq j \text{ and } \{v_i, v_j\} \in E \\ 0, & \text{otherwise} \end{cases} \tag{4.6}$$

Equivalently,

$$L^{symm} = D^{-\frac{1}{2}} L D^{-\frac{1}{2}}, \text{or } L^{symm} = I - D^{-\frac{1}{2}} A D^{-\frac{1}{2}} \tag{4.7}$$

where D is the degree matrix, L is the unnormalized Laplacian matrix of G defined in Equation 4.1, and A is the adjacency matrix of G.
We note that there exist left and right normalized versions of the Laplacian matrix as well.

$$L^{left} = D^{-1} L, \tag{4.8}$$

and

$$L^{right} = LD^{-1}. \tag{4.9}$$

However, we work with the symmetric normalized form defined in Equation 4.7, as it has additional desirable properties over the other two normalized forms. These include symmetry, positive semidefiniteness and easier, faster mathematical computations. Moreover, all the eigenvalues of the symmetric normalized Laplacian matrix lie between 0 and 2, which makes comparison between graphs significantly easier regardless of size and connectivity patterns.

Theorem 4.2.1. *L^{symm} is a positive semidefinite (PSD) matrix.*

Proof. By definition $L^{symm} = D^{-\frac{1}{2}}LD^{-\frac{1}{2}}$. We need to show that $x^T L^{symm} x \geq 0, \forall x \in \mathbb{R}^n$. Let $z = D^{-\frac{1}{2}}x$. Then, by the symmetry of the diagonal matrix $D^{-\frac{1}{2}}$, $z^T = x^T D^{-\frac{1}{2}}$, and thus, we can write

$$x^T L^{symm} x = x^T D^{-\frac{1}{2}} L D^{-\frac{1}{2}} x = z^T L z \geq 0, \tag{4.10}$$

where the last inequality is given by Lemma 4.1.1. □

4.2.1 Spectral decomposition of the symmetric normalized Laplacian matrix

Now, we study the spectral decomposition of the symmetric normalized Laplacian. In the rest of the thesis, we will work with this Laplacian matrix form. Thus, from this point forward, the terms Laplacian matrix, Laplacian spectra and the notation L refers specifically to the symmetric normalized Laplacian matrix and its spectra, unless stated otherwise. Similarly, unless stated otherwise, a graph G refers to a simple undirected graph. Additionally, when referring to the eigenvalues of a matrix, we use the notation λ_i to refer to the $i+1$-st smallest eigenvalue.
Previously, we showed that L is symmetric and PSD, which implies that all of its eigenvalues are real and non-negative. However, there are several additional properties of the Laplacian eigenvalues that are well-known. In the following, we present and prove these further characteristics, which we later use to gain deeper insights into the structural properties of biological networks.

Properties of the Laplacian eigenvalues

One of the key properties of the Laplacian eigenvalues that makes the comparison of the spectra of two graphs easier, is presented in the following theorem, with proof inspired from [Chu97].

Theorem 4.2.2. *The Laplacian eigenvalues of any graph lie between 0 and 2.*

Proof. The lower bound trivially follows from the positive semidefiniteness of L. To prove the upper bound of 2, we introduce the Rayleigh quotinent.

Definition 4.2.1. *The Rayleigh quotient of a given symmetric matrix $A \in \mathbb{R}^{n\times n}$ and a nonzero vector $x \in \mathbb{R}^n$ is a multivariate function defined as*

$$R(A, x) = \frac{x^T Ax}{x^T x}. \tag{4.11}$$

In the next step of the proof we will show that the maximum value of the Rayleigh quotient for a given symmetric matrix A is the largest eigenvalue λ_{n-1} of A, attained by the corresponding eigenvector v_{n-1}.

Theorem 4.2.3. *For a given symmetric matrix A,*

$$\max_{x\in\mathbb{R}^n:x\neq 0} \frac{x^T Ax}{x^T x} = \lambda_{n-1}, \quad \textit{and} \quad \arg\max_{x\in\mathbb{R}^n:x\neq 0} \frac{x^T Ax}{x^T x} = v_{n-1}, \tag{4.12}$$

where $\lambda_0 \leq \lambda_1 \leq ... \leq \lambda_{n-1}$ are the eigenvalues of A, and v_{n-1} is the eigenvector corresponding to λ_{n-1}.

Proof. First, we note that trivially, the Rayleigh quotient is scale invariant, i.e.,

$$R(A, cx) = \frac{(cx)^T A(cx)}{(cx)^T(cx)} = \frac{x^T Ax}{x^T x} = R(A, x). \tag{4.13}$$

This means that it can be assumed w.l.o.g. that $||x||_2^2 = 1$.
Let $A = Q\Lambda Q^{-1}$ be the spectral decomposition of A. Then, for $x \in \mathbb{R}^n : ||x||_2^2 = 1$,

$$x^T Ax = x^T Q\Lambda Q^{-1}x = (x^T Q)\Lambda(Q^{-1}x) = y^T \Lambda y, \tag{4.14}$$

where $y = Q^{-1}x$ and $||y||_2^2 = x^T(Q^{-1})^T Q^{-1}x = x^T x = 1$. Now, the expression becomes

$$\max_{y\in\mathbb{R}^n:y\neq 0,||y||_2^2=1} y^T \Lambda y = \max_{y\in\mathbb{R}^n:y\neq 0,||y||_2^2=1} \sum_{i=1}^{n} \lambda_i y_i^2 = \lambda_{n-1}, \tag{4.15}$$

where the last equation is attained with $y = e_n$. Thus, the vector x that maximizes the expression in Equation 4.12 is $x = Qy = Qe_n = v_{n-1}$. □

Now, all that remains to show to complete the proof of Theorem 4.2.2 is that

$$R(L, x) \leq 2. \tag{4.16}$$

To prove this, we rewrite the formula of $R(L, x)$ w.r.t. the unnormalized Laplacian. We will use the notation $\mathcal{L}$ for the unnormalized Laplacian matrix. Recall from Equation 4.7 that $L = D^{-\frac{1}{2}}\mathcal{L}D^{-\frac{1}{2}}$. Then,

$$\begin{aligned} R(L, x) = \frac{x^T Lx}{x^T x} &= \frac{x^T D^{-\frac{1}{2}}\mathcal{L}D^{-\frac{1}{2}}x}{x^T x} = \frac{f^T \mathcal{L} f}{f^T (D^{\frac{1}{2}})^T D^{\frac{1}{2}} f} \\ &= \frac{\sum_{i,j:\{v_i,v_j\}\in E}(f_i - f_j)^2}{\sum_{i=1}^n f_i^2 d_{v_i}} \leq \frac{\sum_{i,j:\{v_i,v_j\}\in E} 2(f_i^2 + f_j^2)}{\sum_{i=1}^n f_i^2 d_{v_i}} \\ &= \frac{2\sum_{i=1}^n f_i^2 d_{v_i}}{\sum_{i=1}^n f_i^2 d_{v_i}} = 2. \end{aligned} \tag{4.17}$$

where $f = D^{-\frac{1}{2}}g$, thus $g = D^{\frac{1}{2}}f$ and the first equality in the second row follows from Equation 4.5.
Putting everything together, we get that

$$\lambda_0 \leq \lambda_1 \leq ... \leq \lambda_{n-1} = \max_{x \in \mathbb{R}^n : x \neq 0} R(L, x) \leq 2, \tag{4.18}$$

which concludes the proof.

□

Figure 4.1b shows the spectral plot of different matrix representations of the 20-node graph from Figure 4.1a. Namely, we plot the eigenvalues computed from the adjacency matrix, degree matrix, standard Laplacian matrix, as well as the symmetric-, left- and right-normalized Laplacian matrices. The plot shows the symmetric and bounded properties of the spectrum of the symmetric normalized Laplacian compared to the other matrix representations.

Theorem 4.2.4. *The Laplacian spectrum of a graph is the union of the Laplacian spectra of its connected components.*

Proof. It is clear to see that given a graph G with k connected components $G_1, ..., G_k$, the Laplacian matrix $L(G)$ can be written as a block-diagonal matrix, where the blocks are the Laplacian matrices of the connected components.

$$L(G) = \begin{pmatrix} L(G_1) & 0 & \cdots & 0 \\ 0 & L(G_2) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & L(G_k) \end{pmatrix}. \tag{4.19}$$

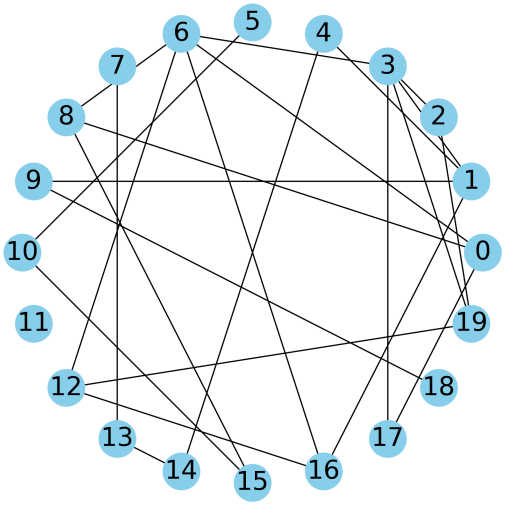
By Lemma 3.2.1, it trivially follows that the spectrum of $L(G)$ is equal to the union of the spectra of $L(G_1), ..., L(G_k)$. □

Theorem 4.2.5. *If graph G is connected, then 0 is an eigenvalue of the Laplacian.*

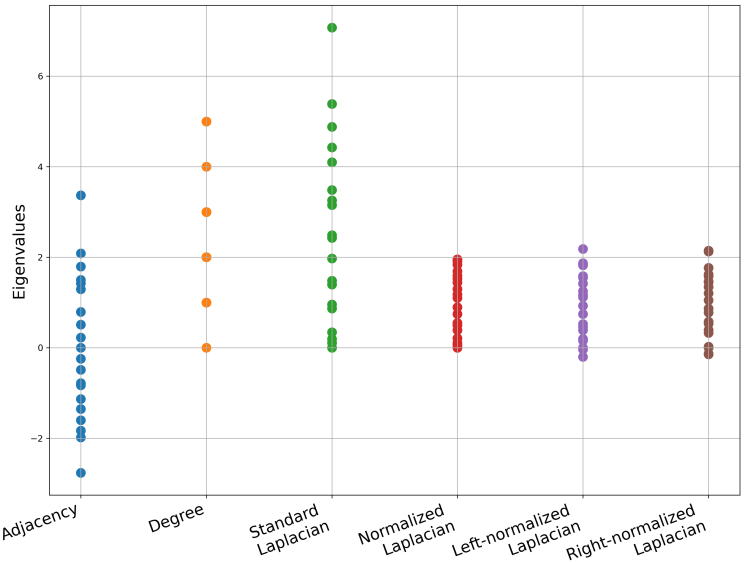
Proof. To prove the statement, we fist need to show that 0 is an eigenvalue of the unnormalized Laplacian of G.
If G is connected, by definition of the unnormalized Laplacian $\mathcal{L}$, the sum of the entries of each row/column of $\mathcal{L}$ will be $\mathcal{L}_{i,\cdot} = \mathcal{L}_{\cdot,i} = d_{v_i} - \sum_{j:\{v_i,v_j\}\in E} 1 = 0$. Terefore, for any constant vector $\mathbf{c} \in \mathbb{R}^n$, $\mathcal{L}\mathbf{c} = \mathbf{0}$, thus 0 is an eigenvalue of $\mathcal{L}$.
Now, we need to prove that 0 is also an eigenvalue of the normalized Laplacian matrix L. For this, we take the vector $\mathbf{x} = D^{\frac{1}{2}}\mathbf{c}$, where $\mathbf{c}$ is an eigenvector of $\mathcal{L}$ corresponding to the 0 eigenvector. Then, we have

$$L\mathbf{x} = LD^{-\frac{1}{2}}\mathbf{c} = D^{-\frac{1}{2}}\mathcal{L}D^{-\frac{1}{2}}D^{\frac{1}{2}}\mathbf{c} = D^{-\frac{1}{2}}\mathcal{L}\mathbf{c} = 0, \tag{4.20}$$

because $\mathcal{L}\mathbf{c} = 0$. Hence, we proved that 0 is also an eigenvalue of L. □



(a) 20-node graph



(b) Spectrum of different matrix representations of the 20-node graph

We now formulate a stronger statement, adapted from [Chu97], that follows from Theorems 4.2.4 and 4.2.5.

Theorem 4.2.6. *The following two statements are equivalent:*

1. $\lambda_i = 0$ *and* $\lambda_{i+1} \neq 0$.
2. G *has exactly* $i + 1$ *connected components.*

From Theorem 4.2.6, it results that if a graph G is connected, then $\lambda_1 > 0$.

We now present a few more important properties of the Laplacian spectrum, stated without proofs, since these can be found in [Chu97].

Theorem 4.2.7. *For a graph with* n *vertices:* $\sum_i \lambda_i \leq n$. *Equality holds if and only if the graph doesn't have isolated vertices.*

Theorem 4.2.8. *For a graph which is not complete,* $\lambda_1 \leq 1$.

Theorem 4.2.9. *For* $n \geq 2$, $\lambda_1 \leq \frac{n}{n-1}$. *Equality holds if the graph is complete.*

Theorem 4.2.10. *For a connected graph* G *with diameter* D: $\lambda_1 \geq \frac{1}{DvolG}$, *where the diameter is the maximum distance between any two vertices of* G *and* $volG = \sum_{i \in \{1,\ldots,n\}} d_i$, *the sum of the degrees.*

Theorem 4.2.11. *Let* G *be a graph with diameter* $D \geq 4$ *and let* k *denote the maximum degree of* G. *Then* $\lambda_1 \leq 1 - 2\frac{\sqrt{k-1}}{k}(1 - \frac{2}{D}) + \frac{2}{D}$.

Remark. λ_1 *tells us how difficult it is to cut up the graph into two disjoint components. Specifically, the smaller* λ_1 *is, the easier it is to partition the graph into two disjoint components. More explicitly, it follows from Theorem 4.2.6 that if* $\lambda_1 = 0$, *then* G *has at least two connected components, thus it can be easily cut into two disjoint graphs. On the other hand, from Theorem 4.2.9, it follows that for a complete graph that is maximally connected, making it resistant to partitioning,* λ_1 *attains its upper bound:* $\lambda_1 = \frac{n}{n-1}$.

Theorem 4.2.12. *For a graph without isolated vertices,* $\lambda_{n-1} \geq \frac{n}{n-1}$

Theorem 4.2.13. $\lambda_{n-1} = 2$ *if and only if a connected component of* G *is bipartite and nontrivial (i.e., consisting of one node and no edges).*

Theorem 4.2.14. *The graph* G *is bipartite if and only if for each* λ_i, *the value* $2 - \lambda_i$ *is also an eigenvalue of* G.

Another important eigenvalue is $\lambda = 1.5$, which corresponds to triangle motifs [Ban08]. Since all degrees are 2, the Laplacian matrix of the triangle graph K_3, is

$$L(K_3) = \begin{pmatrix} 1 & -1/2 & -1/2 \\ -1/2 & 1 & -1/2 \\ -1/2 & -1/2 & 1 \end{pmatrix},$$

with eigenvalues 0 (with multiplicity 1) and 1.5 (with multiplicity 2). A corresponding eigenvector to eigenvalue 1.5 is $\mathbf{v} = (0, 1, -1)^T$. Now, when a triangle is attached to a graph G by one of its nodes, 1.5 becomes an eigenvalue of the graph $(G \cup K_3)$ as well, with corresponding the eigenvector obtained from $\mathbf{v}$ by padding it with zeros. Hence, eigenvalues of 1.5 indicate the presence of local triangular structures in the graph.

Theorem 4.2.15. *If a graph with n nodes is t edges away from being a complete graph K_n, then $\frac{n}{n-1}$ will be an eigenvalue with multiplicity at least $n - 2t - 1$.*

Theorem 4.2.16. *If G is a graph of $n+m$ vertices and is obtained from $K_{n,m}$ by deleting at most t edges, then 1 will be an eigenvalue of G with multiplicity at least $m+n-2(t+1)$.*

The Laplacian eigenvalues of some elementary graphs

Studying the Laplacian spectrum of certain fundamental graphs can provide insights into the structure of general graphs and help identify whether biological networks exhibit characteristics of these special types.

Theorem 4.2.17. *The eigenvalues of the complete graph K_n are 0 with multiplicity 1 and $\frac{n}{n-1}$ with multiplicity $n - 1$.*

Proof. From Theorem 4.2.5, it is clear that 0 is an eigenvalue with multiplicity 1 and the corresponding eigenvectors are of form $D^{\frac{1}{2}}\mathbf{c}$, where $\mathbf{c} \in \mathbb{R}^n$ is any constant vector. In the case of a complete graph, all the diagonal entries of $D^{\frac{1}{2}}$ are the same, thus the eigenvectors of form $D^{\frac{1}{2}}\mathbf{c} = \mathbf{c}'$ are also constant. To show that $\frac{n}{n-1}$ is an eigenvalue with multiplicity $n - 1$, we use use the fact that the eigenvectors of a symmetric matrix are orthogonal. L is symmetric, thus, this implies that every other eigenvector $\mathbf{y}$ corresponding to one of the other eigenvalues, has to be perpendicular to the constant vector,

$$\mathbf{yc} = [y_1, y_2, ..., y_n][c', c', ..., c']^T = c' \sum_{i=1}^{n} y_i = 0, \tag{4.21}$$

which implies $\sum_{i=1}^{n} y_i = 0$. Now, looking at the formula $Ly = \lambda y$, by definition of the Laplacian matrix, the i-th row of Ly is

$$L_i y = y_i - \frac{1}{n-1} \sum_{j \neq i} y_j = \frac{n}{n-1} y_i - \frac{1}{n-1} \sum_{j=1}^{n} y_j = \frac{n}{n-1} y_i, \tag{4.22}$$

thus $Ly = \frac{n}{n-1} y$, meaning that the only nonzero eigenvalue of L is $\lambda = \frac{n}{n-1}$. □

Similarly to the complete graph, one can also prove the following theorem:

Theorem 4.2.18. *The eigenvalues of*

- *the complete bipartite graph* $K_{n,m}$ *are* 0, 1 *with multiplicity* $n+m-2$ *and* 2.
- *the star* S_n *are* 0, 1 *with multiplicity* $n-2$ *and* 2.
- *the path* P_n *are* $1-\cos\frac{\pi k}{n-1}$ *for* $k=0,1,...,n-1$.
- *the cycle* C_n *are* $1-\cos\frac{2\pi k}{n}$ *for* $k=0,1,...,n-1$.
- *the n-cube* Q_n *on* 2^n *vertices are* $\frac{2k}{n}$ *with multiplicity* $\binom{n}{k}$ *for* $k=0,1,...,n$.

4.2.2 Analysis of the regimes of the Laplacian spectrum

After studying the Laplacian eigenvalues of different graphs and showing some theorems about specific eigenvalues and their occurrence, in this section, we analyze the different regimes of the eigenvalue spectrum and what they generally encode about the underlying graph's structure.

We have already showed in Theorem 4.2.2, that all the eigenvalues lie between 0 and 2. Different regimes of this spectrum encode specific structural information about the graph. We now focus on three significant regimes, namely near 0, at the end of the spectrum close to 2, and the middle region around 1. As the previous theorems and discussion have already pointed out, the eigenvalues close or equal to these three numbers are the ones describing the most structural information of the underlying graph.

Just as 0 is an eigenvalue as many times as the number of connected components of the graph, the eigenvalues close to 0 encode the global connectivity of the graph. This region captures large-scale structural features such as connectivity structure and overall separability. Low first eigenvalues correspond to highly cohesive structures and well-separated subgraphs, implying slow diffusion speeds. In the corresponding eigenvectors, neighboring nodes have similar values. The larger the small eigenvalues are, the better connected the graph is, with not so pronounced modules. Conversely, when many eigenvalues accumulate around 0, this usually reflects fragmentation or less coherent global organization.

On the other hand, eigenvalues near the end of the spectrum, i.e., close to 2, are associated with local or global bipartite-like structures as already mentioned earlier, where neighboring nodes strongly disagree in the corresponding eigenvectors. A connected graph has an eigenvalue 2 if and only if it is bipartite, and otherwise, if the spectrum is symmetric, the closer the last eigenvalue is to 2, the closer the graph is to being bipartite as well, or having bipartite-like modules. In scale-free networks, such strongly separated patterns often correspond to local structural characteristics, such as leaf nodes and other sparse, low-degree subgraphs attached to hub nodes.

Finally, eigenvalues concentrated around 1 constitute the spectral bulk. The presence of symmetries, such as the duplication of a node and all of its edges show in this region.

Specifically, each such copy produces an eigenvalue 1. Hence, this regime reflects local structural regularities, such as similar repetitive connectivity patterns, and high density of the spectrum around 1 can signal symmetry at a node or neighborhood level. On the other hand, many eigenvalues around 1 are also specific for random-like connectivity, a non-structured, noisy part of the graph. However, a pronounced elevation above this random redundant bulk indicates the presence of the local symmetries described.

4.2.3 Computational complexity of the Laplacian and its decomposition

Computing the Laplacian matrix of a graph $G = (V, E)$ with $|V| = n$ and $|E| = m$ in a sparse representation requires the computation of the degree of each node in order to get the $-\frac{1}{\sqrt{d_{v_i} d_{v_j}}}$ values. The sparse representation has exactly $2m+n$ nonzero entries, which correspond to the ones from the diagonal (n in total), and the $2m$ fractions computed from the degrees. Getting the degrees of the nodes requires $\mathcal{O}(n + m)$ time by going through all nodes, and for each node, through its edges. Hence, it takes a total time of $\mathcal{O}(n + m)$ to compute the normalized Laplacian matrix of the underlying graph.

