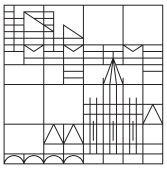
Analysis of Network Ensembles

Dissertation zur Erlangung des
akademischen Grades eines Doktors der Naturwissenschaften

vorgelegt von
Uwe Nagel

an der

Universität
Konstanz



Mathematisch- Naturwissenschaftliche Sektion

Fachbereich Informatik und Informationswissenschaft

Tag der mündlichen Prüfung: 7. Dezember 2011

1. **Referent: Prof. Dr. Ulrik Brandes**
2. **Referent: Prof. Dr. Michael R. Berthold**

Deutsche Zusammenfassung

Graphen bieten einen Beschreibungsformalismus der sonst üblichen Beschreibungen durch Vektoren oder Statistiken in einigen Anwendung überlegen ist, da die Ausdrucksmöglichkeiten im Vergleich deutlich größer sind. Speziell relationale Zusammenhänge finden geeignete Beschreibung als Graphen oder Netzwerke, teilweise mit zusätzlichen Attributen, die Knoten oder Kanten detaillierter beschreiben.

Die Nutzung dieses Formalismus führt für eine Anzahl von beschriebenen Objekten zu einer Menge von Graphen oder Netzwerken, die hier als *Ensemble* bezeichnet werden. Um mit Hilfe von Data-Mining Algorithmen Strukturen in solchen Objektsammlungen zu finden oder Klassifikationen zu lernen, ist es notwendig eine Vergleichbarkeit der Graphen bzw. Netzwerke herzustellen.

Ein Beispiel ist die Vorhersage von mutationsauslösenden Eigenschaften bei chemischen Substanzen: Ausgehend von einer Menge von Molekülen, bei denen bekannt ist, ob sie Mutationen in Zellen auslösen, soll für neue, bislang unbekannte Moleküle genau diese Eigenschaft vorhergesagt werden. Ein Ansatz hierzu ist, die Moleküle als Graphen zu beschreiben, also z.B. Atomen durch Knoten zu modellieren und Verbindungen zwischen Atomen im Graphen durch Kanten abzubilden. Ausgehend von der Annahme, dass *ähnliche* Moleküle sich auch im Hinblick auf ihre mutationsauslösenden Eigenschaften ähneln, kann dann anhand besagter Ähnlichkeit eine Vorhersage getroffen werden.

Eine grundlegende Vorraussetzung für diese und ähnliche Anwendungen ist, dass zwischen den Graphen eine Ähnlichkeit definiert und berechnet werden kann, die für die entsprechende Anwendung geeignet ist. Solche Ähnlichkeiten bzw. Distanzen zwischen Graphen und Netzwerken sind Thema dieser Arbeit.

Unterschieden werden hierbei Graphen, bei denen keine zusätzlichen Attribute gegeben sind und Netzwerke, bei denen zusätzlich Attribute für Knoten bzw. Kanten mit semantischem Bezug zum Problem gegeben sind. Darauf aufbauend werden drei grundlegende Ansätze zur Bestimmung von Ähnlichkeiten entwickelt.

In einem ersten Ansatz wird davon ausgegangen, das zu untersuchende Netzwerke Knotenattribute besitzen, die im Zusammenhang mit ihren strukturellen Eigenschaften stehen. Knoten werden anhand ihrer Attribute in Gruppen gruppiert (geclustert), wobei angenommen wird, dass die entstehende Gruppenaufteilung einen semantischen Bezug zum eigentlichen Problem besitzt. Ausgehend von dieser Aufteilung werden die Netzwerke des Ensembles auf die Gruppen projiziert: Die Knoten jedes Netzwerks werden anhand ihrer Gruppenzugehörigkeit zusammengefasst und aus den entstehenden Gruppen sowie den Verbindungen zwischen den einzelnen Gruppen können für die einzelnen Netzwerke statistische Werte abgeleitet werden, die das jeweilige Netzwerk in Bezug auf die Gruppenstruktur beschreiben. Anhand dieser Statistiken können dann Netzwerke miteinander verglichen werden, wodurch wird eine Struktur im Ensemble erkennbar wird,

die sich auf die vorher bestimmte Knotengruppierung bezieht.

In den weiteren Ansätzen werden Graphen ohne Attribute betrachtet. Zunächst wird angenommen, dass die betrachteten Graphen eine Blockstruktur besitzen: Knoten können in Gruppen eingeteilt werden, so dass die Wahrscheinlichkeit von Verbindungen zwischen Knoten nur noch von ihrer Gruppenzugehörigkeit abhängt. Auf dieser Grundlage wird zunächst ein Verfahren entwickelt, bei dem die Eigenwerte, abgeleitet aus der Spektralzerlegung von Adjazenzmatrizen, benutzt werden können, um Graphen anhand ihrer zu Grunde liegenden Blockstrukturen zu klassifizieren.

In Kapitel 6 wird auch diese Vorannahme fallen gelassen und Graphen werden nur noch anhand ihrer Eigenwerte verglichen. Graphen können durch diverse Matrizen repräsentiert werden und zwischen den Verteilungen der Eigenwerte dieser Matrixrepresentationen und den strukturellen Eigenschaften des Graphen besteht oftmals ein Zusammenhang. Darauf aufbauend wird eine Distanz zwischen Graphen definiert, die in der Lage ist Eigenwerte von verschiedenen Graphen miteinander zu vergleichen. Vorteile gegenüber bereits vorgeschlagenen Ansätzen, die ebenfalls auf Eigenwerten basieren, ist die effiziente Berechenbarkeit dieser Methode und ihre Fähigkeit, einerseits Graphen unterschiedlicher Größe (also auch unterschiedliche Anzahlen von Eigenwerten) ohne Adaption zu vergleichen, wobei andererseits die vorhandene Informationsmenge fast vollständig genutzt wird. Unabhängig von der Größe der zu vergleichenden Graphen fließen sämtliche Eigenwerte in den Vergleich ein, was zum einen eine große Flexibilität ermöglicht, zum anderen aber auch die Anwendbarkeit auf sehr kleine Graphen erlaubt.

Die dargestellten Eigenschaften werden durch beispielhafte Anwendungen auf realistische Problemstellungen illustriert, wobei auch ein Vergleich mit bereits bekannten Ansätzen stattfindet.

Teile dieser Arbeit, im Wesentlichen der Inhalt der Kapitel 4 und 5, wurden bereits in Brandes, Lerner, Nagel, and Nick (2009b), Brandes, Lerner, Lubbers, McCarty, Molina, and Nagel (2010) und Brandes, Lerner, and Nagel (2011) veröffentlicht.

Der Autor wurde während der Erstellung dieser Arbeit von der Deutschen Forschungsgemeinschaft im Rahmen des Graduiertenkollegs "Explorative Analysis and Visualization of Large Information Spaces" (GRK 1042) und im Rahmen des 7th Framework Programme (FP7-ICT-2007-C FET-Open, contract no. BISON-211898) finanziell unterstützt.

Contents

Figures & Tables

Figures

Tables

1 Introduction and Overview

Subject of this dissertation is the assessment of graph similarity. The application context and ultimate aim is the analysis of network ensembles, i.e. collections of networks, in the sense of identifying structure among them, e.g. groups of highly similar networks. Structure is in this context understood as some form of regularity or description of the similarities among the considered networks. As an illustration, consider a collection of two types of networks, where networks of the same type are very similar, while networks of different types are very dissimilar. These two groups form some kind of similarity that is of interest when the ensemble is the object to be analyzed. Consequently, graphs are in this situation the elementary entities and the main interest is the measurement of structural similarities between them.

The interest in graphs as opposed to e.g. vectors as basic objects is motivated by their descriptive capabilities: some objects, e.g. electric circuits, social networks, comprehend important structural properties that can be expressed directly by modeling them as graphs. They have also found to be a powerful description mechanism for objects that do not incorporate an obvious relational structure as for example in image recognition. Using graphs to describe objects leads to sets or collections of graphs on which problems of supervised and unsupervised learning are to be solved. A fundamental prerequisite in such approaches is the ability to compare the elementary objects, i.e. assess similarity or dissimilarity between them. For a number supervised and unsupervised learning algorithms a similarity or distance on the objects of analysis is even the sole prerequisite for their application, a prominent example given by support vector machines (c.f.Vapnik (1998)). Motivated by these considerations, three approaches for assessing and measuring similarity between graphs are developed in this dissertation.

In the following, an overview of the contents is given, followed by a more detailed description of the contributions and motivation. The chapter is concluded by a discussion of different aspects of structural similarity between graphs, intended to provide context for the approaches presented in the remainder.

1.1 Overview

The main contribution of this dissertation are three different methods to measure similarity between graphs. Following the introduction in this chapter and some preliminaries in Chapter 2, related approaches are reviewed in Chapter 3. Thereafter, in Chapters 4 to 6 the individual methods are described in detail. Their order is derived from the assumptions made about the underlying data.

Networks incorporating attributes connected to the network structure are analyzed in

Chapter 4. The attributes attached to nodes are used to find a common structure that all networks in an ensemble can be related to. This relation to the common structure is then exploited to derive a vectorial representation of networks accompanied by a suitable distance.

In Chapter 5, it is assumed that the considered graphs incorporate some block structure, i.e. nodes can be grouped such that connection tendencies related to group membership become visible. It is shown that such block structures are under certain conditions reflected in the eigenvalues of the networks and this relation is exploited to distinguish them by their underlying block structure.

Finally, the method proposed in Chapter 6 assumes nothing about the input graphs. Considerations and experiments are limited to undirected graphs, though the proposed method is shown to generalize to directed graphs. Basis of this approach are relations between the distribution of eigenvalues that can be derived from matrix representations of a graph and various structural properties connected to this distribution. Main assets of the method proposed in Chapter 6 are that it is almost lossless and sensitive to small perturbations. The approach is motivated and supported in Chapter 7 by a number of relations between structural network properties and eigenvalue distributions reviewed therein. The variants of the defined graph distance are then examined and compared empirically in Chapter 8. Finally, a conclusion of is given in Chapter 9.

1.2 Contributions

The method of projecting attributed graphs shown in Chapter 4 is - to the knowledge of the author - the first approach that is capable to construct a vector space that *directly* reflects structural properties of networks while integrating semantic features encoded by node and edge attributes.

While a number of approaches for graph comparison by eigenvalue distribution have already been proposed, the approaches proposed in Chapters 5 and 6 provide additional value. In contrast to the approaches reviewed in Section 3.5, the method of Chapter 5 provides a clear objective of comparison, i.e. graphs are differentiated with respect to a precisely defined structural feature - their block structure - and mathematical proof is provided explaining why this approach succeeds asymptotically.

The distance proposed in Chapter 6, lacks such a clear objective of distinction and targets general structural dissimilarity. The additional value in comparison to approaches previously proposed lies in its precision and simplicity. Eigenvalue distributions are compared without loss of information by approximation or conversion. At the same time, no numerical parameters need to be determined to tune the distance to a particular application.

All of the presented methods were developed in cooperation with others and some of them published previously. Chapter 4 is based on a previous publication on the application of the method in Brandes et al. (2010) and its detailed description in Brandes et al. (2011). Brandes et al. (2009b) is the basis of the content of Chapter 5. The contribution of the author to the development of this approach concentrates on the design

and execution of the empirical examination of the established results. The contents of Chapters 6 to 8 is as of yet unpublished.

1.3 Motivation and Applications

In the following, some example applications of graph comparison are sketched. This is not intended as a survey on applications of graph similarity and does not claim any completeness in coverage of application areas. The number and variety of fields in which this problem arises illustrates the importance of the problem and thereby motivates the development of means to measure graph similarity. Note that the number of examples for the individual fields is not meant to indicate the impact of graph similarities in the corresponding field, but examples are chosen to be as widely spread as possible for illustration.

Examples for fields of application not elaborated in more detail are given by natural language processing as in Suzuki, Sasaki, and Maeda (2006), the analysis of office interior plans in Hanna (2007) or the comparison of street networks in Hanna (2009).

1.3.1 Social Network Analysis

The term *social network* is often used to denote graphs that describe relations between actors. In these, relations encode e.g. acquaintances or business contacts, while actors can be persons, companies, states or even animals as for example in Croft, James, and Krause (2008).

One concrete definition of what a social network is, is given in Wasserman and Faust (1994, p.20): "A *social network* consists of a finite set or sets of actors and the relation or relations defined on them. The presence of relational information is a critical and defining feature of a social network." The following will neither rely on such a definition nor attempt to give one, but restrict considerations to graphs describing social relations.

The purely structural description of such networks is often additionally enriched with attributes that further describe actors, relations, or both. A subset of social networks that are of special interest are *ego networks*. An ego network describes a person (the ego) by relations to and between actors (alters) of its social environment, i.e. the actor of interest is described by the structure of its social embedding. The considered network is then composed of alters which are in direct contact to the ego (omitting the ego) and relations are those between alters.

The reason why a description of a social setting via networks is considered beneficial in some aspects is that the social setting (in the case of ego networks) or society (in the general case) is reflected by regularities in the relations, which in turn are expressed in the observed relations (c.f. Nadel (1957)). Polygamy, for example, may be allowed or banned in a certain society resulting in differently structured marriage relations. The assumption that networks are a descriptive expression of the structure of a society or social setting allows to compare societies or social settings by a comparison of the corresponding networks. An example is the comparison of ego networks for persons from countries

with differing cultural backgrounds in Höllinger and Haller (1990), where the cultural difference between societies is examined empirically by the comparison of ego networks of people from those societies.

1.3.2 Chemistry, Biology, and Biochemistry

Examples for objects in these areas that have been modeled as networks are: molecules, metabolic networks, and protein-protein interaction networks. Therein molecules are often modeled with atoms as nodes and bonds between them as edges. An example is Borgelt and Berthold (2002), where chemical compounds are classified by a set of discriminating subgraphs, being either contained or not contained in a particular molecule. Another example is given by Fröhlich, Wegner, Sieker, and Zell (2006) comparing molecules by constructing (partial) correspondences between the node sets of derived networks.

Metabolic networks model systems of chemical processes occurring within cells. They are assembled of *metabolic pathways*, which can be defined as: "any sequence of feasible and observable biochemical-reaction steps connecting a specified set of input and output metabolites", as done by Koffas, Roberge, Lee, and Stephanopoulos (1999). Similarities between the resulting networks have been examined for their applicability in the reconstruction of evolutionary relationships e.g. in Banerjee (2009).

Protein-protein-interaction networks are derived (e.g. in Chen and Yuan (2006)) by modeling proteins as nodes and their interactions as edges. Evolutionary changes in such networks are considered e.g. in Wagner (2001), with a focus on the stability of certain structures throughout the evolutionary process. Banerjee and Jost (2007) compare these networks to the outcome of a number of random models, indicating that they can be distinguished from such by their eigenvalue distributions.

1.3.3 Image Recognition

Problems of image recognition and computer vision include the identification of objects in a picture or the determination whether two images show the same object or not. In this context, graphs have been identified as a formalism for image description that is useful in the resulting tasks. The basic idea is to identify "important" points such as corners in a picture (e.g. in Wilson, Hancock, and Luo (2005)) and form a graph from these with edges describing their spatial relations. Similarity between depicted objects then hopefully results in similarity of the derived graphs and thus a comparison of these graphs yields a comparison of the depicted objects. Other approaches try to recognize hand-written letters as in the example application in Riesen and Bunke (2007) where graphs are not only compared directly but placed in a vector space spanned by similarities to prototypes.

1.4 The Notion of Structural Similarity

The common topic of all contributions presented in the remainder is the assessment of graph similarity using properties of their structure and sometimes additional attributes. Therefore, a discussion about the different is expedient and will be given in this section. This discussion is intended to give an overview and provide some orientation for the more detailed considerations following. Thus most references and details are omitted here and provided in the more detailed, technical discussions of Chapters 3 and 7.

1.4.1 Equality

A very simple similarity on graphs considers only the structure and dichotomizes: two graphs are either "equal" or "different". In the absence of additional attributes, the order of nodes is usually seen as insignificant and two graphs are considered equal if nodes can be reordered to achieve equality, i.e. if there exists an isomorphism between the two graphs. To decide whether two graphs are isomorphic, i.e. decide *graph isomorphism*, is a problem for which no efficient algorithm is known, though it has not been shown to be $\mathcal{NP}$-complete. A more detailed treatment of this topic is given in Köbler, Schöning, and Torán (1993).

In the assessment of structural similarity approached here, however, equality is only an extreme case and the detection of isomorphic graphs will not be approached. In the remainder, two graphs are considered *identical*, if their nodes can be reordered such that equality in connectivity (nodes have identical neighbors) and - if present - attributes is achieved.

1.4.2 Aspects of Similarity

Though comparison of graphs by their structure is the dominant topic in the remainder, no rigid definition of "structural similarity" will be given, since it is hard - if possible at all - to capture the various interpretations of this concept in a single definition. There are, however, different notions of similarity between graphs which will be referred to as *aspects of structural similarity* in the following. Such an aspect can refer to the difference in a concrete statistical measure such as the (normalized) number of edges in a graph, but could also capture a more complicated notion as e.g. the maximal size of a subgraph contained in two graphs. Considering every possible structural difference at the same time is often not compatible with the application at hand, which might be only interested in certain aspects while neglecting others. Therefore, an explicit consideration of these aspects helps to assess the applicability of a concrete distance or similarity.

Independent of these considerations, a common feature of all distances developed in the following, is zero distance for identical graphs. Non-identical graphs should in contrast not result in total dissimilarity as in the dichotomy considered above, but in a measure somehow valuating their difference. This, however, does not collide with the former considerations about the problem of graph isomorphism, since two non-isomorphic graphs could still be equal in all considered structural aspects.

Diminishing similarity between graphs can be described referring to different aspects of similarity, e.g. two graphs could exhibit very different degree distributions while having the same number of edges. These are not necessarily orthogonal in the sense that a change in a graph usually results in a diminishing similarity under various aspects. A change in the density of a graph will for example necessarily result in a change of its degree distribution.

In the following, some aspects of similarity are reviewed in more detail to provide a context for the assessment of the graph distances developed in the remainder. This collection of considered aspects is not complete but guided by available results and their use in the remainder. References and technical details are omitted for the most part in the following, since the intention is to give an introductory, qualitative discussion about the notion of structural similarity. They will be given in the more detailed discussions in Chapters 3 and 7 and the more detailed discussion of random graph models in Section 2.4.

The particular aspects of structural similarity that are considered fundamental in the remainder are:

- edit distance
- common subgraph similarity
- similarity by statistics
- model similarity

All of these aspects have been used before - sometimes implicitly - to establish a method of network comparison. In the following, these aspects will be discussed briefly.

Edit Distance. *Edit distance* is an aspect of similarity based on graph altering operations. The idea is to measure the difference between two graphs by the number of "small changes", e.g. the removal of a single edge, needed to transform one into the other. Considering unweighted graphs, the addition or removal of a single edge or node is an atomic edit operation. Therefore, the minimal difference (in this view) between two graphs is introduced by a single edit operation. Along the same lines, the similarity is diminished by addition or removal of more nodes or edges, i.e. the more edit operations are executed the larger is the resulting dissimilarity. For two arbitrary graphs the minimal number of edit operations needed to convert one into the other is then defined to be their *edit distance* and used as a measure of their similarity. Consequently, only isomorphic graphs will result in a zero edit distance and the determination of edit distance is at least as hard as the isomorphism test itself. There is a connection between edit distance and the maximal common subgraph of two graphs, as graphs that differ only in a few nodes and edges tend to have a large common subgraph. Additional variations can be introduced by assigning costs to the different edit operations and the concept can be further generalized to networks with additional attributes in different ways.

Common Subgraph Similarity In contrast to the maximum common subgraph concept, *common subgraph similarity* considers small substructures appearing simultaneously in the graphs under comparison. The structural aspect considered here is the frequency of appearance of particular substructures. Consequently, two graphs are regarded as similar under this aspect if they contain a certain set of subgraphs with comparable frequency. The particular choice of the evaluated subgraphs allows a further refinement and adaption to the task at hand. In some cases, not the frequency but the mere existence of particular subgraphs is of interest. An example for the latter case is given by the characterization of planar graphs by forbidden subgraphs ($K_{3,3}$ and K_5). This example also illustrates that the concrete definition of "subgraph" has to be considered. One example are induced subgraphs, i.e. a subset of the node set and all edges connecting these nodes in the original graph. Other possibilities include induced subgraphs with removed edges or subdivisions (i.e. nodes of degree two can be transformed to edges) as in the characterization of planar graphs.

Statistical similarity The aspect of *statistical similarity* refers to the comparison of graphs by derived statistics. Statistics can be derived on the graph level, e.g. the number of nodes, edges, or connected components, but can also be derived from statistics defined on the node (or edge) level. For example the clustering coefficient (c.f. Section 3.1) of a graph can be defined as an aggregation of the clustering coefficients of its nodes. This can be generalized even more by aggregation of statistics about subsets of the node set. The triad census for example considers the frequencies of all (directed) graphs on three nodes.

As an alternative to aggregation, the statistics for nodes or edges can be employed unaltered for a comparison. A prominent example is the degree distribution, which was found to be highly skewed for many observed graphs. The comparison of distributions avoids the loss of information introduced by the aggregation of values to a single number. On the other hand, it introduces the problem of comparing two (multi-)sets which are not necessarily of the same size.

Model similarity *Model similarity* is based on random graph models. Some random graph models are described in more detail in Section 2.4, but for now it is sufficient to consider them as probability distributions over the set of graphs. Despite mathematically defined random graph models, natural or technical processes or measurements producing graphs (e.g. the metabolic networks), can be interpreted as random graph models. In this context a random graph model is assumed to assign a major probability weight on graphs exhibiting a certain structural commonality.

Model similarity tries to capture these structural commonalities by assigning to each graph the model that it was most likely produced by and assigns a similarity based on these models. Technically, such an assignment of a graph to a concrete underlying model is often not possible, e.g. due to graphs with equal probabilities in different random models. However, when the considered set of models is sufficiently restricted, the general idea can be applied. In other cases, model similarity is only approached indirectly, i.e.

instead of identifying the generating model for a graph it is ensured that graphs produced by similar models result in a higher similarity than those produced by different models.

This notion becomes especially interesting, when graphs of different size are compared. While such graphs are not necessarily similar in measurable structural aspects, they can still be produced by random graph models that are identical, except for the parameter determining the size of the produced graph.

The aspects of similarity considered above are not complete in the sense that no others can exists. In contrast, they are merely examples for the possibilities of approaching the notion of similarity among graphs. Later chapters will, however, refer to these aspects, using them as guideline to assess distance measures by their sensibility to these aspects.

As mentioned before, similarity between graphs is a notion that is hard to grasp in a definition due to the many aspects that could be considered. The various aspects above will be applied in the remainder as basic ideas and sometimes be identified with certain concrete properties or construction processes. These are, however, not to be understood as formalizations of the concepts presented here, but only as interpretations in a given context.

2 Terminology and Definitions

For a simple distinction between cases where attributes are present and considered or absent respectively ignored, the terms *graph* and *network* are used distinctively. The term graph will always refer to a graph without node and edge attributes, while the term network denotes an attributed graph, i.e. a graph with values attached to nodes, edges or both. The terms node and vertex are used interchangeably without intended difference in meaning, as are edges and links.

Collections of graphs are often referred to as *ensembles*, usually denoted as $\mathcal{E}$. This is not intended as a pure synonym to collection or set but refers to a special kind of composition. A collection of networks or graphs is considered as an ensemble if its members have substantial commonalities that render them comparable. Examples are networks, that originate from a common natural or technical process such as e.g. repeated measurements, networks generated from questionnaires or graphs drawn from random graph models with possibly varying parameters. In ensembles incorporating networks, these should share a common set of attributes that are exploitable in the analysis.

2.1 Definitions

The graphs considered in the remainder are usually simple, undirected, and involve a finite number of nodes. That is, edges connect only two distinct nodes (in contrast to hypergraphs and graphs with self loops), two nodes are either connected by a single edge or not connected at all and an edge has no assigned direction. A graph G is denoted as a tuple $G = (V, E)$ with a finite set of nodes V and a set $E \subseteq \binom{V}{2}$ of undirected edges. If several graphs are considered, $V(G)$ denotes the set of vertices and $E(G)$ the set of edges of the graph G. The size of a graph, refers to $|V(G)|$, i.e. the number of nodes. For convenience of notation, edges $\{u, v\} \in E$ will alternatively denoted by uv. $d(v)$ denotes the degree of node v, i.e. the number of incident edges. The distance of two nodes within a graph, is considered to be the number of edges on a shortest path between them, i.e., the distance between u and v is k if $\{u, p_1, \ldots, p_{k-1}, v\}$ is a shortest path between v and u.

Graph Names For graphs following certain systematic constructions the following standard notations will be used: P_n is the path, C_n the cycle, and K_n the complete graph on n nodes. $K_{k,l}$ denotes the complete bipartite graph with k nodes on one and l nodes on the other side.

Partitions In a number of contexts, sets of objects are divided into pairwise disjoint groups completely covering the considered set. Such a division will be referred to as a partition of the set and the groups will be called classes. Thus, whenever something is denoted by *partition*, it matches the following definition:

Definition 2.1 (Partition)**.** *A partition* $\mathcal{C} = \{C_1, \ldots, C_k\}$ *of a set* S *is a division of* S *into non-empty classes* C_i *such that* $S = \bigcup_{C_i \in \mathcal{C}} C_i$ *and* $\forall i \neq j \; C_i \cap C_j = \emptyset$*.*

Matrices, Vectors and Norms Matrices will be denoted by capital letters (A,M,...), vectors by bold letters ($\mathbf{a}, \mathbf{v}, \ldots$) and functions by lowercase characters ($f, g, \ldots$). As usual, the components of matrices and vectors are accessed by lower indices: A_{ij} is the element in row i and column j of the matrix A and analogous $\mathbf{a}_i$ is the ith component of the vector $\mathbf{a}$. To construct diagonal matrices, $\mathrm{diag}(\cdot)$ will be used, i.e. for $A = \mathrm{diag}(x_1, \ldots, x_n)$:

$$A_{ij} = \begin{cases} x_i & \text{, if } i = j \\ 0 & \text{, else.} \end{cases}$$

Further, A^T denotes the transpose of A, $\det(A)$ the determinant and $I = \mathrm{diag}(1, \ldots, 1)$ is the identity matrix of appropriate size.

$\|\cdot\|_p$, refers to the standard p norm for vector spaces:

$$\|\mathbf{x}\|_p = \left(\sum_{i=1}^{n} |\mathbf{x}_i|^p \right)^{\frac{1}{p}} ,$$

where n denotes the dimension of the vector $\mathbf{x}$. Norms on functions refer to the corresponding norm formulated as the integral of the function:

$$\|f\|_p = \left(\int_{-\infty}^{\infty} |f(x)|^p dx \right)^{\frac{1}{p}} .$$

2.2 Matrix Representations of Graphs

Alternative to the representation by a tupel of sets (V, E), a graph can be represented by different matrices. A number of possible representations are shown in Table 2.1 using the degree matrix D is defined as:

$$D = \mathrm{diag}(d(v_1), \ldots, d(v_n)) .$$

Whenever the considered graph is not clearly indicated by the context, it will be added as a parameter, e.g. referring by $\mathcal{L}(G)$ to the Laplacian matrix of G.

The adjacency matrix is the most common representation of a graph in matrix form, it indicates edges without any further manipulation and is in some form contained in most other representations. The Laplacian, also known as Kirchhoff matrix or admittance matrix has a popular application in Kirchhoff's theorem on the number of spanning trees

adjacency matrix	$A_{i,j} = \begin{cases} 1 & \text{, if } v_i v_j \in E \\ 0 & \text{, else} \end{cases}$
Laplacian	$\mathcal{L} = D - A$
normalized Laplacian	$\overline{\mathcal{L}} = D^{-\frac{1}{2}} \mathcal{L} D^{-\frac{1}{2}}$
degree normalized Laplacian	$\overline{\mathcal{L}}' = D^{-1} \mathcal{L} D^{-1}$
signless Laplacian	$\lvert\mathcal{L}\rvert = D + A$

Table 2.1: Matrix encodings of graphs.

of a graph which can be directly related to a product of the non-zero eigenvalues of $\mathcal{L}$. It has further applications in spectral clustering approaches which is also true for its normalized variant $\overline{\mathcal{L}}$. An introduction to spectral clustering approaches and reasoning about which matrix representation should be used can be found in von Luxburg (2007). The degree normalized Laplacian $\overline{\mathcal{L}}'$ was examined for the effect of "motif doubling" on its spectrum in Banerjee and Jost (2008), which will be reviewed in Chapter 7. The signless Laplacian was introduced in Desai and Rao (1994) in the study of a measure of nonbipartiteness of graphs.

Besides their use in literature reviewed in the next chapter, a number of these representations are examined for their applicability in the context of graph similarity by comparison of spectra in Chapters 5, 6, and 8.

2.3 Spectral Decomposition

The term "eigenvalues of G" refers to the eigenvalues of one of the matrix representations, usually the adjacency matrix, of G. Due to the extensive use of eigenvalues in the remainder, some of the properties of the spectral decomposition of a matrix will be summarized in the following. This summary is restricted to the facts needed in the remainder with the intention to fix notation rather than giving formal definitions. A more detailed treatment of this topic is given for example in Golub and Van Loan (1996).

For a matrix $M \in \mathbb{R}^{n \times n}$, $\det(M - \lambda I) = 0$ is a polynomial of λ with degree n, called the *characteristic polynomial*. By the fundamental theorem of algebra, this polynomial

has n roots when counting multiplicities and these roots are the eigenvalues of M. These will be referred to as $\lambda_i(M)$ or λ_i when M is clear from context with $i \in \{1, \ldots, n\}$ and $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$, eigenvalues appearing more than once are repeated in this enumeration. The term *spectrum* of M refers to the multiset of the λ_i, the vector of sorted eigenvalues is denoted by $\boldsymbol{\lambda}(A) = (\lambda_1, \ldots, \lambda_n)^T$. An important property for the applications in the remainder is $M = M^T \Rightarrow \lambda_i(M) \in \mathbb{R}\ \forall i$, i.e. all eigenvalues of a symmetric matrix are real. With respect to the matrix representations defined above, this yields that for all undirected graphs the spectra of all matrix representations defined in Table 2.1 contain only real eigenvalues.

In addition, each distinct eigenvalue λ_i has an associated *eigenspace*. Vectors $\mathbf{v}$ in this eigenspace fulfill the condition $M\mathbf{v} = \lambda_i\mathbf{v}$. For symmetric matrices, the dimension of this eigenspace corresponds to the multiplicity of its eigenvalue. In the following, eigenvalues and vectors are assumed to be paired one by one. In case of eigenvalues with multiplicity $k > 1$, each of the k repeated eigenvalues is assigned a vector $\mathbf{v}_i$ such that all $\mathbf{v}_i$ together form a basis of the corresponding eigenspace. Finally, every symmetric, real valued matrix M can be decomposed such that $M = UVU^T$ with the eigenvectors $\mathbf{v}_i$ of M as the columns of $U = (\mathbf{v}_1|\ldots|\mathbf{v}_n)$ and $V = \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$. Alternatively, this can be expressed as a weighted sum of matrix products:

$$A = \sum_{i=1}^{n} \lambda_i \mathbf{v}_i \mathbf{v}_i^T \ .$$

This sum of products motivates the usage of the spectral decomposition for the approximation of a matrix. The eigenspace corresponding to the zero eigenvalue can be omitted from the sum without loosing information. Further, the omission of eigenvector/value pairs with small eigenvalues results in correspondingly small deviations of the result from the original matrix.

Computational Complexity The spectrum of a matrix cannot be determinated exactly in a guaranteed polynomial time complexity due to the possibility of irrational eigenvalues. Consequently, a number of algorithms were proposed for the numerical approximation of spectral decompositions some of them described e.g. in Golub and Van Loan (1996). Unfortunately, statements about the complexity of these approximations are seldom to be found, see e.g. Mohar and Poljak (1993, Appendix A) for a discussion. Of special interest in the context of graphs are approximations for sparse matrices since the considered graphs are often sparse in the sense that only a small number of edges are present, leading to sparse matrices representing them. This property is exploited by methods such as the Lanczos-Algorithm and others described e.g. in Saad (2003), to speed up the computation.

The construction of a complete spectral decomposition or only the determination of eigenvalues have to be considered as expensive operations that limit or at least hinder the application of methods based thereon.

Application for Projections: Multidimensional Scaling In the examination of the distribution of distances on a set of objects it is often useful to gain a visual impression of the implied spatial distribution. A visual impression of this structure can be given by producing a set of coordinates, such that the distance is approximated by the euclidean distance between the coordinates assigned to the objects. This can be achieved using spectral decomposition. The method employed for this purpose is a variant of *multidimensional scaling* described as classical scaling in Cox and Cox (2001). This method will be used to project distances into two or more dimensions, referred to as MDS in the remainder. The most important features of MDS exploited here are the following:

- **input:** distances between objects, number of dimensions k to project to
- **output:** coordinates in $\mathbb{R}^k$ for each object
- projection to k orthogonal directions (derived from eigenvectors)
- the kth direction corresponds to kth direction of maximal variance in distance
- relative information content of each dimension can be derived from the associated eigenvalue.

The main assets of this method include a guaranteed optimal approximation and information about the fraction of information being preserved by the projection. Besides the spatial distribution produced, for each projection the fraction of preserved information and the aspect ratio of the bounding rectangle for the projected objects. Purpose of the application of MDS in the remainder is the visualization of distance distributions between objects. Often a projection to some lower dimensional space allows to visually identify trends or groupings that are not obvious when the distances are considered directly.

2.4 Random Graph Models

Random graph models were already considered shortly in Section 1.4.2 as arbitrary probability distributions over the set of graphs. In contrast, the random graph models employed in the remainder refer to distributions that can be defined in form of *generation schemes*. A generation scheme describes a distribution as the output of an algorithm involving randomized elements which provides all means necessary to draw sample graphs. The advantage over arbitrary probability distributions is that their description readily provides a method to draw samples. In the following, four different generation schemes will be described. The selection of those is not representative and by no means complete but oriented to their employment in the remainder.

The $G(n, p)$ and the preferential attachment models were among the earliest proposed random graph models. Therefore, many of their properties have been examined in great detail and they provide good illustrative examples for tests on synthetically generated data sets.

All of the described generation schemes involve parameters that directly influence the resulting distribution to a great extend. This will be exploited in later experiments to explore the aspect of model similarity. In the remainder, a *random graph model* denotes an instance of a generation scheme with partially fixed parameters.

2.4.1 G(n,p)

One of the most studied network models is the $G(n, p)$-model introduced in Gilbert (1959). An alternative version with a fixed number of edges instead of a probability for edge appearance is defined in Erdős and Rényi (1959). Both variants are often referred to as Erdős-Rényi-model. $G(n, p)$ has a number of applications in statistical mechanics illustrated by the review in Wigner (1967) and quantum physics as shown by the review in Guhr, Müller-Groeling, and Weidenmüller (1998).

A random graph can be drawn from a $G(n, p)$ by creating an empty graph on n nodes and connecting each pair of nodes with probability p, i.e. each edge is drawn randomly, independently and with identical probability p. Consequently, the density of the resulting graph is directly influenced by the parameter p which is the expected density. A more detailed treatment with additional properties of this model can be found e.g. in Bollobás (2001).

2.4.2 Preferential Attachment

The principle of "the rich get richer" was applied early in network analysis by Price (1976) to explain the highly skewed degree distribution in citation networks. It gained popularity due to its application in random network generation by Barabási and Albert (1999). The idea of this generation scheme is to simulate a graph growing process that applies the principle of preferential attachment in every step. The original description advises to begin with a small number of nodes and add in each iteration a new node which is then connected randomly to m of the already existing nodes. The preferential attachment principle is therein implemented by the way neighbors for the new node are chosen. The choice is random, influence only by the current node degrees, i.e. new nodes are connected with a higher probability to nodes of large degree.

This generation scheme will be denoted $\mathrm{pa}(n, m, \alpha)$, with n being the number of nodes, m the number of edges added in each iteration and α a parameter for the distribution used in the node selection process. The graph generation begins with m unconnected nodes and continues to add one new node in each iteration which is then connected by m edges to randomly chosen peers. An exact description of the generation scheme used here is given by Algorithm 1. In the algorithm, the outer loop iterates over nodes to be inserted in the existing graph while the inner loop implements the random sampling of nodes which are to be connected to the newly inserted node. The random sampling of nodes implements the preferential attachment effect by the employed probability distribution, where Z is used for normalization. In contrast to other implementations, e.g. the implementation for the R environment (c.f. R Development Core Team (2010)) described in Csárdi and Nepusz (2006), the random choice of neighbors used here ensures

Algorithm 1: Preferential Attachment Graph Creation

Input: n, m, α
Result: random graph
$V := \{v_1, \ldots, v_m\}$
$E := \emptyset$
for $i \in \{m+1, \ldots, n\}$ **do**
  for $j \in \{1, \ldots, m\}$ **do**
    $Z := \sum_{V \setminus \{u_1, \ldots, u_{i-1}\}} d(v_i)^\alpha + 1$
    draw u_i from $V \setminus \{u_1, \ldots, u_{i-1}\}$ with $P(v_i) = \frac{d(v_i)^\alpha + 1}{Z}$
  $V := V \cup \{v_i\}$
  $E := E \cup \{v_i u_1, \ldots, v_i u_m\}$

that a node is never drawn twice as a neighbor for the newly inserted node. Since this algorithm does not match the original definition exactly, results based on that definition may not always hold for the resulting distribution. However, the advantage of this creation scheme is the avoidance of multi-edges (i.e. two or more edges connecting the same pair of nodes) and loops (edges connecting a node with itself) leading to the direct determination of the number of edges by the parameters m and n. Another effect is the avoidance of star-like structures that tend to be the output of the original definition for certain parameter settings, e.g. large values of α, followed by the removal of loops and multi-edges.

2.4.3 Small World Networks

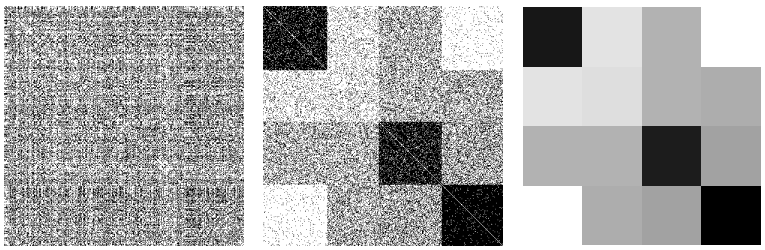
The small world model was proposed in a study on the dynamics of networks with a focus on the effect of the "small-world" property, denoting the phenomenon of arbitrary nodes being connected by surprisingly short paths. Watts and Strogatz (1998) propose a construction scheme that starts from a ring lattice $G = (V, E)$ of n nodes $v_1, \ldots, v_n$ connected such that $v_i v_j \in E$ iff $(i - j) \mod n \leq d$. As a next step, each edge is rewired with probability p by choosing a random node from a uniform distribution on V to replace one of its adjacent nodes. Finally, multi-edges and loops introduced by the rewiring are removed. This construction scheme will be referred to as $\mathrm{sw}(n, d, p)$.

The regular structure in the beginning ensures a high clustering coefficient of the resulting graphs. In addition, the edge rewiring process shortens the characteristic path length by randomly connecting nodes which (by chance) had a much larger distance in the initial graph.

2.5 Blockmodels

Due to their importance in Chapters 4 and 5, *blockmodels* are treated in more detail than the random graph models before. The notion "blockmodel" refers to the idea that

nodes of a network can be partitioned into classes such that connectivity of the graph can be described on the class level without reference to individual nodes. Rephrasing from Holland, Laskey, and Leinhardt (1983): any probability statement about such a model can be modified by exchanging a node with another node from the same block without modifying its probability. As an example, one could specify for each pair of classes the exact number of edges connecting a node from one class to a node from the other. Another possibility is illustrated by the example shown in Figure 2.1 where the differing densities of edges between pairs of classes and within classes form a block structure in a graph. The graph in this example is represented by its adjacency matrix



(a) initial (b) grouped (c) summarized

Figure 2.1: Two views of the same matrix unmanipulated (a), with rows and columns permuted to emphasize block structure (b) and a description of its class structure (c).

to emphasize the blocks that appear when nodes are ordered properly. Figure 2.1(a) shows the graph with nodes ordered randomly with a black pixel in row i and column j representing an edge between the nodes i and j a white pixel marking the absence of such an edge. It also illustrates one of the problems connected to block models: the identification of the node partition. Figure 2.1(b) shows the same graph visualized in the same way only with nodes ordered by class membership, thereby enabling a visual identification of the structure. This reordering illustrates that nodes of identical class have common tendencies to connect to nodes of other classes. The nodes of the first class and last class for example are densely connected to nodes in the same class, indicated by the dark areas of the upper left and lower right corner. At the same time, they are only sparsely connected to nodes of the opposing class, as indicated by the nearly white areas in the lower left and upper right corner. A comparison of the connection patterns (i.e. rows or columns) of individual nodes shows a strong similarity for nodes within the same class but a dissimilarity for nodes from different classes. This similarity is located at the class level, i.e. nodes do not tend to have the same neighbors but rather similar numbers of neighbors in the other classes. Figure 2.1(c) finally illustrates how such a situation can be summarized: since the information about edge probability is given only with respect to class membership the graph can be summarized by describing the density of edges for each pair of classes. Another important parameter of such a description (ignored throughout the example) are class sizes, i.e. the number or fraction of nodes in the individual classes. Both properties can be described in different ways, e.g. by

providing exact numbers or describing them as probabilities.

The remainder of this section is organized as follows: after relating blockmodels to some graph theoretic problems, their application in the context of social network analysis is sketched. Finally, they will be set into the context of ensemble analysis for further use in Chapters 4 and 5.

Related Problems

Some well studied problems in graph theory can be described in terms of blockmodels with specific constraints. In the two examples considered here, admissible blockmodels are described by constraints on connections within or between classes while a concrete node partition obeying these constraints has to be found.

The objective of Graph Partitioning is the minimization of edges between nodes of different classes. Given a maximum K for the number of nodes in each class and a restriction J for the number of edges, a partition of the node set has to be found such that there are no more then J edges connecting nodes of different classes and no class contains more then K nodes. Since the total number of edges is constant, the edges within classes are maximized and this is one translation of the clustering problem into a graph theoretic context.

Vertex Coloring asks for partitions with a minimum number of classes where no edge may connect two nodes of the same class. Alternatively, the vertex set is colored with a minimum number of colors such that no adjacent vertices have the same color.

In both cases, the associated decision problems and a number of their variants are known to be $\mathcal{NP}$-complete. Exact definitions and further variants are described in Garey and Johnson (1979) and the references therein. Since these problems can be interpreted as specialized blockmodels, they illustrate that the identification of constrained blockmodels is a complex problem.

Roles, Positions and Blockmodeling

In social network analysis, the interest in blockmodels is strongly connected to the notions of social role and social position as considered e.g. in Nadel (1957). The approach developed therein tries to describe rules or tendencies existing in a certain society by analyzing positions that members of a society may occupy and the relations between such positions. In terms of blockmodels, positions are described by classes. Positions, however, are not only related to connection tendencies, but in addition to attributes of the corresponding actors. Gender of an actor and its influence on certain relation is a prominent example for such an attribute. This connection of classes and attributes will be considered in more detail in the method developed in Chapter 4. In the context of social networks, a number of different approaches to formalize blockmodels have been proposed as illustrated e.g. in Wasserman and Faust (1994, Chapter 10). Some of those formalizations concentrate on binary modelings: connections between the nodes of two classes are considered as “one-blocks” (connected) or “zero-blocks” (disconnected). The

question to be answered is then whether the connection between two classes should be considered as zero- or one-block.

A more flexible modeling mechanism is provided by *stochastic blockmodels* as proposed in Holland et al. (1983). These model relations between classes as connection probabilities between nodes depending on their class membership. In this mechanism, noise and irregularities, i.e. deviations from underlying rules, do not disturb structural summaries of a given network as they would in stricter settings. In the abstraction of social structure, this can be translated to a division of actors into positions describing their connections in a probabilistic way on the class level. In an example with teacher and students as actors, the teacher has a certain relation to all its students, while relations among students may vary.

Problem Settings

Using the idea of a stochastic blockmodel, a graph $G = (V, E)$ can be summarized by a partition $\mathcal{C} = \{C_1, \ldots, C_k\}$ of V and a density description $e : \binom{C}{2} \to \mathbb{R}$ such that the density (e.g. the number of edges divided by the possible maximum) of edges between class C_i and C_j equals $e(C_i, C_j)$. This example will be used for the illustration of problem settings in the following. Instead of an abstract description of an existing graph, this could also be interpreted as a random graph model. The latter variant is considered in Chapter 5.

Given a graph and a partition of its nodes, a blockmodel as described above can be inferred by simply determining $e()$ from the observation. A first problem could then be formed by omitting the partition, i.e. find a partition for a given graph that results in a "meaningful" blockmodel. This in turn raises the question, whether a partition is meaningful with respect to the aim of blockmodeling, i.e. a measure for assessing the descriptive power or fitting quality of a blockmodel in relation to a concrete graph. If no such measure is defined, any partition of the node set would yield a corresponding blockmodel not necessarily resulting in an informative description. The problem can consequently be summarized in the following way: given a measure of quality and a graph, a partition that optimizes the measure is to be found. Variants of this problem have been the focus of a number of approaches as illustrated by Wasserman and Faust (1994, Chapter 16) and references therein, but will not be approached in the remainder.

Another variant assumes that a graph has a somehow associated blockmodel describing it. Given a number of graphs, these could be compared indirectly by comparing their associated blockmodels. Variants of this problem are considered in Chapters 4 and 5.

3 Approaches to Graph Comparison

This chapter reviews approaches for graph comparison. Some of them are directly related to one of the aspects of similarity described in Section 1.4.2, while others can not be related to specific aspects as for example the spectral distances.

3.1 Network Indices

Network indices map graphs to a single number, e.g. size, density, or a distribution, e.g. degrees. These numbers or distributions can then be used as a basis to compare the associated graphs by the specific aspect measured.

Such statistics can for example be derived by counting nodes (graph size) and edges. The *density* of a graph refers to the fraction of possible edges that are actually present. Consequently, it is defined as the number of edges divided by the number of node pairs:

$$\rho(G) = \frac{2m}{n(n-1)},$$

where n is the number of nodes and m the number of edges in G. Alternatively one could consider the average degree of nodes:

$$\langle d \rangle = \frac{1}{n} \sum_{v \in V} d(v) = \frac{2m}{n} .$$

The latter is useful in situations, where the number of edges is limited by the connections maintainable by individual nodes. The two indices differ in the normalization factor which grows in n^2 for the density while only in n for the expected degree, consequently density often vanishes for large, sparse graphs. The number of edges, the density, and the average degree each represent a single statistic that neglects the distribution of edges among the nodes. Especially the degree distribution among nodes, i.e. how many nodes in the graph have certain degrees, has been examined extensively. For a more detailed discussion of degree distributions and their relation to special features of observed graphs see Li, Alderson, Doyle, and Willinger (2005) and references therein.

A statistic incorporating subgraphs is the *clustering coefficient* proposed in Watts and Strogatz (1998). Here the number of triangles in the neighborhood of a node is considered:

$$\mathrm{cc}(v) = \frac{\Delta(G[\{v\} \cup N(v)])}{\binom{d(v)}{2}}$$

with $\Delta(G)$ denoting the number of triangles (i.e. complete subgraphs with 3 nodes) in a graph G and $G[U]$ is the subgraph of G on the nodes U and all edges of G connecting

two nodes of U. Though Watts and Strogatz (1998) consider only the average clustering coefficient of all nodes in a graph, the distribution of the individual values might be of interest as well.