Once the Laplacian matrix is constructed, its spectral decomposition must be computed, i.e., its eigenvalues and eigenvectors. For a symmetric matrix such as the normalized Laplacian, this corresponds to finding an orthogonal matrix $Q \in \mathbb{R}^{n \times n}$ and a diagonal matrix Λ such that

$$L = Q \Lambda Q^T,$$

where the columns of Q are the eigenvectors and the diagonal entries of Λ are the corresponding eigenvalues [Str06]. In general, computing the full spectral decomposition of a dense matrix requires $\mathcal{O}(n^3)$ time, making it impractical for large graphs.

However, we have already seen that the most important structural properties are encoded in the smallest and largest eigenvalues and their corresponding eigenvectors. Therefore, in many applications, it is sufficient to compute only a small number of extreme eigenvalues and their associated eigenvectors, rather than the full spectrum.

A standard approach for that is the Lanczos algorithm [Lan50], which is specifically designed for large sparse symmetric matrices. It constructs a sequence of orthogonal basis vectors based on which it builds a much smaller tridiagonal matrix whose eigenvalues iteratively converge to the extreme eigenvalues of the original matrix, and its eigenvectors also approximate the corresponding original eigenvectors.

The computational cost of the Lanczos algorithm is approximately $O(km+nk^2)$ [CT16], where k is the number of computed eigenpairs, making it significantly more efficient for large sparse graphs, although its performance depends on the convergence speed. In this thesis, whenever we only need to compute k eigenpairs, and not the full spectrum, we use the Lanczos-based implementation of the SciPy Python package (scipy.sparse.linalg.eigsh) [VGO+20].

5 The Laplacian spectrum of random graphs

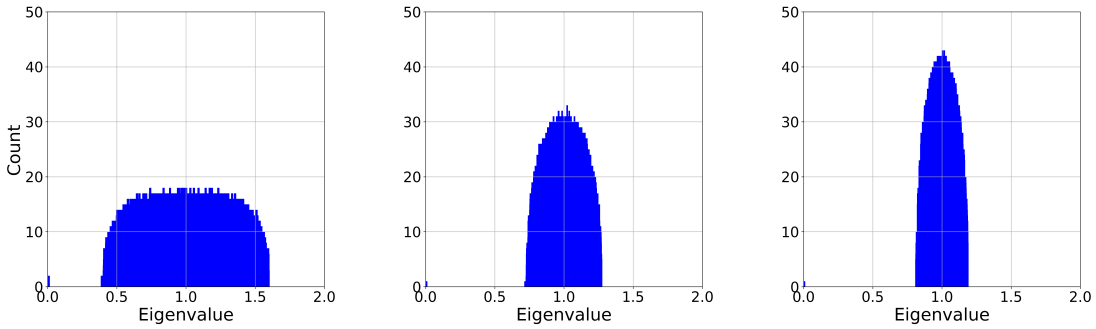
In this chapter, we analyze the Laplacian spectrum of graphs constructed with one of the three well-known random graph models: the Erdős-Rényi model, the Barabási-Albert model, and the Watts-Strogatz model. We experiment with different parameter choices and discuss the outputs. The structure of these three types of networks is widely studied and understood, therefore the goal is to be able to identify different known structural properties in their spectral plots. As many real world networks are known to have structures similar to the random graph models, by understanding the spectra of the random graphs, we may be able to also draw conclusions about the real world networks similar to them.

5.0.1 Erdős-Rényi graphs

We start by analyzing the Laplacian spectrum of the Erdős-Rényi (ER) graph for different values of the edge inclusion probability p. Figure 5.1 shows the obtained plots for inclusion probabilities p of 0.01, 0.05 and 0.1 overlayed on 100 ER graphs with 1000 nodes. The first observation is that the spectra are symmetric around 1 (except the small peak at 0, which is always an eigenvalue with multiplicity at least 1 and indicates the number of connected components of the graph). Observe that, except for the peaks at 0, the eigenvalue distributions closely resemble the Wigner semicircle distribution [Wig57], shifted to be centered around 1. It is also clearly visible that as p increases, and thus the graph becomes denser, the spectrum becomes more tightly centered around 1. If we let p to go until 1, i.e., if we constructed a complete graph, then, by Theorem 4.2.17, all the eigenvalues, except of one having value 0, would be $\frac{1000}{999} = 1.001$. This aligns with the tendency of centering around 1 of the plots in Figure 5.1. The increase in the smallest eigenvalues means faster diffusion times, which is expected with the graph becoming more and more connected. And the decrease in the largest eigenvalues moves the graph further from being bipartite, which is also expected for increased connectivity. In general, ER graphs are homogeneous, hence, real world homogeneous graphs can be expected to have similar spectra.

5.0.2 Barabási-Albert graphs

The next category we consider is a collection of Barabási-Albert (BA) graphs. We plot the Laplacian spectra over 100 realizations of BA graphs with 1000 nodes and different numbers m of edges that are inserted with the addition of a new node. Figure 5.2 shows



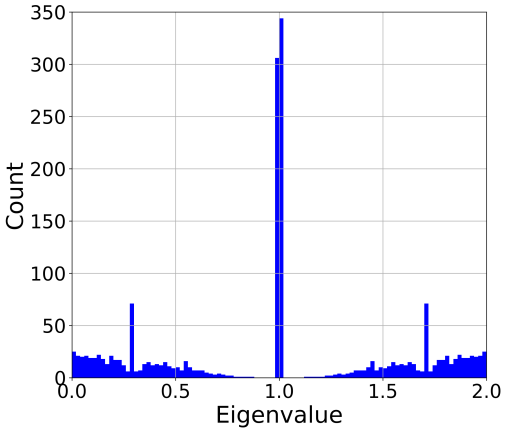
(a) ER graph with $p = 0.01$ (b) ER graph with $p = 0.05$ (c) ER graph with $p = 0.1$

Figure 5.1: Plots of the Laplacian spectra of ER graphs with different edge inclusion probabilities. The number of nodes is fixed to 1000 and each plot is obtained by overlaying the results of 100 repetitions of the same experiment

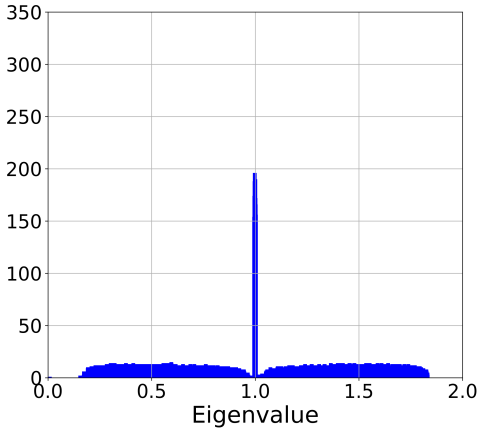
the obtained plots. The first observation is that, similarly to the ER graphs, these plots are also symmetric around 1. Additionally, is is also true that the higher m is, the closer the spectrum gets to 1. However, peaks around different values can be seen, in contrast to the ER bell shaped smoother spectra. The peaks are the most pronounced for $m = 1$. Here, the highest eigenvalues are close (or equal) to 2, meaning that together with the symmetry, the graph is very close to being bipartite. By Theorem 4.2.14, a graph is bipartite if and only if for each eigenvalue λ_i, the value $2 - \lambda_i$ is also an eigenvalue. From the plot, this property seems to hold with just a few exceptions, and is expected, due to the scale-free property, where most nodes are connected to a small number of hubs. Hence, partitioning the graph into two sets, one containing the hubs and the other the remaining nodes, will almost result in a bipartition, with only a few connections present within each set. As we increase m, the highest eigenvalues move closer to 1, and therefore the graph gets further and further away from being bipartite. The smallest eigenvalues also move closer to 1, indicating better connectedness of the graph. Is is also visible that with the increase in m, the peaks around 0.25, 1, and 1.75 gradually smooth out. The peak around 1 indicates the presence of local motifs, where most nodes share similar connections, specifically to the hubs. As m grows, while the hubs still dominate the network's structure, their influence becomes less pronounced. This results in fewer local symmetries and causes the peak around 1 to smooth out. For $m = 10$, the spectrum has a bell-shaped form, similar to ER graphs, with no peaks, having a smoother distribution. This aligns with the higher randomness and more homogeneous structure of the graph, compared to cases with lower m values.

5.0.3 Watts-Strogatz graphs

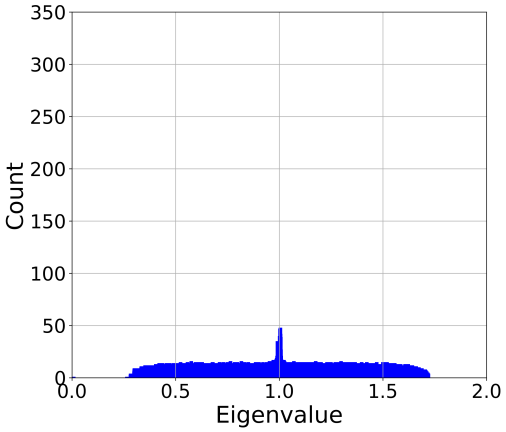
Watts-Strogatz graphs are characterized by two tunable parameters, k, which represents the number of nearest neighbors each node is initially connected to in ring topology, and



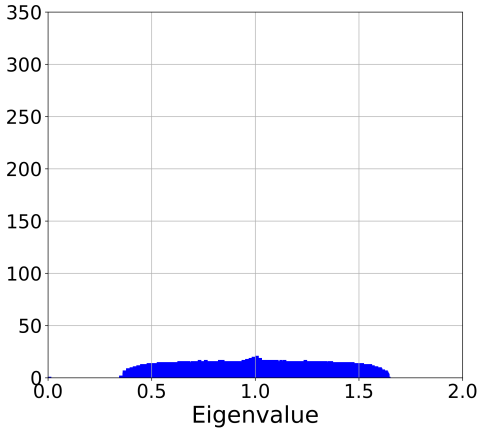
(a) BA graph with $m = 1$



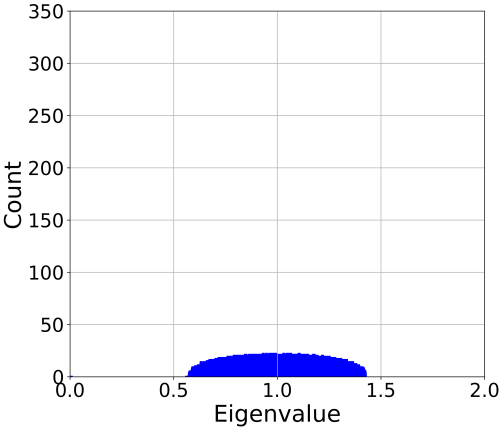
(b) BA graph with $m = 2$



(c) BA graph with $m = 3$



(d) BA graph with $m = 4$



(e) BA graph with $m = 10$

Figure 5.2: Plots of the Laplacian spectra of BA graphs with different numbers m of edges to connect. The number of nodes is fixed to 1000 and each plot is obtained by overlaying the results of 100 repetitions of the same experiment

p, the rewiring probability. Therefore, we will first start by analyzing graphs with fixed k value and changing rewiring probabilities. We start with $k = 2$. Figure 5.3 shows the plots obtained by choosing different rewiring probabilities. Figure 5.3a shows the spectrum of a WS graph without rewiring, which essentially corresponds to a cycle. As stated in Theorem 4.2.18, its eigenvalues are $1 - cos\frac{2\pi k}{n-1}$, with $k = 0, 1, \dots n-1$, which we overlay in red.

By increasing the rewiring probability, peaks start to form around 0.25, 1, and 1.75, similarly to BA graphs with $m = 1$. This proofs that graphs with different degree structures can still have a similar spectrum. However, this also shows that even when degree distributions differ, the spectra can reveal that the graphs actually have similar overall connectivity.

The eigenvalue distribution spans the whole $[0, 2]$ interval for every rewiring probability. This indicates that, on one hand, with $k = 2$, the community structure remains low regardless of the rewiring probability, and on the other hand, the graphs are close to being bipartite for every value p.

Moving forward with the analysis to $k = 4$, Figure 5.4 shows a completely different behavior in this case, compared to $k = 2$. The spectrum becomes heavily asymmetric. By increasing the rewiring probability, the plot becomes more symmetric, while the biggest eigenvalues get closer to 2, the smallest ones move further away from 0. It is not surprising that compared to $k = 2$, the graphs are significantly farther from being bipartite. This is because when each node is connected not only to its immediate neighbors in a ring topology, but also to the nodes that are two steps away, the bipartite structure is completely disrupted, especially in the case of no rewiring or small rewiring probabilities.

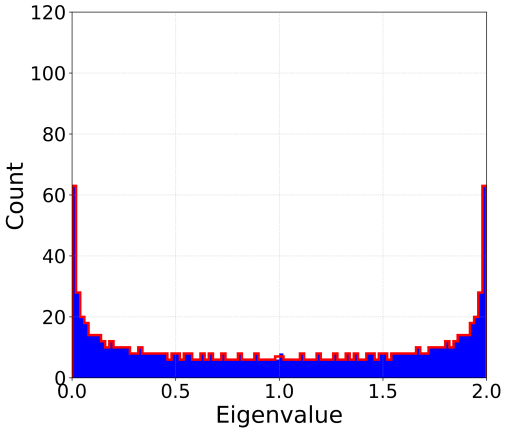
The high peak at 1.5 that slowly smoothens out corresponds to the large number of triangle structures in the graph that progressively get destroyed with higher rewiring probabilities.

The peak at 1 in Figure 5.4a, in contrast to its meaning in the BA graphs, now represents the high global symmetry of the graph. If we gave values of 1, 0, -1, 0 to the nodes of the graph, then the influence from a node's immediate neighbors cancels the influence from its second-order neighbors, resulting in an eigenvalue of 1.

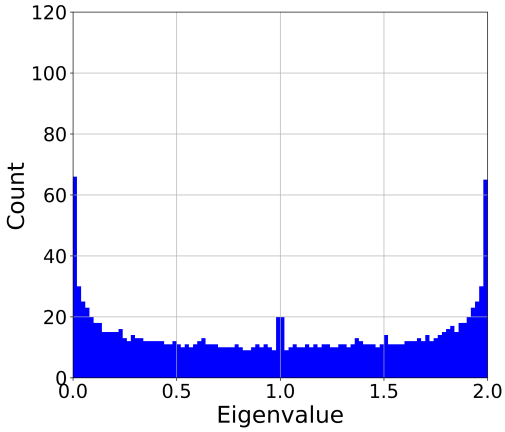
Figure 5.5 shows that for higher values of k that allow for both local clustering and relatively good global connectivity, the behavior is quite similar to the $k = 4$ case. The spectrum is highly asymmetrical with some peaks. The smallest eigenvalues are initially distributed across the entire $[0, 1]$ interval and shift towards 1 as the rewiring probability increases and the graph becomes more random-like. Simultaneously, the largest eigenvalues decrease with increasing p, and the sharp peaks observed in the spectrum become more reduced.

5.0.4 Adding leaf nodes to random graphs

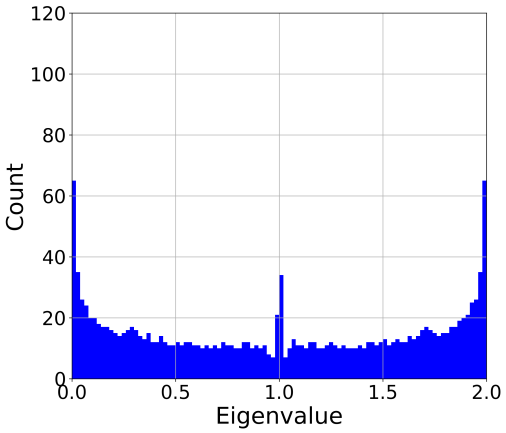
Biological networks often exhibit a specific structure which consists of central hub nodes connected to a large number of leaf nodes. To better understand how this structural property influences the Laplacian spectrum, this section will examine the changes in



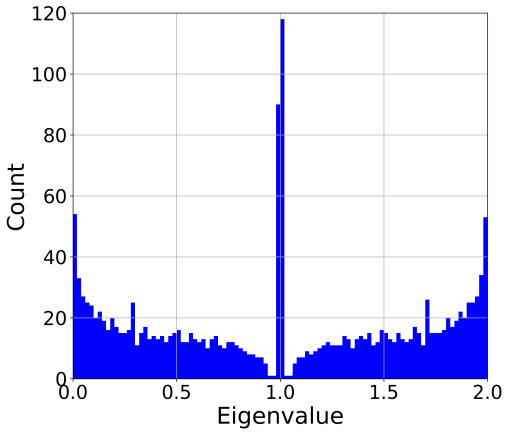
(a) WS graph spectrum with $k = 2, p = 0$



(b) WS graph spectrum with $k = 2, p = 0.05$

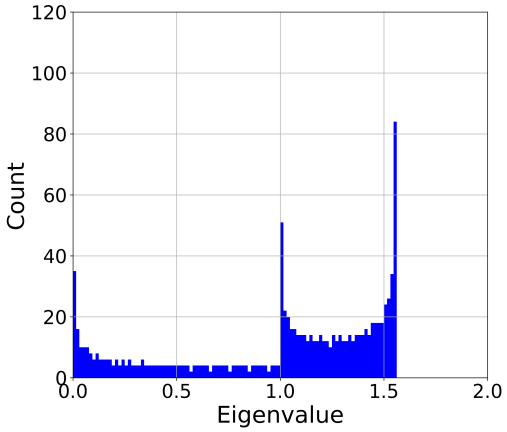


(c) WS graph spectrum with $k = 2, p = 0.1$

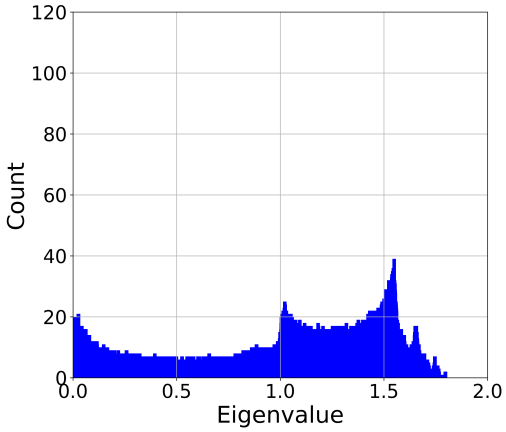


(d) WS graph spectrum with $k = 2, p = 0.5$

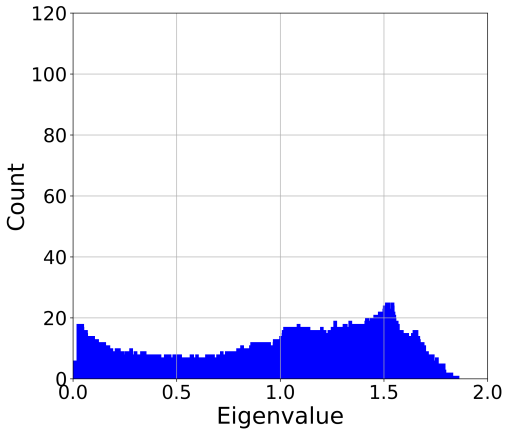
Figure 5.3: Plots of the Laplacian spectra of WS graphs with $k = 2$ nearest neighbors each node is initially connected to and varying rewiring probabilities. The number of nodes is fixed to 1000 and each plot is obtained by overlaying the results of 100 repetitions of the same experiment



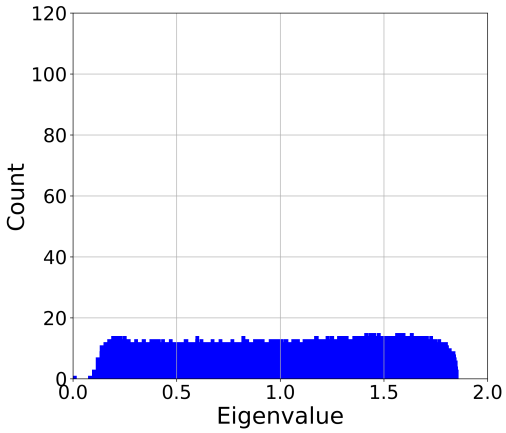
(a) WS graph spectrum with $k = 4, p = 0$



(b) WS graph spectrum with $k = 4, p = 0.05$

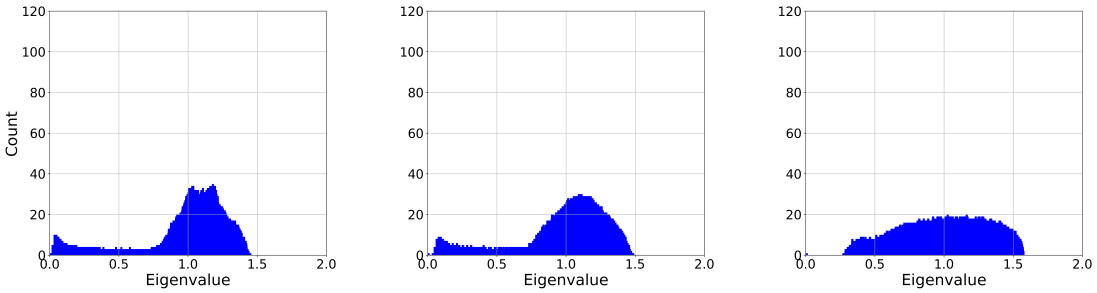


(c) WS graph spectrum with $k = 4, p = 0.1$



(d) WS graph spectrum with $k = 4, p = 0.5$

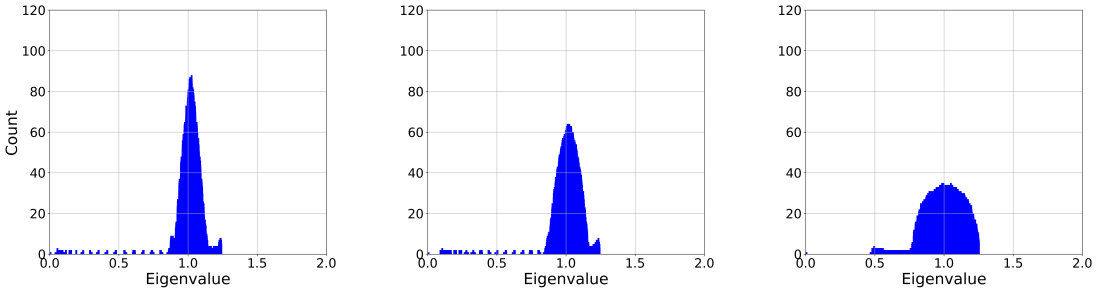
Figure 5.4: Plots of the Laplacian spectra of WS graphs with $k = 4$ nearest neighbors each node is initially connected to and varying rewiring probabilities. The number of nodes is fixed to 1000 and each plot is obtained by overlaying the results of 100 repetitions of the same experiment



(a) WS graph spectrum with $k = 10, p = 0.05$

(b) WS graph spectrum with $k = 10, p = 0.1$

(c) WS graph spectrum with $k = 10, p = 0.5$



(d) WS graph spectrum with $k = 50, p = 0.05$

(e) WS graph spectrum with $k = 50, p = 0.1$

(f) WS graph spectrum with $k = 50, p = 0.5$

Figure 5.5: Plots of the Laplacian spectra of WS graphs with $k = 10$ and $k = 50$ nearest neighbors each node is initially connected to, and varying rewiring probabilities. The number of nodes is fixed to 1000, and each plot is obtained by overlaying the results of 100 repetitions of the same experiment

the spectrum of random graphs when leaf nodes are added to some of the already most connected nodes.