A statistic referring to the connectivity is *average shortest path length*. Let $p(u, v)$ be the distance (number of edges on the shortest path) between u and v in G, then the average path length is the expected value $\langle p(u, v) \rangle$ over all pairs of nodes in G. The average path length, in relation to the size of the graph can be used to describe its connectivity, i.e. reachability among nodes. In Watts and Strogatz (1998) it was used to argue that "typical" network have a high clustering coefficient and at the same time a low average path length. The related Wiener index was introduced in a chemical context by Wiener (1947) to establish a relation between the structure of molecules and the boiling points of the related chemical compounds. Here, however not the average, but the sum of distances between all pairs of nodes is considered.

This section lists only a small number of examples, with no intention for a complete review of graph statistics, a number of additional statistics is listed e.g. in Brinkmeier and Schank (2005).

3.2 Edit Distance and Graph Matching

Edit distances were introduced in Bunke and Allermann (1983) and Sanfeliu and Fu (1983). Basic operations are the addition and deletion of nodes and edges with costs assigned to each operation. The edit distance of two graphs is then defined as the minimum cost of a chain of edit operations that transforms one graph into the other. Consequently, two graphs are considered similar, if one of them can be constructed from the other by a small number of edit operations. Surveys on edit distances can be found in Bunke and Jiang (2000), Bunke (2000) or Conte, Foggia, Sansone, and Vento (2004).

A major problem in the determination of edit distance is the identification of an initial (partial) assignment between the nodes of the two considered graphs. Given such a mapping, missing or superfluous edges can be determined as the difference of the node adjacencies. Missing or additional nodes are given directly by the assignment of nodes as those without a source or target in the mapping. The determination of edit distance is at least as hard as graph isomorphism, since isomorphic graphs have zero edit distance. Strongly related to the identification of an initial node assignment is the problem of finding a maximum common subgraph as discussed in Bunke (1997). A review of graph matching methods is provided by Bunke (2000).

Related to edit distance is the direct assessment of graph matchings, i.e. instead of changing one graph into the other, nodes of the two graphs are mapped onto each other and the resulting mapping is evaluated. In this evaluation, the nodes without source or target, differences in edges, or even comparisons of attributes of nodes and edges in the two graphs can be integrated. An example for the comparison of chemical compounds is developed in Fröhlich et al. (2006), where the atoms of the smaller molecule are mapped onto those of the larger. Similarity is then derived by comparing attributes of the atoms which are in part derived from the chemical structure, e.g. membership to

certain substructures.

Robles-Kelly and Hancock (2005) suggest to reduce edit distance between two graphs to the problem of edit distance between two strings. Due to the linear structure of strings, edit distance is in this case a simpler problem. On the other hand, the encoding of a graph, in this case its adjacency matrix, as a string - called *seriation* - poses a problem on its own. Robles-Kelly and Hancock (2005) suggest to approach this using a spectral decomposition of the adjacency matrix in particular by deriving a node order from the first eigenvector.

3.3 Similarity by Model

Comparisons of graphs by random graph models are suggested by Faust and Skvoretz (2002) and Bezáková, Kalai, and Santhanam (2006). Faust and Skvoretz (2002) fits a fixed graph model (a p^* model in this case) to each graph. The fitted models are used to predict edges of all other graphs in the collection and graph similarities are derived by the comparison of predictions and actual edges. Bezáková et al. (2006) interpret random graph models as probability distributions and determine the likelihood that a graph was produced by a certain model. Though the objective in this work is not directly the comparison of graphs, the approach could probably be adapted.

3.4 Subgraph Based Approaches

Induced subgraphs form a feature that is especially popular in data mining approaches on chemical compounds. An example for the use of frequent subgraphs and a compact review of related methods is given in Deshpande, Kuramochi, Wale, and Karypis (2005). Comparison of chemical compounds is in such approaches often based on the existence of specific subgraphs as well as their frequency of appearance. As noted in Gärtner, Flach, and Wrobel (2003), this involves an $\mathcal{NP}$-complete problem when subgraphs are not limited either in size or in structure. Consequently, most methods restrict the considered subgraphs. Examples are walks as in Gärtner et al. (2003), Neuhaus and Bunke (2006), Suzuki et al. (2006) or cycles as in Horváth, Gärtner, and Wrobel (2004) or Horváth (2005). Node labels are sometimes integrated by deriving similarity from the number of sequences (walks or cycles) with equal label sequences. Borgelt and Berthold (2002) and Rückert and Kramer (2007) suggest to identify which substructures are important or most discriminating, rather than limiting the considered structures a priori.

Many of these approaches were proposed in the context of data mining on molecules, their integration into standard approaches of data mining can be divided into two categories: graph kernels for the application of support vector machines and feature vectors for most other methods. As mentioned before, the application of support vector machines is only supported by a similarity of the underlying objects. This similarity is denoted *kernel*, thus these methods are called kernel methods, examples are the walk and cycle based methods mentioned above. In alternative approaches, feature vectors

are constructed as basis for the application of data mining algorithms. These are derived from either the existence or frequency of the considered substructures, each feature corresponding to a substructure.

3.5 Spectral Distances

Recall, that graphs can be represented by a number of symmetric matrices and that symmetric matrices can be decomposed into a sum of submatrices weighted by real values - the eigenvalues of the associated graph. Spectral distances are based on the assumption, that structural information about a graph is contained in the distribution of these eigenvalues. This assumption is consolidated by a number of concrete results connecting structural properties to properties of the eigenvalue distribution of a graph. A number of such results are described in Chapter 7. Besides these arguments for the employment of eigenvalues for graph comparison, there is the technical asset that node permutations have no influence on the eigenvalues of a graph, at least with respect to the matrix representations considered here.

In spectral approaches, the comparison of two graphs is reduced to the comparison of their eigenvalue distributions, i.e. (multi)sets of real numbers. A drawback is posed by missing knowledge about the relation between structural aspects and their reflection in eigenvalue distributions. A concrete measurement of similarity would optimally assess one or more known (or even controllable) aspects of structural similarity. Therefore, it would be important to understand how structural graph properties are expressed in the corresponding spectra, which is not always the case. In a second step, these expressions of structural properties in the different spectra are to be translated into a concrete similarity measure, while maintaining the knowledge how differences in the expressed structural properties influence the resulting distance which is also an unsolved problem. For these reasons, the approach taken in all of the measures reviewed in this section is to first define a distance or similarity and then assess whether the structural aspects of interest can be measured. The same approach is taken for the distance presented in Chapter 6, while Chapter 5 gives an analysis of the influence exerted by a concrete structural property on the spectrum and derives a distance on graphs based on this structural property.

Unless comparisons are limited to graphs of the same size, a concrete distance definition on spectra has to cope with sets of different sizes. Since the approach to this problem is usually a major design decision in the development of such a distance, the following review is structured by this aspect. A previous review and comparison of distances based on graph spectra is given in Jurman, Visintainer, and Furlanello (2010). Based on this, the following review will focus on finding a common framework to set the different approaches in relation.

A Common Framework

Approaches to spectral comparison of graphs can be categorized by the following design decisions:

1. matrix representation,
2. conversion into metric space,
3. normalization/weighting,
4. distance measure.

The choice of matrix representation should consider the graph properties expressed in the matrix under consideration. Unfortunately, the knowledge of which characteristics are optimally expressed in which matrix representation is quite limited and provides no detailed guideline for this choice. Therefore, the representation is either chosen for technical reasons, justified by some analogy to physical systems, or derived by empirical arguments. Some approaches benefit from a fixed interval in which the eigenvalues appear, which is provided by the normalized Laplacian ($\overline{\mathcal{L}}$).

The conversion of spectra to objects in a common measurable space is the step which enables the application of distances in the resulting space. Examples for such conversions are the derivation of vectors from a subset of eigenvalues, binning or the derivation of density functions. The proposed approaches of conversion, however, often suffer from information loss in this step. Constructing, for example, vectors from a fixed-size subset of eigenvalues neglects the information contained in the unconsidered part.

Following various motivations, some distances weight or normalize the conversion result. Others integrate a normalization directly into the distance definition, thus the strict separation of conversion and normalization is not always present. Depending on the space that spectra are converted into, two different dimensions can be normalized: the range of the eigenvalues and their number or density. The range can be normalized by the choice of matrix representation, e.g. exploiting the fixed range of eigenvalues provided by the normalized Laplacian, or by direct normalization. For the approaches involving distance measures on probability densities it is sometimes necessary to normalize the overall density of the derived function. An example for a semantically motivated normalization is given in Fay, Haddadi, Moore, Mortier, Uhlig, and Jamakovic (2010a), where feature vectors derived from binning are normalized against the expected number of induced N-cycles.

After conversion, the distance measure or norm can be chosen freely, limited only by the metric space. For vector spaces, a number of standard distances are available, while for conversions into density functions distances for probability functions can be employed.

name	matrix	type of space	convolution/conversion	distance	proposed in
D1	$\mathcal{L}$	$\mathbb{R}^N$	$\frac{\sum_{i=N-k}^{N-1}(\lambda_i-\mu_i)^2}{\min\left\{\sum_{i=N-k}^{N-1}\lambda_i^2,\sum_{i=N-k}^{N-1}\mu_i^2\right\}}$	$\|\mathbf{a}-\mathbf{b}\|_2$	Pincombe (2007)
D2	$\mathcal{L}$	$\mathbb{R}^\infty$	$\frac{\gamma}{x+\gamma^2}$	$\|\mathbf{a}-\mathbf{b}\|_2$	Ipsen and Mikhailov (2002)
D3	A, H, Δ	$\mathbb{R}^N$	$\lambda=(\lambda_1,\ldots,\lambda_N)^T$	$\|\mathbf{a}-\mathbf{b}\|_2$	Zhu and Wilson (2005)
D4	$\mathcal{L}$	$\mathbb{R}^N$	$\lambda=(\lambda_1,\ldots,\lambda_N)^T$	$\|\mathbf{a}-\mathbf{b}\|_2^2$	Comellas and Diaz-Lopez (2008)
D5	$\overline{\mathcal{L}}$	$\mathbb{R}^N$	weighted histogram	$\|\mathbf{a}-\mathbf{b}\|_2$	Fay et al. (2010a)
D6	$\overline{\mathcal{L}}$	$\mathbb{R}^\infty$	$\frac{1}{\sigma\sqrt{2\pi}}\exp\left(\frac{-x^2}{2\sigma^2}\right)$	$\mathrm{JS}(\mathbf{a},\mathbf{b})$	Banerjee (2009)

Table 3.1: Distances comparing graph spectra. $\mathbb{R}^\infty$ denotes the vector space of real valued functions and distance denotation assumes a conversion into vectors or functions $\mathbf{a}$ and $\mathbf{b}$. $\mathrm{JS}(\mathbf{a},\mathbf{b})$ denotes the Jenson-Shannon Divergence, defined in the description of **D6**.

3.5.1 Overview of Spectral Distances

An overview of the compared distances is given in Table 3.1, summarizing distances proposed by Pincombe (2007), Fay et al. (2010a), Ipsen and Mikhailov (2002), Zhu and Wilson (2005), Comellas and Diaz-Lopez (2008), Banerjee (2009).

For reference in the following and in the empirical comparisons of Chapter 8, the individual distances are denoted by **D1** to **D6**, following the numbering in Jurman et al. (2010). In the remainder of this section, the individual approaches are described in detail, organized according to the formalism used for comparison.

Dominating Eigenvalues

A number of approaches for the comparison of spectra are based on their conversion into a common vector space. Since the spectrum is a set with cardinality corresponding to graph size and no canonical order, this conversion results in two problems to be solved: eigenvalues have to be assigned to the dimensions of the resulting vector space and, in the case of differing graph sizes, a common dimension has to be ensured. The assignment of eigenvalues to dimensions is actually a matching problem. As an example, two identical spectra could be arranged into vectors with different sequences of eigenvalues resulting in different vector distances. This problem is usually resolved using the order on the real numbers, i.e. creating vectors from the sorted sequence of eigenvalues. To ensure comparability, all spectra have to be converted into a vector space of common dimension. When all graphs are of equal size, this dimension can be derived by simply taking all eigenvalues into consideration. In the case of different graph sizes, a fixed number of eigenvalues has to be chosen to represent the corresponding spectra. Some approaches argue that the largest influence is exerted by eigenvalues of highest magnitude and choose a fixed number of eigenvalues by magnitude for comparison.

Zhu and Wilson (2005) construct vectors from the ordered eigenvalues of equally sized graphs and use euclidean distance for comparison. Since they only compare graphs of equal size, the problem of choosing representative eigenvalues does not arise and is not addressed. Limiting this approach to the first N eigenvalues of the adjacency matrix, i.e. representing each graph G by the vector $\boldsymbol{\lambda}_N(A(G)) = (\lambda_1, \ldots, \lambda_N)$ allows the application of this distance to graphs with different sizes. Choosing N as the size of the smaller graph, this distance will be denoted **D3**:

$$\mathbf{D3}(G, H) = \|\boldsymbol{\lambda}_N(A(G)) - \boldsymbol{\lambda}_N(A(H))\|_2 \ .$$

A similar approach using the eigenvalues of the Laplacian is proposed by Demirci, van Leuken, and Veltkamp (2008) for indexing purposes in graph databases. In this approach dissimilarity is measured as the square of the euclidean distance:

$$\mathbf{D4}(G, H) = \|\boldsymbol{\lambda}_N(\mathcal{L}(G)) - \boldsymbol{\lambda}_N(\mathcal{L}(H))\|_2^2 \ .$$

Hanna (2007, 2009) proposes to compare $\boldsymbol{\lambda}_{100}(A(G))^T$ or $\boldsymbol{\lambda}_{100}(\mathcal{L}(G))^T$, i.e. first 100 eigenvalues of the adjacency or Laplacian matrix with euclidean or manhattan norm without any normalization. The application is the comparison of room plans by graphs

derived from desk arrangements in Hanna (2007) and the relation of cultural backgrounds to axial graphs of cities in Hanna (2009). Axial graphs are derived by modeling e.g. road segments as nodes and their intersections by edges and used to describe urban areas as graph structures. Due to the similarity to **D3** and **D4**, this distance is not considered separately in later experiments.

Pincombe (2007) and Sarkar and Boyer (1998) present applications where only the largest eigenvalues are integrated in the comparison. While Pincombe (2007) considers structural differences in graphs derived from traffic in computer networks for the analysis of change detection over time, Sarkar and Boyer (1998) suggest an approach for the comparison of aerial images to detect changes over time. Both of them integrate a normalization factor into the distance definition:

$$\mathbf{D1}(G,H) = \sqrt{\frac{\|\boldsymbol{\lambda}_N(G) - \boldsymbol{\lambda}_N(H)\|_2^2}{\min(\|\boldsymbol{\lambda}(G)\|_2^2, \|\boldsymbol{\lambda}(H)\|_2^2)}}\,.$$

Summarizing, vectors are often constructed from either all eigenvalues or a subset, using value or magnitude as criterion. Besides different choices of matrix representations to derive the eigenvalues from, the distance to be applied is an important design decision. In addition to standard vector distances such as the ones derived from Manhattan or euclidean norm, normalization factors are included e.g. in **D1** to account for the size of the involved graphs. The reviewed approaches can be characterized by their choice of matrix representation, the method of choosing a subset of the eigenvalues, their arrangement (ordering) into a vector and finally the actual distance measurement including normalization procedures.

Binning

Approaches involving binning of eigenvalues convert, analog to the proposals of the previous section, the spectrum to a vector. Here, however, the components of the resulting vector are not filled by a selection of eigenvalues, but are rather describing the number of eigenvalues in a certain range of the real line. The set of eigenvalues is interpreted as a distribution of values on the real line and captured by a histogram, partitioning the range of values into a number of disjoint regions called *bins* and counting the number of eigenvalues in each bin. When histograms for the individual spectra use the same borders for the bins, the vectors consisting of the number of eigenvalues in each bin is comparable among different graphs and thus spectra are converted into a common vector space.

As Fay, Haddadi, Thomason, Moore, Mortier, Jamakovic, Uhlig, and Rio (2010b) point out, this approach has the computational advantage that the number of eigenvalues in a given interval can be computed considerably faster than their exact values. The main parameter in this approach is the choice of bins, i.e. their size and position. Therefore, an a priori knowledge of the range in which eigenvalues can appear is necessary. This can be achieved by using the normalized Laplacian $\overline{\mathcal{L}}$ as matrix representation, fixing the range of possible eigenvalues to the interval $[0, 2]$.

Fay et al. (2010b) propose to divide the range of eigenvalues of $\overline{\mathcal{L}}$ into N equally spaced bins $b_1, \ldots, b_N$ and derive a feature vector $\mathbf{f}(G)_i = |b_i(G)|$ from the number of eigenvalues of $\overline{\mathcal{L}}(G)$ in the corresponding bin. They further suggest to weight these features according to the normalized sum of k-cycles (closed walks on k nodes), defining the weighted spectrum of a graph as:

$$f(G,k) = \sum_{i=1}^{k} |b_i|(1-p_i)^k \ ,$$

where p_i is the position of bin b_i. This weighting is motivated by the relation of the weighted sum over all k-cycles in the graph to an expression involving the eigenvalues of its normalized Laplacian:

$$\sum_i (1-\lambda_i(\overline{\mathcal{L}}(G)))^k = \sum_{C_k \in G} (d(u_1)d(u_2)\ldots d(u_k))^{-1} \ ,$$

with summation over all k-cycles $C_k = \{u_1, \ldots, u_k\}$, with $u_i u_{i+1} \in E$ and $u_k u_1 \in E$. Finally, the difference between two graphs G and H is defined on these weighted spectra as the squared $\|\cdot\|_2$ norm:

$$\mathrm{dist}(G,H,k,N) = \sum_{i=1}^{N} (|k_i(G)| - |k_i(H)|)^2 (1-p_i)^k \ .$$

In the exerted experiments parameters are chosen as $N = 70$ bins and $k = 4$ as weighting parameter. Using those parameters, the resulting distance $\mathbf{D5}(G,H) = \mathrm{dist}(G,H,4,70)$ will be used for comparison in the experiments of Section 8.5.

Density Function Construction

In avoidance of the problems appearing with binning, several approaches have been proposed to construct a continuous density function that describes the distribution of the eigenvalues. Analogous to kernel density estimation proposed in Rosenblatt (1956) and Parzen (1962), the eigenvalues are considered as a set of samples from an unknown distribution and the corresponding density function is estimated. The parameters of kernel density estimation are twofold: the kernel $K(\cdot)$ and the bandwidth h. For a set of n samples λ_i the density estimate $f(x)$ is derived as:

$$f(x) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{x-\lambda_i}{h}\right) \ .$$

Depending on the kernel, the factor $1/nh$ may have to be replaced by some other normalization, ensuring that $\int_{-\infty}^{\infty} f(x)dx = 1$. Those density functions can then be compared using distances for probability distributions, which leaves matrix representation, kernel, bandwidth and distance as open parameters.

A first instance of this approach is proposed in Ipsen and Mikhailov (2002) using a Lorentz kernel:

$$K_L(x) = \frac{\gamma}{x^2 + \gamma^2}$$

and bandwidth $h = 1$. The motivation for this approach is derived from the vibration frequencies of a dynamical system related to the graph, which correspond to the square roots of the eigenvalues of the Laplacian: $\omega_i(G) = \sqrt{\lambda_i(\mathcal{L}(G))}$. Difference between the resulting density estimation is measured by $\|\cdot\|_2$ using $\gamma = 0.08$ as parameter:

$$\textbf{D2}(G, H) = \left(\int_{-\infty}^{\infty} (l(G)(x) - l(H)(x))^2 dx \right)^{\frac{1}{2}} ,$$

with $l(G)(x) = \frac{1}{\kappa} \sum_i K_L(\omega_i(G))$ and κ chosen, such that $\int_{-\infty}^{\infty} l(G)(x)dx = 1$. Ipsen and Mikhailov (2002) apply this distance for the reconstruction of graphs from given spectra.

Another variant of this approach is described in Banerjee (2009), using a Gaussian kernel:

$$K_G(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{x^2}{2\sigma^2}\right) .$$

The proposed parameters are $\sigma = 0.01$, bandwidth $h = 1$ and the spectrum of the normalized Laplacian as basis for the density estimation. The distance between the resulting functions is measured by Jenson-Shannon divergence:

$$\mathrm{JS}\,(f, g) = \frac{1}{2}\left(\mathrm{KL}\left(f, \frac{f+g}{2}\right) + \mathrm{KL}\left(g, \frac{f+g}{2}\right)\right) ,$$

where KL is the Kullback-Leibler divergence:

$$\mathrm{KL}(f, g) = \int_{-\infty}^{\infty} f(x) \log \frac{f(x)}{g(x)} dx .$$

Thus

$$\textbf{D6}(G, H) = \mathrm{JS}\left(\sum_{i=1}^{n} K_G(\lambda_i(\overline{\mathcal{L}}(G))), \sum_{i=1}^{n} K_G(\lambda_i(\overline{\mathcal{L}}(H)))\right) ,$$

again using the parameters proposed in Banerjee (2009). The resulting distance is originally applied to compare metabolic networks for the reconstruction of phylogenetic trees describing evolutionary relations.

3.5.2 Polynomials

Wilson et al. (2005) develop a framework to derive polynomials from graphs which are invariant under node permutations. Coefficients of these polynomials are then used as features for graph comparison. In contrast to the other methods working with spectral decompositions, both, eigenvalues and eigenvectors, are used and thereby the full information contained in the matrix is integrated into the comparison. The comparison of graphs differing in size is enabled by filling up the smaller graph with disconnected

nodes. Making full use of the available information is clearly an advantage over the other methods. The approach is, however, not comparable to the others with respect to the employed information and will therefore not be considered in the experiments of Section 8.5.

4 Projecting Networks to Node Partitions

This chapter considers network ensembles, i.e. collections of graphs with attributes attached to nodes and edges, in contrast to the graphs without attributes considered in the other methods. It is further assumed that node attributes are *semantically* connected to the structure of the network. Semantically meaning here that the relation between node attributes and structure is meaningful in the context of the application and not only systematic in frequency or probability as in e.g. nodes with equal attributes having by trend more relations than others. Consequently, there are two major preconditions assumed in the remainder of this chapter: (i) all nodes in the considered ensemble are described by a common set of attributes placing them in a common attribute space[1] (ii) locations in this feature space, i.e. coordinates, relate to the existence or absence of edges in a semantically meaningful way. For simplicity, it is further assumed that edges in the networks describe a single kind of relation.

Considering such a network ensemble, each network relates certain coordinates in the attribute space (i.e. actor attributes) to each other by connecting the corresponding actors with an edge. This distribution of nodes and edges in attribute space described by the individual network is the basis of similarity assessment in the proposed method. Different networks may have different distributions of nodes in attribute space or may connect different regions of attribute space. This can be exploited for comparison, using distance in attribute space as well as node and edge distribution.

The resulting problem is the comparability of these distributions, i.e. assessing the similarity of two distributions in attribute space while in addition considering the presence and absence of relations. That is, two aspects are involved: the distributions of the actors in a network in attribute space and the edges among them. The first aspect could be addressed using techniques for distribution comparison, of which some were discussed in Section 3.5, but that would leave the aspect of edge distribution unconsidered. The binning approach, however, can be adapted to consider both, distribution of nodes in attribute space and distribution of edges between locations in attribute space. Consider therefore a fixed set of "bins" (e.g. hypervolumes) in attribute space, such that every node in the ensemble is contained in exactly one of them, i.e. nodes are partitioned into *classes*. Using such a partition, node distributions of two networks could be compared not directly but by comparing the number of nodes in the individual classes. In addition, this allows the consideration of edge distributions which are also not considered individually but only by the classes they connect.

[1] Attribute space is also often denoted as feature space.

This leads to a two step approach: (i) bin the attribute space, i.e. determine a partition of all nodes in the ensemble and (ii) for each network, derive descriptive statistics of the nodes in each class and the edges among the classes. The resulting statistics can then be compared between different networks, since the set of classes is unique throughout the whole ensemble and thus individual statistical measurements can be identified between different networks. The networks are thus placed in a feature space made up of statistical measures relative to the employed node classes. The parameters in this approach can be summarized by the choice of statistics used in the second step and the node partition.

The choice of node partition is therein of major importance, since it controls the way data is reduced into a comparable form. In a certain sense, networks are projected to the classes of the node partition, hence the name *network projection*. Figure 4.1 illustrates this idea by showing the projection of a graph to 4 classes.

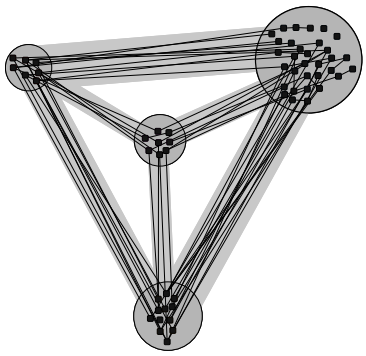


Figure 4.1: Projection of a graph to four classes. Classes are depicted as large circles, whereas vertices of the original graph are depicted as black dots and placed inside their class.

In addition to the comparability, the vector representation of networks allows the application of standard data mining techniques designed for the analysis of feature vectors and methods for visual exploration. This will be exploited for the examination and visualization of the ensemble structure on the level of networks, i.e. the ensemble will be analysed by identifying groups of similar networks and visualizing them.

Since the motivation for the development and an exemplary application of the method described here are embedded in a setting of social network analysis, this context will be elaborated in more detail in the following. This is, however, not intended as a limitation of applicability to scenarios of social science.

In the following section, first some theory on the connection between attributes and network structure in the social sciences will be considered, followed by a description of methods that can be related to the approach proposed here. Section 4.2 then formulates the actual method, followed by an illustrative application on concrete observed data in Section 4.3.

4.1 Methodological Motivation and Related Approaches

The method proposed in this chapter combines two ideas from social science. One is the formalization of social interactions into those between classes or roles of actors. The second is that actors in a social setting are placed into an attribute space constituted of measurable properties - i.e. *Blau space*.

Nadel (1957) develops the idea of describing structure in terms of classes of actors. In his formalization, a society can be described by its inherent rules which can in turn be described by allowed/encouraged or forbidden/discouraged relations between certain roles actors assume. That is, relations (e.g. marital, advise giving) are seen in certain contexts (e.g. wealth, position in a company). The context provides for each actor a role and typical relations in a society can then be described as stronger or weaker, positive or negative connotations of relations depending on the roles of the participants.

The term Blau space was coined by McPherson (2004) who attributes it to Blau (1977). Blau describes social positions in terms of measurable attributes of the actors occupying them. This consequently leads to actors in social structure being distributed in an attribute space (i.e. Blau space) made up of social positions which influence social structure and thus relations between actors.

In the following, these two notions are combined: assuming a single context and thus a unique distribution of actors to classes or roles, different settings are distinguished by examining distributions of actors to classes and relations among actors of different classes following Nadel (1957). The initial assignment of actors to classes in this setting follows Blau (1977), in that a homogeneity of attributes is assumed among the actors of an individual class.

Related Approaches Examples for random graph models using vertex attributes to determine edge probabilities are given in Boguñá, Pastor-Satorras, Diaz-Guilera, and Arenas (2004) and Grabowski and Kosiński (2006). In both proposals, vertices are placed in some attribute space and the probability of an edge between two vertices is directly related to their proximity in this space. A common idea is that homophily and transitivity are directly incorporated by the model specification. Homopholy is the positive correlation between edge probability and proximity in attribute space and therefore directly integrated. Transitivity describes that actors related to the same actor are more likely to be related each other, i.e. friends of a friend know each other with higher probability than random people. Since related actors tend to be close in the attribute space, by geometric argument common neighbors are also close in attribute space. That is, assuming both neighbors are close to the central actor, the triangle inequality indicates a certain proximity between the neighbors themselves and thereby a certain tendency of a relation. This is of course only a sketch and the effects are highly dependent on the actual parameters used in the particular model (e.g. distance, spatial distribution, ...). However, this may be useful in some contexts, but may also be limiting in settings where homophily or transitivity are not desired.

In contrast, the two step approach taken here avoids this dependency by the derivation of positions from attributes and modeling of relations between positions independently

of distance in attribute space. This links the approach described here to blockmodels as proposed e.g. in Holland et al. (1983) and the planted partition model that is examined in Chapter 5. A lossless transformation can be achieved by assigning a class to each distinct position in social space and derive the connection probabilities from the corresponding distances. By such a transformation every social space model can be expressed as a blockmodel. Additionally, blockmodels permit probabilities that are not embeddable in a metric space, thereby allowing to drop the assumptions of homophily and transitivity of edges. Finally, the summarization of vertex positions by classes reduces noise in the data and establishes a natural matching of vertices between different graphs that can be used as the basis for a comparison.

In Butts and Carley (2001) Butts and Carley develop a framework called "inter-structural analysis". Based on the idea that some nodes are equal according to a certain theory derived from expert knowledge they develop the concept of structural exchangeability. This refers to nodes whose exchange does not change edge probabilities with respect to the theory. From the exchangeable pairs of nodes, possible node permutations are derived and similarity of two graphs is measured as the minimum Hamming distance between their adjacency matrices over all accessible permutations. Deriving classes of nodes that are pairwise exchangeable directly relates this to the class extraction employed in this chapter. The theory about node equality employed in this approach can be interpreted as a node partition.

A method to compress graphs for visualization is proposed in Tian, Hankins, and Patel (2008). The authors develop a procedure to visualize large networks by summarizing nodes having equal attributes. Their application, however, is not the comparison of networks but solely the visual inspection of a single network.

4.2 Node Partitions and Network Projection

In the following, the considered ensemble will be denoted as $\mathcal{E} = \{G_1 = (V_1, E_1), \ldots, G_N = (V_N, E_N)\}$. Therein $V_i = \{v_1^i, \ldots, v_{n_i}^i\}$ is the set of vertices of the ith network, vertex sets need not be disjoint and $\mathcal{V} = \bigcup_{i=1}^{N} V_i$ is the set of all vertices in the ensemble. It is further assumed, that a set of common node attributes is given by a function $\mathbf{a} : \mathcal{V} \to A$ mapping nodes to some attribute space A which is equipped with a distance. Note that this last requirement is highly dependent on the meaning of the data under consideration and can be reduced even further as discussed in the following.

The remainder of this section describes the parts of the proposed method in the order of their appearance in the resulting algorithm. Due to its strong connection with the data at hand, only general considerations of the node partitioning process are given in the next section. Following that, the processes of network projection (Section 4.2.2), the derivation of statistical features (Section 4.2.3), the clustering of networks (Section 4.2.4), and the visualization of groups of networks (Section 4.2.4) will be discussed in detail.

4.2.1 Vertex classes

As already mentioned, the node partition is a crucial building block of the proposed method since it controls the projection of networks and thus the point of view from which networks are compared. At least three approaches can be distinguished immediately: manual classification by expert knowledge, regular division of the attribute space, and unsupervised learning of classes.

Depending on the data at hand and the research interest, a node partition by expert knowledge can be used to directly express the research question. An example for such a case is shown in Section 4.3. The attribute space can be divided by certain attributes, e.g. using a subspace spanned by a categorical attribute, or nodes can even be assigned arbitrarily to classes, depending on the context.

In other contexts it may be desirable to divide the attribute space regularly into hypercubes, by splitting every dimension into intervals of equal length. This could be appropriate if no a priory knowledge about node positions in attribute space exists, or if a uniform distribution is expected.

Considering the ideas of social position and role discussed earlier, it may be possible to identify positions reflected in attribute space. That is, regions of the attribute space populated by actors forming a prototype are searched. Actors would qualify for the same prototype when their tendencies to form relations are comparable and differences in these tendencies are not explained by their attributes but rather by personal behavior. Assuming there are such "typical" positions in attribute space, these should be populated more densely than their untypical counterparts. Consequently, one approach to identify such typical positions is the identification of densely populated regions. The projection in the following step, however, demands an assignment of every node to a class, i.e. a partition of the node set. Thus the problem at hand can be interpreted as a clustering task.

Following this line of thought leads to a wealth of clustering algorithms to choose from as the extensive reviews in Jain, Murty, and Flynn (1999) and Xu and Wunsch (2005) illustrate. The underlying motivation, however, suggests some desired properties for the resulting a partition. Since the nodes of each cluster are in the following step contracted, the amount of information neglected in this step should be minimized. Besides creating small clusters, with clusters containing only nodes of identical attributes in the limit, this can be achieved by minimizing pairwise distances between all members within the individual classes. In certain cases - the social network scenario is one example - there may be additional information about typical distributions of certain attributes in a position. Consequently, the distribution of nodes in attribute space could be assumed to be drawn from a mixture of distributions, each describing some location in attribute space and typical derivations from it in the various attributes. Here, the approach reviewed by Fraley and Raftery (2002) may be appropriate, where such a mixture of models is estimated. Given a set of observations in some space, a mixture of models in the corresponding space is derived, that could, in the context considered here, be interpreted as each model describing a position. However, still some complete partition of the node set must be created for the purpose of the following section.

Whichever method of classification is chosen in the concrete application, the result must be a partition $\mathcal{C} = \{C_1, \ldots, C_C\}$ of $\mathcal{V}$ into C classes. This partition is the building block for the following steps and therefore assumed to be given in the following.

4.2.2 Projections

The partition $\mathcal{C}$ of the set of all nodes in the ensemble can be employed to create a projection of each network in the ensemble. That is, the nodes and edges of each network are mapped to a new graph P, the projection target, with node set $\mathcal{C}$ and edges $\{\{C_r, C_s\}\}$ including self loops, i.e. $r = s$. For an individual network $G_i \in \mathcal{E}$ the nodes mapped to the individual classes of $\mathcal{C}$ are simply the nodes in the intersection of its node set and the corresponding C_r, whereas the edges mapped to $\{C_r, C_s\}$ are those that connect nodes being mapped to those classes. Since in the following, descriptive statistics are defined on the source of this mapping, it is useful to define functions that map the parts of P to the nodes and edges of each graph:

$$P_i(C_r) = V_i \cap C_r$$

are the nodes of G_i being mapped onto C_r while

$$P_i(\{C_r, C_s\}) = \{\{u, v\} \in E_i : u \in C_r \wedge v \in C_s\}$$

yields the edges of G_i that connect nodes in C_r and C_s, i.e. the edges being mapped onto $\{C_r, C_s\}$ by the projection of G_i.

Note that P_i projects each graph to the same structure P and thus yields a constant number of possibly empty subsets of the nodes and edges for each graph. Further, the different sets of P_i are indexed by the components of P and are thus identifiable throughout all networks of the ensemble.

4.2.3 Features

The node partition into C classes together with the projections defined in the previous section result in $C + C(C+1)/2$ sets[2] that each graph is projected to. Consequently, a set of features can be defined on these sets that place all networks of the ensemble into the same feature space. Further, properties can be derived from the network directly, e.g. the number of nodes and edges.

Besides these categories (network, nodes, edges), structural and attribute based features can be distinguished. Structural features are based on the underlying graph, e.g. degree distribution in the graph, the number of nodes in a class or the degree of connectedness between two classes. Attribute based features involve the distribution of attributes as the mean and variance of a certain attribute value among all nodes of the network or among the nodes or edges projected to a certain part of P.

[2] That is a set for each class and one for each possible class-class connection including connections between a class and itself.

In the following, some exemplary features are discussed. These are, however, neither complete nor very extensive in their covering of the classification proposed above. Depending on the application context other features might be useful.

Structural features The distribution of vertices to classes in G_i is an important structural aspect for graph comparison, especially since classes are induced by vertex attributes and therefore provide a specific substantive interpretation. In consideration of networks that differ in the number of incorporated nodes, it is useful to consider relative instead of absolute frequency, analogous for edges. For nodes the *relative node frequency* can be formalized as

$$s_r(G_i) = \frac{|P_i(C_r)|}{|V_i|} \ .$$

That is, one set of basic (structural) features is given by the proportion of nodes (edges) being projected to a certain class (pair of classes). In some contexts it may also be useful to consider the balance of the individual class sizes, e.g. the variance of $|P_i(C_s)|$ for the classes C_s.

In consideration of the distribution of edges between the different classes, absolute or relative frequency may not be the ideal measurement. An approach typically used is the that of density i.e. the number of edges divided by the number of overall possible edges. The number of possible edges between two sets of nodes is given by the product of the two set sizes or, in the case of intra class edges, $\binom{k}{2}$ with k being the class size. However, in the context of social networks and thus the example considered in Section 4.3 nodes are expected to have constant average degree, i.e. larger groups do not necessarily lead to a higher average degree. A normalization approach accounting for this is the division by geometric mean of the sizes of the involved classes:

$$e_{r,s}(G_i) = \frac{|P_i(C_r, C_s)|}{\sqrt{|P_i(C_r)| \cdot |P_i(C_s)|}} \ .$$

This way, the number of edges between vertices of classes C_r and C_s is reweighted such that the ratio of edge weights scales with average degrees. When dealing with networks of constant average degree, this scaling behavior is considered advantageous over that of standard density and will be denoted *average degree* in the following.

Attribute based features Attribute features consider the distribution of attributes among the elements of the individual sets created by the projection. Besides the complete set of nodes and edges in an individual graph G_i, the subsets $P_i(C_s)$ of V_i and $P_i(C_r, C_s)$ of E_i induce sample distributions for each dimension of attribute space. This can even be connected to attribute space, by considering the joint distribution that is induced by the edges considering each edge as a pair of nodes and thus attributes. These examples deliver an array of sets that can be used to derive samples of distributions for the variables in attribute space. In the choice of sets that are to be used in the analysis it has to be considered that some of these attributes actually induced the partition and thus introduce a bias on the resulting distributions.

Independent of the considered sets is the method of comparison. For each instance there are two sets of nodes or edges that induce a certain sample of values in attribute space. A possible approach, at least for numerical attributes, is the derivation of the moments of the distribution at hand, resulting in a vector of mean, variance and so on. In addition, the approaches discussed in Section 3.5, can analogous be applied for the purpose here. For example creating histograms and deriving features from the number of nodes in each bin results directly in features that can be used for comparison among different networks.

As will be seen in the following, except for the visualization parts, the method proposed here can be applied on distances without an explicit representation in a vector space. Therefore, a comparison between two classes does not necessarily involve a conversion into some vector space.

Distance derivation The feature groups sketched above are meant as examples and as an illustration of the wealth of statistical measurements derivable from the network projections. Many others are conceivable using combinations of the categories above. The resulting feature vectors provide an embedding of the ensemble into a common space, consisting of subspaces for each group of property. Moreover, they can be utilized as signatures of graphs and, because of their compatibility, prototypical signatures representing subsets of graphs can be derived as well. This is illustrated in the example application in 4.3.

One caveat is in place, however. The usage of raw feature vectors as constructed in the last section, results in groups of features forming subspaces of vastly different dimensionality and extent. A distance on the combined space is therefore prone to be dominated by one or several of the subspaces. In addition, it may be desirable to use individual distances on the subspaces or to emphasize the influence of certain features.

Network ensemble clustering using feature vectors from class structures is in this respect similar to other approaches based on vectors with inhomogeneous components, as e.g. considered in Milligan and Cooper (1988). To account for the possibly different nature of each subspace, these should be normalized and thereafter weighted to control individual influences. For the sake of flexibility, it is here assumed that each of them is equipped with a corresponding distance and the normalization and combination employs only these distances, neglecting the original feature space. Thereby, the necessity of a concrete vector space representation is avoided and additional freedom in the creation of features is gained since only the distances need to be provided.

In the following, distances are normalized such that there is an expected unit distance between two networks in every subspace and weights are used to control the influence of each subspace in the distance combination. Note that this is not the only possibility, but a choice that seems to be sufficient for now.

Let F be the set of features defined on the projections and for two networks G_i, G_j let $\delta_f(G_i, G_j)$ be the distance in the feature $f \in F$. Then the normalized, weighted distance

between G_i and G_j is defined as follows:

$$\hat{\delta}(G_i, G_j) = \sum_{f \in F} \frac{\alpha_f}{\langle \delta_f \rangle} \cdot \delta_f(G_i, G_j)$$

where α_f is a weight for the feature f and $\langle \delta_f \rangle$ is the average of δ_f over all pairs of graphs.

Note that in the case where all features can be described as real numbers and standard distances (e.g. euclidean) apply, the normalization and weighting can be applied directly on these vectors and the following steps might benefit from the vector space representation.

4.2.4 Clustering and Visualization

In addition to the distance on networks developed above, this section aims to provide means to support the visual analysis of the structure of the given ensemble. The approach proposed here is the summarization of structural homogeneous parts of the ensemble and their visualization. That is, parts of the ensemble are to be identified that, corresponding to the employed distance, have approximately the same structure and the resulting groups of networks are visualized. Consequently, a first step, given the normalized and weighted distance between networks, is the identification of subsets of structurally pairwise similar networks.

Using the representation of networks as vectors or distances as derived in the last section, the identification of structurally similar groups of graphs can be reduced to the problem of unsupervised learning on vectors. Consequently, it can be approached using methods proposed in data mining, i.e. clustering algorithms. The intention of this clustering process results in special demands on the partition that is produced in the clustering step. Due to the intended summarizing visualization, the deviation of represented networks within a cluster is to be minimized, which corresponds to compact clusters.

Clustering approaches such as k-means or the estimation of mixtures of Gaussian distributions as described in Fraley and Raftery (2002) generally seem appropriate in this scenario. Even if clusters are not well-separated, such methods yield reference points relative to which the individual networks can be interpreted.

On the basis of the derived feature vectors, a group of similar networks can be described by some representative feature vector, e.g. an average over the vectors of all networks. In addition, the visualizations developed in Brandes, Lerner, Lubbers, McCarty, and Molina (2008) can be extended to use these features for the construction of visual representations. Figure 4.1 gives an idea how this can be done by showing the quotient graph and visually encoding class size as node size and average degrees as edge strength. In more detail, a drawing of the target P of the projection can be used to visually express the mean values of the individual features. Since these features are derived in relation to the individual parts of P, in a visualization they can be attached to these parts and thus exploit this relation for a visual summary. Depending on the application, it may be desired to visualize additional features not even contained in the feature vector, e.g.

histograms of attribute distributions. An example of a basic visualization is given in the application in following section.

An additional question arising for this visualization is the derivation of a mean value. For features represented directly as vectors, this could be the mean value in each dimension while other options have to be considered for more complicated features.

4.3 An Example Application on a Real-World Dataset

This section illustrates the proposed method by application on a real world ensemble. Note that besides the illustrative purpose, there is a concrete interest in the analysis of this particular network targeting acculturation strategies of immigrants.

4.3.1 Data Description

The considered ensemble consists of 504 personal networks derived from interviews with immigrants to Spain and the USA using the EgoNet[3] software. Each network describes the social environment of a migrant originating from a South-American, Middle-American, African, or East-European country. The data was already described in Brandes et al. (2008), but for convenience its main features will be reproduced.

Each respondent was asked to provide the following four types of information:

1. **(questions about ego)** 70 questions about the respondent herself, including age, skin color, years of residence, questions from traditional acculturation scales and health related questions

2. **(name generator)** A list of 45 persons (referred to as *alters*) personally known to the respondent. The alters are the nodes in the respondent's personal network.

3. **(questions about alters)** 12 questions about each of the 45 alters, including country of origin, country of residence, skin color, and type of relation to ego.

4. **(ties between alters)** For each of the 990 undirected pairs of alters, ego specified "What is the likelihood that Alter1 and Alter2 have a relationship independent of you?" by choosing "very likely," "maybe," or "unlikely." The relations have been binarized with "very likely" as the threshold.

While the data on each individual is rich already, the full data set is even more complex. From a purely statistical point of view, each migrant is described by a feature vector with several hundred entries. The present analysis concentrates on the structure of personal networks and considers as attributes only the countries of origin and residence of the alters compared with those of ego. In other words, alters are distinguished only with respect to their immigration situation relative to ego.

[3] See http://www.egoredes.net/ for a description of the project and http://sourceforge.net/projects/egonet/ for a description of the software.

In the following, it is attempted to characterize the individuals by assigning them to different strategies of acculturation. Basis of this analysis in the present semantical domain is the theoretical work provided in Berry (1997). He proposed different modes of acculturation which are used here as context for these personal networks. As elaborated before, the view on the network ensemble established by this method is influence to a large extend by the node partition that induces all following aspects considered in the projections. In the following, this will be exploited by deriving the initial node partition from expert knowledge.

Node Partition In the social setting considered here, the classes of the node partition relate to roles as considered in the framework for social interaction described in Nadel (1957). Compared to the fine-grained role structure presumed by Nadel, here a rather simplistic and abstract view of roles and relationships is employed. This view concentrates on high-level aspects of acculturation, so that roles are determined completely by a small subset of known attributes describing alters relative to ego. This leads to a very small set of stylized roles that apply to the entire ensemble, and provides the partition that will be used as basis for the projection.

The driving proposition is that alter roles are defined by individual attributes, but that their structural consequences are moderated by traits of ego. In other words, alters enacting the same role in the personal networks of distinct egos may be forming alternative relational patterns, and thus reflect back on characteristics of the reporting egos.

With this proposition in mind, the composition and structure of personal networks will be related to the modes of acculturation proposed in Berry (1997). Variance in the relation between roles and interaction patterns can then be interpreted as differences in the immigration situation of the corresponding egos.

The derivation of roles from attributes follows the approach of Brandes et al. (2008) in distinguishing alters by their countries of origin and residence, both relative to those of the ego. In the first step, the alters of all networks are thus partitioned into four classes:

- **origin** - the alter stems from the the same country as the ego and still lives in that country
- **fellows** - the alter stems from the same country as the ego and also immigrated
- **host** - the alter lives in the country the ego immigrated to and stems from that country
- **transnationals** - all other.

Note that this is not the only valid or best node partition for an assignment of this data but rather an expression of the main focus of this application, i.e. characterizing the egos in terms of acculturation strategies. For notational convenience, this node set partition will be denoted $\mathcal{C} = \{O, F, H, T\}$ corresponding to the initials of the introduced node classes. As a first step, the resulting node partition will be used to summarize the complete ensemble at once, followed by a more detailed analysis.

4.3.2 Analysis

An important decision in the proposed method, are the statistics to be derived from the projections. The statistics that will be used in this application are the two examples introduced in Section 4.2.3, namely relative node frequency in each class and average degree between classes. That is, for each network G_i a feature vector consisting of $s_X(G_i)$ and $e_{X,Y}(G_i)$ will be considered for all $X, Y \in \mathcal{C}$.

As a first step of the analysis, an overview of the ensemble is provided by a summary of the contained networks. This serves additionally as an introduction to the visual network summaries, which are an extension of the network visualizations introduced in Brandes et al. (2008). The visualizations of groups of networks derived in the following are based on the mean vector, i.e. the vector derived by taking the mean value of each feature individually.