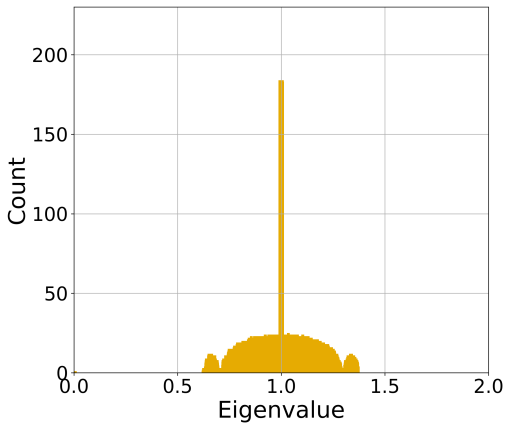
We begin the analysis with the ER graph. To keep the number of eigenvalues fixed at 1000, we start with an 800-node ER graph and then add leaves to the most connected nodes. Specifically, we add 5 leaves to the 40 most connected, 10 nodes to the 20 most connected, 20 leaves to the 10 most connected, and finally, 40 nodes to the 5 most connected nodes. Figure 5.6 shows the results.

The first observation is the appearance of a high peak at 1, compared to the random ER graphs, which corresponds to the added leaf nodes. Moreover, two smaller bumps are seen, symmetric to 1, which are moving further away from 1 as we increase the number of leaves and decrease the number of nodes we add them to. This makes sense, since the graph remains relatively homogeneous in terms of connectivity when we add a smaller number of leaves to a higher number of nodes. Moreover, a slight increase in the peak at 1 can also be observed. This means that the duplication of a node for 5 times, done for the 40 most connected nodes produces less eigenvalues around 1 than 40-40 node duplications for the 5 most connected nodes.

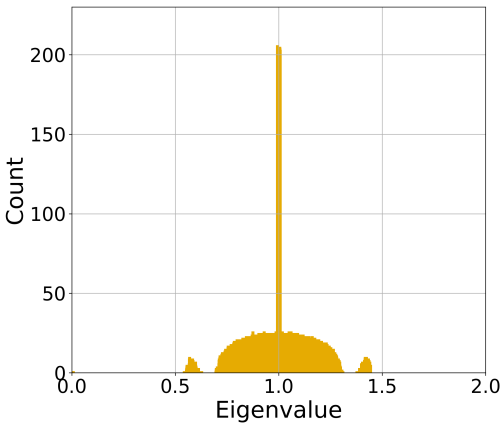
Compared to Figure 5.1b, which shows an ER graph with $p = 0.05$, the difference is in the three peaks and in the slightly more spread shape, implying better community structure, which aligns with our expectations.

If we run the same experiment with BA and WS graphs, the same results can be concluded, i.e. the high peak forming around 1, the small peaks symmetric to 1, and otherwise, the rest of the spectrum being similar to the plots without leaf nodes. Hence, we do not plot the figures obtained from these tests.

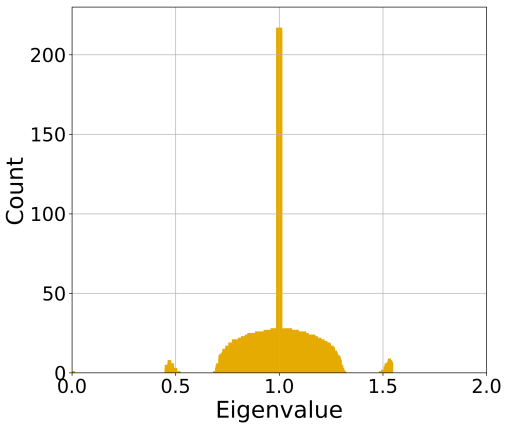
Overall, the experiments with leaf nodes clearly show what we have already expected, namely a high algebraic multiplicity of eigenvalue 1, which corresponds exactly to these leaves. Moreover, the addition of these nodes increases the community structure of the previously more homogeneous graphs, which can be seen in the small peak forming and moving towards 0. The graph becomes closer to having a bipartite strucutre as well, which is also expected, and is shown by the other peak shifting towards 2.



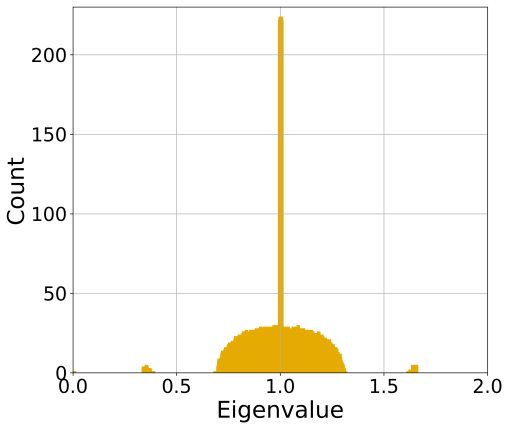
(a) ER graph spectrum with 5 leaves added to the 40 most connected nodes



(b) ER graph spectrum with 10 leaves added to the 20 most connected nodes



(c) ER graph spectrum with 20 leaves added to the 10 most connected nodes



(d) ER graph spectrum with 40 leaves added to the 5 most connected nodes

Figure 5.6: Plots of the Laplacian spectra of ER graphs with 500 nodes and $p = 0.05$, with either 5 leaves added to the top 20 most connected nodes, 10 leaves added to the top 20, 20 leaves to the top 10, or 40 to the top 5 nodes. Each plot is obtained by overlaying the results of 100 repetitions of the same experiment

6 Materials and methods

In this chapter, we describe the data and methodologies used throughout this thesis. We present the spectral-based approaches developed in this work, as well as the state-of-the-art methods used for comparison in the experimental evaluation. We provide the methodological foundation necessary for understanding the analysis and results presented in the next chapter.

6.1 Data

The human protein-protein interaction (PPI) network

The primary dataset used in this work is a human protein-protein interaction (PPI) network constructed from data from four sources: BioPlex [HBP+17] and HuRI [LKL+19], which contain experimentally validated protein-protein interactions, and selected interactions from BioGRID [SBR+06] and IntAct [OAA+14], which include interactions with lower confidence levels. In this network, nodes represent proteins and edges correspond to interactions between pairs of proteins. The number of proteins (nodes) in this network is 18068, while the number of connections (edges) is 306914.

Protein-protein interaction networks provide a structural representation of cellular organization. However, proteins rarely act in isolation but instead interact to form functional units. Hence, PPI networks exhibit characteristic structural properties such as modular organization, heterogeneous degree distributions, and the presence of densely connected subgraphs corresponding to protein complexes. These structural features make PPI networks particularly suitable for spectral analysis, as the Laplacian matrix and its spectral decomposition encode information exactly about these characteristics.

Protein complexes

A protein complex is a group of proteins that physically bind and function together to perform a specific biological task. In the PPI network, protein complexes typically appear as densely connected subgraphs. In this thesis, we use a dataset of protein complexes obtained from the CORUM database [RBD+08].

Disease modules

Next to protein complexes, we also consider disease modules within the PPI network. A disease module is a subset of proteins whose perturbations are associated with a particular disease. From a network perspective, disease modules correspond to subgraphs of the PPI

network whose nodes are involved in the same medical condition. The disease associations were downloaded from disgenet.com [PCR+26].

The disease ontology

To assess semantic similarity between diseases, we utilized the Disease Ontology [SAN+11] obtained from [lBJ+26]. The Disease Ontology is a structured, hierarchical system that organizes diseases based on biological and clinical characteristics. It is represented as a directed acyclic graph, where nodes correspond to disease terms at different levels of specificity, and edges encode parent-child relationships.

This ontology provides a framework for measuring semantic similarities between diseases, which we use to evaluate how well structural and semantic similarities align in disease modules.

We now proceed with the detailed description of the methodology and approaches used in this thesis. All the described algorithms, methods, and experiments are implemented using Python (version 3.8.5).

6.2 Eigenvector localization

In this section, we present how a subgraph $G_s \subseteq G$ of a larger network G can be localized on one or more eigenvectors of G. Each eigenvector represents a structural mode of the network. Hence, if a subgraph corresponding for instance to a protein complex aligns strongly with one of these modes, then a significant fraction of that eigenvector's energy should be concentrated on the nodes of the complex.

Since the eigenvectors computed in Python by NumPy [HMvdW+20] (which we use for eigenvector calculation) are ℓ_2-normalized, meaning that

$$\sum_{i \in V} v_i^2 = 1,$$

a natural localization score of the subgraph G_s on an eigenvector v of graph G is

$$Loc(G_s, v) = \sum_{i \in G_s} v_i^2.$$

This measures how much of the eigenvector's total energy is supported by the nodes of the subgraph. If this value is large, then the eigenvector is strongly concentrated on that subgraph.

For a subgraph, we select the eigenvectors (and corresponding eigenvalues) for which

$$Loc(G_s, v) = \sum_{i \in G_s} v_i^2 \geq k,$$

meaning that at least a fraction k of the total mass of the eigenvector lies on that subgraph.

We use this approach in Section 7.1.1 for different applications related to localizing protein complexes on the PPI network.

6.2.1 Protein complex prediction via eigenvector localization

We propose a novel method for protein complex prediction based on eigenvector localization. To the best of our knowledge, this procedure has not been presented before.

Given a graph G, which is the PPI network in our case, we first compute the inverse participation ratio (IPR) of each eigenvector. The IPR of an eigenvector v is defined as

$$\mathrm{IPR}(v) = \sum_{i=1}^{n} v_i^4.$$

Since the eigenvectors returned by NumPy are ℓ_2-normalized, we have $\sum_{i=1}^{n} v_i^2 = 1$. Hence, a completely delocalized eigenvector distributes its mass uniformly across all n nodes, with entries around $1/\sqrt{n}$, leading to a small IPR. In contrast, a fully localized eigenvector, with value 1 on a single node and 0 on each of the other nodes, attains maximal IPR of 1.

After computing the IPR values, we sort the eigenvectors in descending order by IPR and consider the first k, most localized ones.

For each of these eigenvectors, we perform a greedy local search:

1. Compute the clustering coefficient $c(i)$ of each node i.
2. Compute the localization $Loc(i, v)$ on the given eigenvector v of each node i.
3. Define the score of each node i as
$$\mathrm{score}(i) = Loc(i, v) \cdot c(i).$$
4. Initialize a subgraph with the node of highest score.
5. Iteratively expand the subgraph by adding the neighboring node with highest score.
6. Terminate the search when either a predefined maximum localization ratio is reached or a maximum complex size is exceeded.

The resulting subgraph is considered a predicted protein complex. This way, each of the first k eigenvectors yields one candidate complex.

Eigenvector localization alone can often highlights hub nodes, since hubs may dominate certain eigenvectors. As hubs are not usually indicative of protein complexes, we balance localization with the clustering coefficient. Hub nodes usually have low clustering coefficients, whereas proteins within true complexes form locally dense and homogeneous structures. By combining localization with clustering coefficient, the method is biased towards densely interconnected subgraphs rather than star-like hub structures. The choice of computing the IPR of the eigenvectors was inspired from [CM11].

Evaluation of prediction performance

To evaluate the performance of our approach, we compare it with two state-of-the-art complex prediction methods.

Markov cluster algorithm (MCL): The Markov Cluster Algorithm (MCL) [EVDO02] is widely used as a state-of-the-art method for protein complex detection. MCL is a flow-based clustering algorithm that simulates random walks on the PPI network. It alternates between two operations, expansion: matrix squaring, to simulate longer random walks, strengthening connections within dense regions, and inflation: raising entries to a power and rescaling columns, which strengthens strong connections and weakens weak ones. Through repeated expansion and inflation, flow becomes trapped in dense regions of the graph. These regions are then the predicted protein complexes.

Spectral clustering: We also compare our approach against spectral clustering. Spectral clustering embeds nodes into a low-dimensional space using the normalized Laplacian's eigenvectors and then applies k-means clustering to partition the graph. This method is described in more detail in Section 6.4, in relation to clustering collections of graphs.

Both MCL and spectral clustering are global clustering methods, meaning that every node is forced to be part of a cluster. However, protein complexes do not necessarily cover all proteins, and clustering may predict irrelevant clusters.

In contrast, our method performs local searches and does not require partitioning the entire graph. It identifies denser subgraphs that exhibit strong eigenvector localization, without enforcing the coverage of all nodes.

Other approaches have been proposed in the literature that do not cluster the entire network. Examples include MCODE [BH03], ClusterOne [NYP12], and COACH [WLKN09], which are based on local density expansion, seed growth, or core-attachment strategies. These methods aim to detect complexes without partitioning all nodes and hence, are conceptually closer to our porposed approach. However, we do not include these methods in our experimental comparison. The main reason of excluding them is reproducibility, clear, well-documented implementations are not available, and some existing implementations depend on external software environments. An extended future study however, could incorporate these additional approaches.

To compare the performance of our method with the other two proposed approaches, we use the precision, recall, and F1-scores. For all three methods separately, each true and predicted complex is assigned its highest similarity match, and precision and recall are computed based on the number of complexes whose best match exceeds 50% similarity. The F1 score is then calculated using the resulted precision and recall values.

More specifically, precision (P) is computed as the fraction of predicted complexes that match at least one true complex with similarity above the threshold. Recall (R) is computed as the fraction of true complexes that are recovered by at least one predicted

complex with similarity above the threshold. Lastly, the F1 score

$$F_1 = \frac{2PR}{P+R}$$

is the harmonic mean of precision and recall, providing a single measure that balances the trade-off between correctly predicted complexes and coverage of true complexes.

6.3 Spectral reconstruction of graphs

In this section, we describe our approach of reconstructing graphs based on their Laplacian eigenvalues and eigenvectors. We note that, as mentioned earlier in the related literature, a given Laplacian spectrum may correspond to multiple non-isomorphic graphs [BG12]. Such graphs are called *cospectral*. However, the Laplacian matrix uniquely defines a graph up to isomorphism, as well as its spectral decomposition.

The reconstruction procedure is given in the following. Let $G = (V, E)$ be an undirected graph with $|V| = n$. We compute its normalized Laplacian matrix, and k eigenpairs of it using the Lanczos method described in Section 4.2.3. These may be the k eigenpairs corresponding to the smallest k eigenvalues, the k eigenpairs corresponding to the largest k eigenvalues, or $k/2$-$k/2$ eigenpairs from both ends of the spectrum.

Then, we approximate the Laplacian matrix by $L_k \in \mathbb{R}^{n \times n}$, where

$$L_k = Q_k \Lambda_k Q_k^T,$$

with $Q \in \mathbb{R}^{n \times k}$ containing the k selected eigenvectors as its columns and $\Lambda_k \in \mathbb{R}^{k \times k}$ being a diagonal matrix with the k corresponding eigenvalues.

From L_k, we reconstruct the normalized adjacency matrix S_k, i.e.,

$$S_k = I - L_k,$$

and then, the weighted adjacency matrix A_k of the graph can be computed as

$$A_k = D^{\frac{1}{2}} S_k D^{\frac{1}{2}},$$

where D is the degree matrix.

In the next step, we construct a matrix with n nodes and no edges. Afterwards, we sort the edges in decreasing order according to their weights in the reconstructed adjacency matrix. Starting from an empty graph, we iteratively add edges following this order. We stop once the reconstructed graph contains the same number of edges as the original graph. This procedure yields the reconstructed undirected, unweighted graph, which we then compare to the original one by computing the overlap in the edges,

$$\frac{|E_r| \cap |E|}{|E|},$$

where E_r denotes the set of edges of the reconstructed graph and E denotes the set of edges of the original graph.

We will use this reconstruction approach in Section 7.2.1 to analyze how the reconstruction of disease modules based on different subsets of eigenvalues resembles the original module.

6.4 Graph-level clustering

In this section, we present two *spectral embedding-based clustering methods* for clustering a *collection of graphs*. Given a set of graphs with potentially varying sizes, the goal is to group them into clusters, such that graphs in the same cluster share similar structural properties.

We describe two clustering algorithms, which are based on embedding the graphs by their Laplacian eigenvalues. We then describe different state-of-the-art clustering methods (i.e., Graph2vec and kernels), which will be used to assess and compare the performance of our Laplacian-based methods.

To cluster the graphs, we follow a two-step procedure: we first embed each of the graphs into a k-dimensional vector, and then, one of three widely used clustering methods, i.e. k-means clustering, hierarchical clustering and spectral clustering is applied to the embeddings. For the first step, we present two spectral methods, which generate embeddings based on the eigenvalues of the graphs' normalized Laplacian matrices. We compare our clustering results with embeddings produced by Graph2vec, a state-of-the-art graph embedding method. Moreover, we evaluate our clustering performance against graph kernels, which are widely used methods that compute a similarity matrix directly from the graphs, rather than relying on explicit embeddings, and use this matrix for clustering.

6.4.1 Graph embedding

We first present two graph embedding methods that are based on the spectrum of the graphs' normalized Laplacian. These are the two main methods that we aim to compare to state-of-the-art clustering approaches.

Spectral embedding

Inspired by [Pin19], in this method, each graph is represented by a vector containing its k selected eigenvalues of the normalized Laplacian matrix. Formally, let G be a graph with normalized Laplacian L. We compute the eigenvalues of L, sort them in ascending order, and select either the smallest k (SM), the largest k (LM), or the smallest $k/2$ and largest $k/2$ (BE), depending on the application. The resulting k-dimensional vector serves as the embedding of the graph, capturing structural properties that can be used for clustering.

A natural question that arises is why we do not compute all the eigenvalues of each graphs. There are two main reasons why it is not always possible or efficient to do this. First, as the sizes of the graphs can differ, computing all eigenvalues would yield vectors of different lengths, which is not suitable for the clustering step. One possible solution would be to pad the vectors with zeros or some other values, but this would introduce false structural information, which we want to avoid. Hence, a more logical solution would be to only compute the k eigenvalues for each graph, where k is the number of nodes of the smallest graph. This way, all the graphs are embedded into vectors of the same length, without the need of padding.

The second issue arises when dealing with large graphs. In this case, even the smallest graph can have many nodes ($|V|$), making the computation of $|V|$ eigenvalues for each graph computationally expensive. At the same time, computing more eigenvalues generally provides a better representation of the graph and hence, it can improve the clustering results. Therefore, while a larger number of eigenvalues is desirable and possible in case of smaller graphs, in practice we often need to restrict the computation to a subset of eigenvalues.

Another natural question that arises, is which eigenvalues (SM, LM, BE) to compute. The choice depends on the structural properties of the graphs that are of interest, as discussed in Section 4.2.2. However, in Section 7.2.2, we will make an experiment where we compare the clustering results obtained using each of these selections.

Binned spectral embedding

In the binned spectral embedding, instead of directly using the vector of the k selected eigenvalues for each graph, we partition the interval $[0, 2]$ into k subintervals (bins). For each graph, we then construct a feature vector that records the proportion of eigenvalues falling into each bin. This results in a k-dimensional representation that captures the distribution of the eigenvalues rather than their exact values. The advantage of this approach is that it does not require all graphs to have the same number of eigenvalues. Consequently, we are not constrained by the number of nodes of the smallest graph in the set, for larger graphs, up to k eigenvalues can still be incorporated into the embedding. This allows for a consistent dimension across graphs of varying sizes without being restricted by the smallest graph in the dataset. We note that for many experiments, we will take k, the (maximum) number of computed eigenvalues as the bin size as well, which need not necessarily be the case. This choice is made for ease of discussion and simplified experimentation setup. In principle, these two quantities can be selected independently and as future research, a more rigorous analysis of clustering performance could be conducted by varying them separately.

This approach should be most effective when eigenvalues are distributed differently across graph types and when a larger number of eigenvalues would be computed regardless. If only a very small number of eigenvalues (e.g., $k = 2$ or 4) is used, the corresponding bin sizes are too large, resulting in significant information loss. For larger values of k, and consequently smaller bins, the method may show strong potential, as it can effectively mitigate the impact of noise arising from small differences in eigenvalues.

Next, we describe a different embedding approach that we will use for comparison with the two spectral methods.

Graph2vec embedding

In [NCV+17], Narayanan et al. propose Graph2vec, an unsupervised embedding method for graphs. Inspired by the Doc2vec framework, Graph2vec treats each graph as a document and the rooted subgraphs as words. A skipgram [MCCD13] model with negative sampling [MSC+13] is then used to learn a vector representation for each graph

by predicting the presence of its rooted subgraphs. The resulting embeddings are of fixed length regardless of the graph sizes and have shown good performance compared to graph kernels in clustering tasks.

6.4.2 Clustering the embeddings

We continue by describing the three state-of-the-art clustering methods that are applied in the second step of our graph clustering procedure. These methods are well established and widely used in clustering tasks.

K-means clustering

K-means clustering partitions n vectors of d dimensions (viewed as points in a d-dimensional space) into k clusters by iteratively minimizing the within-cluster variance, i.e. the sum of squared distances between each data point and the centroid of its assigned cluster. In each iteration, every point is assigned to the cluster with the nearest centroid. Afterwards the centroids of the clusters are recalculated as the mean of the assigned points. The process is repeated until convergence.

The performance of k-means clustering is highly dependent on the initialization To address this issue, the k-means++ initialization scheme [AV07] is used The k-means++ algorithm initializes the first centroid uniformly at random, and subsequently selects each new centroid with probability proportional to its squared distance from the nearest already chosen centroid. This encourages a well-separated initialization of the clusters and significantly improves the stability and performance of the algorithm.

K-means clustering is particularly effective for data with roughly spherical and well-separated clusters in Euclidean space, where the notion of a mean is meaningful.

Agglomerative hierarchical clustering

Agglomerative hierarchical clustering is a bottom-up clustering approach that builds a hierarchy of clusters. Initially, each data point is treated as a separate cluster. At each iteration, the two closest clusters are merged according to a predefined distance measure, and this process is repeated until a desired number of clusters is reached.

Agglomerative hierarchical clustering does not require an explicit initialization and is less sensitive to initial conditions than k-means. However, it is typically more computationally expensive, especially for large datasets, due to the repeated computation of inter-cluster distances.

Spectral clustering

Spectral clustering is a graph-based clustering method that leverages the spectral properties of a similarity graph constructed from the data. Given a set of data points, a weighted graph is formed where nodes represent data points and edge weights encode pairwise similarities. From this graph, the normalized Laplacian matrix is constructed.

The key idea of spectral clustering is to compute the first k eigenvectors of the Laplacian matrix and use them to embed the data points into a lower-dimensional space. In this spectral embedding, the structure of the graph is preserved in such a way that clusters become more easily separable. Finally, a standard clustering algorithm, such as k-means, is applied to the rows of the resulting eigenvector matrix to obtain the final partition.

Spectral clustering is particularly effective at capturing non-convex cluster structures and can detect complex relationships that are not well handled by methods such as k-means. However, it requires the computation of eigenvectors, which can be computationally demanding for large datasets.

6.4.3 Evaluation metrics for clustering performance

Clustering is an unsupervised learning task, as it aims to group data without access to ground truth labels. However, in our experiments, we consider collections of graphs with known labels, which allows for a quantitative evaluation of the clustering results by comparing the obtained clusters to the ground truth. For this, we consider two widely used metrics: the Adjusted Rand Index (ARI) and the Normalized Mutual Information (NMI).

ARI measures the similarity between two clusterings by considering all pairs of data points and counting those that are assigned consistently in both clusterings. A value close to 1 indicates a high agreement, while a value close to 0 corresponds to random clustering. NMI, on the other hand, is based on information theory and quantifies the mutual dependence between the predicted clusters and the true labels. It is normalized to take values between 0 and 1, where higher values indicate better agreement. ARI is frequently used as a default, especially for, but not limited to, spherical clusters. NMI is generally better when dealing with complex, non-linear cluster shapes.

By evaluating clustering performance on datasets with known labels, we obtain insights into the general effectiveness of the considered methods. In scenarios where no ground truth is available, the quality of clustering must instead be assessed using internal evaluation measures, such as the Silhouette Coefficient [Rou87], which captures the compactness of clusters and the distinctness between them.

Next, to prepare a more rigorous comparison framework, we introduce graph kernels, which are slightly different approaches, widely used for clustering collections of graphs.

Graph Kernels

We now describe clustering based on graph kernels, a state-of-the-art clustering approach, that we will use for performance comparison to our spectral embedding-based methods.

Graph kernels provide a way to quantify structural similarity between graphs. For a collection of N graphs, a graph kernel computes all pairwise similarities, resulting in a positive semi-definite kernel matrix $K \in \mathbb{R}^{N \times N}$, where each entry K_{ij} measures the similarity between graphs G_i and G_j. This kernel matrix can be interpreted as an affinity (similarity) matrix and directly used as input for clustering methods, enabling the

grouping of graphs according to the structural properties captured by the chosen graph kernel.
Next, we briefly introduce three graph kernels that we will later employ in our experiments.

Weisfeiler Lehman kernel: In the Weisfeiler-Lehman (WL) kernel [SSJ+10], the main idea is to update node labels by progressively incorporating information from their local neighborhoods, thereby capturing increasingly rich structural information. Let $G = (V, E)$ be a graph with initial node labels. If no labels are provided, a constant label is assigned to all nodes. At each iteration h, every node updates its label by concatenating its current label with the multiset of labels of its neighboring nodes. This multiset is sorted and compressed into a new unique label. Repeating this procedure for H iterations produces a sequence of refined labels that encode subtree-like neighborhood structures (up to height H). Thus, the WL kernel measures similarity by counting matching subtree patterns across graphs.

Shortest path kernel: The shortest-path graph kernel [BK05] is a graph kernel that measures similarity between two graphs by comparing all their pairwise shortest paths. Given graphs G and G', each is first transformed into its shortest-path graph, where every pair of nodes is connected by an edge labeled with their shortest-path distance in the original graph. The kernel is then defined as the sum, over all pairs of these "shortest-path edges" e in G and e' in G', of a base kernel $k^{(1)}_{walk}(e, e')$. The kernel explicitly counts matching shortest paths that share identical lengths and identical endpoint labels, capturing global connectivity patterns.

Graphlet sampling kernel: The graphlet sampling kernel [SVP+09] measures similarity between two graphs by comparing the frequencies of small induced subgraphs, called graphlets, of a fixed size k (typically $k \in 3, 4, 5$). Each graph is represented by a feature vector whose entries count the occurrences of each distinct isomorphism type of graphlets of size k. The kernel is then defined as the inner product of these two feature vectors.

Once a similarity matrix is computed using a graph kernel, a clustering algorithm is required to partition the graphs. Among the three clustering algorithms considered earlier, only hierarchical and spectral clustering can directly operate on a precomputed similarity matrix. K-means clustering is based on feature vectors and requires explicit graph embeddings. We note that other relational clustering methods, i.e., approaches that cluster objects based on pairwise similarities or dissimilarities rather than feature vectors, could also be applied. However, for simplicity, we restrict to hierarchical and spectral clustering.

For the Graph2vec embedding, k-means, hierarchical and spectral clustering, as well as for the graph kernels, we will use precomputed implementations from the Scikit-learn [PVG+11], Karate Club [RKS20] and GraKeL [SNL+20] python packages.

All the above described methods are compared in Section 7.2.2, where the main goal is to cluster cancer-relate diseases. However, experiments are conducted on synthetic data and other biological networks as well, in order to assess the clustering power of the

spectral embedding-based methods.

7 Results and discussion

In this chapter, we analyze the eigenvalues and eigenvectors of the human protein-protein interaction network (PPI) with the goal of uncovering meaningful structural properties that may provide insights into the organization and function of the network. We perform eigenvector localization of the protein complexes as described in 6.2 and interpret the results. Afterwards, we propose an algorithm for predicting new protein complexes based on eigenvector localization.

After the analysis of protein complexes, we turn to disease modules. Here, we examine the spectrum of diseases of different types and sizes, and perform spectral reconstruction described in Section 6.3 of the methodology. We then cluster a collection of cancer related diseases using the clustering methods described in Section 7.2.2 of the previous chapter and interpret the results.

7.1 The human protein-protein interaction network

The protein-protein interaction network (PPI), is a 18068-node and 306914-edge graph that models the protein interactions within the human body. Sets of proteins function together, forming complexes that are responsible for the various biological processes such as signaling or DNA replication. The nodes of the PPI graph represent the proteins, and the edges are the interactions between them.

The PPI network is composed of two connected components, one only containing four nodes and three edges. We remove this small component and proceed by only analyzing the one large connected component of the network from this point onward. The minimum node degree of the component is 1, while the maximum node degree is 2344.

We first plot the eigenvalue distribution of the PPI network in Figure 7.1. The eigenvalues span the $[0.25, 1.75]$ interval, with a very large peak at 1. The symmetric, highly concentrated distribution around 1 slightly resembles the ER graph model's spectrum, which would suggest a quite homogeneous and weakly structured network. However, the high peak at 1 clearly makes the PPI network different from a random ER model.

Such a peak also appeared in the BA graphs with small m values. In the case of BA graphs, the peak reflects the scale-free nature of the graphs with many low-degree nodes attached to a few high-degree hubs. This suggests that the PPI network also contains many small-degree protein nodes connected to central hub proteins.

The spectrum does not contain very small or very large eigenvalues, which clearly shows that the network is not well separable into communities, having overlapping modules, and is also far from being bipartite. This differs significantly from BA graphs with small m,

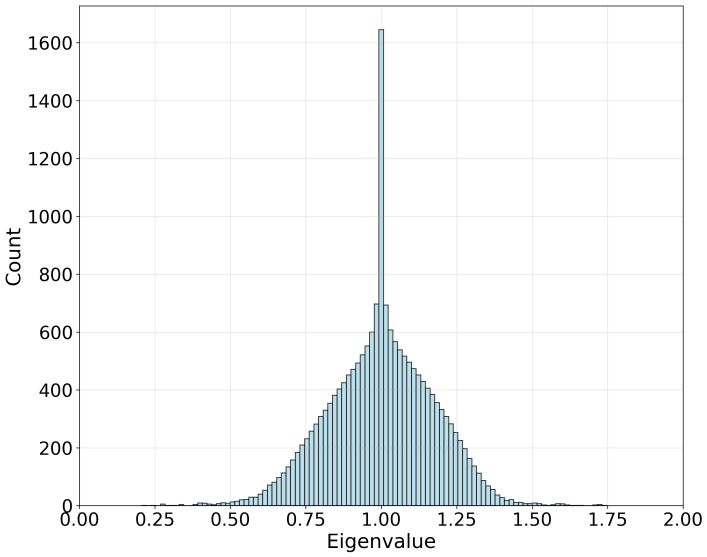


Figure 7.1: The spectrum of the PPI network

where, in addition to the peak at 1, we can observe many eigenvalues near both ends of the spectrum. These observations suggest that the PPI network is scale-free, with many low-degree, possibly leaf nodes attached to hubs, but also has characteristics of a random graph with weak community separation.

7.1.1 Locating protein complexes on the spectrum

In this section, we analyze protein complexes, which are groups of proteins that physically interact to perform a specific biological function. In contrast to arbitrary subnetworks, protein complexes represent experimentally validated functional units. We download a set of 5513 annotated human protein complexes from the CORUM database [RBD+08].

After analyzing the eigenvalue distribution of the PPI network, we use the eigenvectors to localize protein complexes on the spectrum. The approach is described in detail in Section 6.2 of the previous chapter.

For each complex, we select the eigenvectors (and corresponding eigenvalues) for which

$$Loc(C, v) = \sum_{i \in C} v_i^2 \geq 0.01,$$

meaning that at least 1% of the total mass of the eigenvector lies on that complex. Even though 1% may seem small, recall that the eigenvector mass is distributed among around 18 thousand nodes in the PPI network.

Out of the 5513 complexes, 1571 have at least one eigenvector where they are responsible

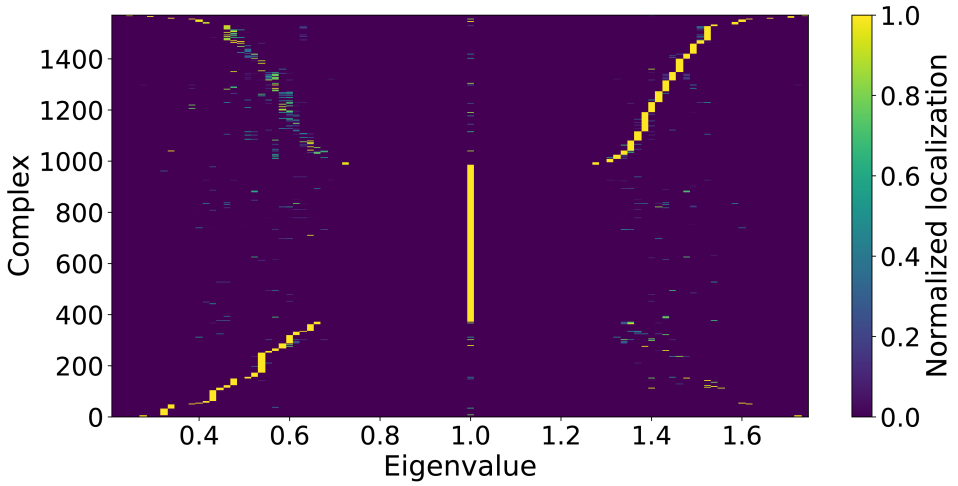


Figure 7.2: Heatmap of the selected eigenvalues for each of the complexes

for at least 1% of the total mass. This already shows that many biological complexes correspond to structural patterns in the network.

We visualize the results in a heatmap in Figure 7.2, where each row corresponds to one of these 1571 complexes and the columns correspond to eigenvalues grouped into 100 spectral bins. The color encodes the strength of localization. To obtain a clearer visualization, we sort the complexes by the bin in which their maximum localization occurs.

The results show that the complexes localize in three regions of the spectrum, namely near the lower end, around 1, and near the upper end.

In the case of the normalized Laplacian, localization in the first region corresponds to well-separated complexes. On the other hand, localization on the other end of the spectrum may indicate complexes that have a bipartite structure, where neighboring nodes strongly disagree in the eigenvector. Localization here means that these complexes bridge different parts of the network or sit in structurally polarized regions. Lastly, since the PPI network has many eigenvalues close to 1, it is natural that we observe some localization here as well. Complexes captured here are neither isolated, nor strongly bipartite, but they are embedded in moderately connected regions with overlapping complexes

However, due to the high peak in the PPI spectrum around 1, an important question that arises is wether localization in this region is biologically and structurally meaningful, or if we would observe similar behavior for random sets of nodes of the same size.

To answer this question, we compare each real complex to random node sets. For every complex, we sample 100 random sets of nodes of the same size and compute their localization scores. Afterwards, for each complex, we compare the maximum localization score of the true complex to the maximum localization score of the random subgraphs by taking the ratio of the two.

This ratio measures the enrichment, i.e., how much more strongly the real biological

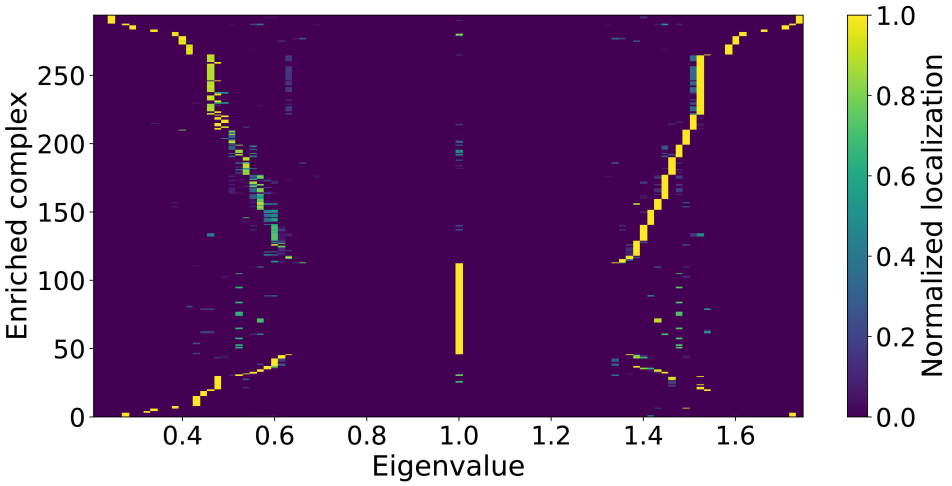


Figure 7.3: Heatmap of the selected eigenvalues for the enriched complexes

complex aligns with a spectral mode compared to random node sets of the same size. If the ratio is large, the localization is not caused by the size of the complex or by random chance.

In Figure 7.3, we plot the 298 complexes where the maximum localization score is at least three times larger than that of random sets. These complexes are spectrally enriched, they capture substantially more eigenvector mass than expected by chance. This indicates that certain protein complexes are aligned with particular structural modes of the PPI network. Observe that, compared to the previous Figure 7.2, many complexes localized to eigenvalues close to 1 were excluded.

Overall, this analysis shows that protein complexes are not randomly embedded in the PPI network, but instead correspond to specific structural patterns that are detectable through spectral localization.

Relation between localization and biological function: A next question regarding complex localization arises with respect to the biological functions of these complexes. For this, we use the CORUM Functional Complex Groups data [RBD+08], where protein complexes are associated with functional classes. We merge the dataset containing the functional classes with our data, and identify 403 protein complexes that appear in both datasets. The 403 complexes are categorized into exactly 30 functional classes, as shown in Table 7.1.

We now consider the three localization regions shown in Figure 7.2, and compute, for each region, the number of complexes whose maximum localization falls into that region for each functional class. The results are presented in Table 7.2.

The table indicates that in many cases complexes of the functional classes are approximately uniformly distributed across the three spectral regimes. Hence, they cannot be clearly separated based on this criterion. This suggests that complexes belonging to the

Functional Complex Group	**Count**
G protein-coupled receptor signaling pathway	154
Nuclear hormone receptors	76
Nuclear transport	17
Inflammasome	17
Vesicle-mediated transport	12
Mitochondrial protein import	12
Signaling of TNF receptor superfamily	11
Cilium assembly	10
Visual perception	10
Interleukin signaling	8
Vascular endothelial growth factor signaling	8
Sodium ion transmembrane transport	7
Toll-like receptor signaling	7
Intraciliary transport	7
RIG-I-like receptor signaling	6
Modulation by pathogen of host process	6
Oxigen transport	5
Calcium ion transmembrane transport	5
Spliceosome	5
Cell adhesion mediated by integrin	4
Regulation of blood pressure	3
Potassium ion transmembrane transport	3
Sensory perception of sound	2
Insulin secretion	2
Serotonin receptor signaling pathway	1
Gamma-aminobutyric acid signaling pathway	1
Acetylcholine receptor signaling pathway	1
Glucose transmembrane transport	1
Chloride transmembrane transport	1
Organic anion transport	1
Total	403

Table 7.1: Distribution of complexes across the functional complex groups

same functional class do not necessarily have similar structural properties, nor encode similar patterns of the PPI. Hence, structural characteristics alone do not appear sufficient to identify similarities among complexes of the same functional class, or the classification of orphan protein complexes (i.e., complexes with unknown function) to the functional classes.

We note that, to support this finding, we also conducted multiple supervised and unsupervised learning experiments to predict the functional classes of the complexes with already known function. The unsupervised approaches included the ones described in 6.4, while the supervised methods consisted of (binned) spectral embedding (as described in Section 6.4.1), as well as unrelated embedding approaches, each combined with basic random forest or support vector classifiers, and even GNN-based models. Across all approaches, we obtained consistent results, indicating that the function of protein complexes cannot be reliably predicted from their structural properties.

However, the localization of protein complexes reveals that certain biological functions or processes still correspond to specific structural regimes of the PPI. In particular, some immune-related processes, such as inflammasome activity, show a strong tendency to localize in the low end of the spectrum. Nodes with high localization at the eigenvectors corresponding to small eigenvalues are usually also globally influential.

In contrast, transport processes appear to be more frequently associated with higher eigenvalues, while for example G protein-coupled receptor signaling pathway, nuclear hormone receptors, or cilium assembly tend to be distributed more broadly, with a significant contribution from the intermediate spectral region as well.

While these observations are not uniform across all functional categories and should be interpreted with caution, they do indicate that spectral properties of the PPI network capture biologically meaningful differences between some functionality classes. Future research could further investigate the properties of complexes falling into the same localization region.