Ensemble Summary

The structural summary of the ensemble is shown in Figure 4.2. In addition to the

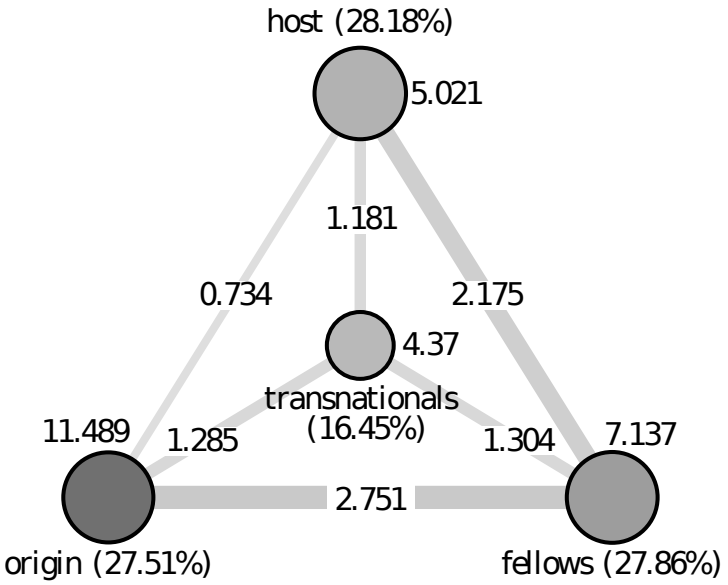


Figure 4.2: Class structure of the ensemble of personal networks. Node sizes express sizes of the corresponding classes, percentage of nodes in each class is given in parenthesis. Edges and nodes are colored by average degree of the connections between the different classes or that of inner class connections.

actual statistical values describing class sizes and average degree, the figure gives a visual impression of the average network structure. Edges are colored according to the average degree described by them, i.e. the darker an edge, the higher is the average degree between nodes of the connected classes in the ensemble. Node colors encode the average degree within a class in the same way, while the average class size within the networks is shown as the size of the corresponding node. Average degrees are shown

close to the corresponding node class or class connection, while relative class sizes are given in percent.

The overview of the ensemble is based solely on an average over all networks, while additional measures such as standard deviation or descriptions of outliers are missing. However, some general trends can be read from this result that give insight into the structure of the ensemble. The individual positions seem not to differ too much in size, except for the category of transnationals. It will later be seen that this balance of class sizes does not hold for all networks. The summary shows, however, that this is not due to the general class distribution in the ensemble. In addition, the average degree of connections within the classes exceeds that between different classes in all cases. In the relations between classes, the connections between origin and fellows and between host and fellows are stronger than all other connections between different classes. These observations agree with a simple explanation: members of the same class live in the same country and the same is true for hosts and fellows. Origins and fellows share a common background, their country of origin, which could explain the strength of this connection.

In the following, the ensemble will be divided into groups of structural similar networks as described above. The visualization just introduced will be the method to enable a visual comparison of the characteristic features for each part. By showing the partition of networks into groups with similar feature vectors and visualizing those, a more detailed view of the structure of the whole ensemble is given.

Weighting and Distance

As mentioned above, for each network G_i the four fractional class sizes $s_X(G_i)$, $X \in \mathcal{C}$ and the ten values for average degree $e_{X,Y}(G_i)$ for

$$(X,Y) \in \{(O,O),(O,F),(O,H),(O,T),(F,F),(F,H),\ldots,(T,T)\}$$

are derived as features.

To allow a free weighting of the individual parts, the distances δ_s and δ_e will be used on the two subspaces and weighting factors for their combination will be introduced. Basis for the distance measurement in the individual subspaces will be the $\|\cdot\|_1$-norm and normalization of distances is implemented as described above by normalizing mean distance to unit length.

For the normalization, the expected distances

$$\langle\delta_s\rangle = \frac{2}{|\mathcal{E}|(|\mathcal{E}|-1)} \sum_{1\le i\le j\le|\mathcal{G}|} \|\mathbf{s}(G_i)-\mathbf{s}(G_j)\|_1$$

and analogous $\langle\delta_d\rangle$ have to be determined first. Since there are only two parts, the weighting factors can be coupled as $\alpha_s = \alpha$ and $\alpha_e = 1-\alpha$. The combined a distance between the networks of the then only depends on α:

$$\Delta^\alpha(G_i,G_j) = \frac{\alpha}{\langle\delta_s\rangle}\|\mathbf{s}(G_i)-\mathbf{s}(G_j)\|_1 + \frac{1-\alpha}{\langle\delta_e\rangle}\|\mathbf{e}(G_i)-\mathbf{e}(G_j)\|_1.$$

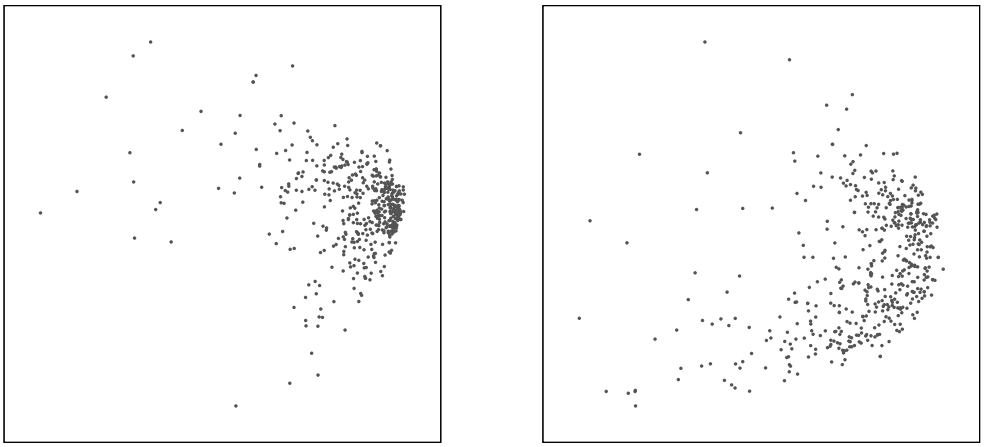
Consequently, α is used in the following to distribute influence between class size distributions ($\alpha \to 1$) and average degrees ($\alpha \to 0$).

Ensemble clustering

Since the clustering of feature vectors strongly depends on the choice of α, different values are explored in the following. The aim is the derivation of a clustering as basis for an analysis. In general, the only constraint on the balance between the influences of class size and average degree is exerted by the employed data and the aim of analysis. On this ground, one could argue which mixture of these is adequate to characterize acculturation strategies in this setting.

Instead, the strategy followed here is to find a mixture of distances such that an "interesting" clustering is the result. Interesting, would be a clustering that divides the networks into different groups of similar networks, thereby showing trends existing in the ensemble. In contrast, a single, large cluster with high variance and no clear subgroups does not indicate further structure and is therefore not helpful in further analysis. Recall, that the overall aim of this method is the identification of trends and consequently the identification of separable subgroups is a necessary condition, since they form those trends in the first place.

Following these arguments, different values of α, namely 0, 0.5 and 1, are examined to find the one that suits best with this goal.



(a) average degree ($\alpha = 0$)

(b) both features ($\alpha = 0.5$)

Figure 4.3: MDS plots of the ensemble showing spatial distributions based on network similarity derived from average degree ($\alpha = 0$) and an equal mixture of the influence of average degree and class sizes ($\alpha = 0.5$).

The distributions resulting from the different values are illustrated in Figure 4.3(a) ($\alpha = 0$), Figure 4.3(b) ($\alpha = 0.5$)) and Figure 4.4 ($\alpha = 1$), all produced by projecting the distances via MDS to two dimensions. Visual inspection reveals no obvious groups in these projections. The visualization are however omitting information, which renders visual inspection insufficient. Consequently all of them were subjected to clustering approaches.

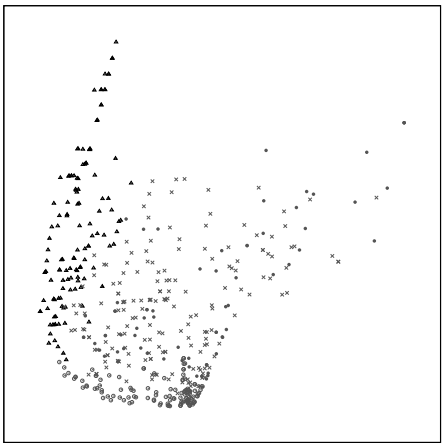


Figure 4.4: Plot of the ensemble based on class size ($\alpha = 1$). A clustering is depicted by the different point types.

Clustering Method For simplicity of implementation and since a detailed examination of different clustering approaches is a large topic in itself, a variant of the k-means algorithm was used to find an appropriate clustering of the networks. k-means starts with k random vectors and iterates by (i) assigning each sample to the cluster represented by its nearest vector and (ii) determining a new representative vector for each cluster by averaging over all vectors in the cluster. A more detailed description of the original algorithm is given in Berthold, Borgelt, Hoeppner, and Klawonn (2010). The number of clusters to produce is a parameter of k-means and therefore has to be determined externally. The approach followed here is to cluster for a range of different k and select the best result. A comparison of the resulting clustering is achieved with the *silhouette coefficient*. The silhouette coefficient (c.f. Berthold et al. (2010)) compares average distances between members of the same cluster to those between members of different clusters and thereby allows the comparison of different clusterings.

The number of clusters k was varied between 2 and 20. To account for the non-deterministic nature of the initialization of k-means, the clustering for each k was repeated 1000 times with random initialization. Experiments show that this leads to reproducible clusterings of the given data set.

Clustering Results For $\alpha = 0$ and $\alpha = 0.5$ only clusterings containing one big cluster and several very small ones could be found. This is additionally supported by the projections in Figure 4.3 in which all networks are concentrated around a single point and no obvious clustering can be observed. As argued above, these clusterings do not highlight structural trends but rather identify some groups of outliers in the small clusters and the majority of networks in the large cluster. Therefore, they are neglected in further analysis.

The situation is different for $\alpha = 1$, i.e. the case considering only the class sizes. The projection in Figure 4.4 already indicates that the distribution is not concentrated around

a single point. Consequently, the remainder of this analysis concentrates on the case of $\alpha = 1$, i.e. the fractional class sizes, using $\delta_S(\cdot, \cdot)$ as distance.

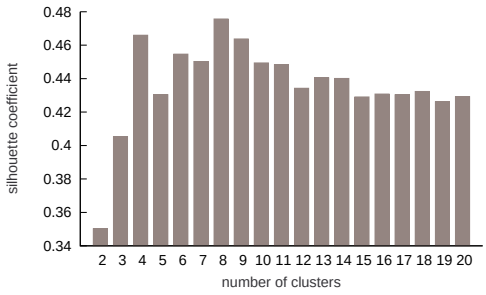


Figure 4.5: Plot of the silhouette coefficient for clusterings differing in the number of clusters. For each number of clusters the clustering was repeated 1000 times and the optimum in each iteration was used for comparison.

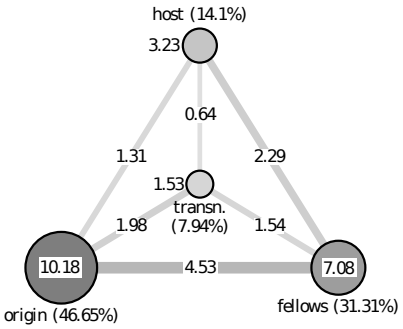
By varying the number of clusters in the application of k-means between 2 and 20, the silhouette coefficients shown in Figure 4.5 are obtained. These hint at two different groupings consisting of four and eight clusters, respectively, and seem to conform to similarities in the ensemble better than the others. The resulting summaries are given in Figures 4.6 and 4.7 and discussed in the following section.

Result Discussion

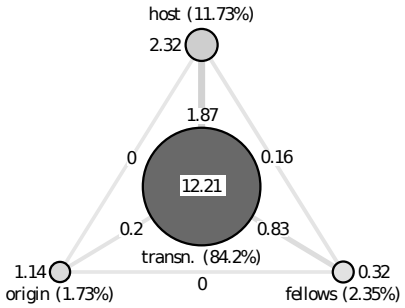
The classifications into four and eight groups can be related to the modes of acculturation proposed in Berry (1997). These modes correspond to four distinct strategies reflecting two different aspects of the integration process an individual chooses (or is forced to choose) to cope with the situation of living in a new, foreign society. The two aspects are maintenance of an individual's identity and cultural characteristics, and maintenance of relationships to the larger society. In the following, the maintenance of *individual characteristics* denotes amplified contact to people having similar cultural backgrounds, thus alters from the origin and fellows classes. Consequently, *contact to the current society* is represented by alters in the host class or transnationals populating the personal network.

The four resulting extremal strategies or modes of acculturation are termed *integration* for maintaining both, individual characteristics and contact to the current society; *assimilation* for emphasis on contact to the new society while abandoning individual characteristics; *separation* for an emphasis on the maintenance of personal characteristics without seeking contact to the host society and *marginalization* for ignorance of both aims.

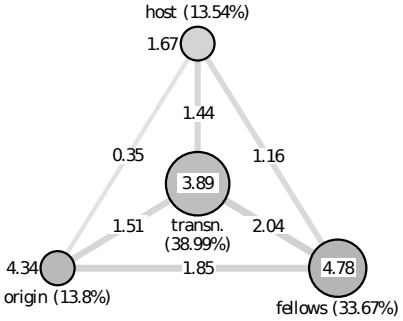
Eight clusters The clustering yielding maximum silhouette coefficient has eight clusters containing 18 to 131 networks and is summarized in Figure 4.6. It is apparent that the clustering exhibits groups that are extreme with respect to class size distributions, which was to be expected for the parameter $\alpha = 1$.



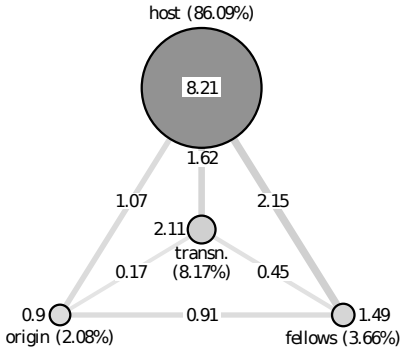
(a) 131 networks



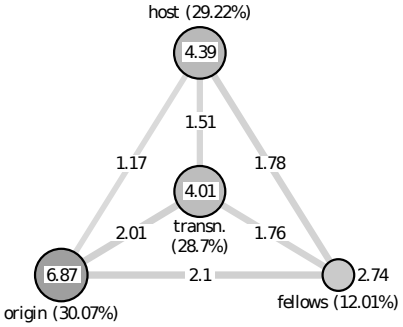
(b) 18 networks



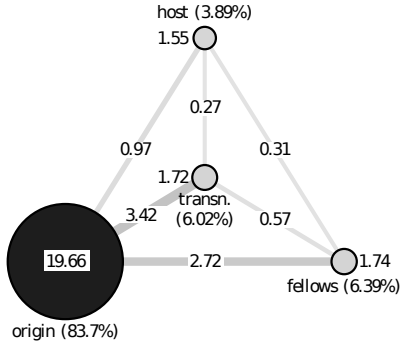
(c) 33 networks



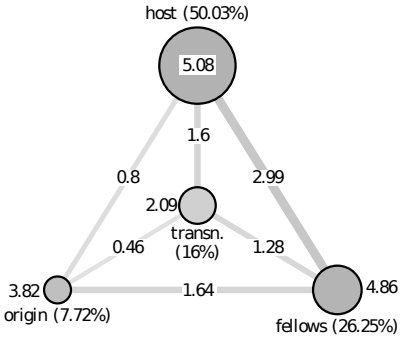
(d) 31 networks



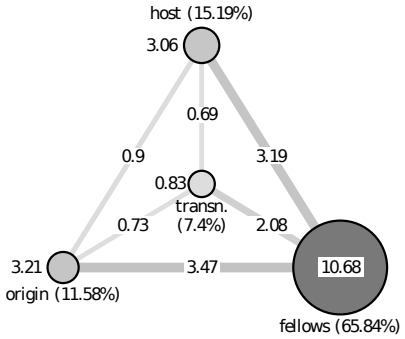
(e) 47 networks



(f) 48 networks



(g) 129 networks

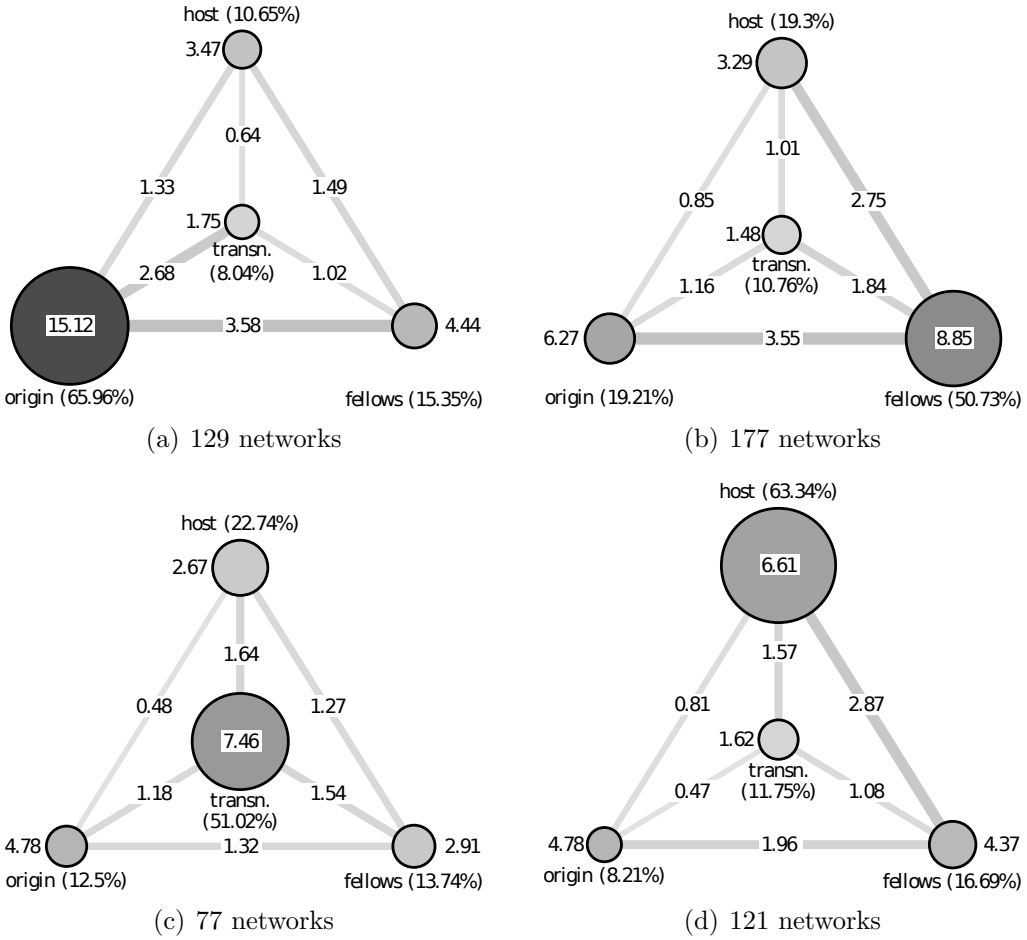


(h) 67 networks

Figure 4.6: Role graphs for a partition into eight clusters of networks. Intra class average degrees appear inside of nodes, average degrees between classes appear on edges. The fractions of actors belonging to the positions is given in parenthesis.

In clusters (b), (d), (f), and (h) one class of alters makes up at least 65% of the complete network. These can partly be seen as extreme instances of the aforementioned modes of acculturation. The networks in the very small cluster (b) for example consist mainly of transnationals, these individuals seem to have nearly no contact to alters originating from the new host society nor can mentionable numbers of alters from the origin or fellows groups be observed. Since this cluster contains only 18 of 504 networks (about 3.6% of the ensemble) it could also be interpreted as a collection of outliers. Cluster (d) shows the behavior described as assimilation, since personal contact is almost limited to members of the host society. The clusters (f) and (h) are good examples for separation (nearly no contact to the host society at all) and marginalization (contact is limited to alters sharing the same origin). The other clusters represent mixtures of these extremes. The clusters containing most of the ensemble are the clusters (a) and (g), covering almost half of all networks in the ensemble and representing nearly opposite modes of acculturation. Cluster (a) shows a strong tendency for separation, expressed by nearly half of all the alters being origins while the contact to alters living in the host society is almost limited to people sharing the same origin and background. In these networks, on average 4 out of 5 alters share the country of origin with the ego. In contrast, cluster (g) could be interpreted as a collection of optimal integration examples. The egos of these networks have on average a lot of contact with the host society which is expressed not only by the size of the host group but additionally in the dense connectedness within this group. This implies an embedding in a group of persons that know each other - a necessary condition for a network of friends, rather than a collection of isolated contacts. At the same time, these individuals maintain a considerable degree of contact to their original culture, showing in about one third of their personal contacts with people of the same cultural background (fellows and origins). Further, the two classes of hosts and fellows are well connected to each other, which can be interpreted as an integration between the two groups. Cluster (c) seems to represent networks suffering at least partly from marginalization. Though there is considerable contact to people of different backgrounds (altogether more than 50%) with transnationals being the largest group, the contact to people originating in the host culture is minimal. This could be due to technical reasons, e.g. sharing of flats. Together with cluster (b) which shows the same effect more prominently and the cluster in Figure 4.7(c) this provides strong support that this is not a random artifact but a meaningful group of similar networks. A cluster representing nearly the total ensemble average is (e). It is hardly assignable to any of the typical modes of acculturation, and the small class of fellows compared to the relatively large class of origins is somehow irritating. A simple explanation is the lack of fellows in the host society, but also a transition between two modes (from separation to assimilation) yields an interpretation. An example would be an individual that follows a strategy of assimilation but is still in the middle of this process.

Four clusters The partition into four clusters (see Figure 4.7), on the other hand, yields fewer outliers in class size, but also lacks the clear interpretability that most of the clusters the previous clustering allow.



(a) 129 networks (b) 177 networks

(c) 77 networks (d) 121 networks

Figure 4.7: Role graphs for a partition into four clusters of networks. Intra-class average degrees appear outside of nodes, average degrees between classes appear on edges. The fractions of actors belonging to the positions is given in parenthesis.

Nevertheless, the clusters match the proposed modes of acculturation almost exactly. The networks summarized in Figure 4.7(a) show strong separation, both with respect to nationality (most of their alters are born in the country of origin) and with respect to place of residence (most of their alters still live in the country of origin). The migrants giving rise to the networks summarized in Figure 4.7(b) know many people living in the host country but still show strong separation with respect to nationality since most of their contacts are classified as fellow immigrants. Figure 4.7(c) exhibits high levels of integration; while there is a considerable number of hosts (about 22%), the remaining alters in these networks are not concentrated on one of the distinguished groups but in the transnationals class. Networks classified into Figure 4.7(d) show strong patterns of assimilation since they know only few alters from their country of origin but most alters stem from the host society.

4.4 Discussion of Applicability and Conclusions

The method presented in this chapter is applicable for the analysis of ensembles containing additional node information which allows assignment or derivation of a meaningful partition. The application example illustrates that it may be reasonable to directly appoint a partition. Another possibility is a consistent (sub-)set of attributes throughout all graphs that can be used to extract such a partition by clustering the nodes. Given this precondition, the aim of analysis can be implemented by two major parameters: the selection of the node partition itself and the selection and weighting of the projection features used to compare the networks.

The selection of the partition is a decision related strongly to the underlying data and should therefore be discussed in relation to the individual context.

Feature Weighting and Selection The second main influence of the outcome is the selection and weighting of the projection properties that are used to compare the networks. Together, the open set of properties to choose from, the distances used to compare them and finally, the weighting of their influence open a vast parameter space. The selection of properties and distances should be guided by expert knowledge about the data at hand and the aim of analysis. On the weighting part, this decision is not settled a priori but depends on the purpose of analysis. Besides the direct setting of influences, which is complicated by their abstract nature, the approach taken in the application example is an interesting option. This could be generalized by optimizing the influence weightings for "clusterability" of the resulting distribution. Structure in the ensemble could then be interpreted as a balance between distance weighting and ensemble clustering on the resulting distances.

The particular example of Section 4.3 also motivates further questions relating to the concrete domain these networks were derived from. The relation between cluster membership and years of residence in the host country is only one example. This illustrates the more general question of the relation between the described network distance and "target" attributes of the networks, i.e. one could consider network attributes that are to be predicted using the introduced distance. To answer such questions, the best approach is probably not using cluster memberships but directly the distances between networks that the clustering was derived from.

5 Differentiation of Blockmodels by Eigenvalues

The last chapter considered ensembles where nodes are grouped by attributes and used - among other features - connection tendencies between those groups to distinguish networks of different structure. This chapter turns to the case where the node classification is not derived from attributes but solely from the structure of the involved graphs. The graphs considered in the following are expected to incorporate class structures that can be identified by the distribution of edges without a given node classification. To differentiate the class structures inferred from node partitions by attributes, the structurally motivated partitions considered in this chapter will be denoted *block structures*.

The following section will define a parametrized random graph model producing graphs with an inherent block structure. Assuming that the observed graphs are produced by instances of this random graph model, Section 5.2 introduces a distance that distinguishes between graphs from different model instances and gives theoretical justification for this distance in the limit of large instances. The applicability to synthetic and real world data, as well as the sensitivity to noise is examined in the experiments described in Section 5.3. Finally, Section 5.4 gives a conclusion.

5.1 Model and Problem Definition

Basis for the following method is a model for random graphs exhibiting a hidden (latent) class structure as defined, e.g., in McSherry (2001) and Golub and Jackson (2008). This model renders the idea of a block structure more precisely.

Definition 5.1 (Planted Partition Model)**.** *A* planted partition model $\mathcal{G}(n, k, \psi, P)$ *is given by a number of vertices* n*, a number of classes* k*, a partition* $\psi\colon \{1, \dots, n\} \to \{1, \dots, k\}$ *of the* n *vertices into* k *classes and a symmetric* $k \times k$ *matrix* P *of edge probabilities* $P_{ij} \in [0, 1]$ *between classes. The probability of a given graph* $G = (V, E)$ *with* n *vertices given the model* $\mathcal{G}(n, k, \psi, P)$ *is*

$$\mathbb{P}\left(G \middle| \mathcal{G}(n, k, \psi, P)\right) = \prod_{\{u,v\} \in E} P_{\psi(u)\psi(v)} \prod_{\{u,v\} \notin E} 1 - P_{\psi(u)\psi(v)}$$

Alternatively, an instance G of $\mathcal{G}(n, k, \psi, P)$ is drawn by including each edge $\{u, v\}$ into G independently with probability $P_{\psi(u)\psi(v)}$. Thus, the probability of an edge between vertices u and v is only dependent on their class-membership.

A planted partition model $\mathcal{G} = \mathcal{G}(n, k, \psi, P)$ is completely defined by its *expected adjacency matrix* which is the $n \times n$ matrix $\overline{M} = \overline{M}(\mathcal{G})$ whose entries are defined by $\overline{M}_{ij} = P_{\psi(i)\psi(j)}$. Note that $\overline{M}$ is indeed the expectation of the adjacency matrices of graphs drawn from $\mathcal{G}(n, k, \psi, P)$.

The subject of analysis in the following are ensembles assumed to be drawn from mixtures of such planted partition models.

Definition 5.2 (Planted Partition Network Ensemble)**.** *A* planted partition network ensemble $\mathcal{E}(N, K, \Psi, \mathcal{G}_1, \ldots, \mathcal{G}_K)$ *is given by a number of graphs* N*, a number of graph models* K*, an assignment* $\Psi\colon \{1, \ldots, N\} \to \{1, \ldots, K\}$ *of the* N *graphs to the* K *models and a family of* K *planted partition models* $\mathcal{G}_1, \ldots, \mathcal{G}_K$*, where* $\mathcal{G}_i = \mathcal{G}(n_i, k_i, \psi_i, P_{(i)})$*.*

Thus, a planted partition ensemble is a set of random graphs drawn from planted partition models. To obtain an instance of $\mathcal{E}(N, K, \Psi, \mathcal{G}_1, \ldots, \mathcal{G}_K)$, the N graphs G_i, $i = 1, \ldots, N$, are independently drawn from the planted partition model $\mathcal{G}_{\Psi(j)}$.

Refining the Problem The model defined above will in the following be assumed to be an unknown source of a given ensemble of graphs. Aim of the developed model is then the partitioning of the ensemble corresponding to the model, i.e. the classification of graphs by common planted partition model. Let $\mathcal{E} = \mathcal{E}(N, K, \Psi, \mathcal{G}_1, \ldots, \mathcal{G}_K)$ be an unknown planted partition ensemble.

> Given an ensemble $(G_1, \ldots, G_N)$ from $\mathcal{E}$, classify the N graphs such that two graphs are in the same class if and only if they are drawn from the same underlying planted partition model.

Obviously, without any further preconditions this problem is not solvable. (For instance, if two of the underlying planted partition models are identical, the graphs generated from these are not distinguishable.)

The next section will, however, propose an efficient algorithm that, given certain preconditions, can decide with high probability whether two graphs are drawn from the same underlying model or not. (The term *with high probability* means "with probability that tends to one as the size of the graphs tends to infinity"; this notion is often employed to assess the quality of heuristic algorithms, compare McSherry (2001).)

5.2 Distinguishing Model Instances

A very simple observation is that if the expectation values $\overline{M_1}, \ldots, \overline{M_N}$ of the corresponding models were provided instead of the adjacency matrices $M_1, \ldots, M_N$ of $G_1, \ldots, G_N$, the problem would be fairly trivial: under the minimal assumption that the planted partition models $\mathcal{G}_1, \ldots, \mathcal{G}_K$ are pairwise different, it follows that their expected adjacency matrices are pairwise different as well. Hence, two graphs G_i, G_j out of $G_1, \ldots, G_N$ are drawn from the same model if and only if the corresponding expected adjacency matrices $\overline{M_i}$, $\overline{M_j}$ are equal.

However, the algorithm does not have access to the expected adjacency matrices. In contrast, the adjacency matrix M_i of an instance graph is rather very far from its expectation value $\overline{M_i}$, since M_i is a zero/one-matrix, while the entries of $\overline{M_i}$ are from the real interval $[0, 1]$; thus the expectation is typically not attainable.

This gap can be bridged by a combination of results from matrix perturbation theory (c.f. Stewart and Sun (1990)) with probabilistic bounds on the eigenvalues of random matrices as provided in Alon, Krivelevich, and Vu (2002) (also compare McSherry (2001)). Basically, these results yield that, even if the adjacency matrix M of an instance graph differs entry wise very much from its expectation $\overline{M}$, the spectrum of M is with high probability close to the spectrum of $\overline{M}$. Consequently, the adjacency matrices of two graphs drawn from the same model have (with high probability) similar spectra and, under the assumption that the spectra of the expected adjacency matrices differ in at least one value, graphs from different models have a larger difference in their spectra.

5.2.1 Method

Let M be an $n \times n$ matrix, with $\boldsymbol{\lambda}(M) = (\lambda_1(M), \ldots, \lambda_n(M))^T$ the vector of its eigenvalues in non-increasing order. An instance of the classification problem is created by randomized drawing N adjacency matrices M_i according to some underlying planted partition models. Each adjacency matrix M_i relates to a corresponding graph G_i, thus the M_i describe a network ensemble $\mathcal{E} = \{G_1, \ldots, G_N\}$.

The classification of the M_i by common model can be solved as a clustering problem: graphs being drawn from the same model should have a spectrum much more similar to each other than graphs drawn from different models and consequently similar graphs should end up in identical classes. Section 5.2.2 shows, that similarity between two graphs in this context should be measured by the supremum norm of their eigenvalue vectors. That is, under certain assumptions $\|\boldsymbol{\lambda}(M_1) - \boldsymbol{\lambda}(M_2)\|_\infty$ should be much greater if the graphs corresponding to M_1 and M_2 are created from different role graphs than if they were from the same role graph.

Consequently, the M_i should by clustered using $\|\cdot\|_\infty$ as vector norm on their eigenvalue vectors. Given objects and distances between them, dense clusters of objects are searched. Standard clustering algorithms can be applied as long as they can be parametrized with a distance measure. Algorithm 2 summarizes the proposed approach.

Algorithm 2: Structural Trends in Network Ensembles

Input: network ensemble $\mathcal{E} = \{G_1, \ldots, G_N\}$
Result: clustering $\{C_1, \ldots, C_k\}$ with $\mathcal{E} = \biguplus_i C_i$

for $G \in \mathcal{E}$ **do**
  determine eigenvalue vector $\boldsymbol{\lambda}(G)$
partition $\{\boldsymbol{\lambda}(G) : G \in \mathcal{E}\}$ using supremum norm

In the ideal case, this method extracts from an arbitrary ensemble a classification of the graphs into groups having the same role graph and thereby solves the stated

algorithmic problem.

Generalization to Weighted Networks For notational simplification, the method sketched above is restricted to binary (unweighted) graphs. A model for ensembles of *weighted* networks (i. e., graphs with real edge-weights) can be defined in almost the same way as in Section 5.1. A *weighted* planted partition model is therein defined as in Definition 5.1 with the difference that when drawing an instance graph one does not include (unweighted) edges with a given probability but rather the weight of an edge $\{u, v\}$ is drawn from a distribution dependent on the classes of u and v. Examples of such distributions would include the normal distribution where the mean value depends on the vertex classes.

The adjacency matrix of a weighted graph is a real matrix whose entries encode the edge weights. Note that the abovementioned method for network classification via the eigenvalues of graphs can be applied to these weighted matrices without any change. Furthermore, the theorems that will be presented in the following also hold for the case of weighted matrices. The application to real world data sketched in Section 5.3 indeed analyzes an ensemble of weighted networks.

5.2.2 Substantiation by Matrix Perturbation Theory

Let $(G_1, \ldots, G_N)$ be an instance drawn from a planted partition network ensemble $\mathcal{E}(N, K, \Psi, \mathcal{G}_1, \ldots, \mathcal{G}_K)$ whose underlying graph models have a common number of vertices n. Building on results from matrix perturbation theory, in this section it will be shown that for sufficiently large n (and ignoring a small number of outliers) the spectra of graphs drawn from the same model have smaller distance than the spectra of graphs drawn from different models.

A planted partition model $\mathcal{G}$ can be associated with a matrix $A(\mathcal{G})$ that encodes the relative class-sizes as well as the edge-probabilities between classes of $\mathcal{G}$. It turns out that the eigenvalues of $A(\mathcal{G})$ correspond—up to a multiplicative constant that is related to the size of the classes—to the non-zero eigenvalues of the expected adjacency matrix $\overline{M}(\mathcal{G})$.

Definition 5.3 (Structure Matrix)**.** *Let $\mathcal{G} = \mathcal{G}(n, k, \psi, P)$ be a planted partition model and denote the proportion of vertices in class $i = 1, \ldots, k$ with*

$$q_i = \left|\{v\,;\ 1 \leq v \leq n \text{ and } \psi(v) = i\}\right| / n$$

The structure matrix *associated to $\mathcal{G}$ is the $k \times k$ matrix $A = A(\mathcal{G})$ whose entries are defined by $A_{ij} = \sqrt{q_i q_j} \cdot P_{ij}$.*

To make the notion *with high probability* precise, a process will now be defined by which the number of vertices in a planted partition model can be increased without changing its structure (more precisely: without changing the relative class-sizes nor the edge-probabilities between classes). Let $\mathcal{G}_1 = \mathcal{G}(n_1, k, \psi_1, P)$ be a fixed planted partition model and $t \in \mathbb{N}_{\geq 1}$ an integer. A planted partition model $\mathcal{G}_t$ that has $n_t = t \cdot n_1$

vertices and the same structure matrix as $\mathcal{G}_1$ can be defined by $\mathcal{G}_t = \mathcal{G}(n_t, k, \psi_t, P)$, where $\psi_t \colon \{1, \dots, n_t\} \to \{1, \dots, k\}$ with $\psi_t(v) = \psi_1(\lceil v/t \rceil)$. Note that $A(\mathcal{G}_t) = A(\mathcal{G}_1)$ holds.

The next theorem shows that the eigenvalues of a planted partition model with fixed structure matrix grow linearly in the number of vertices.

Theorem 5.1. *Let $\mathcal{G}_1 = \mathcal{G}(n_1, k, \psi_1, P)$ be a planted partition model, $t \in \mathbb{N}_{\geq 1}$ an integer, and set $n_t = t \cdot n_1$. Each eigenvalue λ of $A(\mathcal{G}_1)$ yields an eigenvalue $n_t \cdot \lambda$ of $\overline{M}(\mathcal{G}_t)$. The remaining $n_t - k$ eigenvalues of $\overline{M}(\mathcal{G}_t)$ are equal to zero.*

Proof. Note first that the expected matrix $\overline{M} = \overline{M}(\mathcal{G}_t)$ is (after reordering the vertices such that vertices in the same class are consecutive) an $n_t \times n_t$ block matrix

$$\overline{M} = \begin{pmatrix} B_{11} & \dots & B_{1k} \\ \vdots & & \vdots \\ B_{k1} & \dots & B_{kk} \end{pmatrix} \text{ with blocks } B_{ij} = \begin{pmatrix} P_{ij} & \dots & P_{ij} \\ \vdots & & \vdots \\ P_{ij} & \dots & P_{ij} \end{pmatrix}$$

of dimension $(q_i \cdot n_t) \times (q_j \cdot n_t)$. (Note that $q_i \cdot n_t$ is indeed an integer which follows from the definitions of q_i and n_t.)

Let $\mathbf{x} = (x_1, \dots, x_k)^{\mathrm{T}}$ be any eigenvector of $A(\mathcal{G}_1)$ associated to eigenvalue $\lambda \in \mathbb{R}$. Spelling out the equation $A(\mathcal{G}_1) \cdot x = \lambda \cdot x$ yields for $i = 1, \dots, k$

$$\lambda \cdot x_i = \sum_{j=1}^{k} \sqrt{q_i q_j} P_{ij} x_j = \sqrt{q_i} \sum_{j=1}^{k} \sqrt{q_j} P_{ij} x_j \ . \tag{5.1}$$

Then the n_t-dimensional vector $\mathbf{y}$, defined by

$$\mathbf{y} = (\underbrace{x_1/\sqrt{q_1}, \dots, x_1/\sqrt{q_1}}_{n_t \cdot q_1 \text{ times}}, \dots, \underbrace{x_k/\sqrt{q_k}, \dots, x_k/\sqrt{q_k}}_{n_t \cdot q_k \text{ times}})^{\mathrm{T}} \ ,$$

satisfies $\overline{M}(\mathcal{G}_t) \cdot \mathbf{y} = n_t \lambda \mathbf{y}$, which shows that $n_t \lambda$ is an eigenvalue of $\overline{M}(\mathcal{G}_t)$ and, thus, yields the assertion of the theorem.

To see that this is true, let v be any integer satisfying $1 \leq v \leq n_t$ and let $i = \psi_t(v)$ (i. e., i is the index of the class of vertex v.) Then

$$(\overline{M}(\mathcal{G}_t) \cdot \mathbf{y})_v = \sum_{j=1}^{k} n_t q_j P_{ij} x_j / \sqrt{q_j} = n_t \sum_{j=1}^{k} \sqrt{q_j} P_{ij} x_j = n_t \lambda x_i / \sqrt{q_i} = n_t \lambda y_v$$

where the third equation follows from Equation (5.1). □

Corollary 5.1. *Let $\mathcal{G}_1$ and $\mathcal{H}_1$ be two planted partition models with the same number of vertices n. Let $t \in \mathbb{N}_{\geq 1}$ and set $n_t = t \cdot n$. Under the assumption that the eigenvalues of $A(\mathcal{G}_1)$ and $A(\mathcal{H}_1)$ differ in at least one value, the distance between the eigenvalue vectors of the expected adjacency matrices of $\mathcal{G}_t$ and $\mathcal{H}_t$ grows linearly in the number of vertices n_t. More precisely*

$$\|\boldsymbol{\lambda}(\overline{M}(\mathcal{G}_t)) - \boldsymbol{\lambda}(\overline{M}(\mathcal{H}_t))\|_\infty = n_t \cdot \|\boldsymbol{\lambda}(A(\mathcal{G}_1)) - \boldsymbol{\lambda}(A(\mathcal{H}_1))\|_\infty \in \Theta(n_t) \ .$$

Finally, the difference between the eigenvalues of the adjacency matrix M of an instance graph and its expectation $\overline{M}$ has to be bound. For this purpose define the perturbation matrix $E = M - \overline{M}$ as the difference between the observed adjacency matrix and its expectation. Further, a result from matrix perturbation theory (c.f. Stewart and Sun (1990)) will be useful.

Theorem 5.2. *Let $M = \overline{M} + E$ be a symmetric perturbation of a symmetric matrix $\overline{M}$. Then*

$$\|\boldsymbol{\lambda}(M) - \boldsymbol{\lambda}(\overline{M})\|_\infty \leq \|E\|_2 \ ,$$

where $\|E\|_2$ denotes the maximal absolute value of an eigenvalue of E.

The second result needed, is a probabilistic bound on the maximal eigenvalue of the difference between the observed adjacency matrix and its expectation.

Theorem 5.3 (McSherry (2001))**.** *Let M, $\overline{M}$ and E be defined as above and let n denote their dimension. Let σ^2 be the largest variance of an entry in M. (Note that if the i,j'th entry of $\overline{M}$ equals p, then its variance is $p - p^2$; the variance is non-zero if p is in the open interval from zero to one.) If $\sigma^2 \gg \log^6 n/n$, then $\|E\|_2 \leq 4\sigma\sqrt{n}$ with probability at least $1 - 2e^{-\sigma^2 n/8}$.*

The assumption $\sigma^2 \gg \log^6 n/n$ is satisfied for sufficiently large n if at least one entry of $\overline{M}$ is different from zero and one. For the remainder of this section this assumption will be taken for granted, excluding only uninteresting cases.

Theorems 5.2 and 5.3 can now be combined into the following corollary:

Corollary 5.2. *Let M and $\overline{M}$ be defined as above and let n denote their dimension. It is $\|\boldsymbol{\lambda}(M) - \boldsymbol{\lambda}(\overline{M})\|_\infty \in \mathcal{O}(\sqrt{n})$ with probability in $1 - o(1)$ (i. e., with probability tending to one as n tends to infinity).*

Combining these results enables the establishment of the following theorem indicating that any reasonable clustering on the vectors of eigenvalues will—apart from a small proportion of outliers—correctly assign the networks into clusters according to the underlying graph models.

Theorem 5.4. *Let $\mathcal{E} = \mathcal{E}(N, K, \Psi, \mathcal{G}_1, \ldots, \mathcal{G}_K)$ be a network ensemble in which the underlying graph models have a common number of vertices n_t. For each $\varepsilon > 0$ there exists $n_0 \in \mathbb{N}$ such that for $n_t \geq n_0$ and any instance of $\mathcal{E}$*

$$\|\boldsymbol{\lambda}(A(G)) - \boldsymbol{\lambda}(A(G'))\|_\infty < \varepsilon \cdot \|\boldsymbol{\lambda}(A(H)) - \boldsymbol{\lambda}(A(H'))\|_\infty$$

for any graphs G and G' drawn from the same model and any graphs H and H' drawn from different models, with probability in $1 - o(1)$.

Proof. The following assertions hold with high probability. By Corollary 5.2 $\|\boldsymbol{\lambda}(A(G)) - \boldsymbol{\lambda}(A(G'))\|_\infty \in \mathcal{O}(\sqrt{n_t})$. Let $\overline{M}$ be the expected adjacency matrix of H and $\overline{M'}$ be the expected adjacency matrix of H'. Then Corollary 5.1 shows that $\|\boldsymbol{\lambda}(\overline{M}) - \boldsymbol{\lambda}(\overline{M'})\|_\infty \in \Theta(n_t)$ and, again Corollary 5.2 can be used to derive $\|\boldsymbol{\lambda}(A(H)) - \boldsymbol{\lambda}(\overline{M})\|_\infty \in \mathcal{O}(\sqrt{n_t})$ and $\|\boldsymbol{\lambda}(H') - \lambda(\overline{M'})\|_\infty \in \mathcal{O}(\sqrt{n_t})$. Together this yields $\|\boldsymbol{\lambda}(A(H)) - \boldsymbol{\lambda}(A(H'))\|_\infty \in \Theta(n_t)$ which implies that for sufficiently large n_t the inequality of the theorem is satisfied. □

Note that Theorem 5.4 makes only assertions for specific numbers of vertices of the form $n = t \cdot n_1$. However, this restriction is only necessary for notational simplification as Section 5.3 illustrates.

Using these results, one could also think of classifications of ensembles consisting of differently sized graphs. It would be necessary to restrict the eigenvalue vector to a size such that it can be determined for all graphs of the ensemble. One would also have to take care of the growth of the eigenvalues which is linear in the number of vertices of the graph. A possible approach is to take the n eigenvalues with maximum absolute value of each graph where n is the size of the smallest graph in the given instance and divide them by the size of the graph. A more efficient method could be inferred by knowledge of the sizes of the underlying models. If known, the maximum number of classes in the planted partition models could be used as a limit without changing the defined distances.

5.3 Empirical Evaluation

Since the provided theoretical justifications are of asymptotical nature only, the limits of applicability in terms of graph size and noise sensitivity will be explored in this section. Experiments start with small synthetic role graphs and extend to more realistic cases. Finally, an application on a real-world dataset is shown.

Minimal Graph Sizes

To estimate the tightness and expandability of the established results, the first experiments are conducted on artificially generated ensembles, i.e. in a controlled evironment. Note that a planted partition model can be derived directly from a weighted graph with additional class sizes specified by interpreting the nodes as classes and the (appropriately chosen) edge weights as edge probabilities. Such graphs, denoted by *role graphs* are used in the following to randomly create and describe planted partition models. Experiments on these examples split up in two major categories. The first is the case where role graphs on two nodes are specified and experiments try to determine the size needed to distinguish graphs drawn from these models by their spectra. For the second part of the study, role graphs were generated from random edge distributions and random group sizes. In all experiments, graphs of different sizes were generated from each model and compared pairwise in terms of maximum norm on their eigenvalue vectors. In the following, the choice of graph sizes is not restricted to cases that make an exact matching of the group sizes possible but in addition extended to graphs where group sizes can only be approximately established.

Although the analytical results apply only to exact matches of the group sizes, the experiments suggest that the method can be used, e.g., in a setting in which the group membership of each node is determined randomly from a distribution where the probability for membership in a class equals the relative class size in the model. This method was used in the experiment on prespecified models and in the second part of the experiment on random models. Here, additionally the case of group sizes matched as exact as possible

is examined.

Experimental results are illustrated by diagrams showing distances between graphs of these examples distinguishing distances between graphs drawn from different models (points in grey) and those from the same model (in black). The expectation is that distances grow faster with graph size for graphs from different models than they do for graphs from identical models. Consequently, the diagrams show the development of distances between graphs in ensembles of growing graph sizes. The size of the graphs is shown on the horizontal axis and the distance between the graphs appears on the vertical axis.

Prespecified Models For illustrative purposes and a first estimation of asymptotic behavior some archetypical partition models are used for graph generation that have been selected for their simplicity and good separation. All planted partition models incorporate two classes of equal size. They differ in the edge probabilities, where every possible symmetric distribution of the two probabilities $p = 0.2$ and $q = 10^{-3}$ was used to generate two-class models. Excluding isomorphic role graphs, this yields six edge distributions to cluster. For random 200-node graphs, each of the six models is illustrated by three samples in Figure 5.1 showing that structural trends are clearly recognizable. While generating instances from these models, class sizes were not matched exactly but vertices were assigned to the two classes by equal probability. The resulting ensembles were then generated with graphs of size $10i$ with $i = 2, \ldots, 100$, and all graphs of the same size forming one ensemble. The main result is shown in Figure 5.2, suggesting that a clear separation by eigenvalue vectors should be possible for graphs having about 300 and more vertices.

Random Role Graphs The next experiments consider five randomly generated role graphs, each consisting of seven nodes. For each corresponding partition model the desired class sizes and the edge distribution were drawn randomly and independently from a uniform distribution over $[0, 1]$. That is, each planted partition model is created from two random matrices, a $n \times 1$ matrix for node distribution to classes and a $n \times n$ matrix for the edge distribution. The graphs drawn from the different models are distinguished by their eigenvalue vectors and those again derive from the corresponding model.

The resulting models are compared in Figure 5.3 by their spectra (the most important information for the employed distance) as well as their pairwise distances and an overview of the similarity distribution by an MDS of the distances.

The comparison shows that the corresponding models do not differ drastically. The table with pairwise distances in maximum norm of the eigenvalue vectors of their structure matrices and an overview of these distances in a two dimensional layout obtained via multidimensional scaling quantify their relative shapelessness.