7.1.2 Protein complex prediction

In the previous section, we observed that protein complexes localize in three well-separated regions of the spectrum of the PPI network. Although, we have also seen that biological function cannot be predicted from spectral position, the existence of the three highly localized spectral regimes suggests that protein complexes have characteristic structural properties detectable from the localization. In particular, many of the known complexes carry a quite high proportion of eigenvector mass on at least one eigenvector, indicating that eigenvector localization may highlights biologically relevant subgraphs. This motivates the idea of predicting previously unknown complexes using eigenvector localization. For that, we propose a new method described in detail in Section 6.2.1 of the previous chapter. Here, we just briefly explain the approach: first, we compute the inverse participation ratio (IPR) of all eigenvectors of the PPI network and select the top k most localized ones. For each selected eigenvector, we score nodes by combining their eigenvector localization and clustering coefficient, then perform a greedy expansion starting from the highest-scoring node. The process stops when a localization or size threshold is reached, and the resulting

	Localization region		
Functional Complex Group	**[0.2,0.8)**	**[0.8,1.2]**	**(1.2,1.8]**
G protein-coupled receptor signaling pathway	42	54	58
Nuclear hormone receptors	16	29	31
Nuclear transport	3	9	5
Inflammasome	17	0	0
Vesicle-mediated transport	2	2	8
Mitochondrial protein import	2	8	2
Signaling of TNF receptor superfamily	6	1	4
Cilium assembly	2	5	3
Visual perception	4	4	2
Interleukin signaling	1	4	3
Vascular endothelial growth factor signaling	1	0	7
Sodium ion transmembrane transport	2	3	2
Toll-like receptor signaling	2	3	2
Intraciliary transport	0	5	2
RIG-I-like receptor signaling	3	0	3
Modulation by pathogen of host process	5	1	0
Oxigen transport	5	0	0
Calcium ion transmembrane transport	0	3	2
Spliceosome	0	3	2
Cell adhesion mediated by integrin	0	3	1
Regulation of blood pressure	0	1	2
Potassium ion transmembrane transport	0	0	3
Sensory perception of sound	1	1	0
Insulin secretion	0	2	0
Serotonin receptor signaling pathway	1	0	0
Gamma-aminobutyric acid signaling pathway	0	0	1
Acetylcholine receptor signaling pathway	1	0	0
Glucose transmembrane transport	0	0	1
Chloride transmembrane transport	0	0	1
Organic anion transport	1	0	0
Total	117	141	145

Table 7.2: Distribution of functional complex groups across the three localization regions

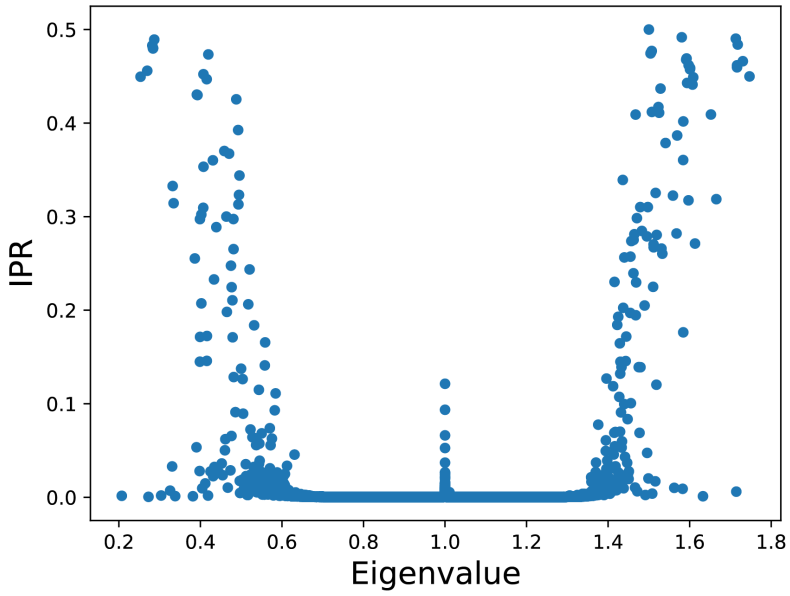


Figure 7.4: IPR scores of the eigenvectors of the PPI

subgraph is the predicted protein complex.

Figure 7.4 shows the IPR values of each eigenvector of the PPI network. We observe that the most localized eigenvectors correspond to eigenvalues located in the same regions where known complexes were previously localized. This further supports the hypothesis that eigenvector localization may capture meaningful structural information relevant to protein complexes.

Experimental results

To evaluate the performance of our approach, we compare it with two state-of-the-art complex prediction methods, namely the MCL algorithm and spectral clustering, both detailed in Section 6.2.1.

First we run the MCL algorithm, which produces 5960 predicted complexes. To ensure comparability, we run spectral clustering with 5960 clusters as well. Moreover, for our method, we also use the first 5960 eigenvectors with highest IPR values, generating one predicted complex per eigenvector. We set the maximum complex size to 50 and the maximum localization ratio to 30%, choosing these parameters to be consistent with the observed known complexes.

This way, each method produces 5960 predicted complexes which are compared against the 5468 known complexes. We compare performance using precision, recall, and F1-score, described in Section 6.2.1.

The results in Table 7.3 indicate the superiority of the proposed localization-based

Measure	MCL	Spectral clusrtering	Localization-based search
Precision	0.39	0.42	0.97
Recall	0.94	0.83	0.55
F1 score	0.55	0.56	0.70
Matched true	5150	4547	3003
Matched predicted	2336	2506	5791

Table 7.3: Performance comparison among the different protein complex prediction methods

method. The MCL finds almost all true complexes, but produces many false positives, similarly to spectral clustering. Our method on the other hand has the highest F1 score, with the best balance between precision and recall. When it predicts a complex, it is almost always correct. While the high F1 makes our algorithm the best overall, the choice also depends on the goal of the user. If one is interested in finding the best coverage of the known complexes, then the other two methods are more favorable. However, when predicting protein complexes, we may prefer to generate fewer but highly reliable predictions, minimizing false positives and increasing the likelihood that the identified complexes indeed correspond to true biological complexes.

It is also important to repeat that we use a similarity threshold of 0.5 to define a match between predicted and true complexes. This corresponds to requiring only a 50% overlap between complexes, meaning that even when a prediction is considered correct, only half of the nodes are guaranteed to be shared with the corresponding true complex. Therefore, the evaluation focuses on detecting partially overlapping structures rather than the exact complexes, which is a common setting in protein complex prediction due to the incompleteness and noise in the PPI network.

The experiment already shows that our method is promising, however, several extensions and more experimentation are possible in future work. For example, extracting multiple complexes per eigenvector by repeating the same process starting from another, not yet included, highly localization node. Or replacing the clustering coefficient with another measure that captures the desired complex structure possibly better. Further tuning the parameters of the model, i.e. the number k of eigenvectors, the maximum complex size, or the localization ratio, may improve performance and provide more biological insights. Moreover, other well-known protein complex prediction approaches [BH03, NYP12, WLKN09] could also be implemented and used for comparison with our proposed method.

7.2 Disease modules

Disease modules are groups of proteins, whose disfunction is associated with a disease. In this section we will use spectral methods based on the normalized Laplacian to analyze the diseases and their structure.

We first compute the largest connected component (lcc) of each disease module. We

restrict our analysis to the lccs of the diseases rather than considering the full modules. The lcc captures the main connected core of disease-associated proteins, where interaction evidence is strongest and most coherent. Smaller disconnected components are more likely to arise from noise, incomplete data, or weak associations. Hence, focusing on the lcc reduces noise and yields a more robust representation of each disease module.

To understand whether the observed lcc of a diseases is meaningful, we compare its size to the lcc of randomly selected subgraphs of the PPI network with the same number of nodes. To quantify this, we compute the z-score:

$$z = \frac{lcc_{disease} - \mu_{random}}{\sigma_{random}},$$

where $lcc_{disease}$ is the size of the largest connected component of the disease module, μ_{random} and σ_{random} are the mean and the standard deviation of lcc sizes of random subgraphs of the same size. A z-score of e.g., 10 means that the lcc size of the disease is 10 standard deviations larger than expected by chance. Figure 7.5 shows very large z-scores for most diseases. This means that the disease-associated genes tend to cluster together in the PPI much more than expected by chance. Hence, disease genes are not randomly scattered across the network, but they form significantly connected modules.

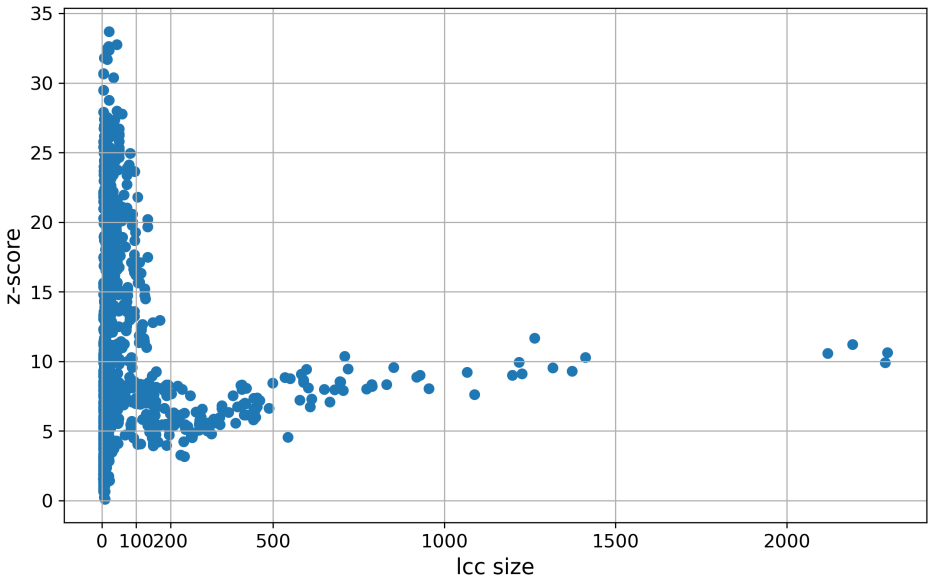


Figure 7.5: Largest connected component size plotted against the corresponding z-score for each disease

Next, we plot the distribution of lcc sizes in a histogram on a logarithmic scale. The plot in Figure 7.6 shows that most diseases have an lcc size smaller than 1000. Based on this observation, we group diseases into different lcc-size regimes and analyze their spectra

accordingly. The spectrum of a graph acts as a fingerprint for it, enabling us to infer important structural properties without the need of visualizing the entire, potentially very large graph.

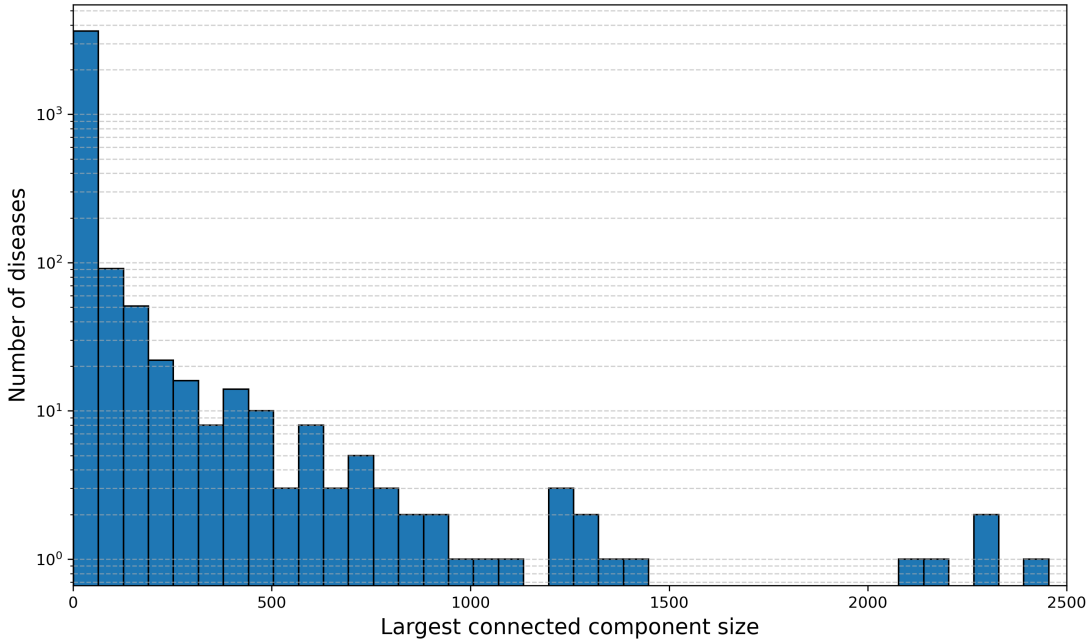


Figure 7.6: The distribution of the size of the largest connected component of each disease

Diseases with very large largest connected component

We start with the few diseases whose lcc size exceeds 1000. These are mostly broad, general disease categories that include many diseases. Figure 7.7 shows the overlaid spectra of some of them. Overall, they share a similar spectral structure, namely a large peak at 1 and a relatively symmetric shape around that peak. However, cancer clearly differs from the others. Its spectrum has a much smaller peak at 1, a sharper increase towards 1, and a symmetric, sharper decrease afterwards. The other diseases show a slower increase towards 1 and a higher peak at 1. This already suggests that cancer modules may have a different internal organization. The narrower spectrum reflects a a well connected structure. Interestingly, the cancer spectrum also closely resembles the spectrum of the whole PPI network shown in Figure 7.1.

Diseases with large largest connected component

Next, we analyze diseases with lcc sizes between 400 and 1000. These are more specific diseases. Their z-scores range between 4 and 11, with two clear outliers showing in Figure

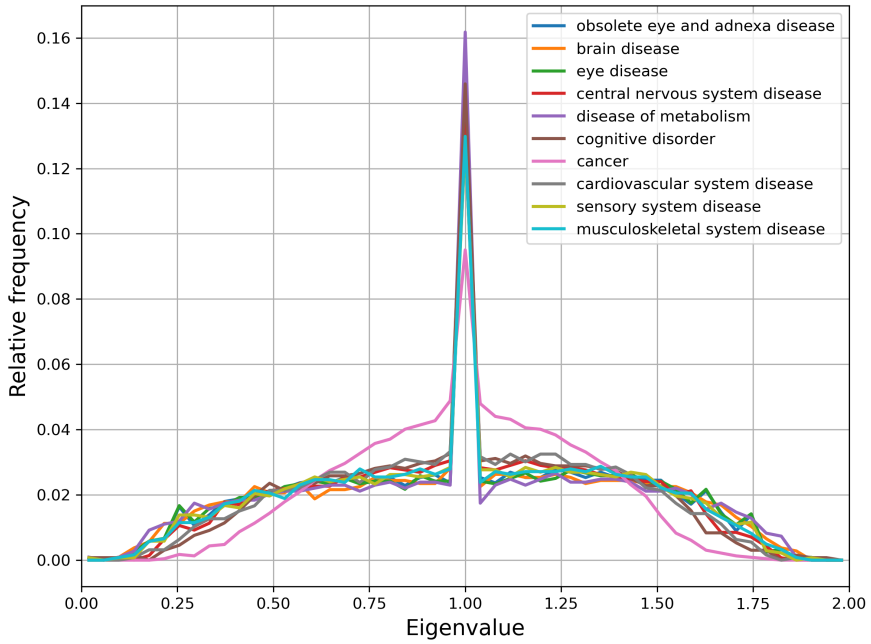


Figure 7.7: Spectra of selected diseases with largest connected component containing more than 1000 nodes

7.5, which are the inherited metabolic disorder ($z = 10.4$) and vascular disease ($z = 4.6$). We plot the spectra of these two together with four other representative diseases from this group. Figure 7.8 again shows that the two cancers in this size range (breast cancer and hematologic cancer) follow the same pattern observed before, i.e., sharper increase and decrease around 1 with a lower peak at 1.

Even though these cancers are biologically very different, their spectra have very similar shapes compared to the other diseases. This strongly suggests that structural similarity in the network may align, at least partly, with semantic similarity.

Diseases with medium-sized largest connected component

We now turn to diseases with LCC sizes between 100 and 200. The z-score distribution in Figure 7.5 shows much larger variability here, ranging from 4 up to 25. We therefore first analyze the higher z-score diseases and then the lower (but still significant) z-score diseases.

Interestingly, many diseases with high z-scores in this size range are neoplasms, either benign or malignant. We therefore select three benign and three malignant neoplasms for comparison. Figure 7.9 shows their overlaid spectra. A clear difference appears, namely that benign neoplasms have smaller peaks at 1 than malignant ones. In contrast, the eigenvalue density right before and after 1 is higher for benign neoplasms. This is interesting because we earlier observed the smaller peak at 1 in cancers, while here, benign

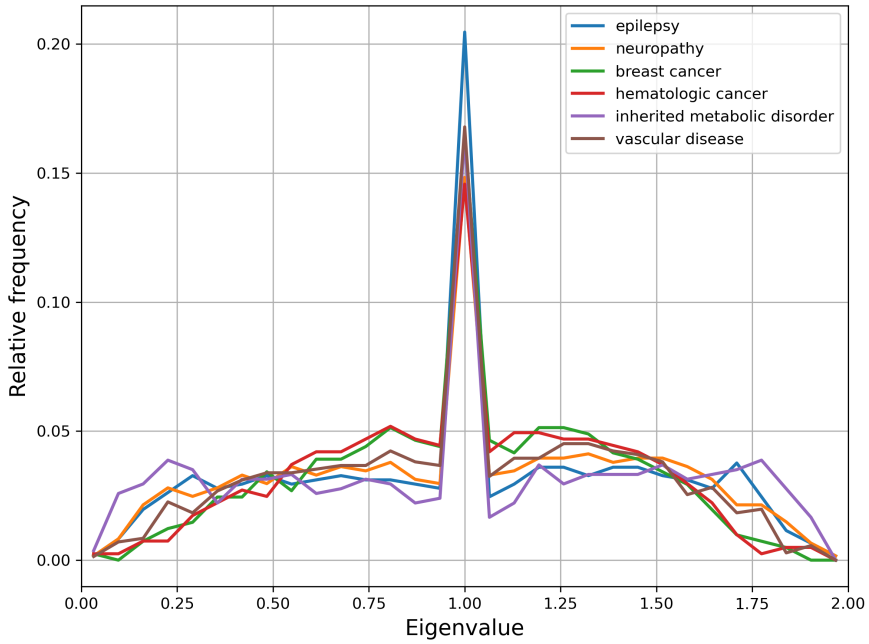


Figure 7.8: Spectra of selected diseases with largest connected component containing between 400 and 1000 nodes

neoplasms have smaller peaks than the malignant ones. However, the benign diseases show peaks around the ends of the spectrum as well (in two of the three cases), which in the case of cancers are usually not present. This indicates stronger, better-defined community structure in benign neoplasms. Overall, the plot shows clearly visible distinction between benign and malignant neoplasms due to the heights of the peaks around 1. This underlines the strength of the eigenvalue fingerprint of diseases, and that structural similarities of the modules seem to align with semantic similarities.

We now consider diseases with the same lcc size (100-200) but lower z-scores. A large fraction of these diseases are ophthalmologic, with only very few cancers appearing in this group. This means that cancer-related diseases generally have bigger largest connected components in comparison to the other diseases, indicating a higher degree of connectivity within their associated modules.

We plot several eye diseases together with two comparison diseases, i.e., colorectal cancer and glomerulonephritis. Figure 7.10 shows the overlayed spectra. The eye diseases share common features, namely smaller peaks around 1, relatively high values near 0, and no eigenvalues close to 2. In contrast, colorectal cancer does not show many eigenvalues near 0, while glomerulonephritis does have more eigenvalues close to 2, compared to the ophthalmologic diseases, and both have a higher peak at 1.

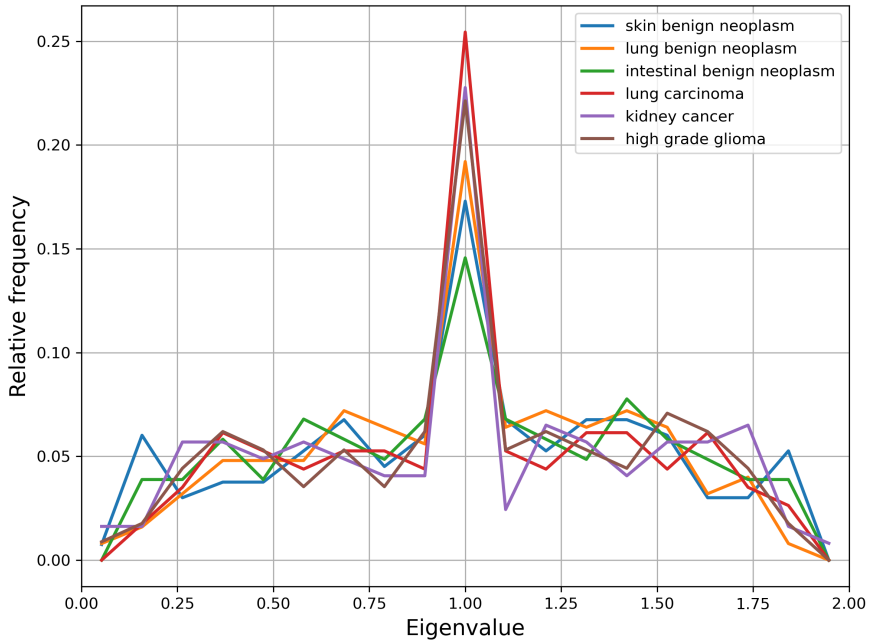


Figure 7.9: Spectra of selected diseases with largest connected component containing between 100 and 200 nodes, and z-score larger than 10

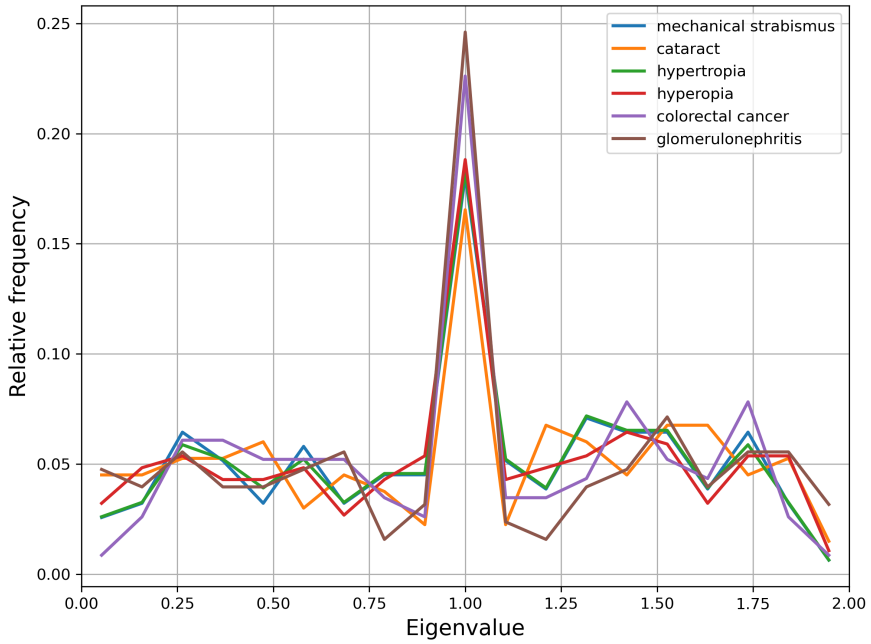


Figure 7.10: Spectra of selected diseases with largest connected component containing between 100 and 200 nodes, and z-score smaller than 10

Diseases with small largest connected component

Finally, we analyze diseases with small lcc. Although many diseases have very small largest connected components, we restrict the analysis to those with lcc sizes between 10 and 50 nodes. Below 10 nodes, the graphs are too small for a meaningful spectral analysis. As seen in Figure 7.5, the z-scores in this size regime span a large interval. We therefore again divide the diseases into two groups: diseases with z-score larger than 10 and diseases with z-score smaller than 10 (but still significantly larger than expected by chance). Since there are many diseases in this size range, the biological variety is much larger. Hence, we select diverse examples spanning different systems.

For the high z-score group, we select two neurological diseases (Alzheimer's disease and Parkinson's disease), one cardiovascular disease (dilated cardiomyopathy), two cancers (lymphatic system cancer and lung adenocarcinoma), and one genetic disease (Duchenne muscular dystrophy). Figure 7.11 shows the spectra. The two neurological diseases have the highest peak at 1, which suggests scale-free-like structure. The cancers again follow the pattern observed earlier, i.e., smaller peak at 1, and increased density of eigenvalues around 0.75 and 1.25, hence a sharper rise and fall around 1. This consistency across size regimes strengthens the idea that cancer modules have a characteristic spectral fingerprint.

Dilated cardiomyopathy, in contrast, shows larger numbers of eigenvalues at the two ends of the spectrum (near 0 and 2), meaning weaker global connectedness and community structure. The most unique is the Duchenne muscular dystrophy, since its spectrum shows very large number of eigenvalues at the endpoints and a relatively small peak at 1. Its spectrum clearly differs from the scale-free-like structures we observed for other diseases and resembles more the spectra of simpler, path-like graphs.

We now turn to diseases with smaller (but still significant) z-scores in the same size range. We again find several ophthalmologic diseases here, underlining the earlier observation that eye diseases tend to have smaller lccs. Hence, we select three eye diseases (retinitis pigmentosa, myopia, and blindness), together with asthma (respiratory), normocytic anemia (hematologic), craniosynostosis (skeletal), and familial periodic paralysis (genetic disease). Their spectra are shown in Figure 7.12.

Interestingly, myopia and blindness have very few eigenvalues at or around 1, but a large peak around 1.5. Eigenvalues of 1.5 indicate the presence of many triangles. Retinitis pigmentosa also does not show a large peak at 1, but rather a relatively uniform distribution between 1 and 1.5, corresponding to tightly clustered motifs.

Familial periodic paralysis has a different spectrum, with a large number of eigenvalues at both endpoints. A similar pattern appears for craniosynostosis. This suggests that both diseases have path-like lcc, which is consistent with Theorem 4.2.18, stating that the eigenvalues of a path on n nodes are $1 - \cos\frac{\pi k}{n-1}$ for $k = 0, 1, ..., n-1$. This formula produces a spectrum that accumulates near the two ends of the interval $[0, 2]$, exactly what we observe in these cases.

Since these graphs are small enough to visualize, we plot the lcc of craniosynostosis and blindness in Figures 7.13a and 7.13b, expected to be path-like and triangle-dense respectively. The graphs confirm what the spectra already showed. Craniosynostosis

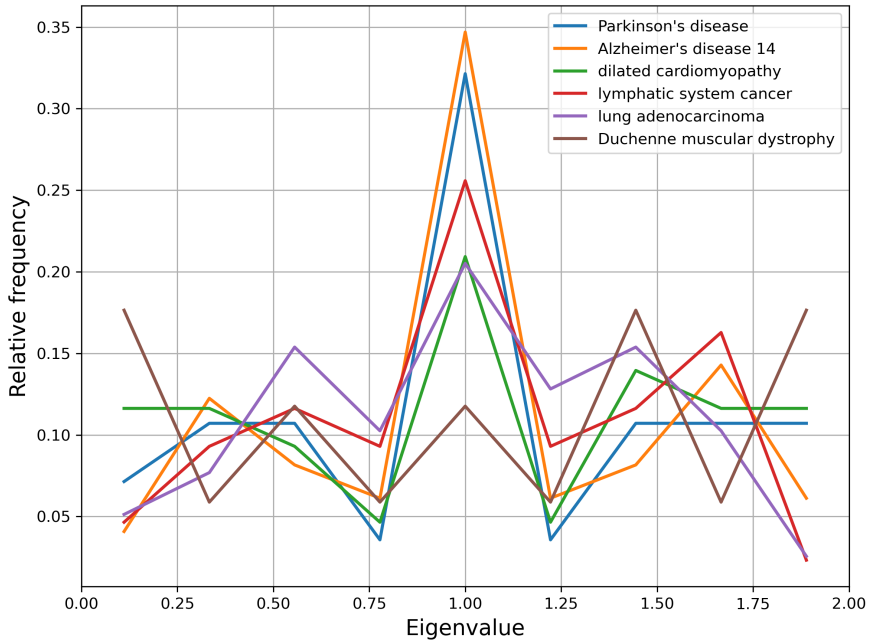


Figure 7.11: Spectra of selected diseases with largest connected component containing between 10 and 50 nodes, and z-score larger than 10

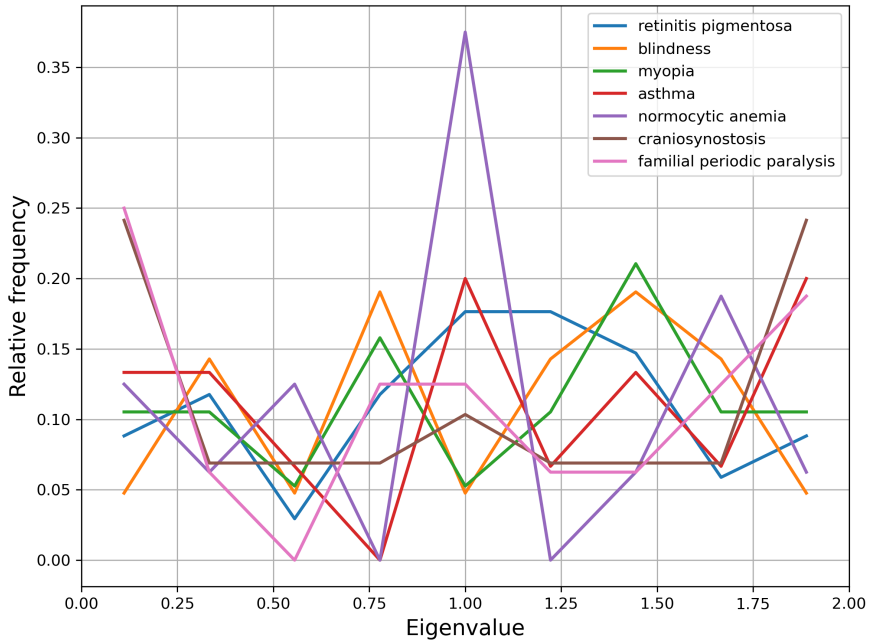
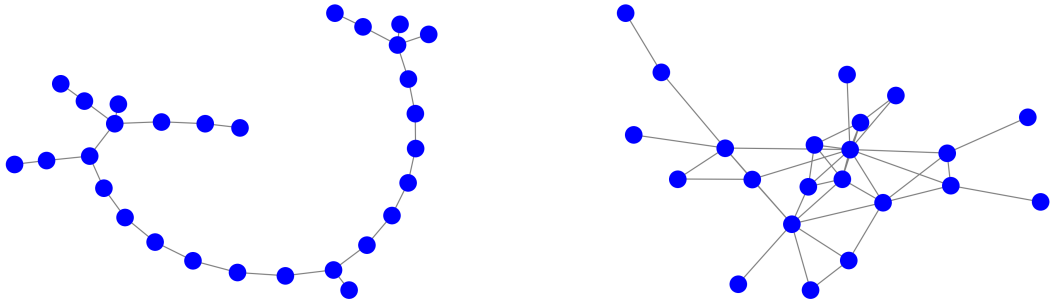


Figure 7.12: Spectra of selected diseases with largest connected component containing between 10 and 50 nodes, and z-score smaller than 10

indeed forms a path-like structure, (its clustering coefficient is 0). In contrast, blindness has many triangle structures (and clustering coefficient 0.4). The important point is that we could already infer these structural properties directly from the spectrum, without visualizing the graphs. This demonstrates how powerful spectral analysis is, especially for larger disease modules that cannot be meaningfully visualized.



(a) lcc of craniosynostosis

(b) lcc of blindness

Overall, across all size regimes, we consistently observe that spectral shapes differ systematically between disease categories. Cancer modules in particular show a characteristic spectral pattern, and even differences such as benign vs malignant neoplasms appear reflected in the eigenvalue distribution. Moreover, important structural properties can be directly derived from the spectra, without the need of visualizing the graphs. This suggests that spectral properties of disease modules capture meaningful structural information that aligns with biological and semantic relationships, and hence, the Laplacian spectrum proves to be a strong tool for the analysis of the diseases.

7.2.1 Spectral reconstruction of disease modules

We have seen that the Laplacian eigenvalues encode important structural properties of the underlying disease graphs. However, computing the full spectrum can be computationally expensive for large graphs, as discussed in Section 4.2.3. This motivates the question of which parts of the spectrum carry the most relevant structural information.

In Section 4.2.2, we described the interpretation of different spectral regimes. Here, we investigate this further by reconstructing disease modules using only subsets of eigenpairs, specifically the smallest and/or largest eigenvalues of the normalized Laplacian and their corresponding eigenvectors. These eigenpairs can be computed relatively fast with the Lanczos method described in Section 4.2.3.

For each disease graph, we consider its largest connected component (lcc) and use k eigenpairs to reconstruct it. We set k proportional to the graph size, specifically $k = \lfloor n/4 \rfloor$, where n is the number of nodes in the lcc. We compare three reconstruction strategies: smallest k eigenpairs, largest k eigenpairs, smallest $k/2$ and largest $k/2$ eigenpairs combined. For each reconstructed graph, we compare its edge set to the

original lcc using the overlap formula

$$\frac{|E_r| \cap |E|}{|E|},$$

where E_r denotes the set of edges of the reconstructed graph and E denotes the set of edges of the original graph.

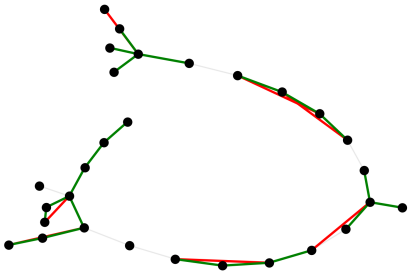
We now reconstruct the graphs analyzed previously. For large disease lccs (with more than 400 nodes), we consistently observe that reconstruction using the largest eigenvalues yields the highest overlap, regardless of disease type. This is on one hand not surprising, since large disease modules contain hub nodes with many attached leaves, which make the graphs have bipartite-like structure. These features are reflected in the high-end eigenvalues of the Laplacian spectrum and the corresponding oscillatory eigenvectors. On the other hand, one might expect that these lccs have high community structure, which would then be encoded in the smallest eigenvalues. This experiment therefore shows that, although community structure may be present, the structural characteristics captured by the largest eigenvalues appear to dominate in these large disease modules. Degree heterogeneity and hub-leaf patterns contribute more strongly to the spectral reconstruction than global organization.

For the medium-sized and smaller lccs, the behavior becomes more dependent on the specific diseases. For example, the eye diseases are reconstructed best using the smallest eigenvalues. On the other hand, the path-like craniosynostosis is best reconstructed using the largest eigenvalues. To illustrate these differences, we plot again the blindness and craniosynostosis lccs on Figure 7.14, showing the reconstructions obtained from the different spectral regimes. However, we omit the mixed (half-half) setting, since overall, it rarely achieves the best performance and is almost always outperformed by the other two cases. The green edges correspond to edges appearing in both the original and reconstructed graphs, red edges are present only in the reconstruction, while light grey edges appear only in the original disease lccs.

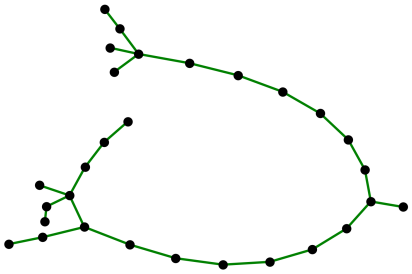
The figure shows that for craniosynostosis, the largest eigenvalues perfectly reconstruct the graph, while reconstruction based on the smallest eigenvalues tries to enforce a community structure, capturing the three more connected regions due to the few larger-degree nodes, and introduces additional edges inside these communities. The enforced segmentation however, disrupts the true organization of the nodes.

For blindness, which essentially consists of one core component containing both hub nodes and leaf nodes, reconstruction based on the first eigenvalues correctly captures this global structure, while it misses many leaf nodes. The largest eigenvalues, on the other hand, capture the leaf nodes well, but connect them to other hub nodes too, which they are not connected to in the original graph. Therefore, the smallest eigenvalues still prove better for the reconstruction of the lcc.

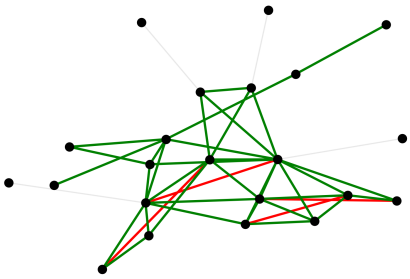
These patterns are in general seen in our experiments and describe our findings well: smallest eigenvalues reliably capture the main dense cores and strongly connected components, but miss many peripheral, or low-degree nodes. Largest eigenvalues better recovered leaf nodes and their connections to hubs, but at the cost of introducing new edges between the leaves and other hub nodes.



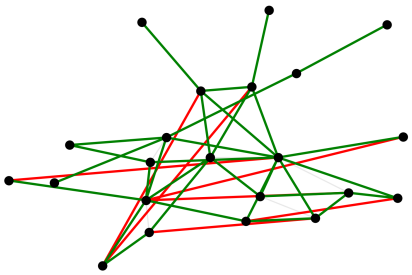
(a) Reconstruction of the lcc of craniosynostosis based on the 7 smallest eigenvalues and corresponding eigenvectors. Overlap 0.75



(b) Reconstruction of the lcc of craniosynostosis based on the 7 largest eigenvalues and corresponding eigenvectors. Overlap 1



(c) Reconstruction of the lcc of blindness based on the 5 smallest eigenvalues and corresponding eigenvectors. Overlap 0.9



(d) Reconstruction of the lcc of blindness based on the 5 largest eigenvalues and corresponding eigenvectors. Overlap 0.82

Figure 7.14: Reconstructions of the craniosynostosis and blindness lccs

We further analyze a subset of 100 diseases with small lccs (between 10 and 50 nodes). Across this subset, in around half of the cases, smallest eigenvalues yield the best reconstruction. In slightly more cases, largest eigenvalues perform best, and only in three diseases does the mixed strategy outperform both. However, when the latter performs best, its advantage from the next best reconstruction is very small. Moreover, in many cases when the largest eigenvalues outperform the smallest ones, the difference is also small. Overall, this suggests that in most diseases, the smallest eigenvalues are a reasonable choice, which capture the frequently occurring community organization of the lccs, but large eigenvalues also prove powerful in the case of more homogeneous networks.

We also observe some biologically meaningful trends. Disease modules best reconstructed with smallest eigenvalues are cancers/neoplasm and blood disorders. These modules tend to exhibit relatively strong global connectivity. On the other hand, modules best reconstructed with largest eigenvalues are cardiomyopathies and many neurological disorders. However, there are many overlaps between the two categories, for example, many cancers are also best reconstructed from the largest eigenpairs, rather than the smallest ones.

The experiment with the smaller graphs support what we observed earlier. Reconstruction based on the smallest eigenvalues performs well when the graph has clear modular substructures or a main connected core. Conversely, largest eigenvalues capture boundary edges and oscillatory behavior, but are also fitting for more homogeneous graphs. However, artificial edges may be introduced between leaf nodes and hub nodes, which makes largest eigenvalues a less appropriate choice for highly heterogenic graphs. We show such an example on the amnestic disorder in Figure 7.15.

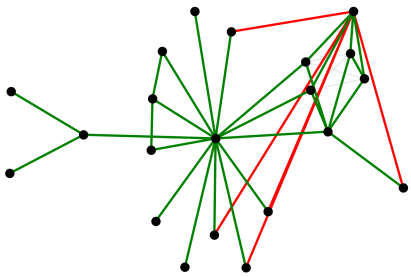


Figure 7.15: Reconstruction of the lcc of amnestic disorder based on the 5 largest eigenvalues and corresponding eigenvectors

Taking both ends of the spectrum usually performs the most poorly. Although the idea behind it was to capture characteristics encoded in both extremes of the spectrum, the mix of fundamentally different structural information cancels out or distorts the other.

Overall, an important finding of the experiments is that across diseases of a wide range of types and sizes, using only $k = n/4$ eigenpairs (where n is the number of nodes in the graph) was sufficient to reconstruct disease lccs with an overlap of around 90%. We also experimented with reducing this number to $k = n/10$, which lowered the overlap to

around 50%.

In future work, a broader range of diseases could be systematically explored, the selection of k should be further investigated, and heuristics for choosing the most informative part of the spectrum based on simple graph properties could be developed.

7.2.2 Clustering cancer diseases

We now consider a different application of the Laplacian spectrum. Rather than analyzing diseases separately, we use spectral information to cluster diseases. More specifically, we investigate how the clustering algorithms described in Section 7.2.2, and most importantly the spectral embedding-based methods perform in clustering the diseases.

To cluster cancer diseases, we take a collection of 64 cancers, or cancer-related conditions. We selected these 64 cancers based on having largest connected components consisting of at least 50 nodes, and restricted our analysis to the largest connected components only. This way, we avoid biasing the clustering algorithms towards the unwanted effects introduced by the disconnected components of the graphs that may give misleading results.

To assess whether structural similarities between disease modules reflect their semantic relatedness, we proceed as follows.

First, since we are not explicitly given a ground truth clustering for cancer diseases, we compute a ground truth clustering in the following way.

We compute the Lin similarity [Lin98] between every pair of the 64 cancers using the hierarchical structure of the disease ontology [SAN+11] downloaded from [lBJ+26]. The formula for the Lin similarity of two diseases c_1 and c_2 is

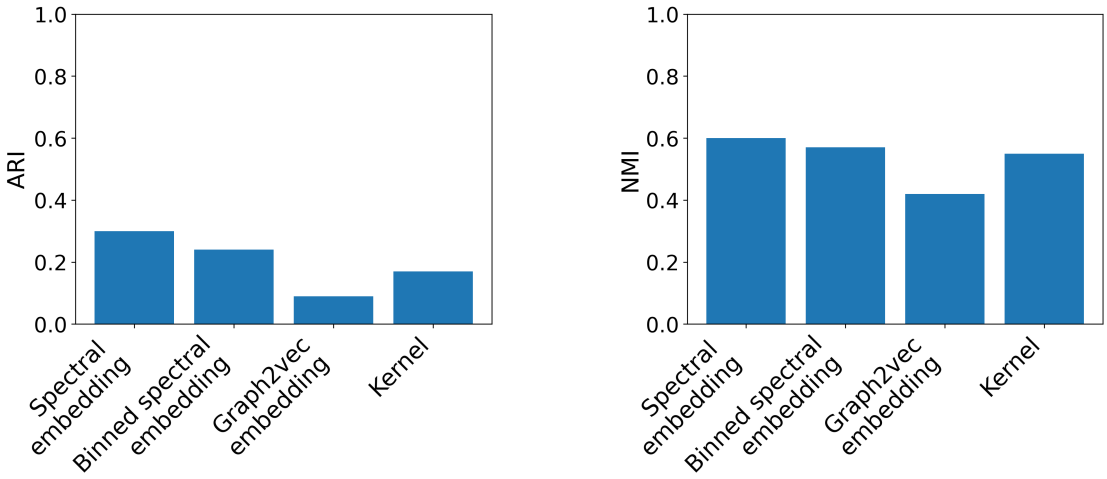
$$Lin(c_1, c_2) = \frac{2IC(MICA(c_1, c_2))}{IC(c_1)_I C(c_2)},$$

where $IC(c) = -\log p(c)$ is the information content of disease c, with $p(c)$ being the fraction of its descendants (including itself) compared to the number of all diseases in the disease ontology, and $MICA(c_1, c_2)$ denotes the most informative common ancestor of the two diseases, i.e. the common ancestor with the highest IC value.

This way, if two diseases have a more specific common ancestor, such as vascular cancer, then their similarity is higher than two diseases that only share a general ancestor, such as cancer.

After computing the pairwise similarities, we cluster the cancers using spectral clustering which directly accepts the similarity matrix S as input. We experiment with different numbers of clusters and select the one that maximizes the silhouette score obtained from the distance matrix $1 - S$. This way, we have 12 clusters, which we can use as a ground truth partition.