Since there are no pointed differences in these spectra, the sampled models can be considered quite typical. In particular, they form a classification instance much harder than the prespecified models considered previously.

For each of the following two experiments, ensembles were created consisting of five

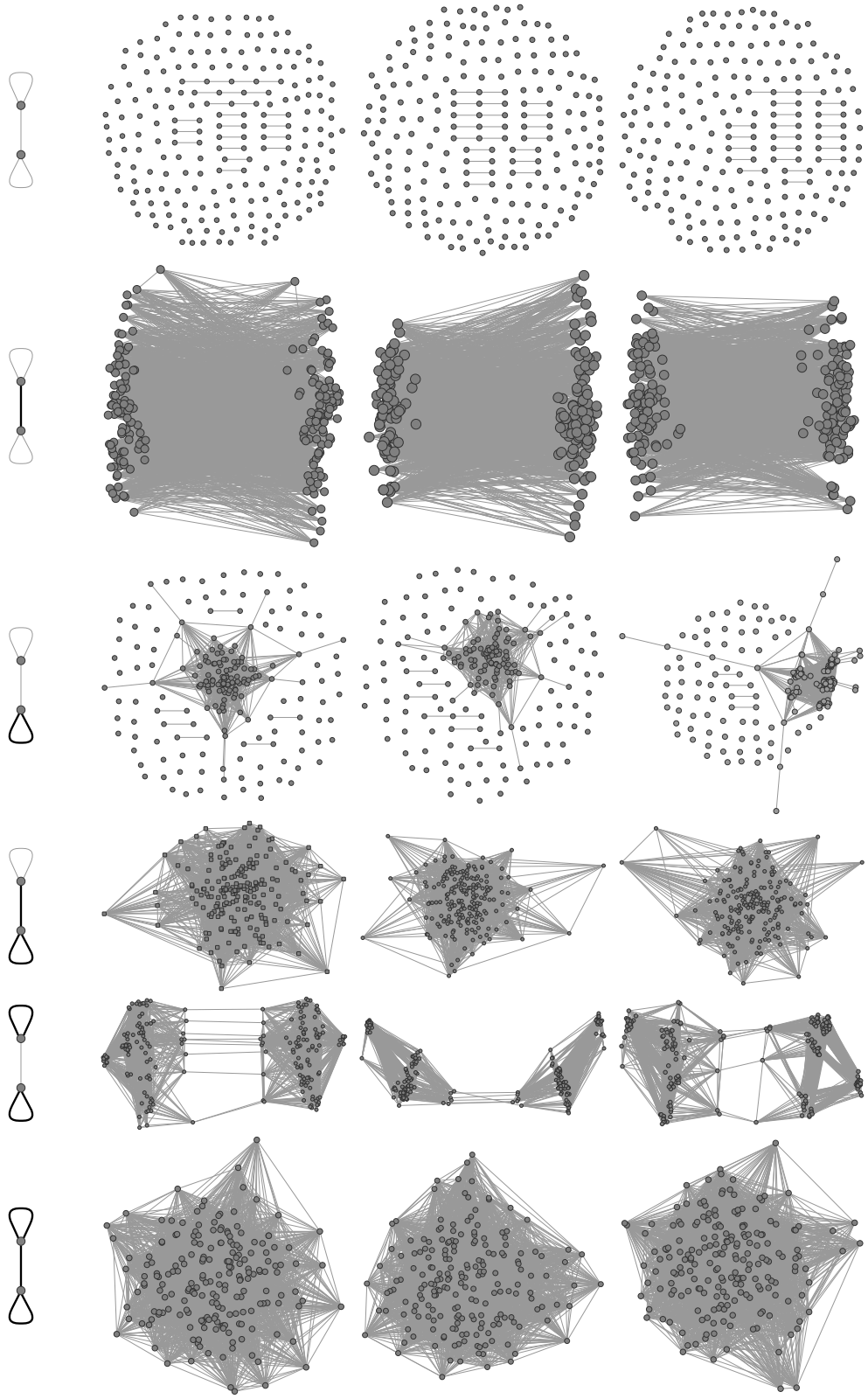


Figure 5.1: Sample graphs with 200 vertices each from six prespecified two-node role graphs.

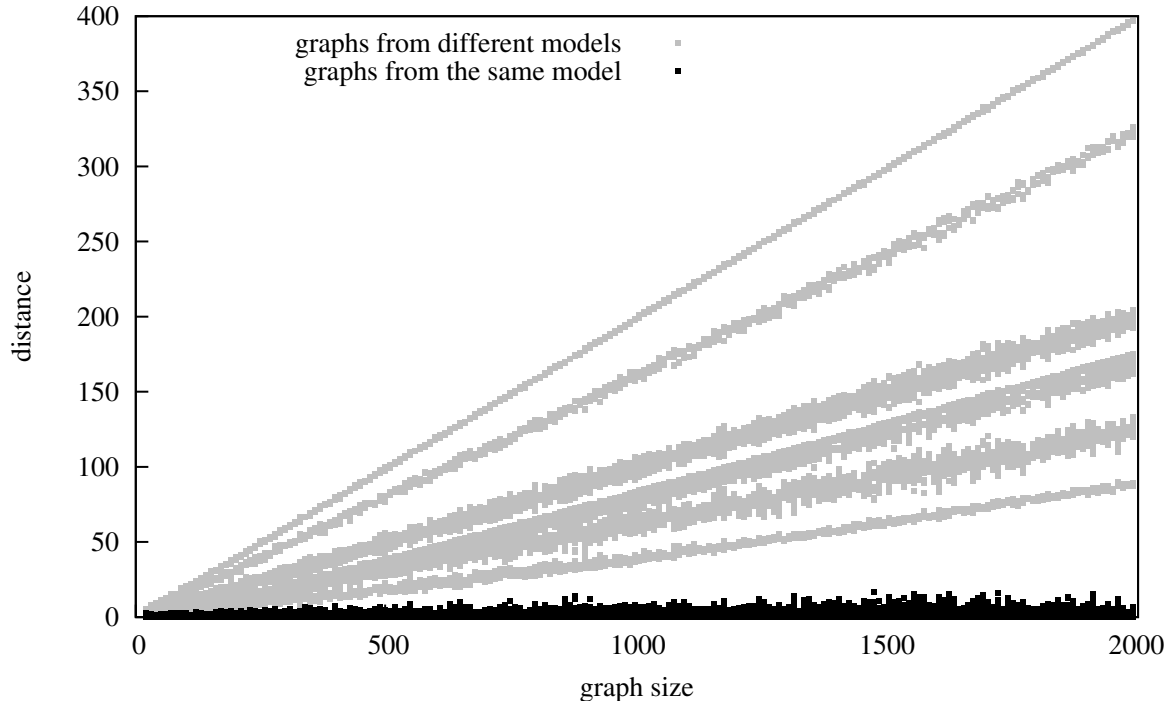
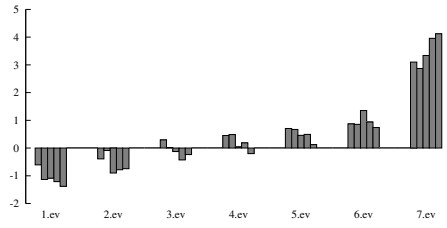


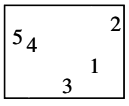
Figure 5.2: Pairwise distances in ensembles generated for six prespecified role graphs with two nodes each. For ensembles with graphs of 300 vertices, a simple distance threshold separates classes well.



(a) spectra

model	2	3	4	5
1	0.528	0.509	0.859	1.022
2		0.810	1.089	1.252
3			0.619	0.782
4				0.388

(b) distances



(c) plot

Figure 5.3: Spectra and pairwise distances of five randomly created role graphs.

graphs with $10i$ vertices for each model and $i = 2, \ldots, 200$, resulting in a sample ensemble with 25 graphs for every i. The difference in the two experiments lies in the assignment of vertices to classes. While in the first part (results in Figure 5.4) partition sizes were matched as exact as possible, in the second part (results in Figure 5.5) the approach described above was used where desired partition sizes are used as a distribution.

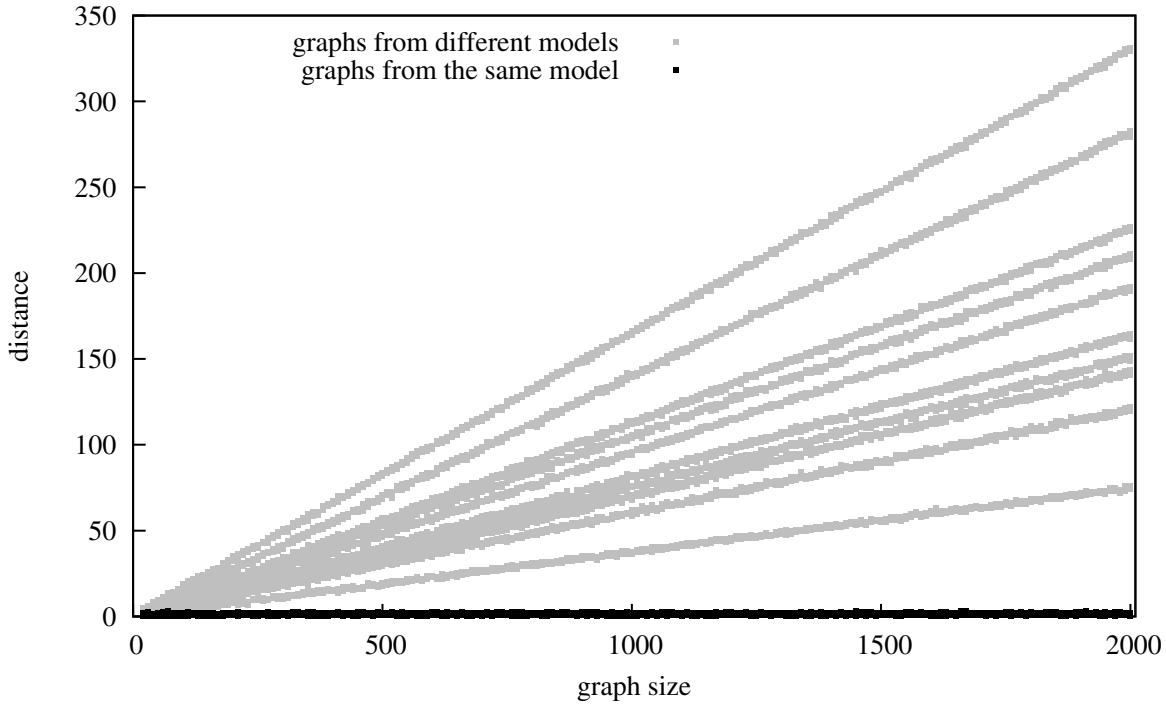


Figure 5.4: Distance development with class sizes matched as accurate as possible.

Figure 5.4 shows how the distances between graphs drawn from different models diverge from each other such that 10 different rays of dots can be seen, which is expected when the distances between the role graphs differ pairwise. Consider a graph with a node for each model and edge lengths defined by the distance between the corresponding models measured as distance between samples from the models. The edge lenghts of this graph growing linearly in the number of vertices of the graphs the ensemble contains plus some random noise are the rays that can be seen in the diagram. The bottom line in black consists of distances between graphs drawn from the same model.

As an unexpected result the distances between graphs corresponding to the same model seem to be constant which could be a hint that the established borders are not tight. A possible explanation could be that the "noisy" parts of the spectra are matched quite well and their differences are minimized by sorting.

The diagram in Figure 5.5 shows how the divergence is weakened by inexact partition sizes. Compared to Figure 5.4 a clear distinction between graphs drawn from different and those drawn from equal models is achieved only with graphs having significantly more vertices, even though a trend towards clear separation can be observed. This illustrates empirically, that the method generalizes to cases where the theoretical model is matched

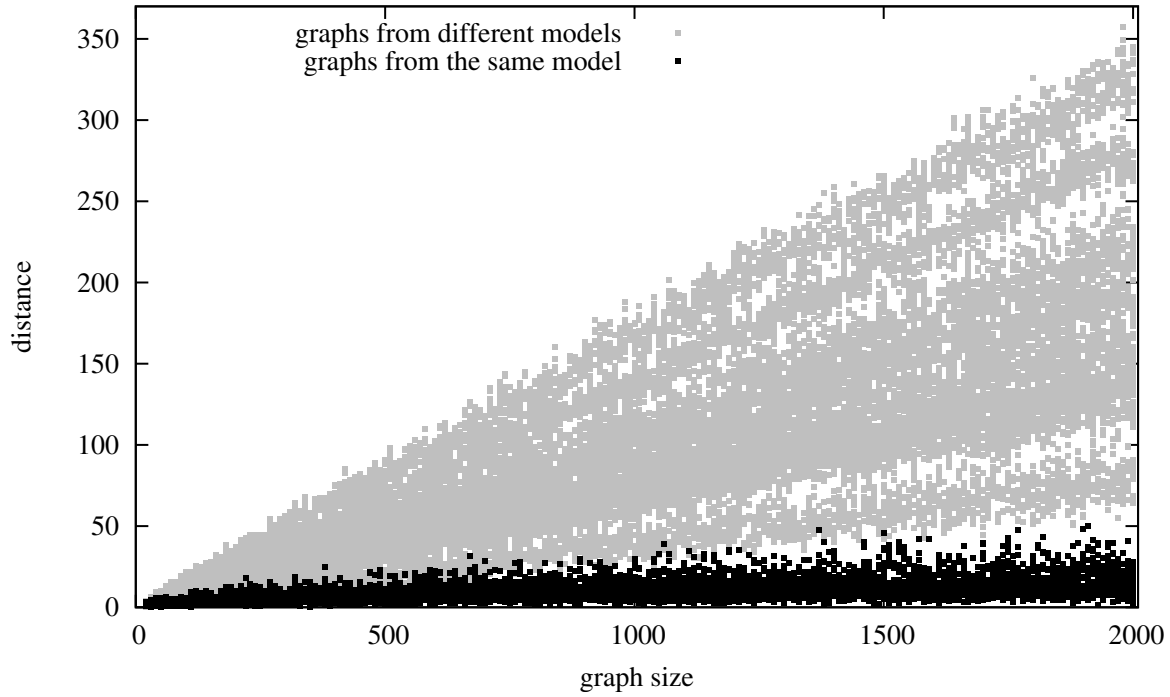


Figure 5.5: Distance development with class sizes as distribution.

only in part with the price of considerably larger graph instances being necessary for a successful distinction.

Real World Data

This section aims to demonstrate the performance of the proposed graph distances in an application on real world data. Object of analysis are edit networks of Wikipedia articles as defined in Brandes, Kenis, Lerner, and van Raaij (2009a). Networks are derived from articles supposed to result in classes of common structure. Therefore, from a set of articles with at least 1000 edits, 60 articles were chosen randomly and 60 were chosen from the set of articles labeled 'featured' by the Wikipedia community.

From the edit logs of these articles a complete graph with a node for each author was created. Each edge was weighted by sums of negative edits between the adjacent authors. A negative edit occurs if either one author deletes words written by the other or if he restores words that were deleted by the other and is valued by the logarithm of the number of words deleted/restored. Since the edit graphs have in general different sizes, comparison was restricted to graphs of at least 500 vertices and vectors consisting of the 500 eigenvalues of maximal absolute value divided by the number of vertices. The number 500 was chosen since the differences are not expressed that clear with smaller values. For greater values the number of graphs being left is not meaningful for class comparison since noise and outliers could dominate the results.

The distance between classes was computed as the average of the pairwise distances

between all graphs of the corresponding classes, while the distances between two graphs was measured as the above described distance on the eigenvalue vector of their weighted adjacency matrices.

The computations yield average distances of $21.7 \cdot 10^{-3}$ within the arbitrarily chosen articles, $15.7 \cdot 10^{-3}$ in the class of featured articles and an average distance of $20.9 \cdot 10^{-3}$ between the two classes. As expected, the featured articles tend toward a structure in their edit graphs that is common among this class and distinguishable from those of arbitrary articles. The inner class difference of arbitrary articles being higher than the distance to the featured articles could be explained by the fact that featured articles are a subclass.

This example represents an even more general case than the one where class memberships are a distribution. Here, graphs have different sizes and a statement on class sizes is impossible. Additionally, graphs with weighted edges were incorporated, which is an additional generalization of the original method. The application of a threshold to convert weighted matrices back to the binary case showed that such an approach destroys too much of the original information, rendering class separation impossible. The obtained results support the decision to employ edge weight matrices and encourages further examination of possible applications in this direction.

5.4 Conclusions

The applicability of the method developed in this chapter can be characterized by two basic conditions. First of all, the assumption about the underlying block structures must hold since it is the basis of the theoretical argumentation. Additionally, a sufficient combination of minimum graph size and fit to the model as shown in Section 5.3 is needed. As the empirical examinations indicate, to some extend there exists a tradeoff between these two conditions, illustrated in the difference between Figures 5.4 and 5.5. The demand on this combination is further influenced by the similarity of the underlying planted partition models, as indicated by Theorem 5.1 and illustrated by a comparison of Figures 5.2 and 5.5.

A drawback of this method is the assumption of dense networks by the planted partition model, which limits its applicability to realistic networks. The reason is that the fixed density (a constant in the model) suggests growing average degrees with growing network size. In contrast, many real networks constrain the maximum degree of a node inherently, as for example in social networks the number of relations an actor is able to keep active is usually limited.

Unfortunately, it is unknown if the presence of planted partitions can be decided purely on the basis of their eigenvalue distribution. In this chapter it was shown that a block structure is reflected in the distribution of eigenvalues. The opposite direction, i.e. some large eigenvalues in the order of n and the rest of low magnitude, is however not necessarily connected to a block structure in the corresponding graph.

6 Structural Graph Difference by Spectrum Transformation Cost

This chapter will motivate and define a distance between graphs that is—analog to the spectral distances described in Section 3.5—based on the comparison of eigenvalue distributions. The following considerations are focused on real valued spectra and thus undirected graphs. The generalization to directed graphs with complex eigenvalues will be considered by sketching the necessary steps for such an application.

Comparison of spectra can be considered as the assessment of similarity between two sets of points: every eigenvalue can be considered as a point in $\mathbb{R}$ respectively $\mathbb{C}$ for complex eigenvalues, i.e. a point on the line or the complex plane. Further, one could assign a certain mass to each point resulting in a model where a spectrum is a distribution of mass in some space. Given two spectra modeled in this fashion, their deviation from each other could be assessed using a physical model: assuming one of them was created by somehow "changing" the other, their difference can be defined as the amount of work incurred by such a change. Consequently, one could measure the necessary change in a physical manner as the product of mass that is moved and the distance it was moved by in the process of changing one spectrum into the other. This involves two important concepts: the weight or mass assigned to the individual eigenvalues and the distances that have to be measured in $\mathbb{R}$ or $\mathbb{C}$ to assess the overall work. The latter will be referred to as *ground distance* in the following. In summary, the difference between two spectra will be measured as the cost of transforming one into the other by the transportation of mass assigned to eigenvalues over distances to be measured in the space they reside in. The resulting dissimilarity will be denoted *spectrum transformation cost*.

This approach involves two major assumptions: (i) the distribution of eigenvalues is an expression of the structural properties of the involved graphs and (ii) the assessment of similarity in the proposed fashion reflects the so expressed structural differences appropriately. The first assumption will be justified by a summary of results relating structural properties to graph spectra in Chapter 7, while the second will be explored by experiments reported in Chapter 8.

Relation to other Approaches The approach proposed here is related to previous proposals, discussed in Section 3.5, in that some of the basic principals are shared. The fact that the same principles have lead to useful distances can be interpreted as a hint at their potential and at the same time the approach proposed here avoids some of the drawbacks of those.

A first example are the dominating eigenvalues: a (fixed size) part of the spectrum is singled out as representation for each graph and eigenvalues of the different graphs are paired up and compared individually by difference. Besides the decision to ignore a part of the spectrum, this is very similar to the approach presented here: the difference between two eigenvalues corresponds to the ground distance, while the unweighted sum over these distances would correspond to a transport of unit weight for each eigenvalue. As will be seen in the remainder, for some instances the approaches employing vectors constructed from sorted eigenvalues actually coincide completely with the proposed transformation cost.

In contrast, the histogram approach of Fay et al. (2010b), where a vector is constructed from a histogram over the eigenvalue distribution, incorporates no notion of ground distance. To stay in the model introduced here, this approach is based on congruences of mass distribution in certain areas while adjacency or ground distance between these areas is not considered. The difference can be illustrated with a small example of three spectra: $A = \{1, 1, 1, 2\}$, $B = \{1, 1, 1, 10\}$ and $C = \{1, 1, 1, 12\}$. Histograms with unit bin size would result in identical pairwise distances between all three spectra, since every distinct value ends up in its own bin. The approach introduced here, however, would suggest that B and C are much more similar to each other than either of them is to A.

The approaches of Ipsen and Mikhailov (2002) and Banerjee (2009) construct density functions by convolution of the eigenvalues with a smoothing kernel. As illustrated by Figure 6.1, this combines the comparison of masses at congruent positions with the idea of a ground distance. Yet the latter is not integrated in the assessment of distance, but

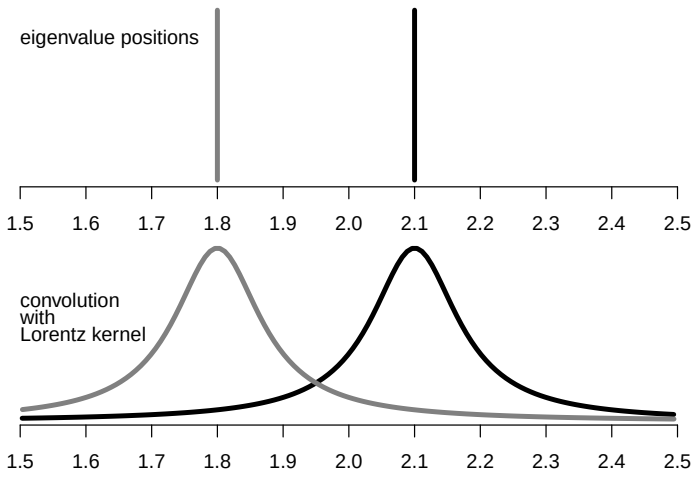


Figure 6.1: An example of two spectra (one eigenvalue each) and the result of their individual convolution with a Lorentz kernel.

only indirectly by the conversion into density functions. As the figure illustrates, the smoothing effect of the kernel achieves that differences in similar eigenvalues of the two spectra do not lead to a “non-match” as they would e.g. in the negative example above for the comparison by histograms. Instead, the continuously diminished value of the kernel around the eigenvalues results in similar function values, where the similarity of

the corresponding functions is related to the distance of the eigenvalues, which can be interpreted as the influence of ground distance.

Transformation and Modeling The idea to measure transformation costs and thus incorporate ground distance into the assessment of difference was already introduced by Shen and Wong (1983) for the comparison of histograms. In this proposal, histograms are *unfolded* by repeating values according to their multiplicity and the resulting vectors are compared entry wise. Figure 6.2 gives an example of this unfolding process and the

$$\begin{array}{lll} H_1 = & (0,1,3,3) & \qquad H_2 = \quad (3,0,2,2) \\ U(H_1) = & (1,2,2,2,3,3,3) & \qquad U(H_2) = \quad (0,0,0,2,2,3,3) \end{array}$$

$$U(H_1) - U(H_2) = (1,2,2,0,1,0,0)$$

Figure 6.2: Example for histogram unfolding and difference between two histograms as described by Shen and Wong (1983). H_i denotes a histogram and $U(H_i)$ its unfolded version.

resulting vector distance. At a first glance, this idea is not related to the approach followed here in an apparent way. However, Werman, Peleg, and Rosenfeld (1985) recognized that the difference of the unfolded histograms corresponds to an optimal matching distance in the sense it is used here. They showed that the unfolding/optimal matching approach can be generalized for histograms on the real line with the help of the δ function, also known as Dirac kernel:

Definition 6.1. *The Dirac δ function $\delta(x)$ is a continuous function satisfying*

$$\int_{-\epsilon}^{+\epsilon} \delta(x)dx = 1 \; \forall \epsilon > 0.$$

Using the δ function for convolution of the eigenvalues, analogous to the approaches of Ipsen and Mikhailov (2002) and Banerjee (2009), Werman et al. (1985) construct functions that allow the formulation of the matching idea in terms of integrals.

The δ function is used in the following to convert point distributions into functions that can be integrated. H_1 from Figure 6.2 would for example be modeled as $f_{H_1}(x) = 1\delta(x-1) + 3\delta(x-2) + 3\delta(x-3)$ and H_2 as $f_{H_2}(x) = 3\delta(x-0) + 2\delta(x-2) + 2\delta(x-3)$. The distance between H_1 and H_2 is then derived from the difference of their cumulative functions as:

$$\int_{-\infty}^{+\infty} |F_{H_1}(x) - F_{H_2}(x)| \, dx,$$

with $F_{H_1}(x) = \int_{-\infty}^{x} f_{H_1}(t)dt$ and $F_{H_2}(x) = \int_{-\infty}^{x} f_{H_2}(t)dt$ being the number of (eigen-)values in H_1 and H_2 having values less then x.

Besides these technical aspects, the modeling of spectra using δ functions allows the explicit integration of transformation and ground distance. By using this approach, mass (i.e. the integral of $\delta(x)$) is positioned in space (i.e. the real line) and transformation can account directly and independently for both. Note that in the proposals discussed so far, the ground distances is usually, often implicitly, assumed to be the one induced by $\|\cdot\|_1$-norm. As will be seen in the following, modeling and measurement of transformation cost in this way corresponds to the distance also known as *earth mover's distance*, defined e.g. in Levina and Bickel (2001). Among a number of other distances coinciding in some settings, in the field of image recognition, earth mover's distance is probably the most common name for this approach. Similar distances compare probability distributions, as e.g. Wasserstein and Mallows' distance. Relations between these are discussed in Section 6.2.

Mass Normalization The considerations about transformation of one distribution into the other assume an equal mass on both sides, i.e. $F_A(\infty) = F_B(\infty)$. In the applications on histograms this assumption corresponds to both pictures being of equal size and therefore having the same number of pixels. In terms of transformations of mass distributions, unequal masses would correspond to the addition or removal of mass during the transformation process. Levina and Bickel (2001), show that using partial matching and not accounting for unmatched mass does, at least for earth mover's distance, not result in a distance in the mathematical sense. There are at least two options to cope with this problem: (i) integrate addition and removal of mass into the distance assessment or (ii) normalize the total mass of both distributions, such that they correspond.

The first of these options would render differing graph sizes as an aspect of dissimilarity, as e.g. in edit distance (c.f. Section 3.2) by assigning costs to mass addition and removal. However, the aim of spectrum transformation cost is the measurement of structural similarity independent of size difference. Therefore, the normalization of total mass, i.e. the second option, will be followed here. Due to a lack of reasoning for the assignment of differing weights, all eigenvalues are assigned equal weights and the normalization will also be uniform, only ensuring corresponding total mass in the two distributions under comparison. For the value of this overall mass the argument of systematic influence of graph size to distance applies again: if e.g. the total masses would be normalized to value depending on the size of the involved graphs, larger graphs would tend to larger distances. To avoid dependency between graph size and distance, all mass distributions are normalized to unit overall weight, ensuring comparability independent of graph size.

As Section 7.4 illustrates, certain structural aspects of graphs result in systematic behavior of their spectra, not only in terms of the number of eigenvalues but also in the width (i.e. minimum and maximum) of the eigenvalue distribution. This aspect can be subjected to a second normalization which is considered in Section 6.4.

In the following section, spectrum transformation cost will be defined using an adaption of earth mover's distance that allows the employment of arbitrary ground distances. Subsequently, Section 6.2 describes an efficient method for its calculation and some of

its mathematical properties are reviewed in Section 6.3. Finally, choices such as matrix representation and additional normalizations being left open by the basic definition are discussed in Section 6.4.

6.1 Definition

Following Rubner, Tomasi, and Guibas (2000), the formal concept used to model the data input for the measurement of transformation cost is the *signature*:

Definition 6.2 (Signature)**.** *A signature S of size n is defined as $S = \{e_1, \ldots, e_n\}$, $e_i = (p_i, v_i)$ being a tuple of position p_i in some metric space (P, ρ) and volume or mass $v_i \in \mathbb{R}_{\geq 0}$.*

In the original definition, the e_i refer to clusters of pixels in a color distribution, e.g. the bins of a histogram, where p_i is the corresponding color value and v_i is the number of pixels in this bin. In the following, p_i will be alternatively referred to as $p(e_i)$ and analogous v_i as $v(e_i)$. As Rubner et al. (2000) show, this formalization allows to use bins of arbitrary position and extension which is more flexible than histograms with a fixed bin distribution. Note that the usage of identical positions $p_i = p_j$ introduces a formal ambiguity, e.g. (p_i, v_i) could alternatively be encoded as $(p_i, v_i/2), (p_j, v_i/2)$ without changing the actual distribution of mass, when $p_i = p_j$. Since the following considerations are only concerned with the mass distribution, two signatures S_1, S_2 are considered equal, if they result in the same total mass at each distinct position, i.e.

$$S_1 = S_2 \Leftrightarrow \forall p \in P : \sum_{e_i \in S_1 : p_i = p} v_i = \sum_{e_i \in S_2 : p_i = p} v_i \ .$$

An eigenvalue distribution will in the following be transformed into a signature, such that each eigenvalue λ_i corresponds to a position $p_i \in P$, while its multiplicity will be reflected by the mass v_i. Consequently, $P = \mathbb{R}$ for undirected and $P = \mathbb{C}$ for directed graphs. In the following considerations, however, P is arbitrary, as long as it is equipped with a distance ρ. The special case of $P = \mathbb{R}$ and $\rho(x, y) = |x - y|$ will be topic of the next section, where it will be exploited for fast computation.

As elaborated above, a prerequisite for the application of this approach is that the two distributions are of equal overall weight, which is ensured by normalizing this weight to unit value. This is expressed by the *normalized signature* defined as follows:

Definition 6.3 (Normalized Signature)**.** *A normalized signature $\overline{S}$ is derived from a signature S as*

$$\overline{S} = \left\{ e_i = \left(p_i, \frac{v_i}{\overline{v}} \right) : i \in 1, \ldots, n \right\}$$

where $\overline{v} = \sum_{i=1}^{n} v_i$.

For a set of eigenvalues $\Lambda = \{\lambda_1, \ldots, \lambda_k\}$ with m_i being the multiplicity of eigenvalue i and $n = \sum_i m_i$, the corresponding normalized signature is then: $S(\Lambda) = \{(\lambda_1, m_1/n), \ldots, (\lambda_k, m_k/n)\}$ modeling each eigenvalue as a position occupied by a mass of $1/n$.

The transformation of one (normalized) signature into another can now be expressed by the flow of mass between the different positions which is formalized by the *transformation flow*. Here, the properties expressed in the physical model are reflected by the constraints applying to admissible flow functions.

Definition 6.4 (Transformation Flow)**.** *For two signatures S_1 and S_2 a transformation flow F is a function*

$$F : \overline{S}_1 \times \overline{S}_2 \to \mathbb{R}$$

marking mass transported between positions with the following properties:

The amount of transported mass is never negative:

$$\forall (e_i, e_j) \in \overline{S}_1 \times \overline{S}_2 : F(e_i, e_j) \geq 0,$$

the mass of $\overline{S}_1$ is completely transported off:

$$\forall e \in \overline{S}_1 : \sum_{e_i \in \overline{S}_2} F(e, e_i) = p_i$$

and $\overline{S}_2$ is the result of this transportation process:

$$\forall e_i \in \overline{S}_2 : \sum_{e_j \in \overline{S}_1} F(e_j, e_i) = p_i \ .$$

The set of all transformation flows from $\overline{S}_1$ to $\overline{S}_2$ is denoted $\mathcal{F}(\overline{S}_1, \overline{S}_2)$. Finally, a transformation flow incurs cost for mass transportation, which will be used to measure the dissimilarity between the compared signatures.

Definition 6.5 (Flow Cost)**.** *The* cost *incurred by a transformation flow F between $\overline{S}_1$ and $\overline{S}_2$ is*

$$\mathit{cost}(F) = \sum_{(e_i, e_j) \in \overline{S}_1 \times \overline{S}_2} \rho(p_i, p_j) \cdot F(e_i, e_j) \ .$$

On this basis, the distance between two signatures is defined as the minimal cost incurred by any transformation flow between their normalized versions.

Definition 6.6 (Signature Transformation Cost)**.** *For two signatures S_1 and S_2 and a ground distance ρ, the distance $\mathcal{STC}^{\rho}(S_1, S_2)$ is*

$$\mathcal{STC}^{\rho}(S_1, S_2) = \min_{F \in \mathcal{F}(\overline{S}_1, \overline{S}_2)} \mathit{cost}(F).$$

With $\mathit{cost}(F)$ being measured on the basis of ρ.

In cases where ρ is clear from context or arbitrary, it will be dropped assuming it to be ρ in the latter case.

An important aspect for the application of the just defined dissimilarity is the distance property. For the case of signatures with equal total weight this was shown in Rubner et al. (2000).

Theorem 6.1. *$\mathcal{STC}$ is a distance on normalized signatures, i.e. for signatures S_1, S_2, S_3:*

(1) $\mathcal{STC}(\overline{S}_1, \overline{S}_2) \geq 0$ *and* $\mathcal{STC}(\overline{S}_1, \overline{S}_2) = 0 \Leftrightarrow \overline{S}_1 = \overline{S}_2$

(2) $\mathcal{STC}(\overline{S}_1, \overline{S}_2) = \mathcal{STC}(\overline{S}_2, \overline{S}_1)$

(3) $\mathcal{STC}(\overline{S}_1, \overline{S}_3) \leq \mathcal{STC}(\overline{S}_1, \overline{S}_2) + \mathcal{STC}(\overline{S}_2, \overline{S}_3)$

holds.

Proof. Let $\overline{S}_1 = \{e_i^1 : i \in 1, \ldots, |S_1|\}$ and analog $\overline{S}_2$ for e_i^2 and $\overline{S}_3$ for e_i^3.

(1) $\mathcal{STC}(\overline{S}_1, \overline{S}_2) \geq 0$ holds due to the definition of flow cost since neither distances nor masses are negative. Assume w.l.o.g., that positions are distinct in both $\overline{S}_1$ and $\overline{S}_2$, meaning all mass at a certain position is in each signature represented by a single entry $e_i = (p_i, v_i)$. If $S_1 = S_2$, a flow can simply be constructed by transporting mass of S_1 to the corresponding position in S_2 resulting in a zero distance and therefore in a flow cost of zero. In the other case, $S_1 \neq S_2$, there must be at least one position in $\overline{S}_1$ with a different mass than the same position in S_2 causing a necessity of transportation of a non-zero mass over a non-zero distance, resulting in flow cost > 0.

(2) Can be shown simply by inversion of the minimum cost flow from $\overline{S}_1$ to $\overline{S}_2$.

(3) Let F_{12} be a cost optimal flow from $\overline{S}_1$ to $\overline{S}_2$ and F_{23} analogous from $\overline{S}_1$ to $\overline{S}_2$. A flow transforming $\overline{S}_1$ directly into $\overline{S}_3$ can be derived by splitting the transported values into portions with exactly one source in $\overline{S}_1$ and one target in $\overline{S}_3$ and then rerouting them by omitting the detour in $\overline{S}_2$. In detail, F_{13} can be constructed as follows: Let for each $e_i \in \overline{S}_2$ the mass be partitioned by the partial sums of incoming (via F_{12}) and outgoing (via F_{23}) mass. That is, v_i is partitioned into intervals following from the partial sums:

$$g_k^- = \sum_{l=1}^{k} F_{12}(e_l^1, e_i^2) \text{ for the incoming and } g_k^+ = \sum_{l=1}^{k} F_{23}(e_l^2, e_i^3)$$

for the outgoing mass. Let further $g_1, \ldots, g_{|S_1|+|S_3|}$ be the ordered seqence of these partial sums. In other words, v_i is partitioned into intervals of sizes $[g_l, g_{l+1}]$, such that each interval has one source $e_{s(l)}^1$ in F_{12} and one target $e_{t(l)}^3$ in F_{23}, where $s(l) = \arg\min_i g_i^- > g_l$ and $t(l) = \arg\min_i g_i^+ > g_l$. Define $v_{s(l),t(l)}(e_i^2) = g_{l+1} - g_l$, then F_{13} can be constructed by setting

$$F_{13}(e_i^1, e_j^3) = \sum_{e_k \in \overline{S_2}} v_{i,j}(e_k^2) \; .$$

In F_{12}, the transport of each $v_{s(l),t(l)}(e_i)$ incurs costs of $\rho(p(e^1_{s(l)}), p(e^2_i))v_{s(l),t(l)}(e_i)$ while the cost incurred in F_{23} is $\rho(p(e^2_i), p(e^3_{t(l)}))v_{s(l),t(l)}(e_i)$. The sum of those is less or equal to $\rho(p(e^1_{s(l)}), p(e^3_{t(l)}))v_{s(l),t(l)}(e_i)$ (the transport cost in F_{13}), since the triangle inequality holds for ρ.

□

In the following, two major variants of P and ρ are considered: the general case, where P is not of concern and ρ is an arbitrary distance function and, more important for further application to undirected graphs, the case where $P = \mathbb{R}$ and $\rho(x, y) = |x - y|$.

6.2 Efficient Calculation

The first formalization of the *assignment problem* in its most general form is often attributed to Monge (1781), c.f. e.g. Gangbo and McCann (1996). Monge's formalization allows very general distributions, described by a density function. The restriction to finite spectra strongly constrains the allowed distributions which substantially simplifies the computation of optimal transformation costs.

Solutions for arbitrary distances In contrast to the general transportation problem, the problem considered here incorporates a limited number of sites where mass resides instead of an arbitrary distribution in space. Consequently, it can be formulated as a combinatorial optimization problem. In the following, two approaches for a solution in arbitrary spaces will be sketched. The main focus of this section is the efficient solution of an even more constrained problem. Therefore, possible solutions for the case of arbitrary ground distance are only indicated and not considered in more detail.

The two solutions considered for arbitrary distance functions can be translated directly to optimization problems on graphs. In addition to the finite number of sites, the first approach additionally exploits the fact that the mass distributions can be modeled alternatively by a finite number of "atomic" masses. That is, given a mass w such that each mass value is an integer multiple of w, each position can be split into the corresponding atomic masses. For two spectra with n_1 and n_2 eigenvalues one possible solution is always $w = 1/n_1n_2$. Consider two normalized signatures $S = \{e^S_1 = (p^S_1, v^S_1), \ldots, e^S_{|S|} = (p^S_{|S|}, v^S_{|S|})\}$ and $T\{e^T_1 = (p^T_1, v^T_1), \ldots, e^T_{|T|} = (p^T_{|T|}, v^T_{|T|})\}$ and let w be the common atomic weight for these two signatures. Then a bipartite graph $G(U \uplus W, E)$ can be constructed with $U = \{v^1_1, \ldots, v^1_{v_1/w}, \ldots, v^{|S|}_1, \ldots, v^{|S|}_{v_{|s|}/w}\}$ derived from S and W analogous derived from T. That is, each of the original sites is split into its atoms of weight w and for each atom a node is created. The edge set can now be used to model the possible mass transfers. Transportation is in principle possible between arbitrary nodes: $E = \{uv : (u, v) \in U \times W\}$ but the cost is dependent on the distance of the corresponding positions: $c(v^x_i, u^y_j) = \rho(p_x, p_y)$. These can be integrated as edge weights, i.e. the weight of edge $v^x_i u^y_j$ equals $c(v^x_i, u^y_j)$. Due to equal overall weights in the signatures S and T, this construction results in a equal number of atoms for both sides. Thus $|U| = |W|$ and the construction of

E ensures that G has a perfect matching. Given such a perfect matching M of minimum weight, the contained edges can be interpreted as transformation function:

$$F(e_i^S, e_j^T) = \sum_{u_k^i v_l^j \in M} c(u_k^S v_l^T) \ .$$

That is, the transported mass (in units of w) corresponds to the number of edges in M connecting nodes derived from the corresponding sites e_i^S and e_j^T. At the same time, the cost(F) is minimized, since

$$\text{cost}(F) = w \sum_{u_k^i v_l^j \in M} c(u_k^S v_l^T),$$

which is minimized by the choice of the matching. In this formulation, the problem could be solved by finding a perfect matching in a bipartite graph. A disadvantage is, however, that the node set can become quite large in comparison to the original sets of eigenvalues due to the modeling with the atomic weight w.

An alternative approach is given by the well known translation of matching problems in bipartite graphs into a *minimum-cost flow problems*. The exact problem definition and the connection between matchings in bipartite graphs and flow problems c.f. Ford and Fulkerson (1962) or, more recent Ahuja, Magnanti, and Orlin (1993). The minimum cost flow problem is given by a graph $G = (V, E)$ with designated source s and sink t, costs $c : E \to \mathbb{R}_{\geq 0}$ and flow capacities $f : E \to \mathbb{R}_{\geq 0}$. The problem to be optimized is then to send a flow of amount d from s to t, such that the amount of flow send over each edge does not exceed is flow capacity and the overall cost, given by $\sum_{uv \in E} F(uv)c(uv)$, is minimized. As before, a bipartite graph is constructed, only here the node set is limited to the actual positions of the two signatures involved and two additional nodes. The additional nodes are the source s and the sink t. The node set is then $V = \{s, t, e_1^S, \ldots, e_{|S|}^S, e_1^T, \ldots, e_{|T|}^T\}$. The source s is connected to all nodes of S and all nodes derived from T are connected to the sink t. These edges se_i^S and $e_i^T t$ ensure the mass transportation limits from above, implemented by their capacities: $f(se_i^S) = v_i^S$ and $f(e_i^T t) = v_i^T$. The actual optimization problem is then encoded by the transportation edges between the nodes of S and T. As before, the complete bipartite graph is created: $\forall i, j : \ e_i^S e_j^T \in E$ and, analogous to the matching approach, distance is expressed by edge costs: $c(e_i^S e_j^T) = \rho(p_i^S, p_j^T)$. Functions for flow capacities and costs are finally completed by setting $f(e_i^S e_j^T) = \infty$ and $c(xy) = 0$ whenever x or y are either s or t.

As noted above, these elaborations are given merely to illustrate that the proposed distance is computable in polynomial time, independently of the underlying space and distance function. Thus especially an application to directed networks and the resulting complex (i.e. two dimensional) eigenvalues is possible without modification of the basic idea.

The one dimensional case In the remainder, only undirected graphs and thus only real spectra are considered and therefore P is limited to $\mathbb{R}$, i.e. the real line. Further,

the only ground distance that will be considered is the absolute of the value difference, i.e. the one induced by the $\|\cdot\|_1$-norm. Given this constrained setting, $\mathcal{STC}$ can be determined in linear time.

As noted before, a normalized signature $\overline{S} = \{e_1 = (p_1, v_1), \ldots, e_n = (p_n, v_n)\}$ can be modeled as a density function describing the probability $P(X = p_i) = v_i$ using the Dirac kernel δ:

$$P(X = x) = f_{\overline{S}}(x) = \sum_{i \in 1,\ldots,n} v(e_i) \cdot \delta(x - p(e_i)) \; .$$

The cumulative distribution of f can then be expressed as a sum:

$$P(X < t) = F_{\overline{S}}(t) = \int_{-\infty}^{x} f_{\overline{S}}(x)dx = \sum_{i:p(e_i)<x} v(e_i) \; . \tag{6.1}$$

For $\rho(x, y) = \|x - y\|_1$, Werman et al. (1985) show, that $\mathcal{STC}^{\|\cdot\|_1}$ can be determined as the integral

$$\int_{-\infty}^{\infty} |F_{\overline{S}}(x) - F_{\overline{T}}(x)|dx.$$

In a more general consideration, Levina and Bickel (2001) show that for normalized signatures on the real line, earth mover's distance equals Mallows distance with $p = 1$. Mallows' distance is originally defined in Mallows (1972) as

$$M_p(f_{\overline{S}}, f_{\overline{T}}) = \left(\int_0^1 \left(\left| F_{\overline{S}}^{-1}(x) - F_{\overline{T}}^{-1}(x) \right| \right)^p dx \right)^{\frac{1}{p}} \; . \tag{6.2}$$

with $p = 2$ and constrained to distributions with zero mean and finite variance. Mallows distance is a special case of Wasserstein distance, c.f. Rüschendorf (2002). Restricting ground distance to $\rho(x, y) = |x - y|$ allows to exploit Mallow's formulation, i.e. Equation (6.2), in conjunction with Equation (6.1) to develop an efficient algorithm.

Due to the discrete positions, the calculation of the integral in Equation (6.2) reduces to the sum of piecewise differences between the cumulative distribution functions. Figure 6.3 illustrates the situation and the notation used in the following.

Let again $S = \{e_1^S = (p_1^S, v_1^S), \ldots, e_{|S|}^S = (p_{|S|}^S, v_{|S|}^S)\}$ and $T = \{e_1^T = (p_1^T, v_1^T), \ldots, e_{|T|}^T = (p_{|T|}^T, v_{|T|}^T)\}$ be two normalized signatures. Let further $\mathbf{e} = (e_1^S, \ldots, e_{|S|}^S, e_1^T, \ldots, e_{|T|}^T)$ their concatenation in a tupel with ith component $\mathbf{e}_i = (p_i, v_i)$ and π an permutation of $\{1, \ldots, |S| + |T|\}$ ordering the entries by their position: $\pi(i) \leq \pi(j) \Rightarrow p_{\pi(i)} \leq p_{\pi(j)}$. Using this notation, Figure 6.3 shows that $F_S^{-1}(x) - F_T^{-1}(x)$ can be derived directly from the partial sums after exchange of coordinates. Consequently, Equation (6.2) reduces to the sum

$$\int_0^1 \left(F_{\overline{S}}^{-1}(x) - F_{\overline{T}}^{-1}(x) \right)^p dx = \sum_{i=1}^{|S|+|T|} \hat{v}_i \cdot (p(\mathbf{e}_{i-1}) - p(\mathbf{e}_i))^p \tag{6.3}$$

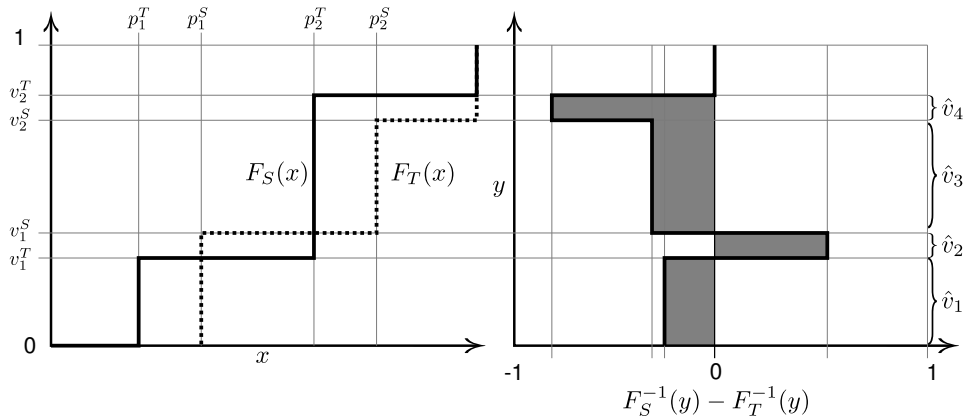


Figure 6.3: Example of distance calculation showing the variables used in the description of the algorithm for distance calculation.

with $p_0 = 0$ for notational simplicity and $\hat{v}_i$ corresponding to the cumulative distribution function:

$$\hat{v}_i = \left| \sum_{e \in \overline{S}(i)} v(e) - \sum_{e \in \overline{T}(i)} v(e) \right| \tag{6.4}$$

where $\overline{S}(i) = \{e \in \overline{S} : p(e) < p(\mathbf{e}_i)\}$ and $\overline{T}(i)$ analogous.