Now, we cluster the cancers using the algorithms described in 7.2.2 with the goal to assess whether structural similarities between disease modules reflect their semantic relatedness. For each approach, we vary the tunable parameters for the embeddings/kernels and the clustering algorithm (k-means, hierarchical, or spectral clustering) and report the best ARI–NMI pairs achieved. The results are shown in Figure 7.16.



(a) Comparison of ARI scores

(b) Comparison of NMI scores

Figure 7.16: Comparison of ARI and NMI scores obtained from the different clustering methods

The results show that, as expected, the clustering performance is generally moderate. Good performance for clustering of diseases is challenging due to the complexity and noisiness of the data. ARI scores are consistently lower than NMI scores across all methods. This is not surprising, since ARI measures pairwise similarity, strongly penalizing disagreements and is sensitive to imbalanced clusters. Hence, when clustering diseases, moderate ARI values are expected even when meaningful structure is present. NMI on the other hand measures information overlap, and is more robust to imbalanced and complex data.

Overall, the two spectral embedding-based methods achieve the best performance. In particular, the spectral embedding method performs best with an ARI of 0.3 and an NMI of 0.6, while the binned spectral algorithm is the second best, with ARI 0.23 and NMI 0.57. They are followed by the kernel-based method that achieves ARI 0.17 and NMI 0.55. While the NMI value is just slightly lower, the ARI score difference is bigger, showing that the eigenvalue-based embeddings preserve better pairwise structural relationships aligned with the semantic clustering. Lastly, the Graph2vec embedding performs the worst, with ARI slightly lower than 0.1 and NMI of 0.42.

These results indicate that structural properties of disease modules are indeed related to their semantic organization in the disease ontology. Even though the clusterings are just partly similar, the ARI and NMI scores demonstrate that structural embeddings encode biologically meaningful information, especially in the case of the spectral embedding-based methods. These appear to be useful for capturing ontology-relevant structural patterns and show potential for future research. This is consistent with the fact that spectral representations encode global connectivity patterns of the network, which may reflect underlying biological relations.

Moreover, structural clustering may help in finding relationships that are not captured in the ontology. For example, if two cancers are grouped together by the spectral embedding method but not by semantic labeling, this may indicate shared network-level properties, even if they are semantically different. Such findings could possibly highlight potential targets for drug repurposing, motivating further investigation by domain experts.

We now examine some of these connections in more detail. In [RGK+20], the authors investigated genetic positive and negative correlations between cancers that would not be considered highly similar according to the disease ontology. For example, they reported the highest correlation between esophageal cancer and non-Hodgkin lymphoma. While in our experiments, these two diseases were indeed not grouped together by the disease ontology, the spectral embedding-based clustering put them in the same cluster.

Another study [RGK+20] also identified several negatively correlated disease pairs, such as esophageal cancer and melanoma, lung cancer and melanoma, and non-Hodgkin lymphoma and prostate cancer. These pairs were likewise assigned to different clusters in our results.

Furthermore, [SWY+15] found a positive correlation between bladder and lung cancers. Our clustering algorithm also grouped bladder cancer together with lung cancers. In addition, [LFBS+17] reported that prostate cancer showed no significant correlation with pancreatic, colorectal, or lung cancer, which is consistent with our findings. The same study also identified positive correlations between pancreatic and colorectal cancers, as well as between lung and colorectal cancers. While colorectal cancer was not clustered with these in our results, lung and pancreatic cancers were indeed grouped together.

Altogether, these observations suggest that spectral embedding-based clustering may be capable of identifying biologically meaningful relationships between diseases and it can give some insights into which connections should be analyzed further. Such examples include the potential relationship between melanoma and musculoskeletal system cancer, or central nervous system cancer with breast carcinoma. However, as already emphasized, such findings should always be interpreted with caution and evaluated by domain experts.

Overall, our results have shown that analyzing diseases from a structural perspective complements semantic labeling, partly recovering the ontology-based similarities, while it may also uncover biologically meaningful relationships that are not yet known.

In the next section, we assess in more depth the performance of the spectral embedding-based clustering methods on synthetic and other annotated biological data as well. This will give us an overview on how well the Laplacian spectrum performs in preserving structural or semantic similarities on collections of graphs where a ground truth grouping is already given, and need not be computed by us. Moreover, it may further strengthen the potential of the spectral methods for clustering collections of graphs, a potential that has already been seen in the case of the cancer diseases in comparison to the other state-of-the-art approaches.

Clustering random graphs

We now present experimental results obtained from the different clustering methods on a synthetic dataset. To assess clustering performance, we again plot the ARI and NMI

scores obtained from the different graph clustering methods. Each experiment is repeated 30 times, and the results are averaged over these runs.

We construct a synthetic dataset of 150 graphs, containing 50 Erdős-Rényi graphs with $p = 0.05$, 50 Barabási-Albert graphs with $m = 5$, and 50 Watts-Strogatz graphs with $k = 4$ and $p = 0.3$. Each graph has 30 nodes. Naturally, we set the number of clusters to 3. We begin with the examination of the different configurations for the two spectral embedding-based clustering methods. This is followed by a comparison of these methods with the state-of-the-art approaches discussed previously, namely Graph2vec-based clustering and kernel-based clustering.

Figure 7.17 shows the eigenvalues of one representative graph of each graph type. The figure clearly illustrates that the smallest eigenvalues differ substantially across graphs, whereas the largest eigenvalues are relatively similar. Hence, we expect clustering methods based on the smallest eigenvalues to yield significantly better separation of the graph types compared to clustering based on the largest eigenvalues.

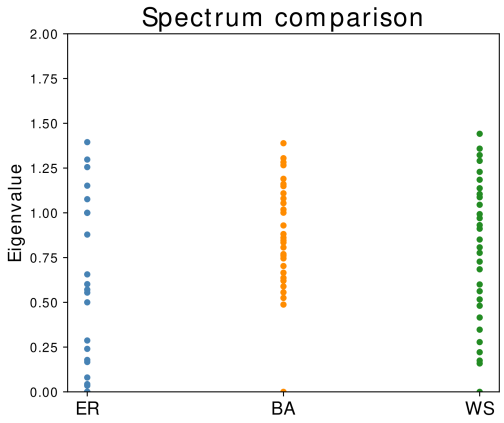
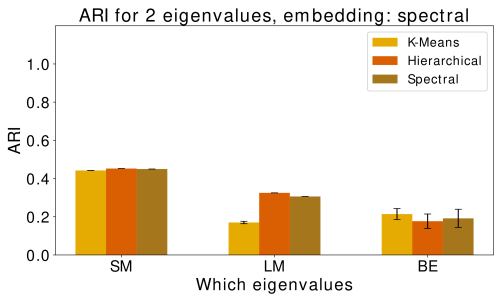
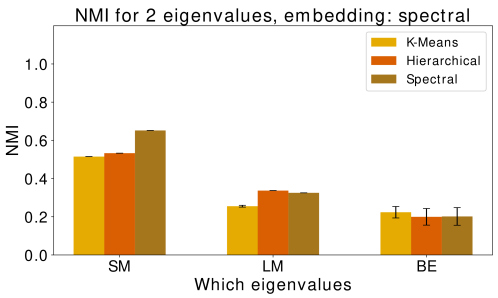


Figure 7.17: Eigenvalues of an ER graph with with $p = 0.05$, a BA graph with $m = 5$, and a WS graph with $k = 4$ and $p = 0.3$

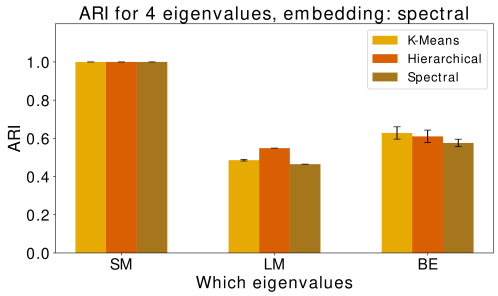
Spectral embedding: We first analyze the results obtained with spectral embedding. Embedding each graph to the vector of all 30 eigenvalues results in a perfect clustering, with ARI and NMI 1. In this example, computing the whole spectrum of the graphs is very fast, hence there is no need to limit to smaller k. Still, we plot the results for $k = 2$ and $k = 4$, to underline the power of the spectral embedding and to confirm earlier observation that the smallest eigenvalues lead to better clustering performance. Figure 7.18 shows that already for $k = 2$, clustering based on the smallest eigenvalues achieves ARI and NMI scores of around 0.5, hence it correctly guesses the labels of around half of the graphs. Moreover, with $k = 4$, a perfect clustering is achieved with kmeans, hierarchical and spectral clustering as well. The other methods (LM and BE) also perform surprisingly well, considering that only four eigenvalues are used out of the total 30.



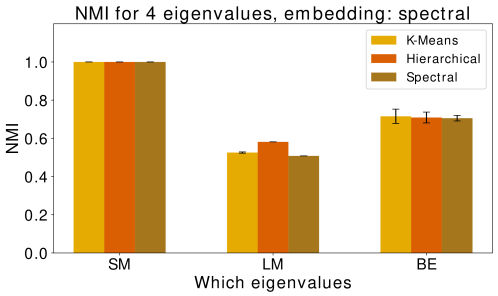
(a) ARI scores obtained from clustering based on spectral embedding with $k = 2$ eigenvalues



(b) NMI scores obtained from clustering based on spectral embedding with $k = 2$ eigenvalues



(c) ARI scores obtained from clustering based on spectral embedding with $k = 4$ eigenvalues



(d) NMI scores obtained from clustering based on spectral embedding with $k = 4$ eigenvalues

Figure 7.18: ARI and NMI scores obtained from clustering with spectral embedding using either two or four eigenvalues

An interesting observation arises in the case $k = 2$. When selecting both the smallest and the largest eigenvalue (BE), the clustering performance is inferior compared to using either the two smallest (SM) or the two largest (LM) eigenvalues. This suggests that the second smallest and second largest eigenvalues play a more significant role in determining clustering quality. This behavior is expected, as the smallest eigenvalue is always 0 and therefore does not provide discriminative information. Hence, in the BE setting, only the largest eigenvalue contributes meaningful variation across graphs. Similarly, in the SM setting, the second smallest eigenvalue varies and thus carries informative structural content. The results also indicate that this second smallest eigenvalue (Fiedler value) used for SM is more informative for clustering than the largest eigenvalue used for BE.

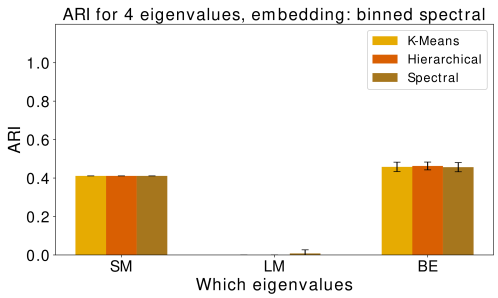
However, for $k = 4$, the BE configuration includes the two smallest and the two largest eigenvalues and, interestingly, its performance is superior to that of the LM configuration for $k = 4$. This means that including the first two eigenvalues (particularly the Fiedler value, as the first one is uninformative), provides more meaningful insight than including the next two largest eigenvalues. These results are consistent with the patterns observed in Figure 7.17.

Binned spectral embedding: Next, we perform a similar analysis for the binned spectral embedding. In this case, using only two eigenvalues is not meaningful, as binning the $[0, 2]$ interval into just two bins fails to capture sufficient information. Figure 7.19 demonstrates that using $k = 4$ eigenvalues already produces meaningful results for SM and BE. In contrast, when clustering is performed based on the largest eigenvalues (LM), $k = 4$ remains insufficient. For $k = 10$ eigenvalues, clustering using the smallest ten eigenvalues (SM) achieves ARI and NMI scores close to one, while the other two methods (LM and BE) also yield reasonably good results. The exception is spectral clustering, which performs poorly for LM and BE in this setting.

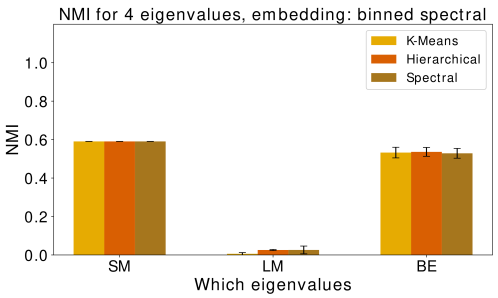
We note that using all 30 eigenvalues yields perfect clustering, with ARI and NMI scores of 1.

Compared to the spectral embedding, it is evident that the binned spectral embedding requires more eigenvalues to achieve comparable results. This is expected, as the number of eigenvalues also determines the number of bins, and too few bins lead to a loss of information.

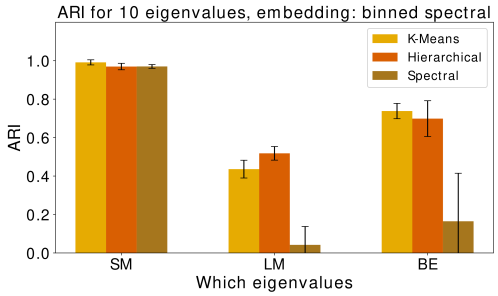
Comparison with state-of-the-art methods: We now proceed to compare these results with the state-of-the-art methods. For the two approaches based on spectral embedding, we report results obtained using all 30 eigenvalues, as computing all of the eigenvalues is very fast for this small dataset. Moreover, for each method, we only plot the highest ARI and NMI score, regardless of the clustering (kmeans, hierarchical, spectral) used. Figure 7.20 shows the obtained results. It is clear that the two spectral embedding-based methods are highly competitive. The runtimes of most methods are quite similar, with the average runtime over 30 runs remaining under one-third of a second for all methods, except for the graphlet kernel-based clustering, which required an average of 46 seconds per run. This experiment provides a solid baseline, confirming that the spectral embedding methods perform well, particularly in comparison to other embedding approach. However, further experiments are necessary to truly evaluate the performance



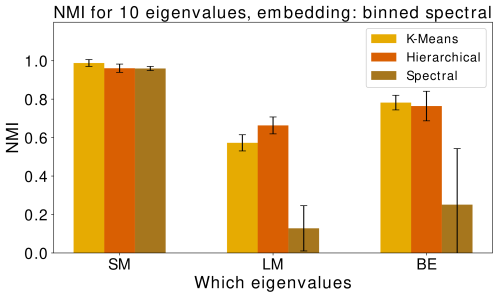
(a) ARI scores obtained from clustering based on binned spectral embedding with $k = 4$ eigenvalues



(b) NMI scores obtained from clustering based on binned spectral embedding with $k = 4$ eigenvalues



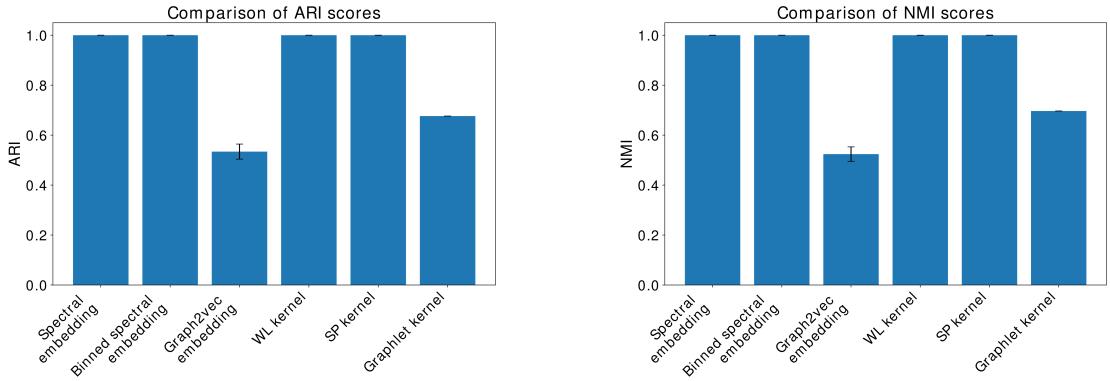
(c) ARI scores obtained from clustering based on binned spectral embedding with $k = 10$ eigenvalues



(d) NMI scores obtained from clustering based on binned spectral embedding with $k = 10$ eigenvalues

Figure 7.19: ARI and NMI scores obtained from clustering with binned spectral embedding using either two or four eigenvalues

of these methods. In this case, the graph classes were relatively easy to separate, but in real-world applications, such clear separations are unlikely.



(a) Comparison of ARI scores

(b) Comparison of NMI scores

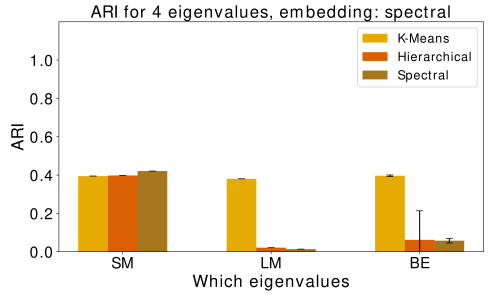
Figure 7.20: Comparison of ARI and NMI scores obtained from the different clustering methods

Clustering other biological networks

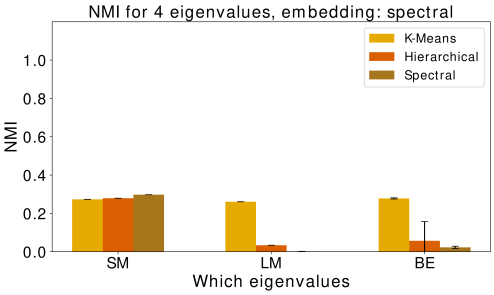
We now extend our study to real-world datasets. To this end, we consider graph collections from [MKB+20], which contain datasets from various domains, including biology, social networks, and computer vision. Each dataset contains a collection of graphs together with associated class labels, which serve as ground truth for evaluating clustering performance. In the following, we focus on graph datasets from biological applications. The primary objective of our experiments is to investigate whether the provided annotations, i.e., the class labels assigned to the graphs, correspond to clusters that can be recovered from structural similarities. Biological networks are inherently complex and highly heterogeneous, and it is by no means guaranteed that structural similarity aligns with semantic labeling. Hence, we cannot expect a strong clustering performance with respect to the ground truth labels.

The MUTAG dataset: We start by analyzing the MUTAG dataset, a benchmark collection of 188 graphs representing nitroaromatic chemical compounds. In each graph, nodes correspond to atoms (e.g., carbon, nitrogen, oxygen) and each graph is labeled according to its mutagenicity, indicating whether it is likely to cause DNA mutations in Salmonella typhimurium or not. The dataset contains graphs whose number of nodes ranges between 10 and 28, with an average of 17.93 nodes per graph. The average number of edges is 37.72. Naturally, we cluster the collection of graphs into two clusters. Each experiment is conducted 30 times, with results averaged over these runs.

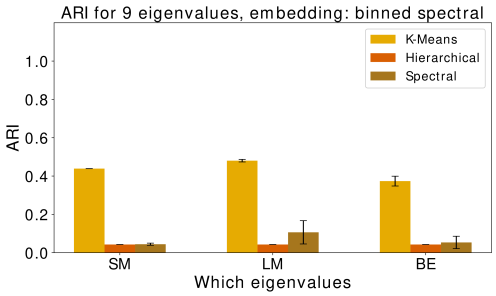
Figures 7.21a and 7.21b show the ARI and NMI scores obtained from clustering based on the spectral embeddings of the graphs, taking either the smallest four (SM), the largest four (LM), or the smallest two and largest two (BE) eigenvalues. Figures 7.21c and 7.21d



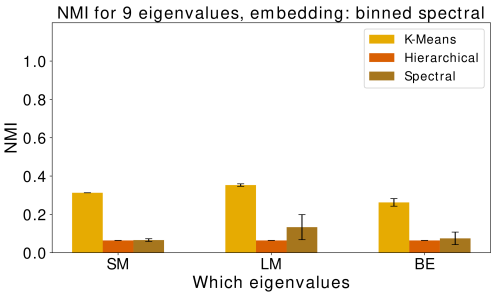
(a) ARI scores obtained from clustering based on spectral embedding with $k = 4$ eigenvalues



(b) NMI scores obtained from clustering based on spectral embedding with $k = 4$ eigenvalues



(c) ARI scores obtained from clustering based on binned spectral embedding with $k = 9$ eigenvalues



(d) NMI scores obtained from clustering based on binned spectral embedding with $k = 9$ eigenvalues

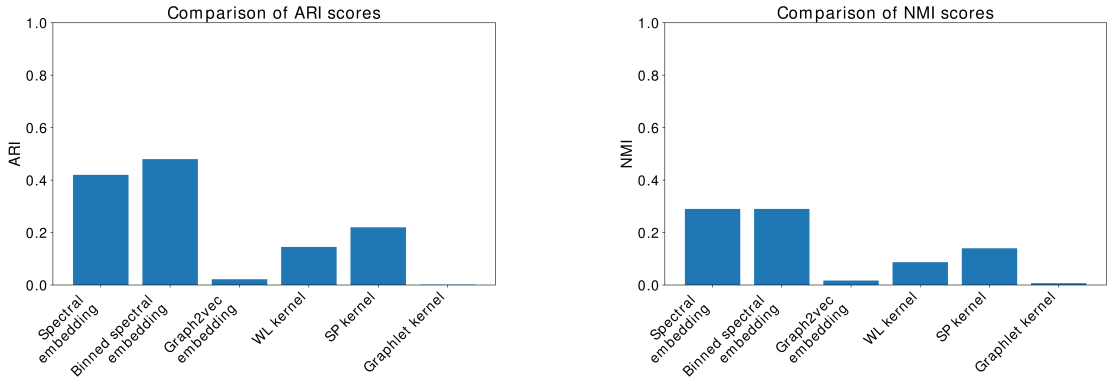
Figure 7.21: ARI and NMI scores obtained from clustering with spectral or binned spectral embedding using four and nine eigenvalues respectively

show the same results for binned spectral embedding and 9 eigenvalues. From all the plots, it is clear that overall, kmeans clustering produces the best results.