Using this algorithm, M_p (c.f. Equation (6.2)) for signatures sorted by positions can be determined in $O(|S| + |T|)$ by sequentially determining $|F_S^{-1}(x) - F_T^{-1}(x)|$ at each v_i. With $p = 1$, this yields an efficient algorithm for determining transformation cost in real spectra of graphs and ground distance $\|\cdot\|_1$. In addition, M_p for $p = 2$ can be determined at the same cost and method. Note that this does not correspond to the usage of $\|\cdot\|_2$ as ground distance. Exploiting this, $p = 2$ will be examined as an alternative in the experiments of Chapter 8. Simplifying notation, $\mathcal{STC}$ will in the following used as $\mathcal{STC}^1 = \mathcal{STC}^{\|\cdot\|_1}$ and $\mathcal{STC}^2$ for the case of $p = 2$.

Thus, the distance calculation for two graphs is dominated by the estimation and sorting of their eigenvalues. However, for a set of graphs the efficiency of the distance estimation is nevertheless important, since for k graphs only k sorted spectra have to be determined while (for the full distance matrix) $\binom{n}{2}$ distances have to be computed.

6.3 Mathematical Properties

One of the advantages of graph comparison via their spectra is the independence of node permutations, automatically considering isomorphic graphs as identical. The reduction of graphs to their spectra, however, entails information loss including the situation of non-isomorphic graphs with identical eigenvalues. These can consequently not be distinguished by approaches based on spectral comparison. Non-isomorphic graphs with

identical spectra exist for different (if not all) matrix representations (c.f. Haemers and Spence (2004)) and are considered in more detail in Section 8.3.

In addition to such pairs, the normalization of masses additionally widens the classes of indistinguishable structures even more. Some of these cases are summarized by the following theorem:

Theorem 6.2. *Let $G_1, \ldots, G_k$ be a set of (different) graphs resulting in identical eigenvalues for some matrix representation $X(G)$ and let $G(I) = \biguplus_{i \in I} G_i$ be the disjoint union of the respective graphs. Here I is a multiset with elements from $\{1, \ldots, k\}$ containing at least one element and the multiplicity of an element is reflected in the union, i.e. $G(I)$ is an arbitrary combination of graphs G_i.*

Then the normalized signature of $G(I)$ with respect to $X(G)$ is independent of I.

Proof. Let $\{\lambda_i : i = 1, \ldots, n\}$ be the set of eigenvalues of G_1 with eigenvalues of multiplicity > 1 repeated, then for any I, the spectrum of $G(I)$ is given by $\{\lambda_i^{|I|} : i = 1, \ldots, n\}$, i.e. every λ_i appears $|I|$ times. The normalized signature $\overline{S}$ of $G(I)$ is

$$\overline{S} = \left\{ \left(\lambda_i, \frac{|I|}{|I|n} \right) : i = 1, \ldots, n \right\} = \left\{ \left(\lambda_i, \frac{1}{n} \right) : i = 1, \ldots, n \right\}$$

and therefore equal to the normalized signature of G_1. □

Note that such unions of disjoint isospectral graphs are rarely relevant for practical applications and therefore the limitation introduced by the normalization process is usually negligible in application scenarios.

These observations show that $\mathcal{STC}$ is not a distance on the set of graphs, since two different graphs can yield a zero distance. This is not surprising, since otherwise $\mathcal{STC}$ could efficiently decide whether two graphs are isomorphic which is still unsolved as discussed in Section 1.4. Apart from that, $\mathcal{STC}$ is, however, a distance on the normalized signatures that are derived from the original graphs and thus on the equivalence classes consisting of graphs having equal signatures. The latter also holds for $\mathcal{STC}^2$.

6.4 Normalization, Variants and Notation

The next chapter will show that the size of graphs influences the eigenvalue distribution not only by determining the number of eigenvalues but also by changing their magnitude. For the structural comparison of graphs such a dependence on the size of the involved graphs is a undesirable. Therefore, despite the introduced normalization of signature weight which could be seen as a "vertical" normalization, an additional "horizontal" normalization will be considered in the following to compensate for this influence of size.

An important choice that has not yet been considered is the matrix representation used to derive the eigenvalues. The different representations result in different distributions of eigenvalues and therefore possibly differ in their suitability for different tasks.

In the following, an approach for the normalization of eigenvalues will be developed, followed by a short discussion of the choice of matrix representation and resulting notation.

6.4.1 Normalization

The theorems of Chapter 5 show that in certain situations the magnitude of some eigenvalues is connected to the size of the graph in a linear relation. In contrast, others have been shown to be related to the square root of the graph size. This illustrates two important properties: (i) graphs exhibiting strong model similarity differ systematically in their eigenvalue distributions due to size differences and (ii) the spectrum is not simply scaled by graph size in a linear fashion.

The aim of this normalization approach is to compensate differences in the spectrum that are introduced by differences in graph size. That is, in the ideal case, two graphs of vastly differing size created using the same planted partition model would result after normalization in nearly identical spectra. Due to the differing relations of eigenvalue magnitudes to graph size discussed above, this cannot be achieved with a linear transformation.

An approach that will be tested in ignorance of these arguments is the normalization of mean and standard deviation. Due to the linear nature of this transformation, the shape of the distribution is left unchanged, except for shifting and rescaling. In addition, the width of the mass distribution is hopefully normalized such that large concentrations of eigenvalues produced by the different random graph models will be mapped to comparable positions for graphs of differing size.

This normalization is in the following considered as an additional variant, differing only in the normalization of the two distributions before determination of distance. These variants will be denoted by $\overline{\mathcal{STC}}$:

Definition 6.7. $\overline{\mathcal{STC}}(S_1, S_2) = \mathcal{STC}(S_1', S_2')$ *where* $S_i' = \left\{ e_i = \left(p_i, \frac{v_i - \mu_i}{\sigma_i} \right) \ : \ e_i \in S_i \right\}$ *and* μ_i *and* σ_i *being the mean and standard deviation of the* v_i*.*

Arguments presented in the next chapter discourage the usage of this normalization step in the general application of the proposed distance. It is, however, intended as a possibility of improvement for the very special situation considering model similarity.

6.4.2 Matrix Representation and Notation

The proposed distances do not incorporate any numerical parameters to be tuned. In their application as a distance on graphs, different variants result from the choice of matrix representation. These alternative matrix representations of graphs discussed in Section 2.2 can be derived from the adjacency matrix since it contains the full information of the original graph. While this could be used as an argument to neglect other matrix representations, it might still be the case that certain aspects of structural similarity are emphasized differently by the different representations. In some matrix representations certain aspects may even be completely suppressed, which could be desirable in certain applications. Unfortunately, information about the relation of aspects of similarity to matrix representations is scarce. An exception is provided in the context of spectral clustering, e.g. in von Luxburg (2007), where different representations have been considered and compared for their applicability.

For these reasons, different matrix representations will be considered separately and the advantages and disadvantages of the individual candidates will be examined empirically in the experiments of Chapter 8. Since the number of possible representations is virtually unlimited, experiments will be limited to representations discussed most frequently in the literature and found to be useful in the derivation of structural similarity from their spectra. In particular, these are the adjacency matrix A, the Laplacian $\mathcal{L}$, and the normalized Laplacian $\overline{\mathcal{L}}$. This seems to be an unnecessary restriction, but with the two variants of $\mathcal{STC}$and their normalized versions already twelve different combinations of distance measurement and matrix representation have to be compared.

For convenience of notation, distances will be applied directly to graphs, implicitly converting them to the corresponding matrix representation and deriving the eigenvalue distribution. If not clearly indicated by the context, the employed matrix representation is denoted denoted by the subscript, e.g. $\overline{\mathcal{STC}}^1_{\mathcal{L}}(G_1, G_2)$ denotes the resulting distance measured by $\mathcal{STC}^1$ between the eigenvalues of the Laplacians $\mathcal{L}(G_1)$ and $\mathcal{L}(G_2)$ after applying the normalization proposed in the previous section.

7 Relation of Eigenvalues to Structural Properties

Approaches that assess structural graph similarity not directly but via their spectra assume that the spectrum of a graph is related to its structure. This assumption then allows to relate the similarity of spectra to similarity in the structure of the corresponding graphs and thus to assess their similarity by comparison of spectra.

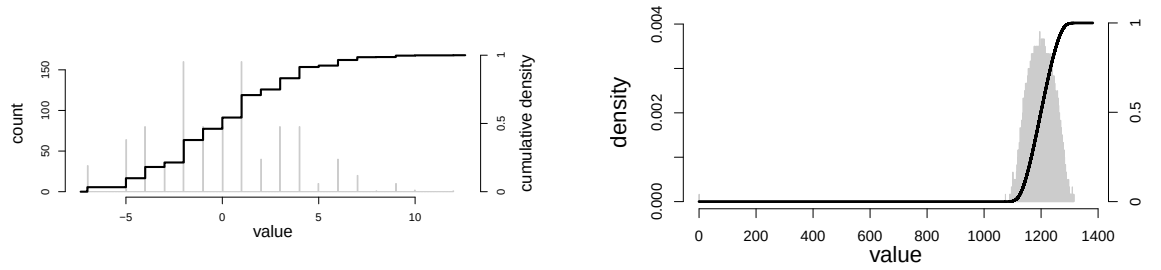
As an example, recall the approach distinguishing graphs by their underlying blockmodels proposed in Chapter 5. In this method, a concrete aspect of similarity, the blockmodel, was singled out and its reflection in the spectrum of the adjacency matrix was described. Based on this relation between underlying blockmodel and spectrum, a distance could be proposed that is tailored for the situation. The example illustrates two important aspects considered in this chapter: (i) the fact that structural property is reflected in the spectrum of a graph and (ii) the characterization of this reflection, i.e. how does this property influence the distribution of eigenvalues. While a number of results are available establishing a general relation, the exact characterization is often complicated by the fact that a number of properties are blended in an individual graph. Consequently, it cannot be guaranteed that different structures result in different eigenvalue distributions, as e.g. illustrated by isospectral graphs.

To justify the reduction of graphs to their spectra and illustrate the idea that comparison by spectra results in a somehow generic structural comparison, a number of results connecting structural properties with the eigenvalue distribution are reviewed in this chapter. In that, the aspects of structural similarity elaborated in Section 1.4 are revisited and extended by additional results that might not fit in the categories discussed there. Note that due to the large amount of material on this topic, this review is necessarily incomplete and intended as illustrative. Further, the results considered are mostly concentrating on the eigenvalues derived from adjacency matrices, following the assumption that other matrix representations yield similar relations.

The review starts with the relation of edit distances to eigenvalue changes in the next section, followed by frequent subgraphs in Section 7.2. Related to the blockmodels treated in Chapter 5 are the equitable partitions considered in Section 7.3. Model similarity in the form of random graph models and their relation to the spectrum is then reviewed in Section 7.4 and the relation between the distribution of the eigenvalues and various graph statistics is considered in Section 7.5.

A Note on the Visualization of Spectra In the discussions of spectra of graphs and their properties in the following, it is often of help to give visual impressions of particular

eigenvalue distributions. This is usually achieved with diagrams of the type shown in Figure 7.1. The two visualizations are smaller versions of Figure 7.5 and Figure 7.6(b)



(a) counts (b) density

Figure 7.1: Two example spectra showing counts of eigenvalues or density values, see text for details.

employed later in the text. They are shown here to demonstrate the two approaches of visualizing distributions of eigenvalues used in the remainder. Both of them share that the distribution is shown as histogram and, in addition, as cumulative density function. The histogram serves the purpose of giving an impression of the general distribution of values, by e.g. showing points of agglomeration. Two variants of these histograms are used throughout the text: the version showing numbers of eigenvalues in the individual bins as in Figure 7.1(a) and the version showing densities as in Figure 7.1(b). The first is used in situations where multiplicities of eigenvalues are of interest. In Figure 7.1(a) for example, each bin contains a single, repeated eigenvalue. When in contrast the relative concentration of values is of interest, relative frequency is shown as in Figure 7.1(b). The intention of using this version is to provide an insight into the often almost continuous distributions of values. Both, frequency and density values being plotted, are indicated on the left vertical axis, while eigenvalues are always mapped to the horizontal axis, with scale shown on the bottom of the diagram.

In addition to the histograms, the cumulative density is shown, its scale indicated on the right vertical axis. Besides providing a help in the visual assessment of the distribution, in some situations, the cumulative distribution highlights properties better than the histogram. For consistency, always both visualizations are provided.

7.1 Edit Operations

The spectrum of a real valued matrix is a continuous function of the matrix itself mapping an $n \times n$ matrix to $\mathbb{C}^n$ or $\mathbb{R}^n$ when the matrix is symmetric. In addition, perturbations of the form $\tilde{M} = M + E$ can be related to changes in the spectrum of M and $\tilde{M}$ using matrix norms. This is captured by Rouché's theorem (c.f. Stewart and Sun (1990, p. 167)):

Theorem 7.1. *Let λ be an eigenvalue of the symmetric matrix M with multiplicity m, $\|\cdot\|$ be any matrix norm and $\tilde{M} = M + E$. For a sufficiently small ε there is a $\delta > 0$ such that if $\|E\| < \delta$ then exactly m eigenvalues of $\tilde{M}$ are in the interval $[\lambda - \varepsilon, \lambda + \varepsilon]$.*

This relates directly to the distance on spectra proposed in the last chapter: interpreting the eigenvalues of M being shifted in the ε-interval to result in the eigenvalues of $\tilde{M}$ then the amount of shifting is measured and related to the change between M and $\tilde{M}$.

For an unweighted, undirected graph, the perturbation matrices describing edge addition or deletion (E in Theorem 7.1) are symmetric and from $\{-1, 1\}^{n\times n}$ and thus not of arbitrary small scale. However, the change in the spectrum caused by such edit operations can be bound by $\|\cdot\|_\infty$ as shown by Theorem 5.2 considering the vector of ordered eigenvalues. A more specific bound using $\|\cdot\|_2$ is given by the following corollary (c.f. Stewart and Sun (1990, p. 205)).

Corollary 7.1. *Let $\tilde{M} = M + E$ with M and E symmetric, then*

$$\sqrt{\sum_{i=1}^{n}(\lambda_i(\tilde{M}) - \lambda_i(M))^2} \leq \|E\|_F \ .$$

The Froebenius norm of a matrix, defined as:

$$\|M\|_F = \sqrt{\sum_{i,j} M_{ij}^2}$$

is for a graph G directly related to the number of edges:

$$\|A(G)\|_F = \sqrt{\sum_{i,j} |A(G)_{ij}|^2} = \sqrt{2|E(G)|} \ .$$

The same holds for the entries of a matrix describing addition and deletion of edges. This can be used to connect the change in the spectrum of G's adjacency matrix and the number of corresponding edit operations.

Corollary 7.2 (Sarkar and Boyer (1998))**.** *Let H be the graph created from G by k operations inserting or deleting an edge, with $\lambda_i^G = \lambda_i(A(G))$ and $\lambda_i^H = \lambda_i(A(H))$, then*

$$\sum_{i=1}^{n} \left(\lambda_i^G - \lambda_i^H\right)^2 \leq 2k \ .$$

Proof. Let $E = A_G - A_H$ be the matrix describing the edit operations with

$$E_{i,j} = E_{j,i} = \begin{cases} 1 & \text{, if the edge } \{i,j\} \text{ is inserted} \\ -1 & \text{, if the edge } \{i,j\} \text{ is removed} \\ 0 & \text{, else} \end{cases}$$

, then

$$\|E\|_F = \sqrt{\sum_{i,j} |E_{i,j}|^2} = \sqrt{\sum_{i,j} |E_{i,j}|} = \sqrt{2k}$$

and substition into Corollary 7.1 results in the inequality to be shown. □

Though these bounds still leave room for large changes, they establish a direct connection between changes in the edge set and the resulting change in the spectrum. In this form, the bound is strict as illustrated by comparison to the empty graph:

Corollary 7.3. *For a graph $G = (V, E)$ with $k = |E|$ and the empty graph $H = (V, \emptyset)$ on the same number of nodes*

$$\sum_{i=1}^{n} \left(\lambda_i^G - \lambda_i^H\right)^2 = 2k$$

holds with $\lambda_i^G = \lambda_i(A(G))$ and $\lambda_i^H = \lambda_i(A(H))$ as before.

Proof. Since all eigenvalues of H are zero, the difference simplifies:

$$\begin{aligned} \sum_{i=1}^{n} \left(\lambda_i^G - \lambda_i^H\right)^2 &= \sum_{i=1}^{n} \left(\lambda_i^G\right)^2 \\ \text{the relation } \lambda_i(A)^2 = \lambda_i(A^2) \text{ further yields:} & \\ &= \sum_{i=1}^{n} \lambda_i(A(G)^2) \\ &= \mathrm{Tr}(A(G)^2) \\ &= 2|E| = 2k \; . \end{aligned}$$

□

The relation between $\mathrm{Tr}(A(G)^2)$ and $|E(G)|$ is a well known result which will be revisited in Section 7.5.

Bounding on the other hand edit distance by any distance on the spectrum is - at least in this form - not possible due to the existance of pairs of nonisomorphic isospectral graphs, i.e. graphs with a non-zero edit distance but identical spectra. The relation between spectral and edit distance in the vicinity of such pairs is examined empirically in Section 8.3.

Besides these extreme cases, empirical evaluations in Zhu and Wilson (2005) indicate an approximately linear relationship between the two distances for different matrix representations. These experiments are, however, only of small scale and limited in the type of graphs under consideration. The small experiment shown in Figure 7.2 illustrates that this bound may not be so tight when none of the involved graphs is emtpy. In the experiment a graph G was drawn from $G(50, 0.5)$. To create samples of known edit distance, a random order of its edges was determined which were then step wise removed. After each removal of an edge, the spectrum was determined and compared to the spectrum of G as indicated in Corollary 7.2. This was repeated 10 times for different orders of edges and the same procedure was applied for addition of edges. The figure shows the average over the 10 repetitions. Standard deviations are omitted due to their small size.

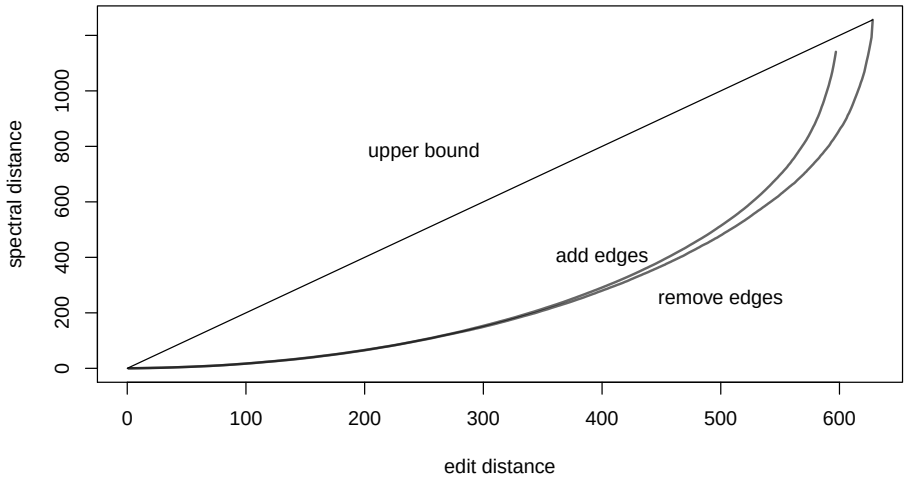


Figure 7.2: Relation between edit distance and $\|\cdot\|_2^2$ on the vectors of sorted eigenvalues. For a graph created from a $G(50, 0.5)$ edges where removed and added in random order until the empty respective full graph was reached. Shown is the upper bound (2k) and distances of the graphs resulting from random edit operations. Basis for the plot are the mean values over 10 repetitions of each experiment, each with a different order of removal/addition.

This is, however, only a very small example, showing that the upper bound established here is not necessarily tight in every situation. The relation between edit distance and spectral distance is explored in more detail empirically in Section 8.1.

An example illustrating that similar results can be achieved for other matrix representations is given for the Laplacian by Godsil and Royle (2001).

Theorem 7.2 (Godsil and Royle (2001))**.** *Let G be a graph with n vertices and let H be obtained from G by adding an edge joining two distinct vertices of G. Then $\lambda_i(\mathcal{L}(G)) \leq \lambda_i(\mathcal{L}(H))$ for all i, and $\lambda_i(\mathcal{L}(H)) \leq \lambda_{i+1}(\mathcal{L}(G))$ if $i < n$.*

That is, edge addition leads in the Laplacian of a graph to monotonic growth of eigenvalues, which is limited by the next largest eigenvalue. In addition, the relation between the trace and the sum of eigenvalues $\mathrm{Tr}(\mathcal{L}(G)) = \sum_{i=1}^{n} \lambda_i(\mathcal{L}(G))$ ensures that $\sum_{i=1}^{n} \lambda_i(\mathcal{L}(H)) = \sum_{i=1}^{n} \lambda_i(\mathcal{L}(G)) + 2$.

Not considered up to this point is the change in the spectrum introduced by addition and removal of nodes. Here, a characterization in terms of vector distances in the same fashion as above is not possible, since the number of eigenvalues corresponds to the number of nodes and thus changes. A qualitative characterization of eigenvalue changes due to node removal is given by Cauchy's interlacing theorem, here in a version adapted for graphs (c.f. Cvetković, Doob, and Sachs (1995)):

Theorem 7.3 (Interlacing Theorem)**.** *Let G be a simple, undirected graph and H be a subgraph, derived by deleting nodes and their adjacent edges from G. Let further $\lambda_1^G \leq \ldots \leq \lambda_n^G$ be the spectrum of $A(G)$ and $\lambda_1^H \leq \ldots \leq \lambda_m^H$ the spectrum of $A(H)$. Then the inequalities*

$$\lambda_{n-m+i}^G \leq \lambda_i^H \leq \lambda_i^G$$

hold for $i \in \{1, \ldots, m\}$.

That is, the deletion of a node results in eigenvalues that are enveloped by the eigenvalues of the original spectrum and thus the change in the spectrum introduced by node deletion is limited by the eigenvalue distribution of the original graph. Since the theorem was originally considering principal submatrices, i.e. the original matrix with row i and column i removed for some i, it holds for all symmetric matrix representations G in which the deletion of a node results in no more change than the deletion of the corresponding row and column.

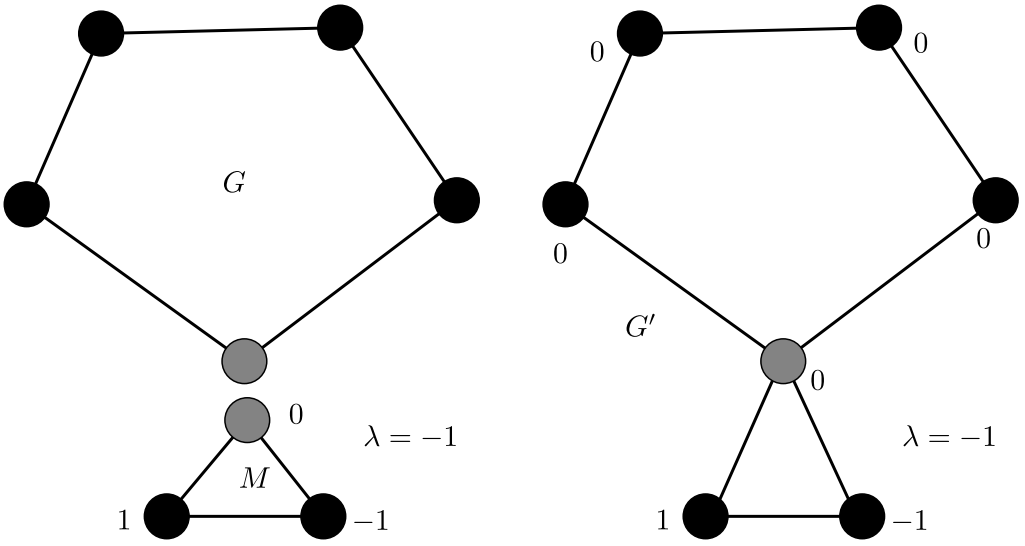
Summarizing, the results reviewed in this section illustrate two major points: (i) graph similarity measured by edit distance can be related to changes in the spectra (ii) the resulting changes can be interpreted as shifts of the involved eigenvalues and thus a measurement of this "eigenvalue movement" relates well to edit distance.

7.2 Frequent Subgraphs

Another structural property of a graph is the frequent ocurrence of certain substructures. In the case that these are not connected and the graph constist in fact of a number of connected components, the resulting spectrum of the adjacency matrix is simply the combination (union of multisets) of the spectra of the individual components. However, this relation is not that simple when the substructures of interest are connected to the rest of the graph. Since this is usually the more interesting case it will be the main focus of this section. While it is hard to establish statements about the influence of substructures on the spectrum of the graph in the general case it is at least possible for two boundary cases: (i) when the substructures are very small and (ii) when the considered substructures are only minimally connected to the rest of the graph. While the first situation is considered in Section 7.5 for edges and triangles, the attachement of structures to single nodes of the graph is described by the notion of *grafting* and considered in the following.

The observation of this effect was first described in Kirkpatrick and Eggarter (1972), and formalized in Baltz and Kliemann (2005) using graph notation. In particular the situation where a component is added to a graph by either merging two nodes from the two graphs or by adding 2 copies of a component and connecting them to the same node, allows to partially describe the resulting changes in the spectrum.

The first variant of attachment is illustrated in Figure 7.3. Theorem 7.4 shows, that under certain circumstances, the eigenvalues of the attached component become eigenvalues of the resulting graph.



(a) Graph G and added component M (b) G' from G joined with M

Figure 7.3: A component M is added to a graph G. The two graphs are joined at the gray node (j and v in Theorem 7.4). Numbers at nodes show the eigenvector $\mathbf{a}$ (left) and $\mathbf{b}$ (right).

Theorem 7.4 (Baltz and Kliemann (2005))**.** *Let $G = (V, E)$ and $M = (V_M, E_M)$ be two graphs with disjoint node sets. Let further λ be an eigenvalue of $A(M)$ with a corresponding eigenvector $\mathbf{a}$, such that $\mathbf{a}_j = 0$ for some $j \in V_M$. Now let $G' = (V', E')$ be obtained by joining M to G merging $j \in V_M$ and some $v \in V$ in the following way:*

$$V' = V \cup (V_M \setminus \{j\}) \tag{7.1}$$

$$E' = E \cup E_M \setminus \{kl : k = j \vee l = j\} \cup \{vl : jl \in E_M\} \tag{7.2}$$

Then λ is an eigenvalue of G'.

Proof. The proof is obtained by constructing an eigenvector $\mathbf{b}$ of G' that is localized on the nodes of V_M:

$$\mathbf{b}_i = \begin{cases} \mathbf{a}_i,\ i \in V_M \\ 0, \text{ else} \end{cases}$$

Consequently, $\mathbf{b}$ is an eigenvector of G' corresponding to the eigenvalue λ. □

The second variant is based on identical subgraphs being connected to the main graph in an equal fashion. It will be denoted *component repetition* and is defined as follows:

Definition 7.1 (Component Repetition)**.** *A subgraph M' repeats a component M in $G = (V, E)$ $\Leftrightarrow$*

1. *M and M' are induced subgraphs of G*
2. *M is isomorphic to M' with an isomorphism $\pi : V(M) \to V(M')$.*

3. *each edge $mv \in E$ between nodes $m \in V(M)$ and $v \in V \setminus (V(M) \cup V(M'))$ has a corresponding edge $\pi(m)v \in E$ and vice versa:*

$$mv \in E \Leftrightarrow \pi(m)v \in E$$

4. *there are no direct edges between the nodes of M and those of M'.*

That is, M and M' are isomorphic, induced subgraphs of G with the same neighborhood and no edges between them.

Baltz and Kliemann (2005) show, that such a component repetition results in a graph having all the eigenvalues of the isolated components at least once, i.e. besides side-effects (other additional eigenvalues) the spectrum of G is extended by the eigenvalues of M. Their result can be be extended to an arbitrary number of repetitions causing repeated eigenvalues with according multiplicities of M's eigenvalues:

Theorem 7.5. *Let M_0 be an induced subgraphs of G and $M_1, \ldots, M_m$ be its repetitions. Then each eigenvalue of $A(M_0)$, is an eigenvalue of $A(G)$ with multiplicity at least m.*

Proof. Let $l = |V(M_0)|$ and $V(G) = \{v_1, \ldots, v_n\}$ with $V(M_i) = \{v_{il+1}, \ldots, v_{il+l}\}$. Further, let $\mathbf{e}$ be an eigenvector of $A(M_0)$ corresponding to the eigenvalue λ and for a vector $\mathbf{a} \in \mathbb{R}^{m+1}$ let

$$\mathbf{e}^{\mathbf{a}} = (\mathbf{a}_1 \cdot \mathbf{e}^T | \ldots | \mathbf{a}_{m+1} \cdot \mathbf{e}^T | 0, \ldots, 0)^T \ ,$$

i.e. the concatenation of $m+1$ times $\mathbf{e}$ (one corresponding to each M_i) weighted by the components of $\mathbf{a}$ and padded with zeros to obtain a vector of length n. Then multiplication with $A(G)$ yields:

$$A(G)\mathbf{e}^{\mathbf{a}} = \mathbf{b} = \left(\lambda \mathbf{a}_1 e^T, \ldots, \lambda \mathbf{a}_{m+1} e^T, \mathbf{b}_{(m+1)l+1}, \ldots, \mathbf{b}_n\right)^T$$

i.e. for the part of $A(G)\mathbf{e}^{\mathbf{a}}$ up to $\mathbf{b}_{(m+1)l+1}$, the eigenvalue relation holds independently of $\mathbf{a}$. Consequently, it is only left to be shown, that this relation also holds for the $\mathbf{b}_i$ with $i > (m+1)l$. These are determined directly by the edges between the M_i and the rest of G:

$$\mathbf{b}_i = \sum_{j=0}^{m} \mathbf{a}_j \sum_{k : v_k \in N_j(v_i)} \mathbf{e}_{(k \bmod l)} \ ,$$

with $N_j(v_i) = N(v_i) \cap V(M_j)$ being the neighbors of v_i in M_j. Since all M_j have identical neighbors in G, the inner sum is independent of j and all $\mathbf{b}_i = 0$ $(i > (m+1)l)$ if the components of $\mathbf{a}$ sum up to zero, i.e. $\mathbf{a} \perp \mathbf{1}$.

Further, for each vector $\mathbf{a}' \perp \mathbf{a}$ yields an orthogonal eigenvector $\mathbf{e}^{\mathbf{a}'}$, since

$$\mathbf{e}^{\mathbf{a}} \mathbf{e}^{\mathbf{a}'} = \sum_{i=1}^{m+1} \mathbf{a}_i \mathbf{a}'_i \left(\sum_{j=1}^{l} \mathbf{e}_j^2 \right) \ .$$

Consequently, there are m pairwise orthogonal vectors $\mathbf{a}^i \in \mathbb{R}^{m+1}$ with $\mathbf{a}^i \perp \mathbf{a}^j$ for $i \neq j$ and all $\mathbf{a}^i \perp \mathbf{1}$. Each $\mathbf{a}^i$ representing an eigenvector of $A(G)$ corresponding to λ and thus the multiplicity of λ is at least m. □

With zero being the eigenvalue of a single node, this also relates to the fact that structural equivalent nodes lead to increased multiplicities of the eigenvalue zero. Structural equivalence (cf. Lorrain and White (1971)), refers to nodes with identical neighbors. In $A(G)$ they induce identical rows and columns, thereby decreasing the rank of $A(G)$.

Due to the change in the diagonal, this theorem is not directly transferable to the variants of the Laplacian. However, Banerjee and Jost (2008) consider a similar effect on the spectrum of the degree normalized Laplacian $\overline{\mathcal{L}}'$. With a technique similar to the proof of Theorem 7.5, they relate the repetition to the multiplicity of eigenvalue 1 of $\overline{\mathcal{L}}'(G)$. More exactly, they show that for a repetition of a component the multiplicity of the eigenvalue 1 is increased by 1.

The contents of this section illustrates that there is a direct influence of repeated substructures to the spectrum of a graph, at least, when these substructures are connected repeatedly to the same node. However, these results consider only the boundary of the graph, since they are based on equalities in the rows and columns of the adjacency matrix belonging to the considered substructures. As mentioned before, for very small substructures (edges and triangles) the influences on the distribution of eigenvalues are considered in Section 7.5.

7.3 Divisors and Equitable Partitions

Chapter 5 relates a stochastic model describing edge probabilities by class memberships of nodes to the spectrum of the expected adjacency matrix of the resulting random graphs. A discrete version of this model can be created by fixing for each node the exact number of neighbors in each class. Fixing the number of neighbors only depending on the class of the node, i.e. each node in a class has the same number of neighbors in each (including its own) class, leads to the notion of an *equitable partition*. This concept is connected to structural node similarities as elaborated in Lerner (2007).

In spectral graph theory, this is captured by the *divisor* of a graph. The following definition and theorem are taken from Cvetković et al. (1995), where this topic is treated in more detail, while this review is restricted to illustrate how it can be used to systematically produce structured graphs with certain eigenvalues. Therefore, the concept of *graph multiplication* is introduced to demonstrate how regularities in the graph structure correspond with eigenvalues.

Definition 7.2 (Divisor)**.** *Let $D \in \mathbb{N}^{d\times d}$ be a matrix and G a simple graph with $n = |V(G)|$. D is called a divisor of G, written $D|G$, if there is a partition $\mathcal{C}$ of $V(G)$ with $d = |\mathcal{C}|$ classes $C_1, \dots, C_d$ such that $\forall j\ \forall v \in C_i : |N(v) \cap C_j| = D_{ij}$ holds.*

In words, D_{ij} is the number of neighbors that each node in C_i has in class C_j. For each graph its adjacency matrix is a trivial divisor. The divisor relation is also transitive: if $A(F)|G$ and $A(G)|H$ then $A(F)|H$.

A divisor that is smaller than the corresponding graph, i.e. $d < n$, reveals information about the structure of the graph by assigning node classes and connection regularities

based on these. This regularity in the graph structure is reflected in its eigenvalues, as shown by the following theorem:

Theorem 7.6 (Cvetković et al. (1995))**.** *Let G be a graph with a divisor D, then the characteristic polynomial of G is a multiple of the characteristic polynomial of D, i.e.*

$$det(D - \lambda I) \,|\, det(A(G) - \lambda I) \ .$$

Consequently, every eigenvalue of D is an eigenvalue of $A(G)$.

Used in the inverse direction, this relation can be employed to produce graphs with systematic properties in their spectra. This operation will be denoted *graph multiplication* and starts from a source graph G and a connection pattern A to produce a new graph in the following way:

Definition 7.3 (Graph Multiplication)**.** *Let G be a d-regular graph and $A = A(P)$ for some simple graph P with k nodes. Then the* product $A \bullet G$ *is defined as follows: Let $G_1, \ldots, G_k$ be k copies of G with G_i and G_j connected by isomorphism $\phi_{i,j}$. Then*

$$\begin{aligned} V(A \bullet G) &= \bigcup_{i=1}^{k} V(G_i) \\ \textit{and} & \\ E(A \bullet G) &= \bigcup_{i=1}^{k} E(G_i) \cup \bigcup_{i,j:A_{ij}=1} \{u\phi_{ij}(u) : u \in V(G_i)\} \end{aligned}$$

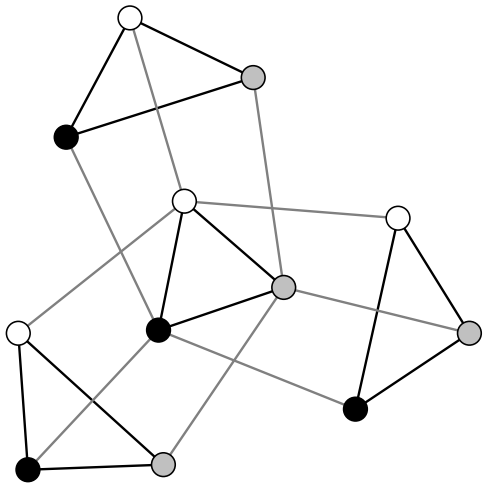


Figure 7.4: A triangle (K_3) multiplied by a star on 4 nodes ($K_{1,3}$).

Consider the example in Figure 7.4, where a K_3 (white, gray and black nodes joined by black edges) has been multiplied by a star $K_{1,3}$. The eigenvalues of K_3 are $-1, 1, 0$ those of the $K_{1,3}$ are $\pm\sqrt{3}$ and 0 two times, while the resulting eigenvalues of $K_{1,3} \bullet K_3$ are $(2 + \sqrt{3}), 2_2, (\sqrt{3} - 1)_2, 2 - \sqrt{3}, -1_4, (-\sqrt{3} - 1)_2$, multiplicities indicated by lower index.

Since G is d-regular, $A' = A + dI$ is a divisor of G and thus for every eigenvalue λ of A there is an eigenvalue $\lambda + d$ of A' and by Theorem 7.6 also of $A \bullet G$. In the example of Figure 7.4, the K_3 is 2-regular and thus the eigenvalues $\pm\sqrt{3}$ and 0 result in $2 + \sqrt{3}$, $2 - \sqrt{3}$ and 2 of $K_{1,3} \bullet K_3$.

If, in addition, P is regular then also $A \bullet G$ is regular since every node is connected to the same number of self-copies. In this case, the multiplication can be applied repeatedly, maintaining the conditions of Theorem 7.6. Such a repeated application of a pattern will be denoted by

$$A \bullet^k G = \underbrace{A \bullet \ldots \bullet A}_{k \text{ times}} \bullet G \,.$$

Exploiting the regularity of P and G, Theorem 7.6 can be used to predict a number of eigenvalues of the result of graph multiplication.

Corollary 7.4. *Let G be a d-regular graph and A be the adjacency matrix of a regular graph. Then the spectrum of $A \bullet^k G$ contains every eigenvalue of G and for every eigenvalue λ of A and $i = 0, \ldots, k-1$, $\lambda + d + i$ is an eigenvalue of $A \bullet^k G$.*

This is a direct consequence of Theorem 7.6 and the fact that $A + \mathrm{diag}(d)$ is a divisor of $A \bullet G_0$ under the conditions of the corollary. Note that this characterization is neither complete with respect to multiplicities nor does it rule out the existence of other distinct eigenvalues. In fact, it renders a special case of a divisor involving even more regularity in the sense that nodes between the different groups are connected systematically, i.e. to copies of themselves, while this is not necessarily the case in a graph having a divisor.

An example of the eigenvalues resulting from repeated graph multiplication is shown in Figure 7.5.

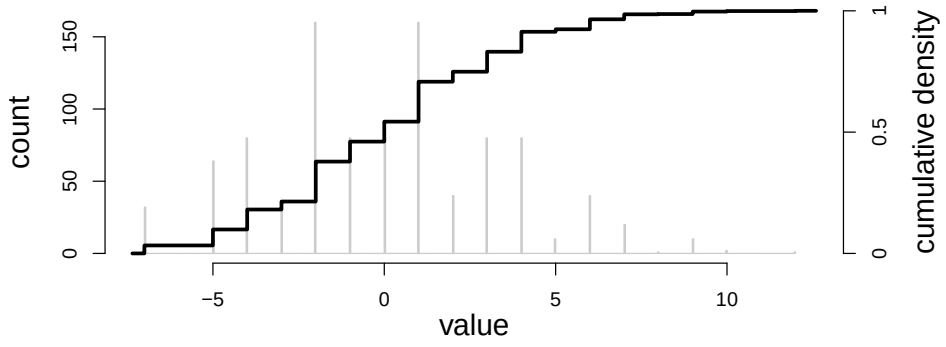


Figure 7.5: The spectrum of $G = C_4 \bullet^5 K_3$.

Similar recursive construction schemes for graphs have been proposed by Dorogovtsev, Goltsev, and Mendes (2002) and Comellas, Fertin, and Raspaud (2004), who show that such constructions are in principle capable of reproducing a number of properties observed

in real-world graphs. In contrast, this is not related to the notion of recursive graphs as defined in Bean (1976).

A prominent example for a family of graphs that can be constructed by this mechanism is the hypercube Q_n. Starting from a graph G_0 with a single node, Q_n can be constructed by repeated application of $A(P_2)$:

$$Q_n = \begin{cases} G_0 & \text{, if } n = 0 \\ A(P_2) \bullet^n G_0, & \text{if } n > 0 \end{cases} .$$

Due to this construction and the transitivity of divisors $A_i = A(P_2) + iI$ is a divisor of Q_n for $i = 1, \ldots, n$ and thus the spectrum of Q_n contains every distinct eigenvalue of A_i, i.e. $i+1$ and $i-1$ at least once for every i. In fact, a Q_n contains the eigenvalue $(n-2i)$ with multiplicity $\binom{n}{i}$ for each $i \in \{0, \ldots, n\}$ (c.f. e.g. Harary, Hayes, and Wu (1988)). In that respect, the existence of a divisor, the construction scheme of Corollary 7.4 and the Q_n reveal an increasing amount of information about the spectrum accompanied by increasingly detailed description of their structure.

The divisor and the construction of graphs using graph multiplication relates to approaches capturing structural node similarities in social network analysis, c.f. Lerner (2005, 2007). Considering the perturbation theorems mentioned in Section 7.1, this relation is also relevant for observed and thus noisy graphs, where a crisp divisor is usually not expected. It thereby connects to the stochastic blockmodels considered in Section 2.5 and Chapter 5.

7.4 Asymptotic Eigenvalue Distributions

A model that can be interpreted as a special case of the planted partition model discussed in Chapter 5 is the $G(n,p)$ - a model with a single partition. In the following, $G(n,p)$-models with constant p and models with $p \sim 1/n$ will be considered seperately. This distinction is based on two related graph statistics: constant p results in an expected constant density while $p \sim 1/n$ results in graphs with a constant expected node degree. In addition, different results are available for the two situations. While for constant p, the eigenvalue distribution can be charaterized asymptotically, for $p \sim 1/n$ only observations can be shown together with an hypothesis about the underlying mechanism.

Constant Expected Density Wigner first characterized the asymptotic behavior of symmetric random matrices with entries from $\{-1, 1\}$ in Wigner (1955, 1958) known as Wigner's Semicircle-Law. In addition, Ding and Jiang (2010) develop extensions of such characterizations for the adjacency and the Laplacian matrix. For a summarization, the following definition is useful:

Definition 7.4 (Distribution and Cumulative Distribution)**.** *For a symmetric matrix M and the Dirac function δ (c.f. Definition 6.1, p. 67)*

$$\hat{\mu}^M(x) = \frac{1}{n} \sum_{i=1}^{n} \delta(x + \lambda_i(M))$$

is the empirical spectral distribution *of* M *and*

$$F^M(x) = \frac{1}{n}|\{\lambda_i(M) : \lambda_i(M) \leq x\}|$$

is the empirical spectral cumulative distribution function *of* M.

Using $\hat{\mu}^M$ and F^M, the results of Ding and Jiang can be summarized in the following two corollaries. The similarity between Definition 7.4 and the considerations for the efficient calculation of spectrum transformation cost in the last chapter directly relate these results to the distance measurement proposed here.

Corollary 7.5 (Ding and Jiang (2010))**.** *Let* $p_n \in [0,1]$ *and* $\alpha_n = \sqrt{(np_n(1-p_n))}$, $\alpha_n \to \infty$ *as* $n \to \infty$ *and* $A_n = A(G(n,p_n))/\alpha_n$. *Then almost surely* F^{A_n} *converges weakly to the semicircle with density* $\hat{\mu}^{A_n}(x) = \frac{1}{2\pi}\sqrt{4-x^2}$.

Note that the constraint on α_n excludes the case of $p_n = p/n$ for some fixed p. In addition, Ding and Jiang (2010) establish for the maximal eigenvalue $\lambda_n^{A_n}$ and constant p_n the relation

$$\frac{\lambda_n^{A_n}}{np_n} \to 1 \text{ almost surely, as } n \to \infty\ .$$

Even sharper estimations for this distribution are given in Füredi and Komlós (1981) suggesting that with probability $1 - o(1)$ only a single eigenvalue will leave the semicircle interval. An approach to compute the moments of the eigenvalue distribution for $G(n,p)$s can be found in Bauer and Golinelli (2001). For the spectrum of the Laplacian a similar result is available:

Corollary 7.6 (Ding and Jiang (2010))**.** *Let* $\mathcal{L}_n = \mathcal{L}(G(n,p_n))$ *and*

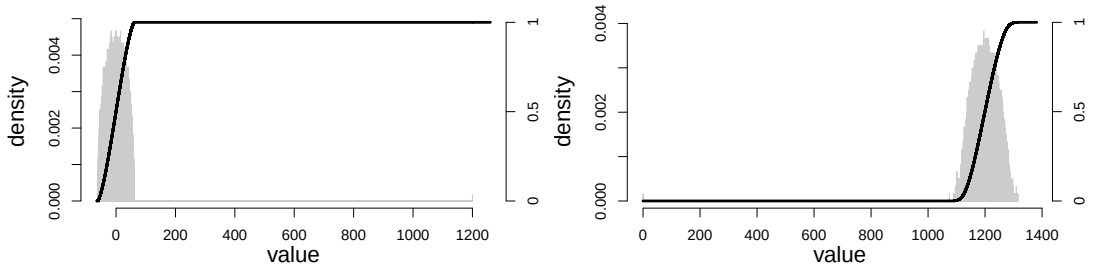
$$\tilde{F}_n(x) = \frac{1}{n}\left|\left\{\frac{\lambda_i(\mathcal{L}_n) - np_n}{\sqrt{n}(p_n - p_n^2)} \leq x\right\}\right|,\ x \in \mathbb{R}\ .$$

Then $\tilde{F}_n(x)$ *converges weakly to the free convolution* γ_M *of the semicircular law and the standard normal distribution. Further, the maximal eigenvalue of* $\mathcal{L}^n$ *is determined by*

$$\frac{\lambda_n(\mathcal{L}_n)}{np_n} \to 1$$

in probability.

γ_M is a distribution that is not directly defined, but derived in Biane (1997) and characterized by its moments in Bryc, Dembo, and Jiang (2006). Again, this characterization is valid for constant p_n, while it does not hold for $p_n = p/n$. Figure 7.6 gives an example for the spectra of the adjacency and Laplacian matrix of a large $G(n,p)$-graph illustrating Corollaries 7.5 and 7.6. The difference in the two distributions, besides the obvious difference in the position of the largest eigenvalues concentration, can be observed at the left and right borders of the respective agglomerations. For the adjacency



(a) adjacency matrix eigenvalues (b) Laplacian matrix eigenvalues

Figure 7.6: Eigenvalue distributions for the adjacency matrix and the Laplacian of a graph drawn from a $G(4000, 0.2)$-model.

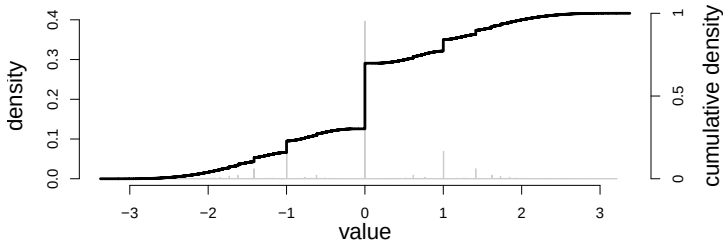
matrix the drop in the density of eigenvalues is more sudden, than in the corresponding position in the density of Laplacian eigenvalues. Alternatively, this can be observed in the cumulative density function, which is considerably smoother for the Laplacian than for the adjacency eigenvalues.

Together, these results establish a relation between the model similarity considered in Section 1.4 and the spectra of graphs drawn from these models. Though these results are restricted to one model and two different matrix representations, an example for a close relation between model similarity and eigenvalue distribution has been established.

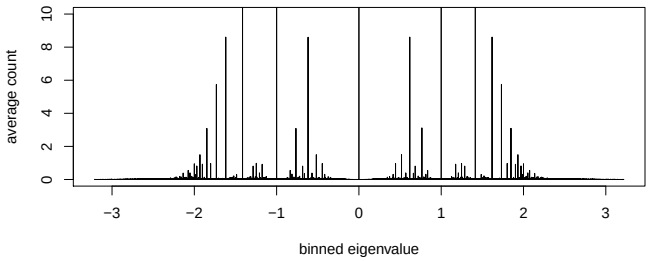
In particular, the adjacency matrix of a $G(n, p)$ tends to have one dominating eigenvalue accompanied by $n - 1$ semi-circle distributed eigenvalues. Chapter 5 shows that for planted partition models the situation is similar in that each partition results in an asymptotically dominating eigenvalue. There, however, the distribution of the smaller eigenvalues is not considered.

For the assessment of model similarity by spectrum transformation cost, these results illustrate the problem with model similarity already discussed in Section 6.4 in more detail. Settings corresponding to this problem will be revisited in the experiments of Section 8.2.

Fixed Expected Degree None of the above characterizations cover the case of fixed expected node degree as in graphs drawn from a $G(n, p/n)$ model. Experiments in Bauer and Golinelli (2001) show that such low density random graphs tend to have "spikes" in their spectral distributions, due to eigenvalues that appear with a much higher multiplicity than others. Figure 7.7 shows an example of this situation in the spectrum of a graph from a $G(1000, 1.2/1000)$. The large view, i.e. Figure 7.7(a), illustrates, that the point wise concentrations of eigenvalues dominate all the others in their multiplicity, i.e. especially they determine to a large extend the shape of the cumulative density function. The second, enlarged version in Figure 7.7(b), then shows at the same horizontal but different vertical resolution details about the points of concentration. Note that to achieve the necessary resolution, larger spikes, e.g. at zero are not shown in full extend, but cut off at ten repeated eigenvalues. This detailed view reveals, that the concentrations of



(a) signature



(b) enlarged version

Figure 7.7: Spectrum of a $G\,(1000, 1.2/1000)$ averaged over 100 repetitions.

eigenvalues on certain points is sharp, i.e. not a large number of values are near zero, but the multiplicity of the eigenvalue zero is large. The same holds for other points of concentration. In addition to the values considered explicitly in the following, this effect seems to be repeated with additional eigenvalues at smaller multiplicities, though this was not explicitly verified and could be a random effect.

An explanation for these spikes due to Bauer and Golinelli (2001) are small trees that appear as isolated components or attached to greater components as in Theorems 7.4 and 7.5. Indeed, the 9 highest spikes (average multiplicity > 5) are exactly the eigenvalues of the trees P_2, P_3, P_4 and $K_{1,3}$ with the larger spikes corresponding to the eigenvalues of the smaller trees, which was also reported by Bauer and Golinelli. The only exception of this scheme is the eigenvalue 0 which can be explained by single nodes, either isolated or appearing as multiple leaves connected to the same node. As Bauer and Golinelli (2001) further show, these spikes appear independent of the graph size while at the same time their normalized densities differ only marginally.

7.5 Moments of the Eigenvalue Distribution

An important characeristic of a distribution are its moments, e.g. mean, deviation, skewness. Considering the eigenvalues of the adjacency matrix A, the moments of their distribution can be related to the structure of the graph via its powers A^n. In the following, this will be exploited to relate the moments of the eigenvalue distribution to graph properties.

Let in the following $A = A(G)$ for an arbitrary (simple, undirected) graph G, $\lambda_i = \lambda_i(A)$ and M_k be the kth central moment of the discrete distribution formed by the λ_i, i.e.

$$M_k = \frac{1}{n}\sum_{i=1}^{n}(\lambda_i - \langle\lambda_i\rangle)^n$$

for $k > 1$ and $M_1 = \langle\lambda_i\rangle$. Based on the relation $\mathrm{Tr}(M) = \sum_{i=1}^{n}\lambda_i(M)$ for $M \in \mathbb{C}^{n\times n}$ (c.f. e.g. Golub and Van Loan (1996)), $M_1 = \langle\lambda_i\rangle = \mathrm{Tr}(A) = 0$ can be inferred directly. With all moments being centered at zero, the general case yields:

$$M_k = \frac{1}{n}\sum_{i=1}^{n}\lambda_i^k = \frac{1}{n}\mathrm{Tr}(A^k)\ .$$

Note that $A^k_{i,j}$ corresponds to the number of walks of length k between nodes i and j and $\{\lambda_i^k : i = 1,\ldots,n\} = \{\lambda_i(A^k) : i = 1,\ldots,n\}$ for $k \geq 1$. Useful are especially the cases $k = 2$ and $k = 3$, since they relate the eigenvalue distribution to the degree distribution and the density (cf. Section 3.1) of G:

$$\begin{aligned} n \cdot M_2 &= \sum_{i=1}^{n}\lambda_i^2 = \mathrm{Tr}(A^2) = 2|E| \\ &= \sum_{v\in V} d(v) = \rho(G)\cdot(n-1)\ . \end{aligned}$$

M_3 then relates the skewness of the eigenvalue distribution to the number of triangles $\Delta(G)$ in G:

$$n \cdot M_3 = \sum_{i=1}^{n}\lambda_i^3 = 6\Delta(G) \tag{7.3}$$

In terms of walks, a triangle is a walk from a node back to itself via one other node, counted in two directions. In addition, walks of this kind are counted for each of the participating nodes resulting in the factor 6.

Chung, Lu, and Vu (2003) consider graphs with an exponential degree distribution, i.e. the number of nodes with degree k is proportional to $k^{-\beta}$ for some β. They show that under certain conditions the exponential distribution of degrees results in an exponential distribution of the largest eigenvalues, thus illustrating that degrees and eigenvalues are related beyond the connection of their moments.

Summarizing, this section shows that the moments of the eigenvalue distribution - a very descriptive property of a distribution - are directly connected to structural features of the graph.

8 Empirical Studies on Spectrum Transformation Cost

The previous chapter summarized a number of arguments, why the spectral distances proposed in Chapter 6 are expected to be sensitive to structural properties of graphs. This chapter will explore the applicability of the proposed distances to a number of aspects of structural similarity in a empirical, quantitative fashion.

Experiments are conducted to explore

1. the relation of $\mathcal{STC}$ to the different notions of structural similarity with a special focus on edit distance and model similarity;
2. the behavior on graphs close to isospectral graphs;
3. the applicability on observed graphs of small size;
4. the relation to alternative distances on graph spectra and comparison of performance on practical problem sets.

The first two sections are motivated by the aspects of structural graph similarity discussed in Section 1.4.2. Experiments in this part are designed to assess how the different aspects, in particular edit distance and model similarity, are reflected by distance measurement using spectrum transformation cost.

As mentioned before, some pairs of different graphs result in identical sets of eigenvalues. Section 8.3 examines the (edit distance-) neighborhood of such graphs in terms of the behavior of the spectra of neighboring graphs.

Related to practical applications are the experiments on graphs involving very few nodes. The fact that the measurement of transformation costs between spectra exploits almost all available information contained in the eigenvalue distribution is supposed to enable their successful application to arbitrary small graphs. This is tested in Section 8.4.

Finally, Section 8.5 compares the distance measurement via transformation costs to the previously proposed spectral distance measurements that were discussed in Section 3.5.

8.1 Relation to Edit Distance

Recall from Section 3.2, that edit distance between two graphs is defined as the number of nodes and edges that have to be removed and inserted in one graph to change it into the other. The fact that graphs of zero edit distance are isomorphic shows that no efficient method is yet known for its determination.

Edit distance is related to change in the eigenvalue distribution by an upper bound as shown by Corollary 7.2 and therefore directly to $\mathcal{STC}_A^2$. The experiments of Zhu and Wilson (2005) and those in Section 7.1 further show that the squared $\mathcal{STC}_A^2$ is not necessarily close to this theoretical upper bound.

In the following, these experiments are extended to other variants of transformation cost. The intention is, however, not to find the variant that provides the best approximation for edit distance but rather to explore its relation to the various variants. Due to the established relation of edit distance and the spectrum of the adjacency matrix, the experiments on this section are limited to this graph representation.

Since the determination of edit distance is a problem in general, the following experiments employ graphs that are constructed by adding and removing nodes and edges to and from an initial graph. Thereby, edit distance to the starting graph can always be derived directly from the construction, as long as the chain of edit operations is limited to either only add or only remove operations. In the measurement of the actual edit distance, no special costs for the individual operations are taken into consideration but every edit operation is charged with unit cost, independent of its type. To incorporate the influence of differing graph sizes in the experiments, all isolated nodes in the constructed graphs are removed before distance calculation.

As elaborated in the last chapter, the expected distribution of eigenvalues is, at least in some cases, strongly connected to the random graph distribution samples are drawn from. Therefore, the following experiments are executed on two differently constructed sets of graphs: one following a uniform distribution of edges among the pairs of nodes, the other using a graph drawn from a preferential attachment model as reference. The idea is to extend measurements to graphs that are dissimilar to those that are expected to be drawn from a $G(n,p)$-model. While edges to be added and removed are chosen uniformly among all possibilities, using a preferential attachment graph as in the second experiment ensures that possible influences of this special graph structure are considered in the experiment. This could also be achieved by an adaption of the edge selection process, but repetitions of the edge sampling process to ensure stability of measurements under random fluctuations are simplified by the approach taken here.

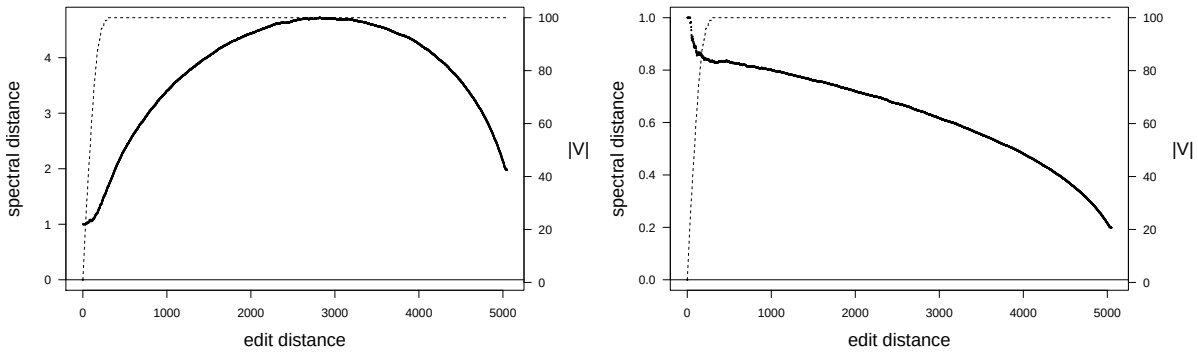
Experiment 1: A Growing $\mathbf{G}(\mathbf{n},\mathbf{p})$

The first test setting is a set of graphs constructed by randomly adding edges to initially isolated nodes in the fashion of the $G(n,p)$ random graph model. Beginning with an empty graph on 100 nodes, samples are constructed by consecutively adding new edges. In each step, a pair of unconnected nodes is chosen with uniform probability and connected by an edge. The sample is formed from the resulting graph by removing all isolated nodes. To ensure comparability between the variants of spectrum transformation cost, all measurements used the same sequences of edge insertions.

Point of reference for this experiment is the graph on a single node, i.e. the spectrum consisting only of a single zero eigenvalue. Note that this is similar to but not exactly a zero element in the algebraic sense. Considering the measurements, all mass distributions have to be transformed such that they are concentrated at zero, which is also the mean of

positions. Therefore, the distances considered in the following are in some way comparable to the (undefined) norm that underlies the employed distances.

Results Figure 8.1 compares edit distance to $\mathcal{STC}^1_A$ (8.1(a)) and its normalized version $\overline{\mathcal{STC}}^1_A$ (8.1(b)). Additionally, in each diagram the number of nodes in the individual sample graphs is shown as a dotted line corresponding to the axis labeling on the right to emphasize the effect of node additions in the beginning of the sampling process. Due to the construction process, the density of the samples is almost a linear function of the edit distance, with deviations only due to node additions.



(a) $\mathcal{STC}^1_A$ (b) $\overline{\mathcal{STC}}^1_A$

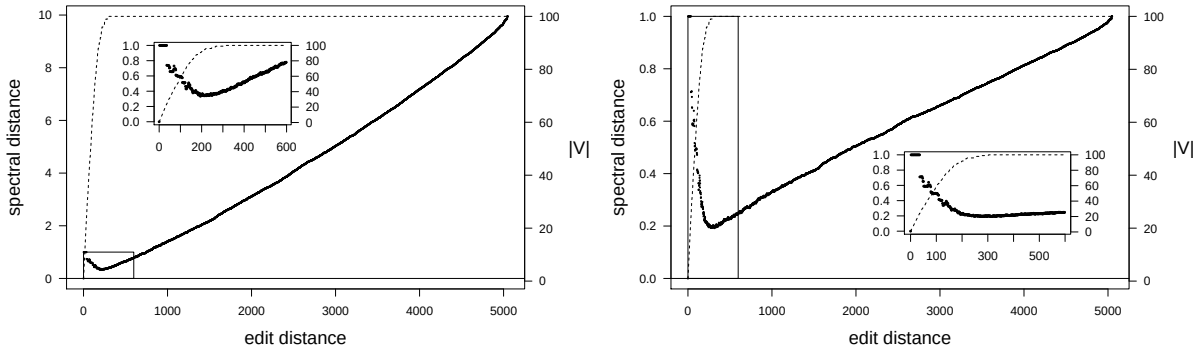
Figure 8.1: Relation between $\mathcal{STC}^1_A$ and edit distance. The dotted line shows the number of nodes in the sample on the scale indicated on the right axis.

Figure 8.1(a) shows that $\mathcal{STC}^1_A$ is far from being linearly related to edit distance, while $\overline{\mathcal{STC}}^1_A$ is negatively correlated in this particular setting. Following a monotonically growing distance until roughly half of the edges are inserted, distances decline while samples approach the full graph. Structurally, this could be related to the homogeneity of entries in A which is minimal, when half of the edges are present and maximized by the full and the (nearly) empty graph. The considerations of the last chapter allow an even more direct explanation. A transition between the concentration of eigenvalues near zero for the near empty graph, the semi-circle with varying radii centered at zero for the intermediate $G(n,p)$-like graphs, and the eigenvalue distribution of the full graph (one eigenvalue $n-1$, $(n-1)$ eigenvalues -1) is most likely the explanation. This assumed transition is not completely backed up by theoretical results, but indicated by the known spectra in the beginning and end, the estimation of the spectra of $G(n,p)$-graphs and the theorems of Section 7.1. Assuming this behavior, the spectra of samples are expected to start by spreading eigenvalues away from zero with growing edit distance up to some point of maximal variance, followed by their contraction to a concentration at -1 except for the single $n-1$ eigenvalue of the full graph. Consequently, the assumption is coherent with the development of distances regarding $\mathcal{STC}^1_A$.

In the relation of edit distance to $\overline{\mathcal{STC}}^1_A$ (Figure 8.1(b)), the behavior observed with $\mathcal{STC}^1_A$ is changed by the normalization procedure. A benevolent interpretation could

suggest that many small components and a larger, sparse structure contains more structural information or structural complexity than a dense structure with uniformly distributed edges. As noted before, the aim of these experiments is not an approximation of edit distance, but rather the exploration of the behavior of transformation cost using edit distance as a basis for comparison. In that respect, the normalized version of $\mathcal{STC}$ could yield an interesting measure for certain aspects of graph structure.

Both variants are not directly related to edit distance but measure some different aspect of graph structure, though in these experiments, the basis of measurement is the empty graph on one node which could be interpreted as a general assessment of structural complexity.



(a) $\mathcal{STC}^2_A$ (b) $\overline{\mathcal{STC}}^2_A$

Figure 8.2: Relation between $\mathcal{STC}^2_A$ and edit distance. The dotted line shows the number of nodes in the sample on the scale indicated on the right axis and the insets show a detailed view of the framed area in the main plot.

In the limited and very special setting considered here, both $\mathcal{STC}^2_A$ and $\overline{\mathcal{STC}}^2_A$, are in a large region almost linearly related to edit distance as Figure 8.2 illustrates. The insets in both diagrams show the strongest deviation from this relation with samples that incorporate less than the full number of possible nodes. Due to the construction process, such samples appear after the first edge insertions and consist of a number of small components. As Figure 8.2(b) shows, this difference is emphasized by the normalization procedure leading to distances of the first samples equal to those of the full graph. These first samples consist of isolated edges, i.e. graphs with maximum density, while the last sample is also a full graph differing only in size, not density. Thus here, the normalization seems to compensate for size differences and shift the focus to density. This interpretation is consistent with the samples of minimum distance to the empty graph which appear in both diagrams near the point where samples start to include all nodes. While the very small components like P_2 and P_3 emerging in the beginning have a very high density, the addition of further edges leads to more nodes included in the graph with density necessarily growing slower than the number of edges.

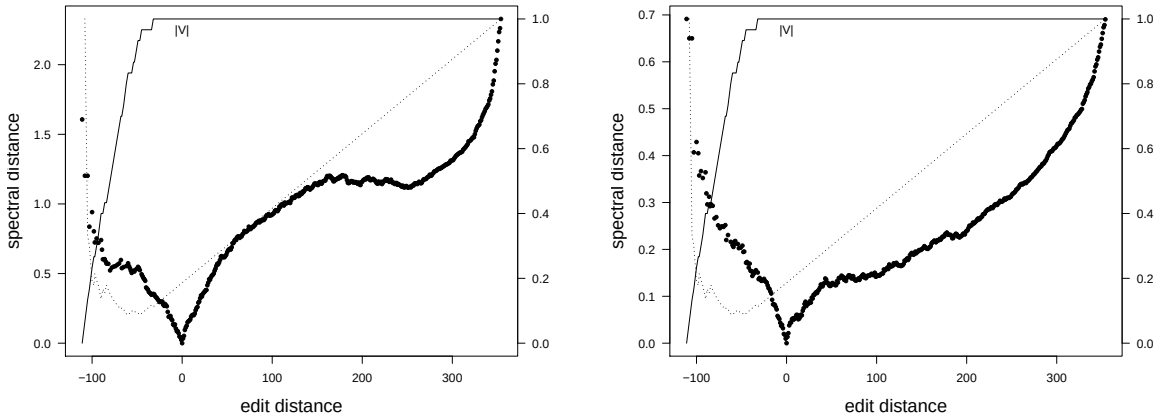
Though a complete examination of the structural properties related to the individual distances is - if possible at all - not undertaken in the following, the hitherto gained

insights are consolidated by a second experiment in the following.

Experiment 2: Distance to a pa$(\mathbf{n}, \mathbf{m}, \alpha)$

The last experiment involved a very special point of reference - the empty graph. In the following, a more realistic setting is created by employing a graph with a more complicated structure as point of reference. The general nature of sample construction, however, results in the full and the empty graph as extremes in terms of edit distance. In addition, the edges are removed and added by uniform, random choice which suggests a tendency to graphs resembling those drawn from a $G(n, p)$-model. The following setting varies from this path of development from empty via $G(n, p)$-like samples to full graph by using a random graph drawn from a preferential attachment model pa$(30, 3, 1)$ as point of reference. This creates a contrast to the last experiment in that such a graph has a very low probability under the conditions of uniform edge distributions. Samples are created as before by removal and addition of edges and nodes and as before isolated nodes are removed. A combined diagram showing distances in relation is created by depicting edit distance resulting from deleted nodes and edges as negative values.

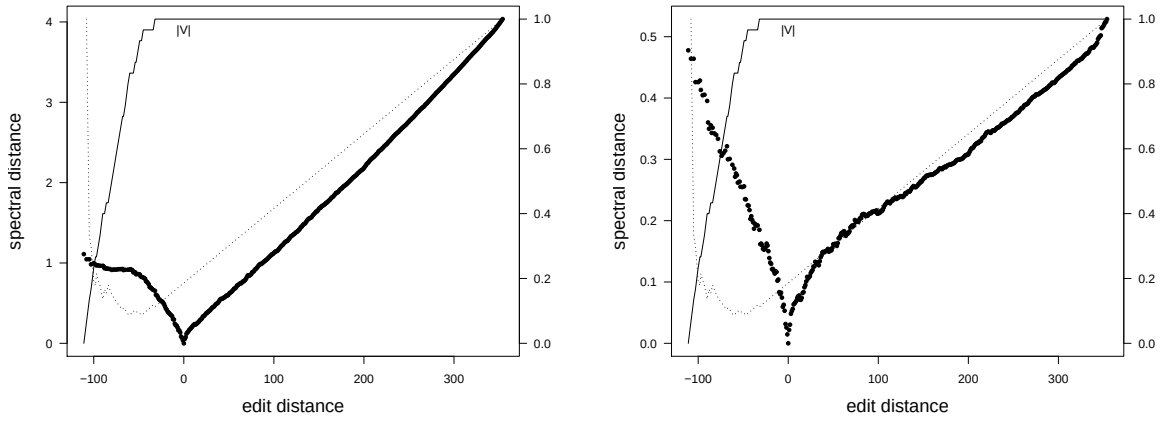
Results Analogous to the previous experiment, all distance variants are compared in the same setting, showing results for $\mathcal{STC}^1_A$ in Figure 8.3 and $\mathcal{STC}^2_A$ in Figure 8.4 for both, the default and the normalized version.



(a) $\mathcal{STC}^1_A$ (b) $\overline{\mathcal{STC}}^1_A$

Figure 8.3: Relation of edit distance and $\mathcal{STC}^A\|\cdot\|_1$ with a preferential attachment graph as point of reference. The solid line shows the number of nodes in the sample with values in $[0, 30]$ (not indicated) and the dotted line indicates the densities of the samples corresponding to the axis labeling on the right.

To some extend, the observations of the previous experiment are confirmed: $\mathcal{STC}^2_A$ and its normalized version show a behavior very close to a linear function of edit distance, except for the samples with less nodes. As before, $\mathcal{STC}^2_A$ seems to be the



(a) $\mathcal{STC}^2_A$ (b) $\overline{\mathcal{STC}}^2_A$

Figure 8.4: Relation between edit distance and $\mathcal{STC}^A\|\cdot\|_2$ with a preferential attachment graph as point of reference. The solid line shows the number of nodes in the sample with values in $[0, 30]$ (not indicated) and the dotted line indicates the densities of the samples corresponding to the axis labeling on the right.

best approximation of edit distance, while $\overline{\mathcal{STC}}^2_A$ again compensates size differences in exchange for similarity in density. The reference graph contains already 83 of the possible 435 edges, yielding an edit distance of 352 to the full graph and 113 to the empty graph, counting in addition the 30 nodes to be removed. The distances of ≈ 1 to the empty and ≈ 4 to the full graph obtained by $\mathcal{STC}^2_A$ show a very similar ratio. Note that this seemingly consequent behavior may well be an artifact of the experimental design and rather indicate that the quadratic versions of transformation cost are strongly correlated with density.

In contrast, $\mathcal{STC}^1_A$ yields similar distances for a number of samples, those with roughly edit distance 200 and on the other side those with about 60 removed edges. This indicates the dependency of $\mathcal{STC}^1_A$ to density is not as strict as that of $\mathcal{STC}^2_A$. In both cases normalization leads to maximal dissimilarity of samples with highest density, independent of their size.

Summarizing, the results of the last experiment are partially confirmed in that edit distance is best approximated by $\mathcal{STC}^2_A$, whereas $\mathcal{STC}^1_A$ is more dependent on other structural aspects of the graphs under consideration. Which exact aspects are involved here, is however not clear from these small experiments due to limits of the experimental design and lack of a wider distribution of samples.

Conclusions

The experiments indicate that for an approximation of edit distance, $\mathcal{STC}^2_A$ is a variant worth of closer examination. $\mathcal{STC}^1_A$ seems to assess additional, structural properties, which are not covered by edit distance. In addition, the normalized variants show

promising results with respect to the compensation of size differences, which will be examined more closely in the experiments of the next section.

8.2 Model Similarity

Analogous to edit distance in the last section, the aspect of *model similarity* and its relation to spectrum transformation cost will now be examined empirically. As elaborated in Section 1.4, model similarity denotes the similarity of graph creation processes reflected in the resulting graphs. Such creation processes can be measurements or random graph distributions. The latter will be employed in the following to produce test sets.

As indicated by the discussion of the previous chapter, samples from different random graph models differ systematically in the distribution of their eigenvalues. In addition, their eigenvalue distributions vary depending on their size. While the problem of distinguishing graphs from different models is simple in the case of equal sizes, the differentiation of samples by the underlying model becomes more complicated when the samples are of considerably different sizes. The objective of the following experiments is to determine which of the variants of spectrum transformation is apt for this situation, i.e. which variant is best capable to distinguish different underlying random graph distributions by distances produced on samples of varying size.

In that, different random graph distributions will be distinguished on two different levels: (i) different construction schemes, such as $\mathrm{pa}(n, m, \alpha)$ and $G(n, p)$ and (ii) different parameters in the same construction scheme, e.g. $G(n, p)$-models with different edge probabilities.

As before, the considerations are constrained to the adjacency matrix as graph representation to limit the number of experiments. Alternatives are, however, considered in the experiments of Section 8.5, where the different variants including different matrix representations are compared to other spectral distances on various settings including the one developed here.

To begin with, the actual problem is highlighted by a discussion of different classes of change in the eigenvalue distribution related to sample size. Following this, experiments are carried out to investigate whether one of the variants of spectrum transformation cost is capable to handle this problem.

8.2.1 Limits of Direct Comparison

Consider as a first example $n = k^2$ nodes arranged on a $k \times k$ grid in the plane with each node connected to his four neighbors as shown in Figure 8.5. This simple construction scheme can be used to produce graphs of different size with considerably structural similarity. Since graph sizes differ, there is a necessary change in the number and thus distribution of eigenvalues for different k. Examples of the spectra for $k \in \{10, 20, 30, 40\}$ are given in Figure 8.6. A comparison of the cumulative density functions for different k suggests, that a certain function is approached in the limit of large k and deviation from this function vanishes with increasing k. The concrete behavior of

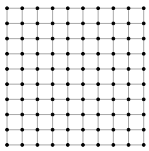


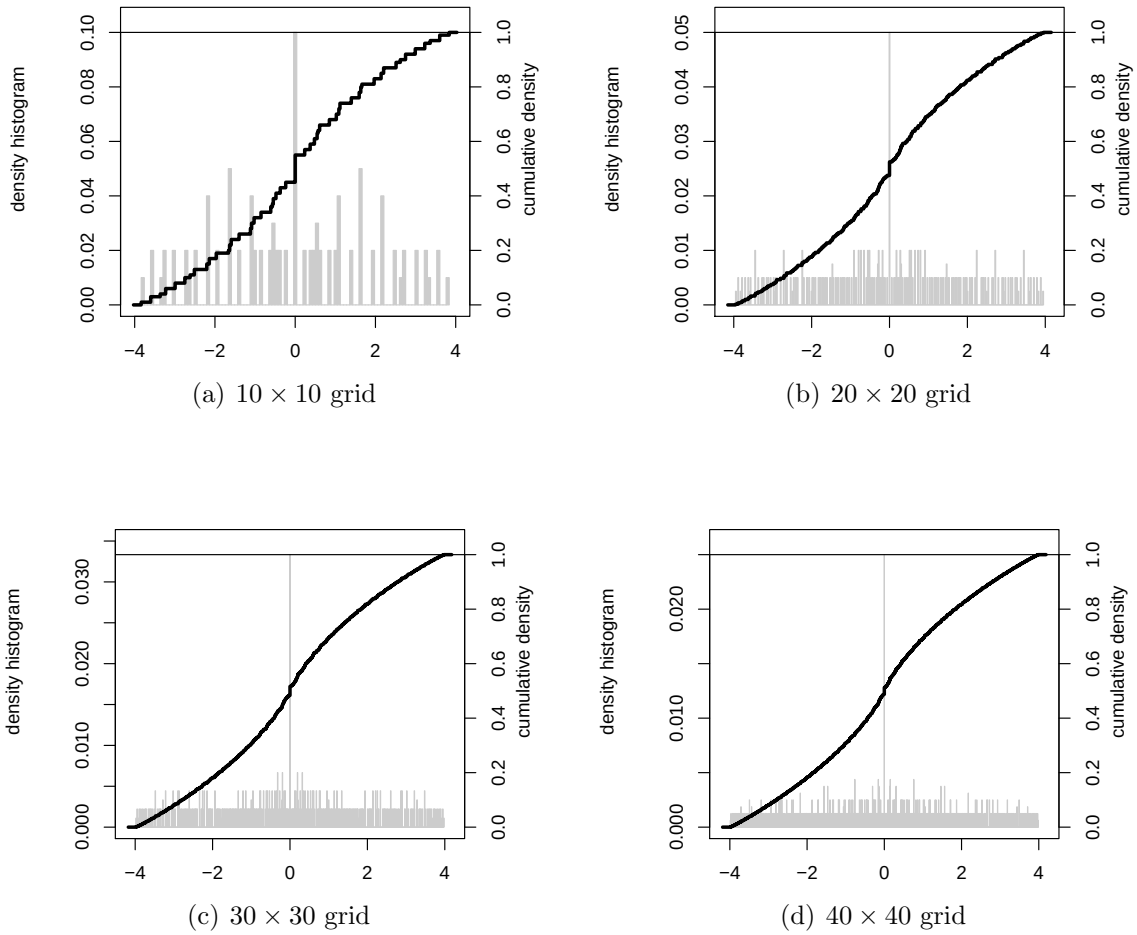
Figure 8.5: A quadratic grid on 10×10 nodes.

these spectra in dependence of k is not of concern here, but it serves as an example for a construction scheme with an - at least seemingly - converging spectrum. With respect to the measurement of model similarity, this is the ideal case, since spectrum transformation cost depends directly on the cumulative density function.

Unfortunately, this behavior does not apply in general. The discussion in Section 7.4 shows, that for some random graph models eigenvalue distributions change systematically with growing graph size while not converging to a common cummulative density function. Following the discussion in the last chapter, the expected spectra of graphs drawn from a $G(n,p)$-model with growing n and fixed p can be described using functions of n and $\sqrt{n}$, i.e. different parts of the spectrum develope with asymptotically different rates depending on graph size. Consequently, the corresponding cumulative density functions and thus distances measured by spectrum transformation cost will undergo such systematic changes. Due to the different rates of change, this cannot be compensated by linear normalization as in the normalized variants of $\mathcal{STC}$ proposed in Section 6.4. Simplifying, this could be described as spectra of the grid exhibiting *static* behavior while spectra of samples from $G(n,p)$-models following a *dynamic* behavior.

A mixed behavior can be observed with samples from preferential attachment models. As Figure 8.7 illustrates, graphs produced by this model (or at least the concrete $\mathrm{pa}(n,2,1)$ used for the illustration) yield a certain shape of distribution with at least three points around which eigenvalues concentrate. Here, three graphs of considerably differing size where drawn from a $\mathrm{pa}(n,2,1)$ model and the figure depicts the spectra of their adjacency matrices aligned for comparison. Note, that due to the extreme high number of zero eigenvalues in these samples the histograms (corresponding to the left vertical axis) are cut of at 25 values, since otherwise the remaining distribution would have been compressed beyond visibility. This example illustrates a development of spectra that will be considered as a third category of interaction between sample size and eigenvalue distribution.

Areas of eigenvalue concentration are located at zero and approximately -1 and 1. Since these positions are constant among the three examples, they appear to be independent of the sample size. Apart from those points of concentration, eigenvalues of larger magnitude appear with growing sample size. In particular, Goh, Kahng, and Kim (2001) relate the largest eigenvalues in a $\mathrm{pa}(n,m,\alpha)$-model directly to the number of nodes. In particular, they show that $\lambda_1 \sim n^{1/4}$ for samples with n nodes and large n in certain models. Due to these partially constant and partially dynamic relations, this



(a) 10×10 grid

(b) 20×20 grid

(c) 30×30 grid

(d) 40×40 grid

Figure 8.6: Spectra of the adjacency matrices of quadratic grids of different size.

kind of interaction between eigenvalue distribution and sample size will can be described as *semi-dynamic*.

The discussion above illustrates, that there are random graph distributions resulting in static eigenvalue distributions, while other yield dynamic distributions with different parts changing with different rates in relation to the sample size and, finally, models producing semi-dynamic spectra combining both properties. As a result, the effect of sample size on eigenvalue distribution is not possibly compensated by a linear normalization of spectra. Even though an apt normalization procedure is not completely ruled out by the arguments above, this approach will not be followed here. Instead, the following experiments will be restricted to the exploration of the limits in which the already proposed distances and their normalized versions are applicable.

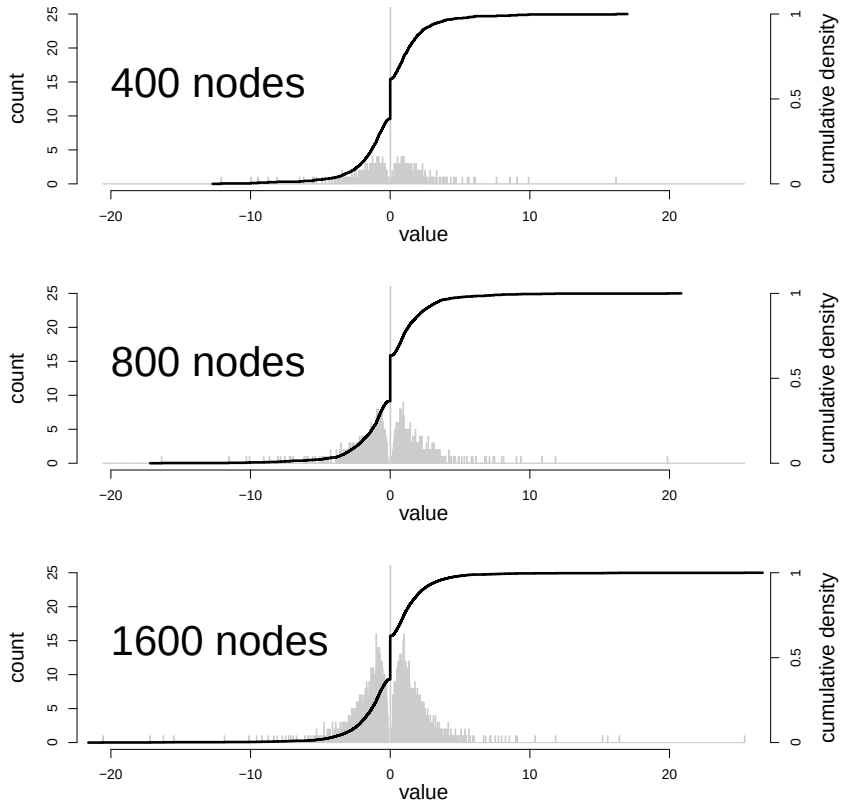


Figure 8.7: Spectra of samples from $\mathrm{pa}(n, 2, 1)$ for $n = 400, 800, 1600$. The diagrams show only a detail on counts up to 25 but the complete value range.

8.2.2 Experiments

As mentioned above, the following experiments are limited to spectra derived from the adjacency matrix. In addition, experiments are limited to samples from distributions created by preferential attachment models and $G(n, p)$-models.

Within these limitations, all of the experiments follow a common procedure: using a single graph of reference drawn from a certain distribution, distances to samples of the same and other distributions will be examined with respect to their development in size. For equally sized samples it is expected that smallest distances are produced by the samples from the same model, while samples of different models are found at larger distances. The limits of application are approached when the size differences between samples and reference graph grows. As a consequence of the change of eigenvalue distributions with size, the samples from the same model are expected to produce distances to the reference graph which increase with difference in size. Eventually, this development could render a classification of samples by originating graph model impossible. The assumption is, that this case arises when distances of samples being drawn from the same distribution as the reference graph yield distances that are comparable to or even larger than those produced by samples from different distributions. The arguments

of the last section indicate, that such a behavior can be expected when unmodified spectra are compared using spectrum transformation costs. It is, however, still open *when* size differences yield the assumed effect and further, how the proposed normalization procedure ($\overline{\mathcal{STC}}$) influences the situation.

Recall, that two levels of model similarity have been distinguished: equality of generation schemes and additional equality of the parameters involved in the generation process. The following experiments will incorporate this by comparing distance developments of samples from the same distribution with (i) identical parameters, (ii) different parameters, and to samples from a distribution defined by the other generation scheme.

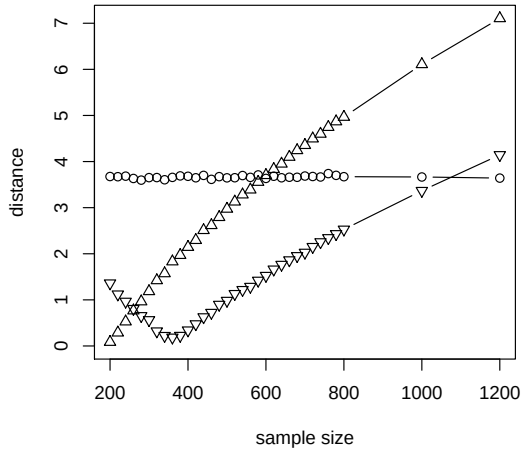
In particular, the first experiment compares a $G(200, 0.2)$ to samples from $G(n, 0.2)$, $G(n, 0.1)$ and $\mathrm{pa}(n, 2, 1)$ for values of n ranging from 200 up to 1200. The second experiment then compares the same set of samples to a reference graph drawn from a $\mathrm{pa}(200, 2, 1)$. For both settings all variants of $\mathcal{STC}_A$ are compared.

Results Distances produced in the two experiments are shown in Figure 8.8 for the reference graph drawn from $G(200, 0.2)$ and in Figure 8.9 for the reference graph drawn from a $\mathrm{pa}(200, 2, 1)$-model. Note that the two experiments differ only in the reference graph while the samples they are compared to are the same in both experiments.

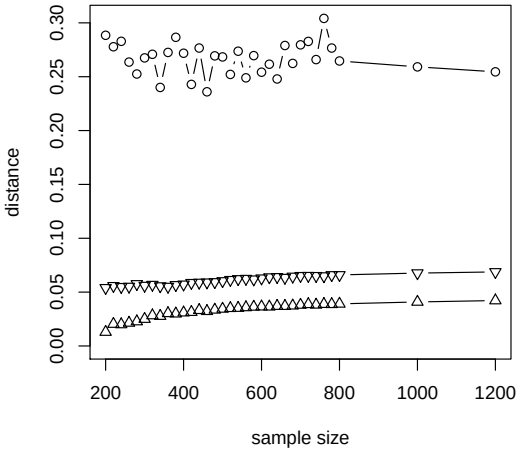
As expected, using any variant of $\mathcal{STC}$, the originating random graph models can be clearly distinguished from each other and the reference graph, when only samples are considered that are of the same size as the reference graph. That is, in the first experiment the sample drawn from the $G(200, 0.2)$ yields the smallest distance to the reference graph for every distance and this holds analogously for the sample drawn from the $\mathrm{pa}(200, 2, 1)$ in the second experiment.

The aim of the experiment is, however, the assessment of the development of distances with growing size difference, i.e. samples produced with larger n. As suggested by the discussion of the last section, both experiments and all distance variants reveal a relation between distance to the reference graph and the sample size which is mostly a distance increasing with sample size. Depending on the employed distance, this relation takes different forms: for $\mathcal{STC}^1_A$ it is close to a linear function of the size, while $\mathcal{STC}^2_A$ and its normalized form show a more complicated behavior. In contrast, the size-distance dependency seems to be nearly constant for $\overline{\mathcal{STC}}^1_A$, though a closer inspection shows even in this case a small increase in distance. For the classification of samples by the random graph distribution producing them, a minimal condition is that distances of samples from the same model (but different size) yield constantly smaller distances than samples produced by other random graph models. Only for sizes where samples from the same model as the reference graph result in equal or higher distances than samples from other models, the classification by random model is becoming more complicated, though not necessarily impossible. The former is the case in the second experiment for all distance variants, i.e. all of them yield constantly larger distances for the $G(n, p)$-samples than for the samples from the $\mathrm{pa}(n, m, \alpha)$-model. For $\mathcal{STC}^1_A$, $\mathcal{STC}^2_A$ and $\overline{\mathcal{STC}}^2_A$, this relation, however, changes in the reversed situation of the first experiment.

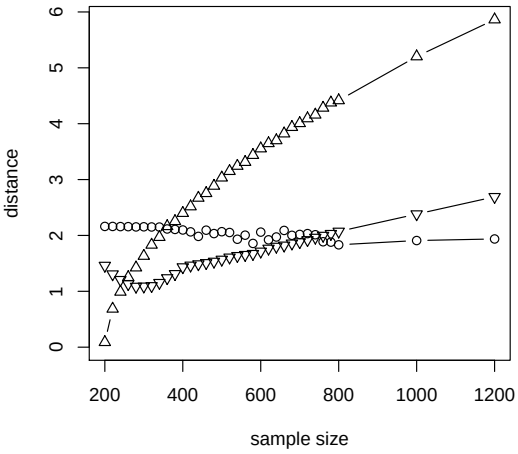
The size range considered in the experiments ($n \in \{200, \ldots, 1200\}$ is sufficient to



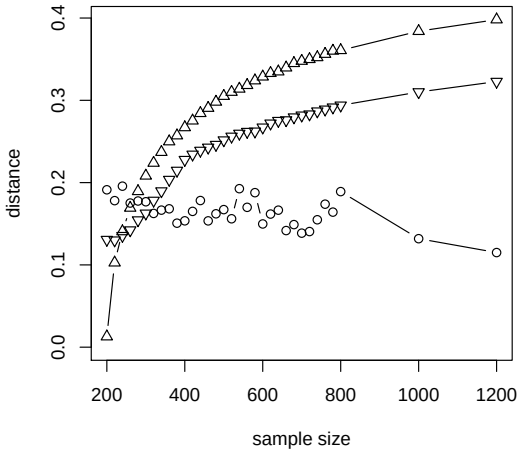
(a) $\mathcal{STC}^1_A$



(b) $\overline{\mathcal{STC}}^1_A$

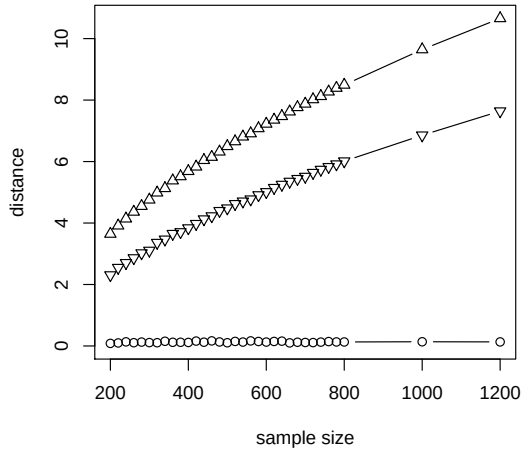


(c) $\mathcal{STC}^2_A$

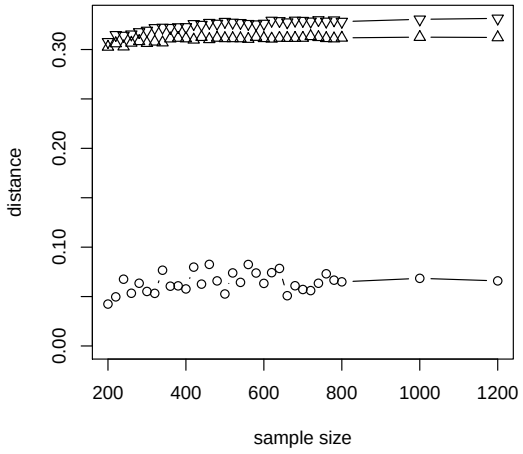


(d) $\overline{\mathcal{STC}}^2_A$

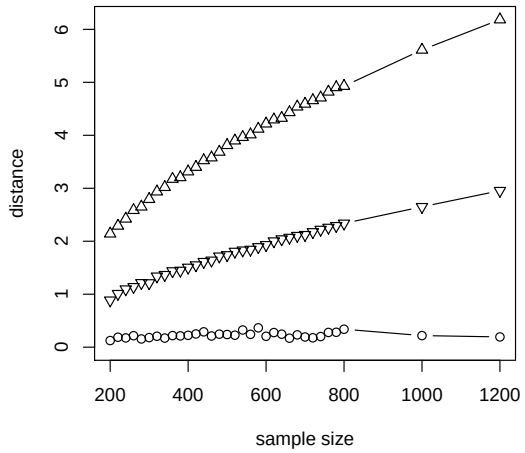
Figure 8.8: Distances of samples from $\mathrm{pa}(n, 2, 1)$ ($\circ$), $G(n, 0.2)$ ($\triangle$) and $G(n, 0.1)$ ($\triangledown$) to a reference graph drawn from a $G(200, 0.2)$. In all diagrams, distances to the reference graph are shown on the vertical axis and compared to the size of the sample under comparison on the horizontal axis.



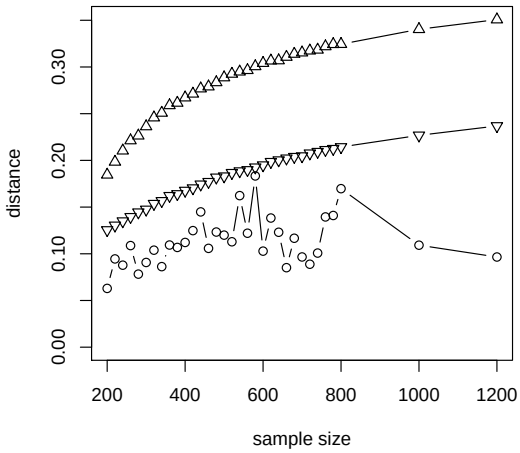
(a) $\mathcal{STC}_A^1$



(b) $\overline{\mathcal{STC}}_A^1$



(c) $\mathcal{STC}_A^2$



(d) $\overline{\mathcal{STC}}_A^2$

Figure 8.9: Distances of samples from $\mathrm{pa}(n, 2, 1)$ ($\circ$), $G(n, 0.2)$ ($\triangle$) and $G(n, 0.1)$ ($\triangledown$) to a reference graph drawn from a $\mathrm{pa}(200, 2, 1)$. In all diagrams, distances to the reference graph are shown on the vertical axis and compared to the size of the sample under comparison on the horizontal axis.

observe the limit in detection of model similarity for all the distance variants in both experiments. Only $\overline{\mathcal{STC}}^1_A$ does not reach this point within the observed range and setting. Keeping in mind the very limited setting of this experiment, this is an indicator, that despite the discouraging arguments of the last section, normalization may be an approach that is helpful with respect to model similarity. Consequently, these results render $\overline{\mathcal{STC}}^1_A$ a promising candidate for further research in this direction.

Additionally, it can be observed that distances of samples from the $\mathrm{pa}(n, m, \alpha)$-model to the reference graph from the $\mathrm{pa}(n, m, \alpha)$-model are nearly constant for $\mathcal{STC}^1_A$ and $\mathcal{STC}^2_A$ in the second experiment. It was shown before, that despite the regions of eigenvalue concentrations at certain, size independent points the model produces eigenvalues of magnitude growing with graph size. This contradiction could be explained by (i) the limited resolution of the experimental observation, i.e. distances will grow in the limit and (ii) by the signature normalization applied, which assigns weight decreasing with graph size to the eigenvalues of large magnitude, i.e. their weight decreases quickly then their distance from other eigenvalues grows. On the other hand, the weight positioned at the fixed points is stable or even growing, due to the increasing number of eigenvalues at these positions.