We also tried increasing the number of eigenvalues used for the spectral embedding to more than four, which resulted only in a slight improvement in ARI and NMI scores, remaining below 0.5. This indicates that, although additional spectral information slightly improves performance, the rate of improvement decreases substantially as more eigenvalues are incorporated.

For the binned spectral embedding, a different behavior was observed. Using significantly more or less than nine eigenvalues led to a decrease in performance. Thus, around nine eigenvalues appear to represent a peak configuration for this embedding approach. This number is close to the smallest number of nodes among the graphs in the dataset.

We now compare these results with those obtained using the other clustering methods. In Figure 7.22, we report only the best performance achieved by each method, without distinguishing between k-means, hierarchical, and spectral clustering. In general, the scores are not very high, nevertheless, this is expected for real-world graphs, which are typically not well separable. The figure, however, clearly shows the superiority of the two spectral embedding methods. Clustering based on the graphlet kernel and on Graph2vec embedding is close to random, while the other two kernel-based methods perform quite poorly as well. Note that graph kernels and the Graph2Vec embedding can incorporate node labels if given. Since the MUTAG dataset provides node labels, these methods could, in principle, have an advantage over the spectral approaches. However, this is not the case here, the spectral embedding-based methods still achieve better clustering performance.



(a) Comparison of ARI scores

(b) Comparison of NMI scores

Figure 7.22: Comparison of ARI and NMI scores obtained from the different clustering methods

Other biological datasets: In the following, we performed similar experiments on several other graph collections from the TU Dataset. The results were usually poor, with ARI and NMI scores close to zero for all methods, including the spectral embedding approaches. These findings highlight a key limitation: while structural similarity can be informative, it is not always sufficient for unsupervised clustering of complex biological

graphs. In contrast to the synthetic dataset, where clustering methods recovered well the meaningful groups, performance on some real-world biological datasets is poor with respect to the ground truth labels. This is due to the fact that class labels in biological applications often reflect functional properties, biochemical characteristics, or semantic similarity, that are not encoded in the graphs' topology. Hence, clustering methods based on structural properties cannot always recover the true labels of the graphs.

However, as observed in both the cancer disease clustering and the MUTAG dataset clustering experiments, the spectral and binned spectral embedding-based clustering methods outperformed the other state-of-the-art approaches. This means that the structural characteristics encoded in the Laplacian spectrum are the most closely related to the semantic similarity of the graphs. For future research directions, these results clearly motivate the potential use of the spectral methods also in clustering problems where the ground truth labels are not provided.

8 Conclusion

In this thesis, we bridged spectral graph theory with biological network analysis by investigating the potential of the symmetric normalized Laplacian matrix and its eigenvalues and eigenvectors in the study of biological networks, in particular the human protein-protein interaction (PPI) network.

We began by collecting known theoretical properties of the Laplacian matrix and analyzing random graph models in order to better understand how different underlying structural properties of graphs are reflected in the Laplacian spectrum.

We then applied eigenvector localization to analyze protein complexes and proposed a novel localization-based approach for predicting new complexes. We assessed the performance of the new approach by comparing it to existing methods, and discussed the high practicality of it.

Afterwards, we examined disease modules from a spectral perspective and investigated to what extent these modules can be reconstructed from the Laplacian eigenvalues and eigenvectors.

Finally, we introduced two Laplacian eigenvalue embedding-based clustering approaches for grouping graphs and applied them to cluster 64 cancer-related diseases. We evaluated their performance against state-of-the-art clustering algorithms on the cancer dataset, as well as on synthetic and other biological datasets. The two spectral embedding-based methods showed superior performance.

Overall, this thesis aimed to uncover how the structural information encoded in the Laplacian matrix can be used to better understand protein interactions and disease mechanisms. The results consistently demonstrated the effectiveness of Laplacian-based methods and provided strong evidence that the Laplacian matrix, its eigenvalues and eigenvectors capture biologically meaningful information.

Since this thesis approached the problem mainly from a mathematical perspective, a further direction is collaboration with domain experts to interpret the findings in a more biologically meaningful way and to formulate additional biologically relevant questions for further investigation.

While there are several directions for future research, many of which outlined throughout the thesis, the present work already clearly demonstrates the significant potential of the Laplacian matrix in bridging structural graph properties with biological characteristics, specifically on the PPI network.

Bibliography

[AV07] David Arthur and Sergei Vassilvitskii. k-means++: the advantages of careful seeding. In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '07, page 1027–1035, USA, 2007. Society for Industrial and Applied Mathematics.

[BA99] Albert-Laszlo Barabasi and Reka Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, October 1999.

[Ban08] Anirban Banerjee. *The spectrum of the graph Laplacian as a tool for analyzing structure and evolution of networks.* PhD thesis, Verlag nicht ermittelbar, 2008.

[Ban12] Anirban Banerjee. Structural distance and evolutionary relationship of networks. *Biosystems*, 107(3):186–196, 2012.

[BC13] Steve Butler and Fan Chung. *Spectral Graph Theory*, pages 811–824. 12 2013.

[BDVRT15] Rodrigo O. Braga, Renata R. Del-Vecchio, Virgínia M. Rodrigues, and Vilmar Trevisan. Trees with 4 or 5 distinct normalized laplacian eigenvalues. *Linear Algebra and its Applications*, 471:615–635, 2015.

[BG12] Steve Butler and Jason Grout. A construction of cospectral graphs for the normalized laplacian, 2012.

[BH03] Gary D. Bader and Christopher W. V. Hogue. An automated method for finding molecular complexes in large protein interaction networks. *BMC Bioinformatics*, 4(1):2, 2003.

[BJ09] Anirban Banerjee and Jürgen Jost. Graph spectra as a systematic tool in computational biology. *Discrete Applied Mathematics*, 157(10):2425–2431, 2009. Networks in Computational Biology.

[BK05] Karsten M. Borgwardt and Hans-Peter Kriegel. Shortest-path kernels on graphs. In *Proceedings of the Fifth IEEE International Conference on Data Mining*, ICDM '05, page 74–81, USA, 2005. IEEE Computer Society.

[But16] Steve Butler. *Algebraic aspects of the normalized Laplacian*, pages 295–315. Springer International Publishing, Cham, 2016.

[Chu97] Fan RK Chung. *Spectral graph theory*, volume 92. American Mathematical Soc., 1997.

[CL04] Fan Chung and Linyuan Lu. Complex graphs and networks. *American Mathematical Society, Providence*, 01 2004.

[CLV03] Fan Chung, Linyuan Lu, and Van Vu. The spectra of random graphs with given expected degrees. *Proceedings of the National Academy of Sciences of the United States of America*, 100:6313–8, 06 2003.

[CM11] Mihai Cucuringu and Michael W. Mahoney. Localization on low-order eigenvectors of data matrices. *ArXiv*, abs/1109.1355, 2011.

[CO07] Amin Coja-Oghlan. On the laplacian eigenvalues of gn,p. *Combinatorics, Probability and Computing*, 16(6):923–946, 2007.

[CT16] Chen Chen and Hanghang Tong. On the eigen-functions of dynamic graphs: Fast tracking and attribution algorithms: Eigen-functions of dynamic graphs. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 10, 06 2016.

[CYP24] Tengjie Chen, Zhenhua Yuan, and Junhao Peng. The normalized laplacian spectrum of n-polygon graphs and applications. *Linear and Multilinear Algebra*, 72(2):234–260, 2024.

[DM03] Sergey Dorogovtsev and José Fernando Mendes. *Evolution of Networks: From Biological Nets to the Internet and WWW*, volume 57. 03 2003.

[ER59] P Erdös and A Rényi. On random graphs i. *Publicationes Mathematicae Debrecen*, 6:290–297, 1959.

[EVDO02] A. J. Enright, S. Van Dongen, and C. A. Ouzounis. An efficient algorithm for large-scale detection of protein families. *Nucleic Acids Research*, 30(7):1575–1584, 04 2002.

[FHU+08] Damien Fay, Hamed Haddadi, Steve Uhlig, Andrew Moore, Richard Mortier, and Almerima Jamakovic. Weighted spectral distribution. *University of Cambridge, Computer Laboratory, Tech. Rep. UCAM-CL-TR-729, Sep*, 01 2008.

[HBP+17] Edward L. Huttlin, Raphael J. Bruckner, Joao A. Paulo, Joe R. Cannon, Lily Ting, Kurt Baltier, Greg Colby, Fana Gebreab, Melanie P. Gygi, Hannah Parzen, John Szpyt, Stanley Tam, Gabriela Zarraga, Laura Pontano-Vaites, Sharan Swarup, Anne E. White, Devin K. Schweppe, Ramin Rad, Brian K. Erickson, Robert A. Obar, K. G. Guruharsha, Kejie Li, Spyros Artavanis-Tsakonas, Steven P. Gygi, and J. Wade Harper. Architecture of the human interactome defines protein communities and disease networks. *Nature*, 545(7655):505–509, 2017.

[HMvdW+20] Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with NumPy. *Nature*, 585(7825):357–362, September 2020.

[Lan50] Cornelius Lanczos. An iteration method for the solution of the eigenvalue problem of linear differential and integral operators. *J. Res. Natl. Bur. Stand. B*, 45:255–282, 1950.

[lBJ+26] lschriml, J. Allen Baron, Claudia Marie Sánchez-Beato Johnson, jbmunro, Elvira, Melody Swen Sue Bello, Anu, Becky Jackson, andrawaag, Chris Mungall, and James A. Overton. Diseaseontology/humandiseaseontology: March 2026 release (v2026-03-31). Zenodo, 2026.

[LFBS+17] Sara Lindström, Hilary Finucane, Brendan Bulik-Sullivan, Fredrick R. Schumacher, Christopher I. Amos, Rayjean J. Hung, Kristin Rand, Stephen B. Gruber, David Conti, Joanna B. Permuth, Hui-Yi Lin, Ellen L. Goode, Thomas A. Sellers, Laufey T. Amundadottir, Rachael Z. Stolzenberg-Solomon, Alison P. Klein, Gloria M. Petersen, Harvey A. Risch, Brian M. Wolpin, Li Hsu, Jeroen R. Huyghe, Jenny Chang-Claude, Andrew Chan, Sonja I. Berndt, Rosalind Eeles, Douglas Easton, Christopher A. Haiman, David J. Hunter, Benjamin Neale, Alkes L. Price, Peter Kraft, PanScan, GECCO, and GAME-ON Network. Quantifying the genetic correlation between multiple cancer types. *Cancer Epidemiology, Biomarkers & Prevention*, 26(9):1427–1435, 2017. Epub 2017 Jun 21.

[Lin98] Dekang Lin. An information-theoretic definition of similarity. In *Proceedings of the Fifteenth International Conference on Machine Learning*, ICML '98, page 296–304, San Francisco, CA, USA, 1998. Morgan Kaufmann Publishers Inc.

[LKL+19] Katja Luck, Dae-Kyum Kim, Luke Lambourne, Kerstin Spirohn, Bridget E. Begg, Wenting Bian, Ruth Brignall, Tiziana Cafarelli, Francisco J. Campos-Laborie, Benoit Charloteaux, Dongsic Choi, Atina G. Cote, Meaghan Daley, Steven Deimling, Alice Desbuleux, Amélie Dricot, Marinella Gebbia, Madeleine F. Hardy, Nishka Kishore, Jennifer J. Knapp, István A. Kovács, Irma Lemmens, Miles W. Mee, Joseph C. Mellor, Carl Pollis, Carles Pons, Aaron D. Richardson, Sadie Schlabach, Bridget Teeking, Anupama Yadav, Mariana Babor, Dawit Balcha, Omer Basha, Suet-Feung Chin, Soon Gang Choi, Claudia Colabella, Georges Coppin, Cassandra D'Amata, David De Ridder, Steffi De Rouck, Miquel Duran-Frigola, Hanane Ennajdaoui, Florian Goebels, Anjali Gopal, Ghazal Haddad, Mohamed Helmy,

Yves Jacob, Yoseph Kassa, Roujia Li, Natascha van Lieshout, Andrew MacWilliams, Dylan Markey, Joseph N. Paulson, Sudharshan Rangarajan, John Rasla, Ashyad Rayhan, Thomas Rolland, Adriana San Miguel, Yun Shen, Dayag Sheykhkarimli, Gloria M. Sheynkman, Eyal Simonovsky, Murat Taşan, Alexander Tejeda, Jean-Claude Twizere, Yang Wang, Robert Weatheritt, Jochen Weile, Yu Xia, Xinping Yang, Esti Yeger-Lotem, Quan Zhong, Patrick Aloy, Gary D. Bader, Javier De Las Rivas, Suzanne Gaudet, Tong Hao, Janusz Rak, Jan Tavernier, Vincent Tropepe, David E. Hill, Marc Vidal, Frederick P. Roth, and Michael A. Calderwood. A reference map of the human protein interactome. *bioRxiv*, 2019.

[LM15] Eric Lewitus and Helene Morlon. Characterizing and comparing phylogenies from their laplacian spectrum. *Systematic Biology*, 65(3):495–507, 12 2015.

[LRH14] Siemon Lange, Marcel Reus, and Martijn Heuvel. The laplacian spectrum of neural networks. *Frontiers in computational neuroscience*, 7:189, 01 2014.

[Lux04] Ulrike Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17:395–416, 01 2004.

[MCCD13] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space, 2013.

[Mil67] Stanley Milgram. The Small-World Problem. *Psychology Today*, 1(1):61–67, 1967.

[MKB+20] Christopher Morris, Nils M. Kriege, Franka Bause, Kristian Kersting, Petra Mutzel, and Marion Neumann. Tudataset: A collection of benchmark datasets for learning with graphs. In *ICML 2020 Workshop on Graph Representation Learning and Beyond (GRL+ 2020)*, 2020.

[MRSG17] Enrico Maiorino, Antonello Rizzi, Alireza Sadeghian, and Alessandro Giuliani. Spectral reconstruction of protein contact networks. *Physica A: Statistical Mechanics and its Applications*, 471:804–817, 2017.

[MSC+13] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality, 2013.

[NCV+17] Annamalai Narayanan, Mahinthan Chandramohan, Rajasekar Venkatesan, Lihui Chen, Yang Liu, and Shantanu Jaiswal. graph2vec: Learning distributed representations of graphs, 2017.

[NJW01] Andrew Ng, Michael Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. In T. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems*, volume 14. MIT Press, 2001.

[NYP12] Tamás Nepusz, Haiyuan Yu, and Alberto Paccanaro. Detecting overlapping protein complexes in protein-protein interaction networks. *Nature Methods*, 9(5):471–472, 2012.

[OAA+14] Sandra Orchard, Mais Ammari, Bruno Aranda, Lionel Breuza, Leonardo Briganti, Fiona Broackes-Carter, Nancy H. Campbell, Gayatri Chavali, Carol Chen, Noemi del Toro, Margaret Duesbury, Marine Dumousseau, Eugenia Galeota, Ursula Hinz, Marta Iannuccelli, Sruthi Jagannathan, Rafael Jimenez, Jyoti Khadake, Astrid Lagreid, Luana Licata, Ruth C. Lovering, Birgit Meldal, Anna N. Melidoni, Mila Milagros, Daniele Peluso, Livia Perfetto, Pablo Porras, Arathi Raghunath, Sylvie Ricard-Blum, Bernd Roechert, Andre Stutz, Michael Tognolli, Kim van Roey, Gianni Cesareni, and Henning Hermjakob. The mintact project—intact as a common curation platform for 11 molecular interaction databases. *Nucleic Acids Research*, 42(D1):D358–D363, 01 2014.

[PCR+26] Janet Piñero, Javier Corvi, Natalia Rykova, Anna Guillem, Amelia Martínez, Jaione Telleria Zufiaur, Ivo Rivetta, Sarah Holmes, Mariandrea Del Boccio, Daniel Shapoval, Macarena De Sarasqueta Szneiderowicz, Felix Slager, Ferran Sanz, and Laura I. Furlong. Disgenet: Accelerating data-driven discovery in disease genomics and therapeutic development. *bioRxiv*, 2026.

[Pin19] Edouard Pineau. Using laplacian spectrum as graph feature representation, 2019.

[PJV13] Victor M. Preciado, Ali Jadbabaie, and George C. Verghese. Structural analysis of laplacian spectral properties of large-scale networks. *IEEE Transactions on Automatic Control*, 58(9):2338–2343, September 2013.

[PVG+11] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[RBD+08] Andreas Ruepp, Barbara Brauner, Irmtraud Dunger, Goar Frishman, Corinna Montrone, Michael Stransky, Brigitte Waegele, Thorsten Schmidt, Octave Doudieu, Volker Stümpflen, and Hans-Werner Mewes. Corum: The comprehensive resource of mammalian protein complexes. *Nucleic acids research*, 36:D646–50, 02 2008.

[RGK+20] Sara Rashkin, Rebecca Graff, Linda Kachuri, Khanh Thai, Stacey Alexeeff, Maruta Blatchins, Taylor Cavazos, Douglas Corley, Nima Emami, Joshua Hoffman, Eric Jorgenson, Lawrence Kushi, Travis Meyers, Stephen Van Den Eeden, Elad Ziv, Laurel Habel, Thomas Hoffmann, Lori Sakoda, and

John Witte. Pan-cancer study detects genetic risk variants and shared genetic basis in two large cohorts. *Nature Communications*, 11:4423, 09 2020.

[RKS20] Benedek Rozemberczki, Oliver Kiss, and Rik Sarkar. Karate club: An api oriented open-source python framework for unsupervised learning on graphs. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, CIKM '20, page 3125–3132, New York, NY, USA, 2020. Association for Computing Machinery.

[Rou87] Peter J. Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987.

[SAN+11] Lynn M. Schriml, Cesar Arze, Suvarna Nadendla, Yu Chang, Mark Mazaitis, Victor Felix, Gang Feng, and W. Kibbe. Disease ontology: a backbone for disease semantic integration. *Nucleic Acids Research*, 40:D940 – D946, 2011.

[SBR+06] Chris Stark, Bobby-Joe Breitkreutz, Teresa Reguly, Lorrie Boucher, Ashton Breitkreutz, and Michael Tyers. Biogrid: a general repository for interaction datasets. *Nucleic Acids Research*, 34(Database issue):D535–D539, 2006.

[SD20] Shaowei Sun and Kinkar Chandra Das. Normalized laplacian spectrum of complete multipartite graphs. *Discrete Applied Mathematics*, 284:234–245, 2020.

[SK10] FRANTIŠEK SLANINA and ZDENĚK KONOPÁSEK. Eigenvector localization as a tool to study small communities in online social networks. *Advances in Complex Systems*, 13(06):699–723, December 2010.

[SNL+20] Giannis Siglidis, Giannis Nikolentzos, Stratis Limnios, Christos Giatsidis, Konstantinos Skianis, and Michalis Vazirgiannis. Grakel: a graph kernel library in python. *J. Mach. Learn. Res.*, 21(1), January 2020.

[SSJ+10] Nino Shervashidze, Pascal Schweitzer, Erik Jan, Van Leeuwen, Kurt Mehlhorn, and Karsten Borgwardt. Weisfeiler-lehman graph kernels. *Journal of Machine Learning Research*, 1:1–48, 01 2010.

[Str06] Gilbert Strang. *Linear algebra and its applications*. Thomson, Brooks/Cole, Belmont, CA, 2006.

[SVP+09] Nino Shervashidze, SVN Vishwanathan, Tobias Petri, Kurt Mehlhorn, and Karsten Borgwardt. Efficient graphlet kernels for large graph comparison. In David van Dyk and Max Welling, editors, *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics*, volume 5

of *Proceedings of Machine Learning Research*, pages 488–495, Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA, 16–18 Apr 2009. PMLR.

[SWY+15] J. N. Sampson, W. A. Wheeler, M. Yeager, O. Panagiotou, Z. Wang, et al. Analysis of heritability and shared heritability based on genome-wide association studies for thirteen cancer types. *Journal of the National Cancer Institute*, 107(12):djv279, 2015. Erratum published in Journal of the National Cancer Institute, 2016, 108(4): djw106.

[UB18] Shashankaditya Upadhyay and Sudeepto Bhattacharya. A comparative study of ecological networks using spectral projections of normalized graph laplacian, 11 2018.

[VGO+20] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020.

[VS20] Renny P Varghese and D Susha. On the normalized laplacian spectrum of some graphs. *Kragujevac Journal of Mathematics*, 44(3):431–442, 2020.

[Wig57] Eugene P. Wigner. Characteristics vectors of bordered matrices with infinite dimensions ii. *Annals of Mathematics*, 65(2):203–207, 1957.

[WLKN09] Min Wu, Xiaoli Li, Chee Keong Kwoh, and See-Kiong Ng. A core-attachment based method to detect protein complexes in ppi networks. *BMC Bioinformatics*, 10:169 – 169, 2009.

[WMDP19] Sam F L Windels, Noël Malod-Dognin, and Nataša Pržulj. Graphlet laplacians for topology-function and topology-disease relationships. *Bioinformatics*, 35(24):5226–5234, 06 2019.

[WS98] Duncan J. Watts and Steven H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393:440–442, 1998.

[XZC16] Pinchen Xie, Zhongzhi Zhang, and Francesc Comellas. On the spectrum of the normalized laplacian of iterated triangulations of graphs. *Applied Mathematics and Computation*, 273:1123–1129, 2016.