Critique and Conclusion The experiments presented in this section represent only initial steps into a comprehensive examination of the possibilities and limits in the assessment of model similarity among graphs of differing size. It has been shown, that all variants of spectrum transformation cost except for $\overline{\mathcal{STC}}^1_A$, can be ruled out immediately even for small size differences. The conducted experiments cover only very simple settings, with graphs from different models expected to be strongly differing in structure, while the models introduce structural commonalities (e.g. degree distributions) being similar for samples of differing sizes. Consequently, a distance failing in this setting can not be considered promising for further tests, while a good performance is by no means a guarantee for applicability in similar settings.

However, promising results are delivered by $\overline{\mathcal{STC}}^1_A$ illustrating that the involved normalization process might help to approach the underlying problem.

Further experiments should also consider different matrix representations, though in each of those probably some asymptotic change with respect to graph size is involved. Though it may be possible that models having a dynamic eigenvalue distribution in their adjacency matrix eigenvalues are converted to a static behavior in the eigenvalue distribution of another matrix representation. Finally, a refined definition of the notion of "model similarity" would be an important step of further research, since the notion has been introduced here only informally, solely substantiated by its interpretation as "being produced by an algorithmic random graph generation scheme".

8.3 Isospectral Graphs

Graph distances based on eigenvalues generally suffer from the fact that there exist *pairs of isospectral non-isomorphic graphs* often referred to as *PING*s. That is, graphs

distinguished by e.g. edit distance result in identical spectra and can therefore not be distinguished by these approaches. Though this section will again be restricted to the eigenvalues of the adjacency matrix, isospectral graph exist for a number of matrix reprentation as shown in Haemers and Spence (2004).

In proposals of spectral graph distances, the existence of PINGs is often ignored, justified by the argument that they appear rarely and are of no concern in real-world applications. Schwenk (1973), however shows, that the number of trees uniquely identified by their spectrum vanishes with growing size. Godsil and McKay (1982) describe a construction mechanism for PINGs of arbitrary size. On the other hand, the empirical study in Haemers and Spence (2004) indicates that the number of PINGs of size n vanishes for $n \to \infty$. Note, that identical spectra for one matrix representation do in general not coincide with identical spectra in other representations. In addition, for the context of structural comparison, the previous chapter summarizes a large number of structural properties to the eigenvalue distribution of graphs, which indicates that from a structural point of view graphs with identical spectra have to be structurally similar to some extend.

The main objective of spectrum transformation cost is the assessment of structural similarity. One of the considered aspects is edit distance and consequently the indistinguishability of graphs with non-zero edit distance is indeed a disadvantage. For other aspects it may be of no concern and some of the arguments above justify to neglect this problem, but for edit distance in these special cases it seems interesting.

Therefore, in the following the relation of spectrum transformation cost and edit distance in the vicinity of PINGs will be explored. Edit distance and spectrum transformation cost imply some space within which graphs are distributed[1]. As the existance of PINGs shows for spectrum transformation cost, this space contains points occupied by more than one graph. In the space implied by edit distance however, these two graphs have a non-zero distance, i.e. occupy different points. The objective of the following experiments is to examine the edit distance vicinity of the two graphs in terms of spectrum transformation costs. One possibility would be that not only the two discrete points in edit distance space are placed together by spectrum transformation cost but that the same applies to their neighborhood. Another possibility would be that graphs in the direct (edit distance) vicinity of the PING quickly diverge in the distance measured by transformation costs.

The second case would the preferred one in the application of spectrum transformation cost and whether it applies or not is the central question of this section. Therefore it is formulated as the hypotheses to be confirmed or confuted:

Hypothesis 8.1. *Given a pair of non-isomorphic, isospectral graphs G, H and graphs $\tilde{G}_i$,$\tilde{G}_j$, $\tilde{H}_i$ in small edit distance from G respective H, with high probability, spectral distances between graphs derived from the same source will be smaller than between graphs from different sources. That is, by tendency $\mathcal{STC}^1_A(\tilde{G}_i, \tilde{G}_j) < \mathcal{STC}^1_A(\tilde{G}_i, \tilde{H}_j)$.*

In the remainder, G and H will denote the graphs of the PING, while $\tilde{G}$ and $\tilde{H}$, with index if needed, will refer to samples derived from G or H respectively.

[1]For a finite set of examples such a space could be created by a projection of the distance using MDS.

8.3.1 Experimental Method

As before, experiments will be limited to the adjacency matrix and in addition only a single variant of spectrum transformation cost, $\mathcal{STC}^1_A$, will be considered. The normalized versions are not considered, since no differences in size are to be accounted for and in addition $\mathcal{STC}^1_A$ seems to be better suited for the task in light of the previous experiments.

PING Construction The first task to be solved in preparation of the following experiments is the gathering of source data, i.e. obtain pairs of non-isomorphic isospectral graphs. Concrete PINGs and families of PINGs have been presented in the literature, for example in Cvetković et al. (1995). Most of these, however, describe pairs of graphs that are constructed to be isospectral and therefore of very special structure. A larger variety can be achieved with the construction method shown in the following.

As mentioned above, Godsil and McKay (1982) present a method that is able to construct PINGs of flexible size and can in general produce a number of such pairs for a given graph size. In the following, their theorem will be used to derive an algorithm for the construction of PINGs involving random elements, i.e. a random distribution of isospectral pairs of graphs. Basis for the construction is the following theorem:

Theorem 8.1 (Godsil and McKay (1982))**.** *Let G be a graph and let $\pi = (C_1, \ldots, C_k, D)$ be a node partition of G such that whenever $1 \leq i, j \leq k$ and $v \in D$*

(a) any two vertices in C_i have the same number of neighbors in C_j

(b) v has either 0, $n_i/2$ or n_i neighbors in C_i, where $n_i = |C_i|$.

Let further H be the graph formed from G by switching: for each $v \in D$ and each $1 \leq i \leq k$ such that v has $n_i/2$ neighbors in C_i, delete the corresponding edges and join v instead to the other $n_i/2$ vertices in C_i. Then $A(G)$ and $A(H)$ have identical eigenvalues.

Note that Theorem 8.1 guarantees G and H to have identical eigenvalues, i.e. the same spectrum, while not guaranteeing that they are non-isomorphic. Consequently, a separate test for isomorphism is needed to ensure that G and H are a PING and not just the same graph. Since efficient isomorphism tests are not available (c.f. Section 1.4), the size of graphs that can be produced on the basis of this theorem is limited by the necessary tests.

On the basis of Theorem 8.1, Algorithm 3 was used to produce candidates from which the actual PINGs were then filtered by an isomorphism test. The concrete instances G and H produced by the algorithm are more constrained than Theorem 8.1 would allow. The method $\mathrm{pa}(n, m, \alpha)$ draws a sample D from a preferential attachment graph using Algorithm 1 described in Section 2.4.2. At this point, an arbitrary graph could be chosen. Using a preferential attachment model, however, yields with high probability a degree distribution of high variance. This hopefully leads to strong changes introduced by the switching of edges changing G into H.

The edges E of G and E' of H are maintained in parallel, differing only in the connections between D and the nodes in the C_i. In the next step, the classes C_i are created.

Algorithm 3: Producing isospectral graphs.

Input: $n, m, \alpha, k, n_1, \ldots, n_k$
Result: isospectral graphs G, H
$D \leftarrow \text{pa}(n, m, \alpha)$;
$E \leftarrow E(D)$;
$E' \leftarrow E(D)$;
for $i \in \{1, \ldots, k\}$ **do**
  $C_i \leftarrow \{v_1, \ldots, v_{n_i}\}$;
  for $j \in \{1, \ldots, i-1\}$ **do**
    $M \leftarrow$`createRegularBipartite`(C_i, C_j);
    $E \leftarrow E \cup M$;
    $E' \leftarrow E' \cup M$;
  if $n_i \ mod \ 2 = 0$ **then**
    for $v \in V(D)$ **do**
      $C \leftarrow$`sample`$(C_i, n_i/2)$;
      $E \leftarrow E \cup \{vc : c \in C\}$;
      $E' \leftarrow E' \cup \{vc : c \in C_i \setminus C\}$;
$G \leftarrow (V(D) \cup C_1 \cup \ldots \cup C_k, E)$;
$H \leftarrow (V(D) \cup C_1 \cup \ldots \cup C_k, E')$;

Their nodes are randomly connected to each other by `createRegularBipartite`(C_i, C_j). In this edge creation it is ensured that (i) edges connect only nodes from C_i to nodes from C_j, (ii) all nodes within the individual class are of the same degree and (iii) neither the empty, nor the complete set of edges is produced. This corresponds to demand (a) in Theorem 8.1.

In addition, each node of D is connected to half of the nodes in C_i within G and to the other half in H, realizing the switching described in Theorem 8.1. Here, the function `sample`(S, n) randomly samples n nodes in S using a uniform distribution. The selection of edges corresponds to the demand (b) of Theorem 8.1, but whenever possible chooses half the nodes in C_i and the empty set only when the size of C_i is odd.

Finally, G and H are created from the nodes in D and all of the C_i with the edges sets E and E'.

Godsil and McKay note that the parameters $k = 1$ and $n_1 = 4$ yield the most non-isomorphic graphs. Experiments show, that the yield is still sufficient for larger parameters. Note that at least some of the n_i have to be chosen as even numbers, since otherwise the switching step will not introduce any change due to the second condition of Theorem 8.1.

Generating Samples in Edit Distance Neighborhood For a concrete PING, the neighborhood in edit distance will be explored by deriving samples from the original graphs with *switching* operations. Switching a certain pair of nodes denotes, as in Theorem 8.1,

the deletion of the edge connecting them if it exists or addition of this edge, if not. Using this operation, the edit distance neighborhood of both graphs can be explored systematically: for both original graphs, node pairs are selected randomly and samples are generated by switching them. Repeating this operation leads to samples at correspondingly larger edit distances, though the number of switching operations, even guaranteeing distinct pairs of nodes does not necessarily lead to graphs in an edit distance identical to the number of switching operations. An appropriate number of such operations, however, surely yields a number of graphs in edit distance vicinity of the original graphs. Even though this correspondence is not necessarily exact, the number of switching operations will be used as an estimate for the actual edit distance in the remainder.

Comparison of Distances Given the PING and samples of approximately known edit distance, the relation between both distances will be illustrated by comparing edit distance and spectrum transformation cost between the samples. Since neighborhoods are considered with respect to the source graph, samples are distinguished by the graph they were derived from. Pairwise distances are compared group wise between samples derived from the same graph and samples derived from different graphs, where the originating graphs together constitute the PING under consideration. In addition, edit distance and spectral distance do not necessarily coincide, i.e. a set of samples in common edit distance from the original graph yields a distribution of distances obtained by spectrum transformation cost. Consequently, this distribution of distance is considered in the analysis and viewed in the evaluation of results by comparing for each distinct edit distance the distribution of spectrum transformation cost between pairs $\tilde{G}_i, \tilde{G}_j$ and $\tilde{H}_i, \tilde{H}_j$ to pairs of $\tilde{G}_i, \tilde{H}_j$.

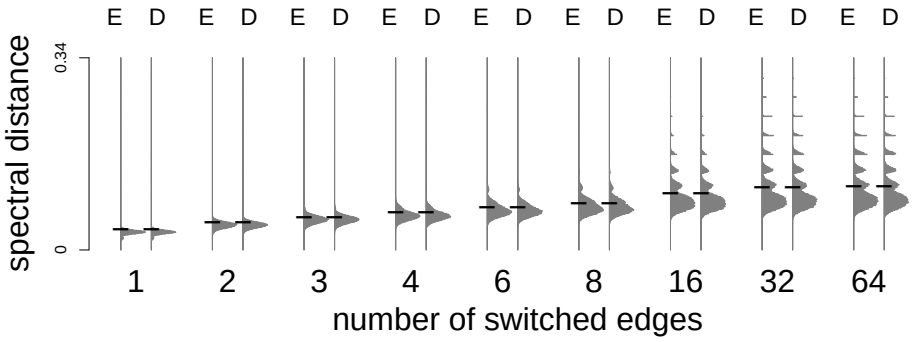
Besides a direct, visual comparison of the resulting distributions, the mean value is provided as a guideline to determine which group tends to larger distances.

Experiments are based on two data sets which are both based on PINGs created with Algorithm 3.

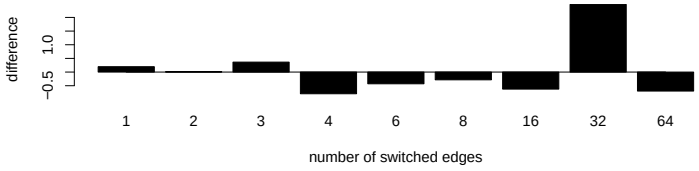
8.3.2 Experiment and Results

The construction parameters common to both examples are: $n = 40$, $m = 3$, $\alpha = 1$, though D was drawn independently in the two cases. For the first example, k was set to 1 and n_1 to 20, i.e. D was connected to a set of 20 nodes with no internal edges. This provides two non-isomorphic graphs with 60 nodes and 988 edges. Results for this first example are shown in Figures 8.10(a) and 8.10(b). To confirm results, a second example with $k = 3$ and all $n_i = 20$ was created, resulting in the measurements illustrated in Figures 8.10(c) and 8.10(d). The two graphs of this PING have 100 nodes and 3594 edges.

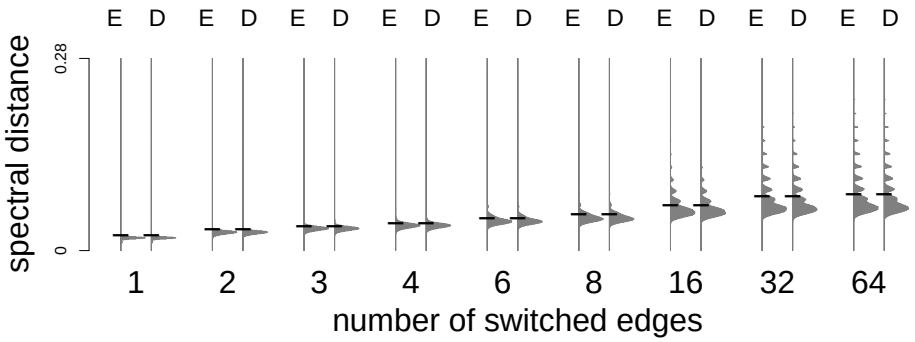
The results show spectral distances between samples after switching 1,2,3,4,6,8,16,32 and 64 edges, i.e. besides showing the direct neighborhood, a number of larger distances are considered. For each source graph and each number of switched edges, 400 samples were generated resulting in 800 samples for each distance step. Of these samples, for each number of switching steps and all pairs of samples, the spectrum transformation cost



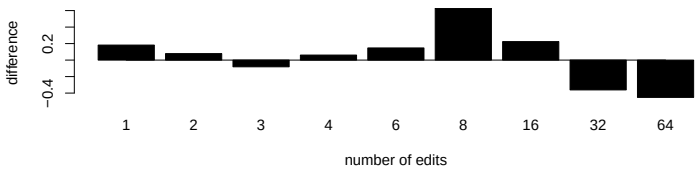
(a) distribution of distances



(b) difference of mean distances·10^4



(c) distribution of distances



(d) difference of mean distances·10^4

Figure 8.10: Distribution of distances in various edit distances (8.10(a),8.10(c)) and the differences of average distance between samples from equal and different sources (8.10(b),8.10(d)). Distance development is shown for samples created from a PING with graphs of 60 nodes and 988 edges in 8.10(a) and 8.10(b) and a second PING with graphs of 100 nodes and 3594 edges in Figure 8.10(c) and 8.10(d).

($\mathcal{STC}^1_A$) was determined. These were grouped by pairs containing two graphs derived from the same source ($\tilde{G}_i, \tilde{G}_j$) or from different sources ($\tilde{G}_i, \tilde{H}_j$).

The distributions of the resulting distances are shown in Figure 8.10(a) for the smaller PING and in Figure 8.10(c) for the larger. It can be observed that spectrum transformation cost increases with additional switching of edges. Visual comparison the two distributions for each step reveals no difference, they seem to be extremely similar. Even comparison of the mean distance for each distribution confirms this observation. Only direct comparison of these mean distances reveals differences on a scale much smaller than the original distances. While distances are for both graphs in $[0, 3.4 \cdot 10^{-1}]$ the magnitude of difference between their mean values is almost always below 10^{-4}. Figures 8.10(b) and 8.10(d) illustrate these differences, again in relation to the number of switching steps. The figures show the difference between mean values for samples from different sources and mean values for samples from equal sources, i.e. $\left\langle \mathcal{STC}^1_A(\tilde{G}_i, \tilde{H}_i) \right\rangle - \left\langle \mathcal{STC}^1_A(\tilde{G}_i, \tilde{G}_i) \right\rangle$, for the individual number of switching steps.

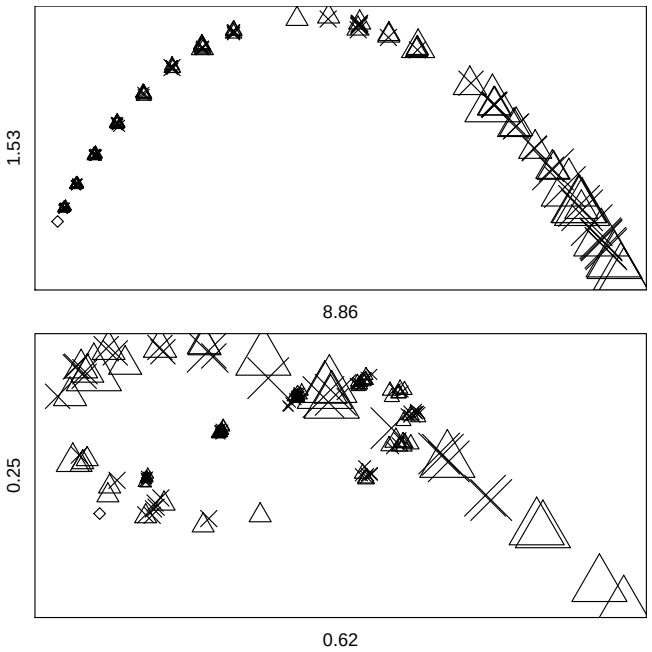


Figure 8.11: Spatial distribution of samples in the edit distance neighborhood of a PING. The graphs forming the PING are depicted as $\diamond$, samples derived from the two graphs as $\triangle$ and $\times$. The number of switching steps corresponds to symbol size, i.e. larger symbols represent more switching steps. Note that aspect ratios are not correct, the amount of distance realized on each axis is shown at the bottom and left of each plot.

With respect to Hypothesis 8.1, this difference should be positive in the beginning and approach zero at some point. The results, however, show no such trend. Differences are in fact so small and unsystematic that they could result from the randomization involved in the experiments. Though these experiments consider only two samples and a limited

range of switching steps, the results reject Hypothesis 8.1, since the suggested effects can not be observed.

This is further supported by the MDS plots of samples resulting from the experiments shown in Figure 8.11. The plots show the first four dimension of the projection for a randomly drawn collection containing one sample for each source graph and switching step. The first axis is strongly correlated with the number of switched node pairs. In addition, it accounts for the major part of distance information available, while the amount of information contained in the following dimensions quickly vanishes. In all four dimensions, no systematic divergence of position between samples derived from the two different source graphs can be observed. Though some difference in their positioning can be observed, it appears random while by Hypothesis 8.1 they should drift systematically from each other. In summary, Hypothesis 8.1 has to be rejected considering the results of these experiments.

8.4 Application to Small Graphs

In the definition of spectrum transformation cost in Chapter 6 it was argued, that this approach should be applicable to graphs of small size. The problem for some of the alternative approaches in the spectral comparison of small graphs is caused by the small number of available eigenvalues. Coarsening these, then may lead to a much larger information loss than for eigenvalue distributions involving many values. The following chapter will examine the applicability of spectrum transformation cost to a set of small graphs observed from different sources. The assumption, that the different sources result in systematic structural differences in the obtained samples connects these experiments to those of Section 8.2 since here also graphs are to be distinguished by their creation process.

8.4.1 Different Domains

The collection to be analyzed in the following consists of 100 randomly chosen graphs from each of the following datasets:

- COIL-DEL (triangulation graphs of images)
- AIDS (chemical compounds, to be tested for activity against HIV)
- mutagenicity (chemical compounds, to be tested for mutagenicity)
- web (graphs created from web documents).
- egoredes (personal networks)

Of these groups, the COIL-DEL, AIDS, mutagenicity and web graphs are taken from the IAM graph database described by Riesen and Bunke (2008) which provides also more

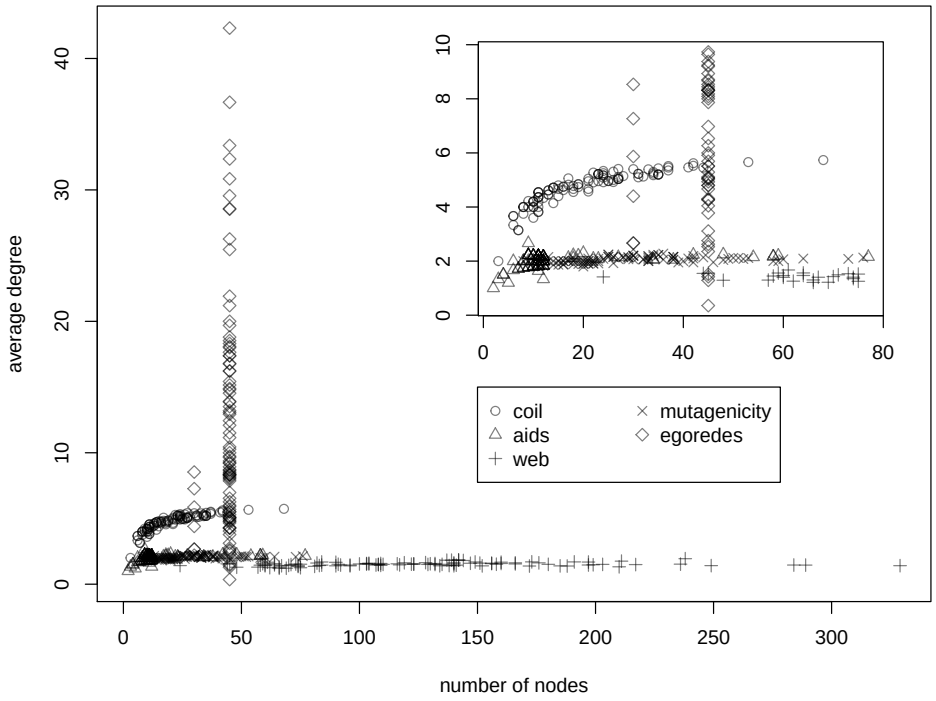
detailed descriptions of these datasets and their method of creation. In addition, the collection is extended by 100 of the personal networks introduced in Chapter 4.

In the remainder, the different classes will be denoted *coil*, *aids*, *mutagenicity*, *web*, and *egoredes*. It is further assumed that graphs of different classes tend to be structurally different, which is reasonable given their different sources and construction processes. An exception are the aids and mutagenicity graphs, since both of them represent chemical compounds and both of them are converted into a graph representation by the same method. The two classes describe, however, different kinds of chemical compounds and therefore a certain tendency to (a somewhat smaller) structural difference is assumed anyway.

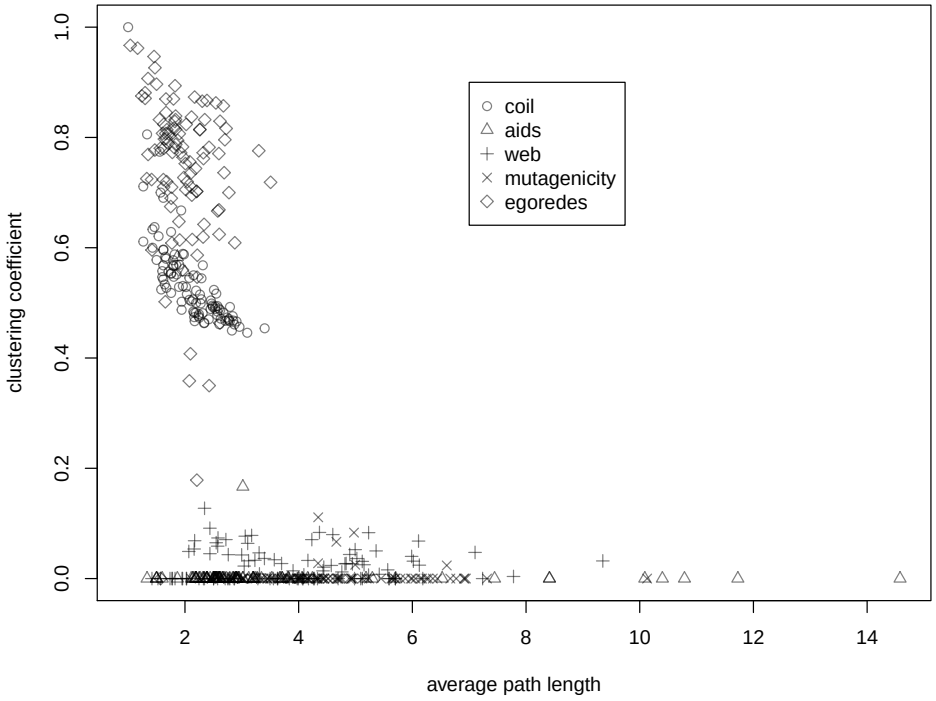
Figure 8.12 shows the distributions of the samples with respect to four graph statistics. It can be observed, that graphs of the individual classes tend to group in certain areas, which confirms the assumption of their structural similarity. Using these and perhaps additional statistical measures, it would probably be possible to classify the samples solely on the basis of the resulting features. The plots, however, illustrate that the structural difference between them is - at least under these statistics - not so extreme that the classification task is rendered trivial. Especially the two classes modeling chemical compounds, aids and mutagenicity, cover with some exceptions almost identical regions.

Due to the acquisition method, all egoredes graphs have either 30 or 45 nodes. The other classes contain graphs with node sets between 3 and 100 nodes with the largest concentration between 3 and 50 nodes. The average degree among all samples is concentrated below 10, with the exception of some of the larger egoredes networks. Except for the egoredes samples, most graphs seem to align to two different relations between average degree and size. While graphs from the class "coil" exhibit average degrees increasing with network size, all others show an approximately constant average degree for the different sizes. No such trend can be observed for the egoredes networks, since they are only in two different sizes available. It can, however, be observed that they occupy the largest range of average degrees. The plot of average path length and clustering coefficient reveals two well separated clusters. These correspond roughly to coil and egoredes having low average path lengths and high clustering coefficients while most of the other graphs have low clustering coefficients and for the most part also small average path lengths. As noted above, a classification of the samples might well be possible using additional statistics. Independent of that, the following experiment examines whether the different - unknown - structural properties are distinguished by spectrum transformation cost.

Experiment and Analysis Instead of formulating an optimization problem, the distances between samples are analyzed visually. Therefore, the MDS method is used to create diagrams approximately realizing the distances inferred from the samples. Recall that for a set of objects and a distance on these objects, an MDS can be used to infer a spatial distribution which approximately realizes object distances in Euclidean space. In the resulting spatial distribution, the first dimensions approximate the original distance due to the fact that these realize the directions of maximal variance. The analysis will



(a) graph size and average degree



(b) average path length and clustering coefficient

Figure 8.12: Distribution of graph size, average degree, average path length and clustering coefficient for the sample graphs of all classes. Figure (a) shows in addition the most densely populated part enlarged in the inset.

mainly exploit the latter property: projection to only the first of these dimensions gives a first insight into the characteristics of the underlying distance and thus allows to judge whether its application is reasonable for the purpose of class separation. For this experiment, again only $\mathcal{STC}^1_A$ will be considered, while an extensive test on all variants is deferred to Section 8.5.

The first four dimensions of the MDS on $\mathcal{STC}^1_A$ of the complete ensemble are shown in Figure 8.13. As can be observed in the first diagram of Figure 8.13, the first dimension

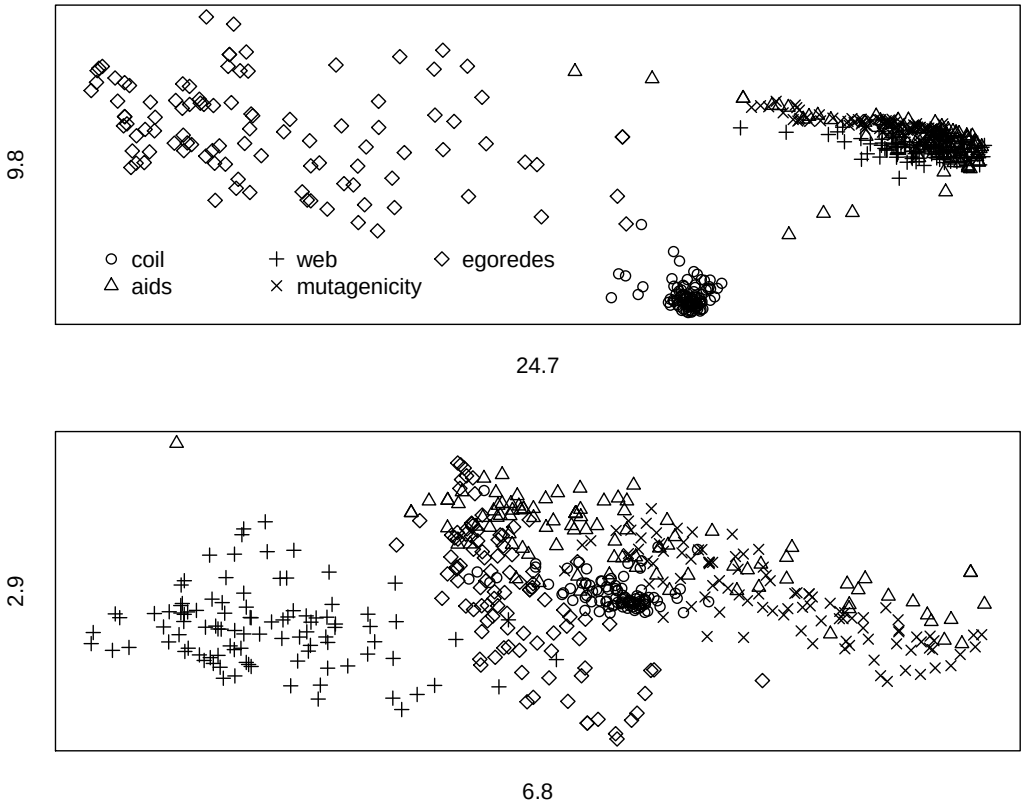


Figure 8.13: MDS projection of spectral distances for all samples. Depicted are the first 4 dimensions, i.e. dimension one and two in the first diagram, dimensions three and four in the second diagram. Axis labels indicate the spectral distance realized on the corresponding axis. Together, the two projections display about 58% of the total distance information.

seperates nearly all of the egoredes graphs from the rest of the ensemble. Their distribution allows the interpretation that these graphs are considerably different from all other classes, though the internal distances of this class seem to be comparable to distances of other classes. The graphs of the coil class are clustered in a very compact region and also almost completely separated from all other graphs on the second dimension. In contrast, the two molecule groups and the graphs produced from web documents appear to be more similar to each other and occupy strongly overlapping regions on the first two

dimensions. The web graphs are, however, separated from the molecule classes and rest of the ensemble on the third dimension.

These results confirm the assumption that the different sources result in systematic dissimilarities which in turn are measurable by $\mathcal{STC}^1_A$. The only classes that are not distinguished in the plots seen so far are the graphs from the mutagenicity and aids classes. Since they represent objects from very similar sources, this result is coherent with the assumptions.

It was, however, assumed that despite their similarity at least some structural difference between the two classes exist. This is explored in more detail in Figure 8.14 in a projection involving only graphs from the mutagenicity and aids classes. The restriction of the MDS

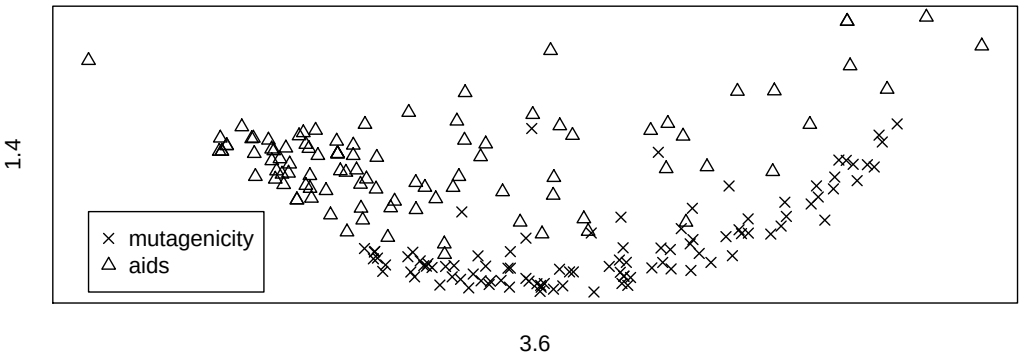


Figure 8.14: MDS projection of spectral distances between samples derived from molecules. Axis labels indicate the spectral distance realized on the corresponding axis. The projection preserves about 30% of the total distance information.

onto these two classes allows the realization of more specific distance information with respect to the objects of the two classes. This is additionally illustrated by the distances realized on the individual axes of the diagrams, which are considerably smaller than those in Figure 8.13. Though there is not such a clear class separation to be observed as in the previous example, the two types of molecules are spatially separated. Note, that the diagram shows only 30% of the total distance information, i.e. the spatial separation could even improve when more information is taken into account.

Altogether, the clarity of this result is surprising since neither is class separation an aim of the MDS method, nor is the class information part of its input. That is, the observable separability of classes is only the result of differences in the spectra of the involved graphs, i.e. the eigenvalue distributions of graphs in these classes are measurably different.

The next section will consider a set of graphs representing molecules of different types, which is in a sense an extension of the setting examined above.

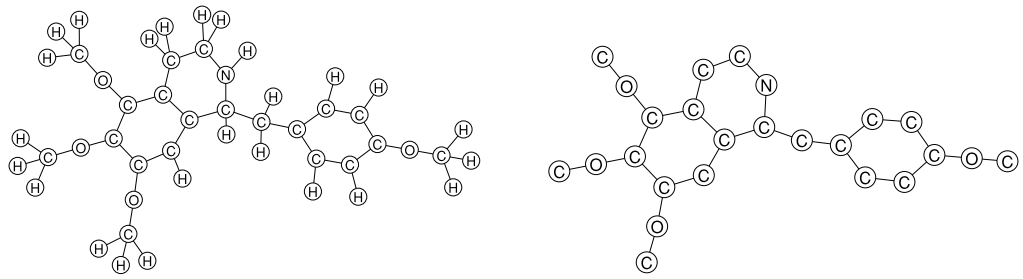
8.4.2 Molecules

This section considers an ensemble of graphs that is comparable to the one from the last section, only with a higher expected similarity since all of the graphs are derived from the same domain. As before, only $\mathcal{STC}_A^1$ will be considered as distance measure.

The assumption of structural difference in the considered graphs in the last sections was derived from their vastly differing source domains and different creation methods. In contrast, in the following setting all samples are representing chemical compounds, modelling the corresponding molecules in the same way. The set of samples consists of 147 molecules from four different chemical groups:

- phenylquinolones
- podophyllotoxins
- steroids
- styryls.

Molecules are modeled as graphs with nodes referring to atoms and links referring to bonds. For practical reasons - it improves class distinctions - all hydrogen atoms and the corresponding bonds were removed. The resulting graphs were then further simplified by removing loops and multiple edges. Figure 8.15 shows an example for a molecule modeled as a graph and its reduced version. The ensemble again consists of small graphs,



(a) original structure (b) reduced structure

Figure 8.15: Original molecule and its reduced version with all hydrogen atoms removed.

illustrated by the size distribution shown in Figure 8.16.

These graphs and the chemical groups their original molecules belong to, form the data for an experiment in the same fashion as that of the last section. Note that the derivation of graphs removed a major part of the information originally provided. Removing the hydrogen atoms actually improves the results, besides that all the information about the involved atom types is completely disregarded. The question examined in the following is, whether purely structural differences distinguish compounds from the different classes and whether these can be measured by spectrum transformation cost.

To test this, the approach of the last section is reiterated by assessing the possible separation visually, using the first dimensions of MDS projections. Here, however,

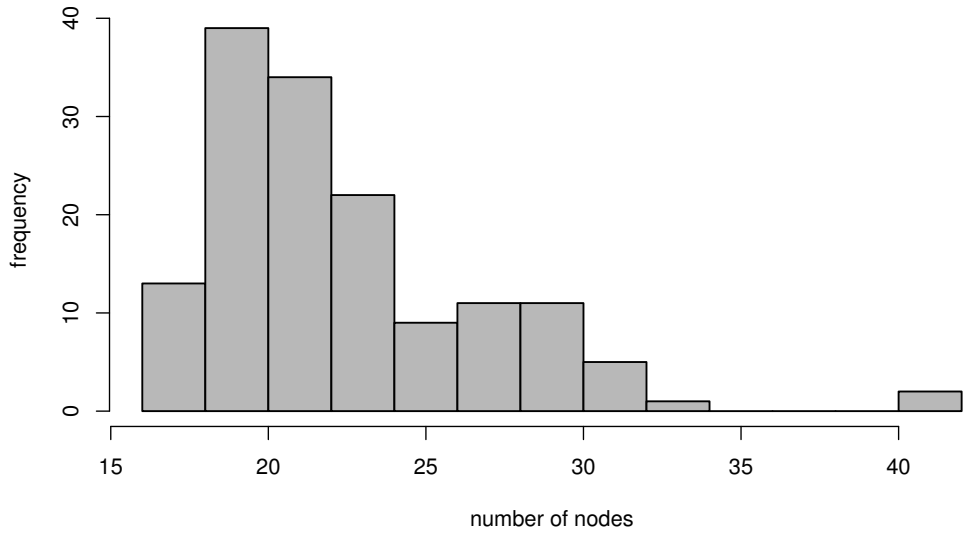


Figure 8.16: Distribution of graph size among samples representing molecules.

structural differences show to be smaller than before. Therefore, the groups are considered pairwise in Figure 8.17 and not all at once as in the last experiment. For each pair of classes, the projection of distances between samples from only these two classes is shown separately. Note further, that the original aspect ratio of the projection is distorted for illustrative purposes. The total distance shown on the individual axes is denoted on the bottom (first dimension) and left (second dimension) axis. The title of each diagram gives the two classes under comparison and the fraction of total distance information preserved by the particular projection.

It can be observed, that the different molecules are not as clearly separated as the classes of stronger structural differences from the experiment of the last section. However, tendencies of spatial separation can be still be observed. Some classes are clearly separated from each other, e.g. podophyllotoxins and steroids, while others seem to be much more similar and harder to distinguish, e.g. phenylquinolones and styryls. Considering the information loss due to the transformation and the fact that only a fraction of the available information is shown which is chosen to minimize information loss and not to maximize class separation, these results encourage the usage of spectrum transformation cost in such scenarios. Note, however, that learning problems on molecules are usually much more specific and the approach presented here is not suggesting the applicability of spectral distances in molecule mining. The experiments rather suggest that structural comparison using eigenvalue distributions can be applied to very small graphs. Considering the setting, the observations also indicate that the considered chemical classes, or at least the considered examples, differ in the (graph) structure of the associated molecules.

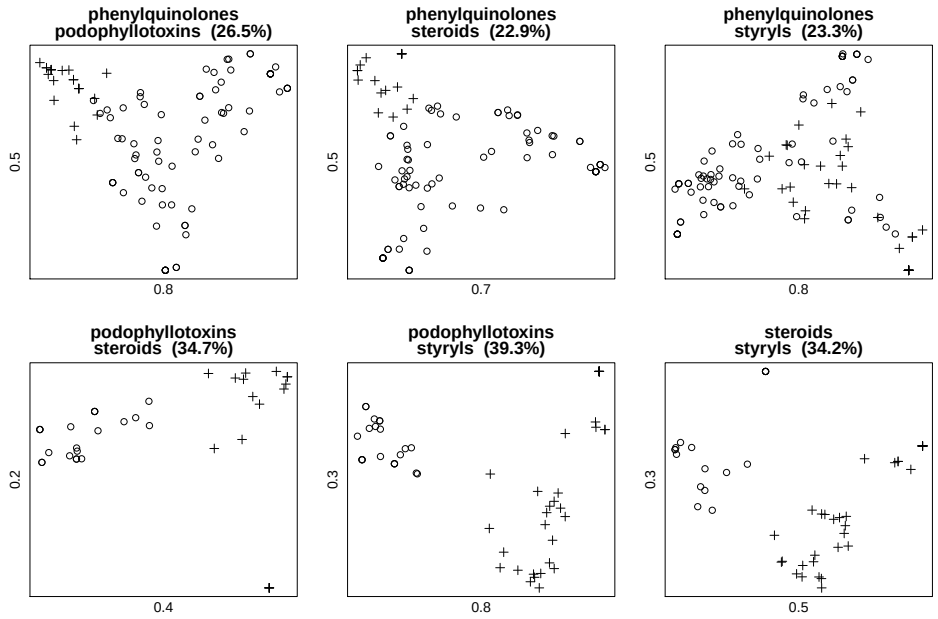


Figure 8.17: Distances between molecules from different chemical classes. For each pair of classes the first two components of an MDS are depicted with ∘ and + symbolizing samples of the first and second class respectively. See text for details.

8.5 Comparison of Distances

The previous sections of this chapter developed a number of scenarios in which spectrum transformation cost was examined for its applicability. Most of them tried to assess whether certain structural properties are expressed in the eigenvalue distribution in a way that can be exploited in the distance measurements by spectrum transformation cost to solve certain tasks. The assessment was, however, mostly of qualitative nature, i.e. illustrations showed that the produced distances are probably apt to solve the task at hand, while no concrete benchmarks were produced.

In addition, Section 3.5 presented a number of alternative approaches to derive distances between graphs from their eigenvalue distributions. Consequently, the following section will not only compare the different variants of spectrum transformation cost in a benchmark setting to each other, but also to the previously proposed approaches of spectral distance measurement. Therefore, the test scenarios introduced in the previous sections of this chapter will now serve as benchmark settings for a comparison of all spectral distances considered so far. To ensure comparability, this is restricted to distances based on the eigenvalue distributions. Some of the tasks at hand might be approached more successfully using other means of graph comparison which is, however, not of concern here. The basic assumption in the following is that the eigenvalue distributions contain sufficient information to solve the provided tests more or less successful. The different distances are then compared by their capability to exploit the available information for

classification tasks.

The considered distances include all variants of $\mathcal{STC}$, i.e. $\mathcal{STC}^1$, $\mathcal{STC}^2$ and their normalized counterparts $\overline{\mathcal{STC}}^1$, $\overline{\mathcal{STC}}^2$, using the spectra of the adjacency (A), Laplacian ($\mathcal{L}$) and normalized Laplacian ($\overline{\mathcal{L}}$) matrix representations as basis for measurement. In addition, the distances introduced in Section 3.5, denoted as **D1** to **D6** will be included in the comparison. For the latter, usually parameters have to be chosen. Instead of fine-tuning these for each individual experiment, the parameters employed for experiments in the original proposal are used. These choices are described in detail in Section 3.5.

8.5.1 Experimental Setup

Each of the graph comparison approaches yields only a set of distances and not an actual method of classification. Therefore, based on these distances a 3-nearest-neighbor voting process will be used for classification. This essentially results in a classification for each sample as the most frequent among the three samples of minimum distance, with ties being broken randomly. The choice of classification algorithm and especially the parameter three is somewhat arbitrary in this setup. Nearest neighbor voting was chosen for its simplicity and for the direct relation to the spatial distribution implied by the distance to be tested. Therein the consideration of only the three nearest neighbors results in a very local view, though the consideration of more neighbors might result in a better performance. To avoid random effects, a leave-one-out cross-validation is applied in every test. Together, this tests for each sample and distance, whether the class of the sample coincides with the majority of classes among its three nearest neighbors and thus whether classes are compact and separated from each other under the currently tested distance.

Consequently, the performance of a distance in relation to the test set is measured as the fraction of incorrect predictions. Due to the high number of distances and test sets, further visualizations of spatial distributions are omitted. The analysis of results will be based solely on the performance of each distance in each test set, measured as the fraction of incorrect predictions.

Test Cases

The test scenarios used in the following incorporate two types of samples: real world graphs as in the experiments of Section 8.4 and synthetic sets of samples drawn from sets of random graph models analogous to Section 8.2. The first two sets were already introduced in Section 8.4. The ensemble of graphs from different domains together with their domain are taken to be labeled with their original domain as the class which has to be predicted. This test set will be denoted *collections*. The second set is composed of the graphs derived from chemical compound of the four different groups, where given a sample the corresponding group has to be determined. This set will be denoted *molecules*.

The second kind of test scenarios is constructed as an extension of the test set used in Section 8.2.

Test Sets from Random Graph Models Extending the idea of Section 8.2 on classification of samples according the random graph model used for their creation, two larger test sets than the ones previously used are created in the following.

To emphasize the influence of size difference, the test sets will describe growing difficulty with respect to size differences. In both, a number of graph generation models with fixed parameters are used to create samples of varying sizes. Due to the prediction scheme, the outcome is highly dependent on the direct surrounding of each object. With respect to the objective of this experiment class predictions based solely on samples of similar size are to be avoided. Consider for example a method of distance measurement that yields small distances for graphs of the same type as long as they are of similar size and large distances if they are of different size. A class prediction by nearest neighbors could exploit neighboring graphs of the same class and size. To avoid this, only a single sample for each combination of random graph model and size is included in the test set.

In a first test set, denoted as *models*, sample sizes are varied only moderately. The employed models are: $M = \{G(n, 0.1)$, $G(n, 0.01)$, $G(n, 3/n)$, $G(n, 10/n)$, $\mathrm{pa}(n, 2, 1)$, $\mathrm{pa}(n, 4, 1)$, $\mathrm{pa}(n, 5, 1)$, $\mathrm{sw}(n, 3, 0.1)$, $\mathrm{sw}(n, 5, 0.1)$, $\mathrm{sw}(n, 5, 0.01)\}$. For each of these ten models samples of sizes $S = \{50, 100, 300, 500, 700, 900\}$ were drawn, resulting in 60 samples with 10 different class labels.

The purpose of the second test set is the amplification of size differences. Due to the computational effort necessary for the determination of the individual distances, the set of employed models is reduced to $M = \{G(n, 0.1)$, $G(n, 0.01)$, $G(n, 3/n)$, $G(n, 10/n)$, $\mathrm{pa}(n, 2, 1)$, $\mathrm{pa}(n, 5, 1)$, $\mathrm{sw}(n, 5, 0.1)$, $\mathrm{sw}(n, 5, 0.01)\}$ in exchange, the size range is extended and samples of sizes $300, 900, 1500, 2100$ and 2700 nodes were generated from each model. This second set is referred to by *models extreme* due to the extreme size differences involved. Again, class labels refer to the random graph model used for construction of the individual sample.

8.5.2 Results

Results of the test for all distances on all test sets are summarized in Table 8.1. The comparison of the error rate from all distances for each experiment shows that some of them form easier tasks such that most distances result in similar prediction rates, as e.g. the collections test set. In contrast, the experiments on random graphs with extreme size differences are highly discriminating, in that only a few distances achieve good prediction rates while most of them are completely inapt for this purpose.

On the *collections* test set, no distance results in an error rate higher than 30%. In fact, the majority of approaches achieve error rates lower than 10% with best results produced by $\mathcal{STC}^1_{\mathcal{L}}$ (3%), $\overline{\mathcal{STC}}^1_A$ (3.4%) and $\mathcal{STC}^1_A$ (3.6%). In addition, **D1** to **D4** also yield error rates below 10%, while **D5** and **D6** perform considerably worse. The small differences in the error rates, however, do not allow to infer a strong tendency from this scenario.

The second test set, *molecules*, is more distinctive in this respect. Error rates are quite low in comparison to the next two test sets, but only a few distances result in error

matrix	distance	collections	molecules	models	modelsextreme
$\mathcal{L}$	D1	6.40	14.97	81.67	50.00
$\mathcal{L}$	D2	5.60	**4.80**	**11.67**	17.50
A	D3	7.40	23.13	93.33	77.50
$\mathcal{L}$	D4	7.40	14.97	86.67	57.50
$\overline{\mathcal{L}}$	D5	19.00	6.80	55.00	27.50
$\overline{\mathcal{L}}$	D6	25.17	23.81	68.33	57.50
A	$\mathcal{STC}^1_A$	**3.60**	8.84	25.00	**10.00**
A	$\mathcal{STC}^2_A$	8.60	18.37	38.33	20.00
A	$\overline{\mathcal{STC}}^1_A$	**3.40**	10.88	35.00	12.50
A	$\overline{\mathcal{STC}}^2_A$	6.60	20.41	46.67	40.00
$\mathcal{L}$	$\mathcal{STC}^1_{\mathcal{L}}$	**3.00**	**4.76**	**21.67**	**10.00**
$\mathcal{L}$	$\mathcal{STC}^2_{\mathcal{L}}$	6.60	12.93	30.00	15.00
$\mathcal{L}$	$\overline{\mathcal{STC}}^1_{\mathcal{L}}$	10.80	**6.12**	38.33	15.00
$\mathcal{L}$	$\overline{\mathcal{STC}}^2_{\mathcal{L}}$	18.40	14.97	46.67	22.50
$\overline{\mathcal{L}}$	$\mathcal{STC}^1_{\overline{\mathcal{L}}}$	7.00	12.93	**18.33**	12.50
$\overline{\mathcal{L}}$	$\mathcal{STC}^2_{\overline{\mathcal{L}}}$	15.80	27.89	53.33	42.50
$\overline{\mathcal{L}}$	$\overline{\mathcal{STC}}^1_{\overline{\mathcal{L}}}$	6.60	14.97	26.67	**5.00**
$\overline{\mathcal{L}}$	$\overline{\mathcal{STC}}^2_{\overline{\mathcal{L}}}$	17.00	27.89	48.33	40.00

Table 8.1: Results of the individual validations for each combination of distance and test set. The four columns correspond to the individual experiments, each showing error rates visually and as percentage. For convenience of comparison, error rates as small diagrams. The black bar indicates the error rate on a scale, shown as gray background. The scale ranges from 0% to the maximal prediction error the corresponding test set. Additionally, for each test set the three lowest error rates are shown in bold.

rates below 10%. In particular, these are $\mathcal{STC}^1_{\mathcal{L}}$ (4.76%), **D2** (4.8%), $\overline{\mathcal{STC}}^1_{\mathcal{L}}$ (6.12%), **D5** (6.8%) and $\mathcal{STC}^1_A$ (8.84%).

Only **D2**, $\mathcal{STC}^1_A$, and $\mathcal{STC}^1_{\mathcal{L}}$ yield error rates below 10% in both settings. In addition, it can be observed that the $\mathcal{STC}^2$ variants of spectrum transformation cost perform in general worse than their $\mathcal{STC}^1$ counterparts. This relation holds constantly for all variants and all test sets and is independent of whether normalization is applied or not.

The following two test sets target model similarity in the setting of samples with differing sizes. This problem is not considered in the design of **D1** to **D6** and the approach of normalizing eigenvalue distributions by variance is merely an idea to approach this problem. Consequently, the average performance is worse than in the test sets before. Error rates on the *models*-setting are dominated by **D2** (11.67%), followed by $\mathcal{STC}^1_{\overline{\mathcal{L}}}$ (18.33%), $\mathcal{STC}^1_{\mathcal{L}}$ (21.67%) and $\mathcal{STC}^1_A$ (25%). Prediction errors of up to 93% and the large range of error rates illustrate, that performance in this setting is strongly depending on the way distances between spectra are measured. The normalization, originally intended to compensate for size differences, has in the setting a negative effect on almost all distance variants except for $\mathcal{STC}^2_{\overline{\mathcal{L}}}$.

Surprisingly, some of the distances result in lower error rates when applied to the *models extreme* setting. Best performance on these samples is shown by $\overline{\mathcal{STC}}^1_{\overline{\mathcal{L}}}$ (5%), $\mathcal{STC}^1_A$ (10%) and $\mathcal{STC}^1_{\mathcal{L}}$ (10%). **D2**, the best performing distance on the previous setting results in an error rate of 17.5% which is in the lower range for this setting. Also, this is the only setting, where one of the normalized versions of spectrum transformation cost yields best results. In general, however, normalization does not provide advantages over distance measurement without normalization.

Section 8.2 elaborated the difficulty of this setting in more detail. Figure 8.18 provides additional data by showing the sources of errors, i.e. the source models of incorrectly classified samples. As the discussion in Section 8.2 suggests, a major problem is posed by eigenvalue distributions strongly varying with graph size as in graphs from $G(n,p)$-models. In particular, no distance is able to correctly classify all samples of this class, while for each other class at least one such distance exists. In general, the distances with best performance produce most errors with samples from $G(n,p)$-models. Further, it can be seen that the normalization in some cases lowers the error rate for $G(n,p)$-samples while it often increases incorrect classifications of samples from other models. An example is provided by $\mathcal{STC}^1_A$ and $\overline{\mathcal{STC}}^1_A$: in the latter, the error made on $G(n,p)$-samples is decreased, while at the same time incorrect classifications of other sample are increased. An explanation could be derived from the massive gatherings of eigenvalues at certain fixed positions in samples from the $\mathrm{pa}(n,m,\alpha)$-model. These are independent of sample size as described in Section 7.4 and Section 8.2.1. While the larger eigenvalues increase transformation costs, these fixed points collect weight growing with graph size and thereby decrease the resulting distances. Normalization in contrast results in shifts of these points, such that they differ between graphs of different sizes, thereby causing additional transformation costs.

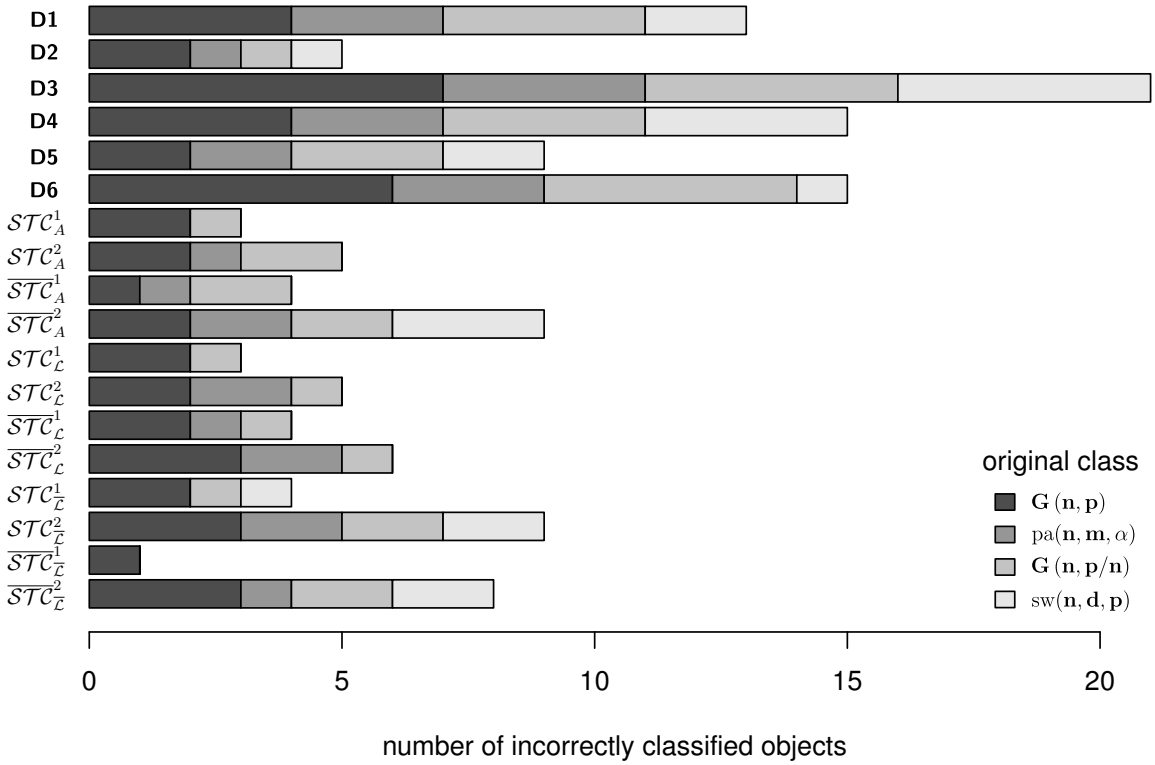


Figure 8.18: Original classes of the incorrectly classified graphs for each distance in the models extreme test set. Random distributions producing the samples are combined into the four categories $G(n,p)$, $\mathrm{pa}(n,m,\alpha)$, $\mathrm{sw}(n,d,p)$ and $G(n,p/n)$, i.e. $G(n,p)$ with constant expected degree.

Comparison The experiments of this section result in a large number of results for combinations of distances and test sets. In addition, the different test sets result in very different distributions of error rates. Therefore, Table 8.1 is only of limited help in an assessment of the overall performance of a single distance or the comparison of performance between different distances. Figure 8.19 provides a different approach of distance comparison: instead of comparing the absolute error rates, for each test set error rates are normalized to a common scale provided by the overall mean and standard deviation of error rates within the particular test. That is, considering each test independently, the mean error rate and the standard deviation of error rates over all tested distances is determined. Using these, each distance can be rescaled such that its performance can be shown as difference from mean error rate in units of standard deviation. Figure 8.19 compares distances by these values, e.g. a unit of "+std. dev." in the diagram reflects that this distance in this test resulted in an error rate higher than the mean error rate by one unit of standard deviation. Consequently, this approach allows to assess performance of a distance in comparison to the performance of the other distances. In addition, distance performance can be compared over the different sets. In the *collections* setting for example, even small differences in the error rate indicate an

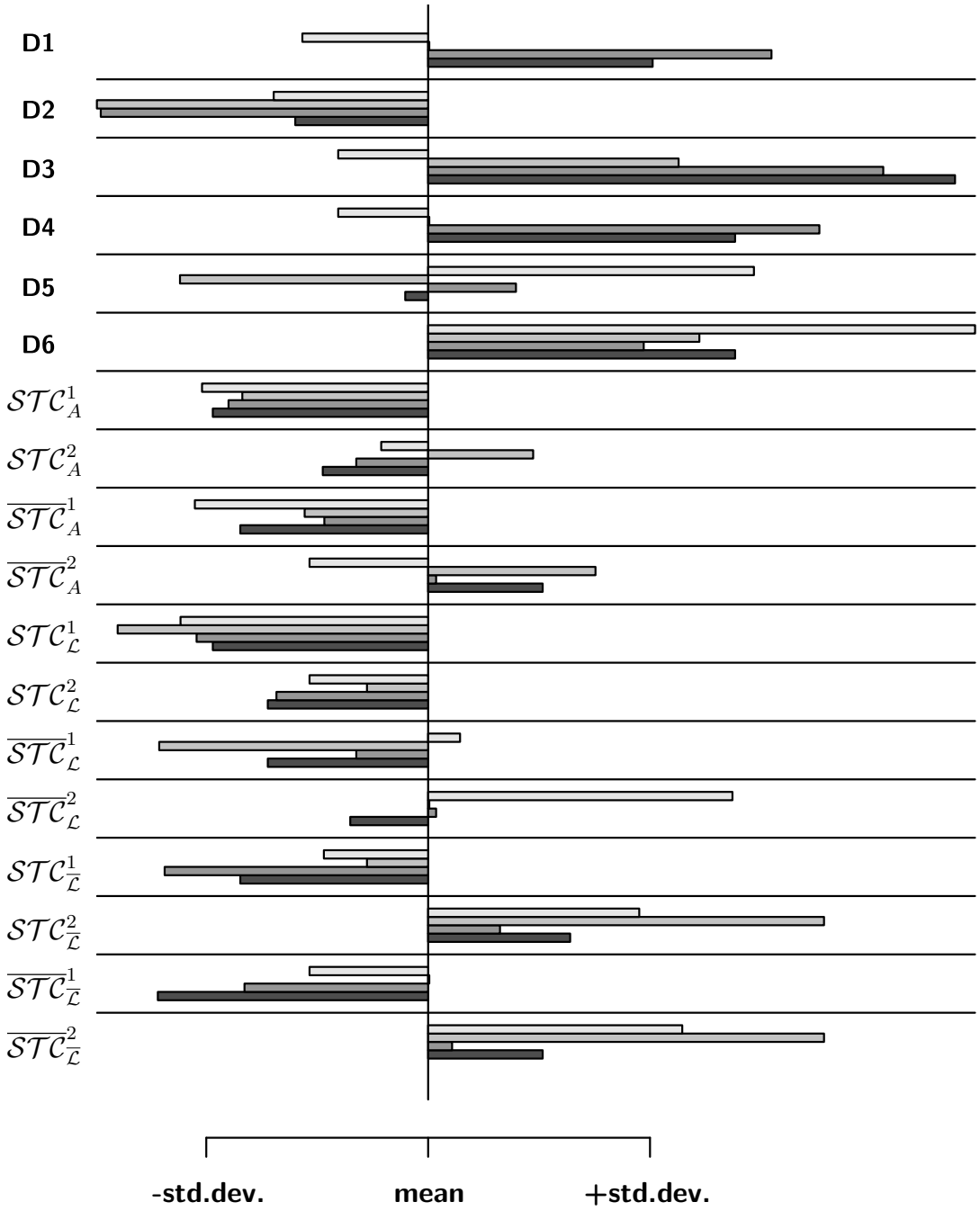


Figure 8.19: Comparison of the validation results for the individual data sets. Performance on test cases is compared by deviation from average performace of all approaches in an individual test case. Deviation is shown in units of standard deviation.

advantage since most distances yield very good results. On the other hand, in the *models* and *models extreme* settings a large range of different error rates can be observed and even multiples of the best error rate of 5% are not extremely bad. A robust distance with respect to all test sets should lead to normalized error rates that are constantly, significantly below the mean.

At first glance, some distances disqualify due to results constantly above the mean error rate as for example **D6**, $\mathcal{STC}^2_{\overline{\mathcal{L}}}$ and $\overline{\mathcal{STC}}^2_{\overline{\mathcal{L}}}$. Examples for the opposing behavior, i.e. being constantly below the average error rate, are **D2**, $\mathcal{STC}^1_A$ and $\mathcal{STC}^1_{\mathcal{L}}$, which show the most stable performance with a slight advantage in favor of $\mathcal{STC}^1_{\mathcal{L}}$. This result is biased due to the experiments regarding model similarity which target a special kind of similarity under special and disadvantageous conditions. Concentrating comparison on the first two experiments, however, favors $\mathcal{STC}^1_{\mathcal{L}}$ due to the poor performance of **D2** in the first experiment. On the real-world datasets considered above, $\mathcal{STC}^1_{\mathcal{L}}$ outperforms all other distances, though **D2** and $\mathcal{STC}^1_A$ perform comparably.

Besides exceptions and outliers, most variants of spectrum transformation cost show results below the average rates, which can not be observed for the distances proposed previously, except for **D2**. These would probably have profited from tuning of parameters, which is not necessary for transformation costs. For the spectrum transformation cost it can be observed that best results are produced by variants of $\mathcal{STC}^1$ without normalization.

In summary, the results of this section indicate that spectrum transformation cost is a method of graph comparison which yields results comparable to if not better than other approaches based on graph spectra. Note finally that the experiments conducted in this section are by no means exhaustive and only indicators for the applicability of spectrum transformation cost to various problem sets.

9 Summary and Conclusion

The approach for the analysis of network ensembles presented here considers networks or graphs as objects whose distribution has a certain structure to be identified. Such structures can be groups of objects of high similarity, leading to clustering problems, the identification of regularities to be exploited for classification or others. These tasks were assumed to be solved by techniques proposed in data mining and artificial intelligence and not solved in a new way. In the application of such methods to sets of graphs or networks, one problem is the comparison of the underlying objects with each other. Identifying structure in a set of objects depends on distance or similarity between them and therefore depends on a concrete measurement of distance or similarity. Considering homogeneous feature vectors, e.g. from $\mathbb{R}^n$, a number of distances and similarity measures have been proposed that can be chosen with respect to the task at hand. Analogous, but more recently, measures incorporating inhomogeneous feature vectors have been examined and methods have been developed to use them for learning and classification tasks. Finally, methods establishing distances and similarities on graphs and networks have been proposed, which are surveyed in Chapter 3.

In this dissertation, three new approaches to measure distances between graphs or networks were introduced. Two of them, introduced in Chapters 5 and 6 consider only the structure of graphs and are based on properties of eigenvalues derived from graphs. In contrast, the method of projecting networks to node partitions described in Chapter 4 relies on attributes available in those networks and thereby relates to the real-world properties encoded by these attributes. Basis of this method is a partition of all nodes in the ensemble to be analyzed. This node partition is derived independently from network structures using only node attributes. For a concrete application, it is required that this partition is meaningful with respect to the underlying data, i.e. the derived groups of nodes have to hold meaning in the context the network data was derived from. Since this initial node grouping completely determines the projections considered in further steps and thereby all features that can be derived from the projected graphs, it is the core decision in the application. Based on this, it was shown how feature vectors can be derived that describe structural aspects of networks in a common feature space. Thereby, the problems on networks were transformed into problems on feature vectors, enabling the use of methods designed for feature vectors.

An approach that distinguishes graphs by a specific difference in their structure was developed in Chapter 5. This method considers only the structure of graphs without any attributes and assumes that every graph in the considered ensemble incorporates some block structure. It was shown that these block structures are systematically connected to eigenvalues of large magnitude and that these eigenvalues can be employed for differentiation by structure. One condition for the applicability of this approach is

the density of the involved blocks, i.e. block density remains constant under growing graph size. This and the sensitivity to noise indicated by the experiments of Chapter 5 severely limits applications of this method outside the assumed scenario.

Finally, in Chapter 6 a second, more general graph distance based on eigenvalues -*spectrum transformation cost*- was proposed. In contrast to the distance distinguishing block structures, this approach does not target a precise structural aspect that renders graphs more or less similar but tries to assess structural difference in general. This general idea of structural comparison by comparison of eigenvalue distribution is shared with a number of previous approaches in the same direction summarized in Section 3.5. Besides these commonalities, the proposed measurement of spectrum transformation cost is distinguished from the other approaches by

- the absence of numerical parameters to be optimized for an individual application scenario,
- the exploitation of almost all the information available in the eigenvalue distribution without coarsening the distribution or cropping of eigenvalues,
- an efficient method of computation without numerical problems, as e.g. in the approaches involving numerical integration.

A common problem of most distance measurements on graphs employing eigenvalue distributions is that interactions between structural properties and eigenvalue distributions are not completely understood. Despite that, large distances derived from the comparison of spectra are in this context usually interpreted as an indication for considerable structural dissimilarity between the two graphs without specifying exactly what structural dissimilarity denotes. Some aspects of structural similarity considered important were distinguished in Section 1.4 and a selection of results on their reflection in eigenvalue distributions of graphs was summarized in Chapter 7. Finally, the measurability of certain aspects of structural similarity by spectrum transformation cost was examined empirically in Chapter 8. In addition, the proposed approach was compared to other spectral graph distances on a number of test scenarios, resulting in comparable, in some cases superior, performance.

A number of points for possible extension have been left open. In the method of graph comparison by projections, the derivation of an initial node partition is accomplished solely on the basis of their attributes. The corresponding chapter argues in a context of social science that within social groups certain attributes are more or less homogeneous and therefore clustering on this basis may reveal social structure. Apart from this context, considering arbitrary graphs with node attributes, it may be of interest to examine the interplay between node partitions derived by attributes and the distribution of the feature vectors resulting from projection to this partition. The described approach assumes an interaction between node partition and the resulting projections only in one direction, i.e. the partition determines the projection. The node partition is therein assumed to capture the group structure which interacts with the relation described by the edges of

the individual networks. This could be extended by assuming that a good node partition in this sense results in "clean" projections. Clean projection denoting here either that connections between the different groups of nodes are homogeneous in the individual networks or that the feature vectors derived from the networks exhibit homogeneous groups, i.e. typical structures. Allowing interaction between node partition and the resulting projections in both directions could yield further insight, e.g. node partitions revealing connection structures in the ensemble and in turn attributes responsible for group formation. While the involved a priory knowledge in the approach is decreased, the gained insight could increase.

In the approach to distinguish graphs by block structure, further work could consider the actual block structures involved. In Chapter 5, it was shown that the distance increases with graph size for graphs from different blockmodels while the actual blockmodels have not been identified. An extension of interest would be the reconstruction of the involved block structures, i.e. identify the classes of nodes forming the block structure in an individual graph. This could possibly be achieved by an analysis of the eigenvectors associated with the eigenvalues analogous to methods of spectral clustering. Exploiting such a relation, it may even be possible to avoid the limitation introduced by the asymptotic nature of the argument. It could in some cases be possible to identify eigenvalue/eigenvector pairs corresponding to block structures not only by magnitude of the eigenvalue but, in addition, by the structure of the corresponding eigenvector. Brandes and Lerner (2010) examine such a relations, though the selection of eigenvalues is an open problem.

The introduction of spectrum transformation cost already mentioned the possible extension to the case of directed graphs, i.e. distributions of complex eigenvalues. Whether this extension results in a similar measure of structural similarity or the behavior is completely different would have to be clarified by further experiments. Another possible extension is the integration of node attributes. Edge attributes can be already be integrated by using the weighted adjacency matrix as long as they are expressed by a real number. The integration of node attributes, however, demands further consideration since they are not directly placed in a matrix representation of the graph. A first idea would be to represent the difference between attributes of two connected nodes as the weight of the edge connecting them and to apply the approach considered for edges. This would, however, introduce a possible bias, since only attribute differences between connected nodes are considered. On the other hand, deriving the matrix with entries consisting of attribute difference between all pairs of nodes, i.e. the completely filled matrix, neglects the structure of the underlying network since edges are not expressed in this construct. Finding a balance between the two approaches or combining them somehow is an interesting possible extension of the method discussed so far.

Bibliography

Ahuja, R. K., **Magnanti**, T. L., and **Orlin**, J. B. *Network Flows*. Prentice Hall, **1993**. 73

Alon, N., **Krivelevich**, M., and **Vu**, V. H. On the concentration of eigenvalues of random symmetric matrices. *Israel Journal of Mathematics*, 131(1):259–267, **2002**. 53

Baltz, A. and **Kliemann**, L. Spectral analysis. In Brandes and Erlebach (2005), pages 373–416. 84, 85, 86

Banerjee, A. Structural distance and evolutionary relationship of networks. *ArXiv e-prints*, **2009**. http://arxiv.org/abs/0807.3185. 4, 24, 25, 28, 66, 67

Banerjee, A. and **Jost**, J. Laplacian Spectrum and Protein-Protein Interaction Networks. *ArXiv e-prints*, **2007**. http://arxiv.org/abs/0705.3373. 4

Banerjee, A. and **Jost**, J. On the spectrum of the normalized graph Laplacian. *Linear Algebra and its Applications*, 428:3015–3022, **2008**. 11, 87

Barabási, A.-L. and **Albert**, R. Emergence of scaling in random networks. *Science*, 286:509–512, **1999**. 14

Bauer, M. and **Golinelli**, O. Random incidence matrices: moments of the spectral density. *Journal of Statistical Physics*, 103:301–307, **2001**. 91, 92, 93

Bean, D. R. Effective coloration. *The Journal of Symbolic Logic*, 41(2):469–480, **1976**. 90

Berry, J. W. Immigration, acculturation, and adaptation. *Applied Psychology*, 46(1):5–68, **1997**. 41, 46

Berthold, M. R., **Borgelt**, C., **Hoeppner**, F., and **Klawonn**, F. *Guide to Intelligent Data Analysis: How to Intelligently Make Sense of Real Data*, volume 42 of *Texts in Computer Science*. Springer-Verlag, **2010**. 45

Bezáková, I., **Kalai**, A., and **Santhanam**, R. Graph model selection using maximum likelihood. In *Proceedings of the 23rd international conference on Machine learning (ICML '06)*, pages 105–112. ACM Press, **2006**. 21

Biane, P. On the free convolution with a semi-circular distribution. *Indiana University Mathematics Journal*, 46(3):705–718, **1997**. 91

Blau, P. M. *Inequality and Heterogeneity*. Free Press, **1977**. 33

Boguñá, M., **Pastor-Satorras**, R., **Diaz-Guilera**, A., and **Arenas**, A. Models of social networks based on social distance attachment. *Physical Review E*, 70(056122), **2004**. 33

Bollobás, B. *Random graphs, Second edition*. Number 73 in Cambridge Studies in Advanced Mathematics. Cambridge University Press, **2001**. 14

Borgelt, C. and **Berthold**, M. R. Mining molecular fragments: finding relevant substructures of molecules. *Data Mining, IEEE International Conference on*, 0:51–58, **2002**. 4, 21

Brandes, U. and **Erlebach**, T., editors. *Network Analysis: Methodological Foundations*, volume 3418 of *Lecture Notes in Computer Science*. Springer-Verlag, **2005**. 135, 136, 140

Brandes, U., **Kenis**, P., **Lerner**, J., and **van Raaij**, D. Network analysis of collaboration structure in Wikipedia. In *Proceedings of the 18th International World Wide Web Conference (WWW2009)*, pages 731–740. ACM Press, **2009**a. 62

Brandes, U. and **Lerner**, J. Structural similarity: Spectral methods for relaxed blockmodeling. *Journal of Classification*, 27(3):279–306, **2010**. 133

Brandes, U., **Lerner**, J., **Lubbers**, M. J., **McCarty**, C., and **Molina**, J. L. Visual statistics for collections of clustered graphs. In *Proceedings of the IEEE Pacific Visualization Symposium (PacificVis'08)*, pages 47–54. IEEE Computer Society, **2008**. 39, 40, 41, 42

Brandes, U., **Lerner**, J., **Lubbers**, M. J., **McCarty**, C., **Molina**, J. L., and **Nagel**, U. Recognizing modes of acculturation in personal networks of migrants. In *Proceedings of the 6th Conference on Applications of Social Network Analysis (ASNA'09)*, volume 4 of *Procedia Social and Behavioral Sciences*, pages 4–13. Elsevier, **2010**. ii, 2

Brandes, U., **Lerner**, J., and **Nagel**, U. Network ensemble clustering using latent roles. *Advances in Data Analysis and Classification*, 5(2):81–94, **2011**. ii, 2

Brandes, U., **Lerner**, J., **Nagel**, U., and **Nick**, B. Structural trends in network ensembles. In *Proceedings of the International Workshop on Complex Networks (CompleNet'09)*, volume 207 of *Studies in Computational Intelligence*, pages 83–97. Springer-Verlag, **2009**b. ii, 2

Brinkmeier, M. and **Schank**, T. Network statistics. In Brandes and Erlebach (2005), pages 293–317. 20

Bryc, W., **Dembo**, A., and **Jiang**, T. Spectral measure of large random Hankel, Markov and Toeplitz matrices. *The Annals of Probability*, 34(1):1–38, **2006**. 91

Bunke, H. On a relation between graph edit distance and maximum common subgraph. *Pattern Recognition Letters*, 18:689–694, **1997**. 20

Bunke, H. Graph matching: Theoretical foundations, algorithms, and applications. In *International Conference on Vision Interface*, pages 82–88. **2000**. 20

Bunke, H. and **Allermann**, G. Inexact graph matching for structural pattern recognition. *Pattern Recognition Letters*, 1(4):245–253, **1983**. 20

Bunke, H. and **Jiang**, X. Graph matching and similarity. In H.-N. Teodorescu, D. Mlynek, A. Kandel, and H.-J. Zimmermann, editors, *Intelligent Systems and Interfaces*, pages 281–304. Kluwer Academic Publishers Group, **2000**. 20

Butts, C. T. and **Carley**, K. M. Multivariate methods for interstructural analysis. CASOS working paper, **2001**. Center for the Computational Analysis of Social and Organizational Systems, Carnegie Mellon University. 34

Chen, J. and **Yuan**, B. Detecting functional modules in the yeast protein-protein interaction network. *Bioinformatics*, 22(18):2283–2290, **2006**. 4

Chung, F. R. K., **Lu**, L., and **Vu**, V. H. Eigenvalues of random power law graphs. *Annals of Combinatorics*, 7:21–33, **2003**. 94

Comellas, F. and **Diaz-Lopez**, J. Spectral reconstruction of complex networks. *Physica A*, 387(25):6436–6442, **2008**. 24, 25

Comellas, F., **Fertin**, G., and **Raspaud**, A. Recursive graphs with small-world scale-free properties. *Physical Review E*, 69:037104, **2004**. 89

Conte, D., **Foggia**, P., **Sansone**, C., and **Vento**, M. Thirty years of graph matching in pattern recognition. *International Journal of Pattern Recognition and Artificial Intelligence*, 18(3):265–298, **2004**. 20

Cox, T. F. and **Cox**, M. A. A. *Multidimensional Scaling*. Monographs on Statistics and Applied Probability. Chapman & Hall/CRC, 2nd edition, **2001**. 13

Croft, D. P., **James**, R., and **Krause**, J. *Exploring Animal Social Networks*. Princeton University Press, **2008**. 3

Csárdi, G. and **Nepusz**, T. The igraph software package for complex network research. *InterJournal*, Complex Systems:1695, **2006**. 14

Cvetković, D. M., **Doob**, M., and **Sachs**, H. *Spectra of Graphs*. Johann Ambrosius Barth Verlag, **1995**. 83, 87, 88, 110

Demirci, M. F., **van Leuken**, R. H., and **Veltkamp**, R. C. Indexing through Laplacian spectra. *Computer Vision and Image Understanding*, 110(3):312–325, **2008**. 25

Desai, M. and **Rao**, V. A characterization of the smallest eigenvalue of a graph. *Journal of Graph Theory*, 18(2):181–194, **1994**. 11

Deshpande, M., **Kuramochi**, M., **Wale**, N., and **Karypis**, G. Frequent substructure-based approaches for classifying chemical compounds. *IEEE Transaction on Knowledge and Data Engineering*, 17(8):1036–1050, **2005**. 21

Ding, X. and **Jiang**, T. Spectral distributions of adjacency and Laplacian matrices of random graphs. *Journal of Applied Probability*, 20(6):2086–2117, **2010**. 90, 91

Dorogovtsev, S. N., **Goltsev**, A. V., and **Mendes**, J. F. F. Pseudofractral scale-free web. *Physical Review E*, 65:066122, **2002**. 89

Erdős, P. and **Rényi**, A. On random graphs I. *Publicationes Mathematicae Debrecen*, 6:290–297, **1959**. 14

Faust, K. and **Skvoretz**, J. Comparing networks across space and time, size and species. *Sociological Methodology*, 32:267–299, **2002**. 21

Fay, D., **Haddadi**, H., **Moore**, A. M., **Mortier**, R., **Uhlig**, S., and **Jamakovic**, A. A weighted spectrum metric for comparison of internet topologies. *ACM SIGMETRICS Performance Evaluation Review*, 37(3):67–72, **2010**a. 23, 24, 25

Fay, D., **Haddadi**, H., **Thomason**, A., **Moore**, A. M., **Mortier**, R., **Jamakovic**, A., **Uhlig**, S., and **Rio**, M. Weighted spectral distribution for internet topology analysis: Theory and applications. *IEEE/ACM Transactions on Networking*, 18(1):164–176, **2010**b. 26, 66

Ford, Jr., L. R. and **Fulkerson**, D. R. *Flows in Networks*. Princeton University Press, **1962**. 73

Fraley, C. and **Raftery**, A. E. Model-based clustering, discriminant analysis and density estimation. *Journal of the American Statistical Association*, 97(458):611–631, **2002**. 35, 39

Fröhlich, H., **Wegner**, J. K., **Sieker**, F., and **Zell**, A. Kernel functions for attributed molecular graphs - a new similarity-based approach to ADME prediction in classification and regression. *QSAR & Combinatorial Science*, 25(4):317–326, **2006**. 4, 20

Füredi, Z. and **Komlós**, J. The eigenvalues of random symmetric matrices. *Combinatorica*, 1(3):233–241, **1981**. 91

Gangbo, W. and **McCann**, R. The geometry of optimal transportation. *Acta Mathematica*, 177(2):113–161, **1996**. 72

Garey, M. R. and **Johnson**, D. S. *Computers and Intractability. A Guide to the Theory of $\mathcal{NP}$-Completeness*. W. H. Freeman and Company, **1979**. 17

Gärtner, T., **Flach**, P., and **Wrobel**, S. On graph kernels: Hardness results and efficient alternatives. In *Proceedings of the 16th Annual Conference on Computational Learning Theory and 7th Kernel Workshop*, volume 2777 of *Lecture Notes in Computer Science*, pages 129–143. Springer-Verlag, **2003**. 21

Gilbert, E. N. Random graphs. *The Annals of Mathematical Statistics*, 30(4):1141–1144, **1959**. 14

Godsil, C. D. and **McKay**, B. D. Constructing cospectral graphs. *Aequationes Mathematicae*, 25:257–268, **1982**. 109, 110, 111

Godsil, C. D. and **Royle**, G. *Algebraic Graph Theory*. Graduate Texts in Mathematics. Springer-Verlag, **2001**. 83

Goh, K.-I., **Kahng**, B., and **Kim**, D. Spectra and eigenvectors of scale-free networks. *Physical Review E*, 64(5):051903, **2001**. 102

Golub, B. and **Jackson**, M. O. How homophily affects communication in networks. *ArXiv e-prints*, **2008**. http://arxiv.org/abs/0811.4013. 51

Golub, G. H. and **Van Loan**, C. F. *Matrix Computations*. John Hopkins University Press, 3rd edition, **1996**. 11, 12, 94

Grabowski, A. and **Kosiński**, R. A. Evolution of a social network: The role of cultural diversity. *Physical Review E*, 73:016135, **2006**. 33

Guhr, T., **Müller-Groeling**, A., and **Weidenmüller**, H. A. Random-matrix theories in quantum physics: common concepts. *Physics Reports*, 299(4-6):189–425, **1998**. 14

Haemers, W. H. and **Spence**, E. Enumeration of cospectral graphs. *European Journal of Combinatorics*, 25:199–211, **2004**. 76, 109

Hanna, S. Representation and generation of plans using graphs spectra. In A. S. Kubat, Ö. Ertekin, Y. I. Güney, and E. Eyüboğlu, editors, *Proceedings of the 6th International Space Syntax Symposium*, pages 099.01–099.13. Istanbul Technical University, **2007**. 3, 25, 26

Hanna, S. Spectral comparison of large urban graphs. In D. Koch, L. Marcus, and J. Steen, editors, *Proceedings of the 7th International Space Syntax Symposium*, pages 039:1–039:11. Royal Institute of Technology (KTH): Stockholm, Sweden, **2009**. 3, 25, 26

Harary, F., **Hayes**, J. P., and **Wu**, H.-J. A survey of the theory of hypercube graphs. *Computers & Mathematics with Applications*, 15(4):277–289, **1988**. 90

Holland, P. W., **Laskey**, K. B., and **Leinhardt**, S. Stochastic blockmodels: First steps. *Social Networks*, 5(2):109 – 137, **1983**. 16, 18, 34

Höllinger, F. and **Haller**, M. Kinship and social networks in modern societies: a cross-cultural comparison among seven nations. *European Sociological Review*, 6(2):103–124, **1990**. 4

Horváth, T. Cyclic pattern kernels revisited. In *Advances in Knowledge Discovery and Data Mining*, volume 3518 of *Lecture Notes in Artificial Intelligence*, pages 791–801. Springer-Verlag, **2005**. 21

Horváth, T., **Gärtner**, T., and **Wrobel**, S. Cyclic pattern kernels for predictive graph mining. In *KDD '04: Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 158–167. ACM Press, **2004**. 21

Ipsen, M. and **Mikhailov**, A. S. Evolutionary reconstruction of networks. *Physical Review E*, 66:046109, **2002**. 24, 25, 28, 66, 67

Jain, A. K., **Murty**, M. N., and **Flynn**, P. Data clustering: a review. *ACM Computing Surveys*, 31(3):264–323, **1999**. 35

Jurman, G., **Visintainer**, R., and **Furlanello**, C. An introduction to spectral distances in networks (extended version). *ArXiv e-prints*, **2010**. http://arxiv.org/abs/arXiv:1005.0103, q-bio.MN/1005.0103-v2. 22, 25

Kirkpatrick, S. and **Eggarter**, T. P. Localized states of a binary alloy. *Physical Review B*, 6(10):3598–3609, **1972**. 84

Köbler, J., **Schöning**, U., and **Torán**, J. *The Graph Isomorphism Problem: Its Structural Complexity*. Birkhäuser Verlag, **1993**. 5

Koffas, M., **Roberge**, C., **Lee**, K., and **Stephanopoulos**, G. Metabolic engineering. *Annual Review of Biomedical Engineering*, 1:535–557, **1999**. 4

Lerner, J. Role assignments. In Brandes and Erlebach (2005), pages 216–252. 90

Lerner, J. *Structural Similarity of Vertices in Networks*. Ph.D. thesis, Universität Konstanz, **2007**. 87, 90

Levina, E. and **Bickel**, P. The earth mover's distance is the mallows distance: Some insights from statistics. In *IEEE Computer Society International Conference on Computer Vision (ICCV'01)*, pages 251–256. **2001**. 68, 74

Li, L., **Alderson**, D., **Doyle**, J. C., and **Willinger**, W. Towards a theory of scale-free graphs: Definition, properties, and implications. *Internet Mathematics*, 2(4):431–523, **2005**. 19

Lorrain, F. and **White**, H. C. Structural equivalence of individuals in social networks. *Journal of Mathematical Sociology*, 1:49–80, **1971**. 87

von Luxburg, U. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, **2007**. 11, 77

Mallows, C. L. A note on asymptotic joint normality. *The Annals of Mathematical Statistics*, 43:508–515, **1972**. 74

McPherson, M. A Blau space primer: prolegomenon to an ecology of affiliation. *Industrial and Corporate Change*, 13(1):263–280, **2004**. 33

McSherry, F. Spectral partitioning of random graphs. In *Proceedings of the 42nd Annual IEEE Symposium on Foundations of Computer Science (FOCS'01)*, pages 529–537. IEEE Computer Society, **2001**. 51, 52, 53, 56

Milligan, G. W. and **Cooper**, M. C. A study of standardization of variables in cluster analysis. *Journal of Classification*, 5:181–204, **1988**. 38

Mohar, B. and **Poljak**, S. Eigenvalues in combinatorial optimization. In R. Brualdi, S. Friedland, and V. Klee, editors, *Combinatorial and Graph-Theoretical Problems in Linear Algebra*, pages 107–151. Springer-Verlag, **1993**. 12

Monge, G. Mémoire sur la théorie des déblais et de remblais. In *Histoire de l'Académie Royale des Sciences de Paris, avec les Mémoires de Mathématique et de Physique pour la même année*, pages 666–704. **1781**. 72

Nadel, S. F. *The Theory of Social Structure*. Cohen & West LTD, **1957**. Reprinted by Routledge, 2004. 3, 17, 33, 41

Neuhaus, M. and **Bunke**, H. A random walk kernel derived from graph edit distance. In *Structural, Syntactic, and Statistical Pattern Recognition*, volume 4109/2006 of *Lecture Notes in Computer Science*, pages 191–199. Springer-Verlag, **2006**. 21

Parzen, E. On estimation of a probability density function and mode. *The Annals of Mathematical Statistics*, 33(3):1065–1076, **1962**. 27

Pincombe, B. Detecting changes in time series of network graphs using minimum mean squared error and cumulative summation. In *Proceedings of the 13th Biennial Computational Techniques and Applications Conference, CTAC-2006*, volume 48 of *ANZIAM Journal*, pages C450–C473. **2007**. 24, 25, 26

Price, D. d. S. A general theory of bibliometric and other cumulative advantage processes. *Journal of the American Society for Information Science*, 27(5-6):292–306, **1976**. 14

R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, **2010**. `http://www.r-project.org`. 14

Riesen, K. and **Bunke**, H. Classifier ensembles for vector space embeddings of graphs. In *Multiple Classifier Systems*, volume 4472/2007 of *Lecture Notes in Computer Science*. Springer-Verlag, **2007**. 4

Riesen, K. and **Bunke**, H. IAM graph database repository for graph based pattern recognition and machine learning. In *Structural, Syntactic, and Statistical Pattern Recognition*, volume 5342/2008 of *Lecture Notes in Computer Science*, pages 287–297. Springer-Verlag, **2008**. 115

Robles-Kelly, A. and **Hancock**, E. R. Graph edit distance from spectral seriation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(3):365–478, **2005**. 21

Rosenblatt, M. Remarks on some nonparametric estimates of a density function. *The Annals of Mathematical Statistics*, 27(3):832–837, **1956**. 27

Rubner, Y., **Tomasi**, C., and **Guibas**, L. J. The earth mover's distance as a metric for image retrival. *International Journal of Computer Vision*, 40(2):99–121, **2000**. 69, 71

Rückert, U. and **Kramer**, S. Optimizing feature sets for structured data. In *ECML 2007*, volume 4701 of *Lecture Notes in Artificial Intelligence*, pages 716–723. Springer-Verlag, **2007**. 21

Rüschendorf, L. Wasserstein metric. In M. Hazewinkel, editor, *Encyclopaedia of Mathematics*. Springer-Verlag, **2002**. 74

Saad, Y. *Iterative Methods for Sparse Linear Systems*. SIAM, **2003**. 12

Sanfeliu, A. and **Fu**, K.-S. A distance measure between attributed relational graphs for pattern recognition. *IEEE Transactions on systems, man, and cybernetics*, 13(3):353–362, **1983**. 20

Sarkar, S. and **Boyer**, K. L. Quantitative measures of change based on feature organization: Eigenvalues and eigenvectors. *Computer Vision and Image Understanding*, 71(1):110–136, **1998**. 26, 81

Schwenk, A. J. Almost all trees are cospectral. In *New Directions in the Theory of Graphs*, pages 275–307. Academic Press, **1973**. 109

Shen, H. C. and **Wong**, A. K. C. Generalized texture representation and metric. *Computer Vision, Graphics, and Image Processing*, 23:187–206, **1983**. 67

Stewart, G. W. and **Sun**, J.-G. *Matrix Perturbation Theory*. Academic Press, **1990**. 53, 56, 80, 81

Suzuki, J., **Sasaki**, Y., and **Maeda**, E. Hierarchical directed acyclic graph kernel. *Systems and Computers in Japan*, 37(10):58–68, **2006**. 3, 21

Tian, Y., **Hankins**, R. A., and **Patel**, J. M. Efficient aggregation for graph summarization. In *SIGMOD '08: Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 567–580. ACM Press, New York, NY, USA, **2008**. 34

Vapnik, V. N. *Statistical Learning Theory*. Wiley, **1998**. 1

Wagner, A. The yeast protein interaction network evolves rapidly and contains few redundant genes. *Molecular Biology and Evolution*, 18(7):1283–1292, **2001**. 4

Wasserman, S. and **Faust**, K. *Social Network Analysis: Methods and Applications*. Cambridge University Press, **1994**. 3, 17, 18

Watts, D. J. and **Strogatz**, S. H. Collective dynamics of "small-world" networks. *Nature*, 393:440–442, **1998**. 15, 19, 20

Werman, M., **Peleg**, S., and **Rosenfeld**, A. A distance metric for multidimensional histograms. *Computer Vision, Graphics, and Image Processing*, 32:328–336, **1985**. 67, 74

Wiener, H. Structural determination of paraffin boiling points. *Journal of the American Chemical Society*, 69(1):17–20, **1947**. 20

Wigner, E. P. Characteristic vectors of bordered matrices with infinite dimensions. *Aequationes Mathematicae*, 62(3):548–564, **1955**. 90

Wigner, E. P. On the distribution of the roots of certain symmetric matrices. *Annals of Combinatorics*, 67(2):325–327, **1958**. 90

Wigner, E. P. Random matrices in physics. *SIAM Review*, 9(1):1–23, **1967**. 14

Wilson, R. C., **Hancock**, E. R., and **Luo**, B. Pattern vectors from algebraic graph theory. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(7):1112–1124, **2005**. 4, 28

Xu, R. and **Wunsch**, II, D. Survey of clustering algorithms. *IEEE Transactions on Neural Networks*, 16(3):645–678, **2005**. 35

Zhu, P. and **Wilson**, R. C. A study of graph spectra for comparing graphs. In *Proceedings of the 16th British Machine Vision Conference*. **2005**. 24, 25, 82, 96

List of Symbols

symbol	description	definition
$\mathcal{C}$	partition	Page 10, Definition 2.1
cc	clustering coefficient	Page 19, Section 3.1
C_n	circle on n nodes	Page 9, Section 2.1
$\hat{\delta}$	distance on projected networks	Page 39, Section 4.2.3
δ	Dirac kernel	Page 67, Definition 6.1
Δ	number of triangles	Page 19, Section 3.1
$\Delta^{\alpha}(G_i, G_j)$	feature vector distance	Page 43, Section 4.3.2
D1	spectral distance 1	Page 26, Section 3.5.1
D2	spectral distance 2	Page 28, Section 3.5.1
D3	spectral distance 3	Page 25, Section 3.5.1
D4	spectral distance 4	Page 25, Section 3.5.1
D5	spectral distance 5	Page 27, Section 3.5.1
D6	spectral distance 6	Page 28, Section 3.5.1
$e_{r,s}()$	average degree	Page 37, Section 4.2.3
$\mathcal{E}$	ensemble of networks	Page 10, Definition 2.1
$\mathcal{E}(\dots)$	planted partition ensemble	Page 52, Definition 5.2
$\boldsymbol{\lambda}()$	vector of sorted eigenvalues	Page 12, Section 2.3
$\lvert\cdot\rvert_F$	Froebenius norm	Page 81, Section 7.1
$\mathcal{G}(n, k, \psi, P)$	planted partition model	Page 51, Definition 5.1
$\lVert\cdot\rVert_{\infty}$	infinity norm	
$\mathrm{JS}(\mathbf{a}, \mathbf{b})$	Jenson-Shannon divergence	Page 28, Section 3.5.1
K_n	complete graph on n nodes	Page 9, Section 2.1
$K_{k,l}$	complete bipartite graph on $k + l$ nodes	Page 9, Section 2.1
P	projection target	Page 36, Section 4.2.2
P_i, $P_{i,j}$	elements of graph mapped to part of P	Page 36, Section 4.2.2
P_n	path on n nodes	Page 9, Section 2.1
ρ	density of a graph	Page 19, Section 3.1
$s_r()$	relative node frequency	Page 37, Section 4.2.3
$\mathcal{STC}$	spectrum transformation cost for ρ	Page 70, Definition 6.6
$\mathcal{STC}^1$	spectrum transformation cost for $\lVert\cdot\rVert_1$	Page 75, Section 6.2
$\mathcal{STC}^2$	Mallows' distance for $p = 2$	Page 75, Section 6.2
$\overline{\mathcal{STC}}$	normalized spectrum transformation cost	Page 77, Definition 6.7

Index