2018

Networked Data Analytics: Network Comparison And Applied Graph Signal Processing

Weiyu Huang
University of Pennsylvania, whuang@seas.upenn.edu

Recommended Citation

Huang, Weiyu, "Networked Data Analytics: Network Comparison And Applied Graph Signal Processing" (2018). *Publicly Accessible Penn Dissertations*. 2826.
https://repository.upenn.edu/edissertations/2826

Networked Data Analytics: Network Comparison And Applied Graph Signal Processing

Abstract

Networked data structures has been getting big, ubiquitous, and pervasive. As our day-to-day activities become more incorporated with and influenced by the digital world, we rely more on our intuition to provide us a high-level idea and subconscious understanding of the encountered data. This thesis aims at translating the qualitative intuitions we have about networked data into quantitative and formal tools by designing rigorous yet reasonable algorithms. In a nutshell, this thesis constructs models to compare and cluster networked data, to simplify a complicated networked structure, and to formalize the notion of smoothness and variation for domain-specific signals on a network. This thesis consists of two interrelated thrusts which explore both the scenarios where networks have intrinsic value and are themselves the object of study, and where the interest is for signals defined on top of the networks, so we leverage the information in the network to analyze the signals. Our results suggest that the intuition we have in analyzing huge data can be transformed into rigorous algorithms, and often the intuition results in superior performance, new observations, better complexity, and/or bridging two commonly implemented methods. Even though different in the principles they investigate, both thrusts are constructed on what we think as a contemporary alternation in data analytics: from building an algorithm then understanding it to having an intuition then building an algorithm around it.

We show that in order to formalize the intuitive idea to measure the difference between a pair of networks of arbitrary sizes, we could design two algorithms based on the intuition to find mappings between the node sets or to map one network into the subset of another network. Such methods also lead to a clustering algorithm to categorize networked data structures. Besides, we could define the notion of frequencies of a given network by ordering features in the network according to how important they are to the overall information conveyed by the network. These proposed algorithms succeed in comparing collaboration histories of researchers, clustering research communities via their publication patterns, categorizing moving objects from uncertain measurmenets, and separating networks constructed from different processes.

In the context of data analytics on top of networks, we design domain-specific tools by leveraging the recent advances in graph signal processing, which formalizes the intuitive notion of smoothness and variation of signals defined on top of networked structures, and generalizes conventional Fourier analysis to the graph domain. In specific, we show how these tools can be used to better classify the cancer subtypes by considering genetic profiles as signals on top of gene-to-gene interaction networks, to gain new insights to explain the difference between human beings in learning new tasks and switching attentions by considering brain activities as signals on top of brain connectivity networks, as well as to demonstrate how common methods in rating prediction are special graph filters and to base on this observation to design novel recommendation system algorithms.

Degree Type

Dissertation

Degree Name

Doctor of Philosophy (PhD)

Graduate Group
Electrical & Systems Engineering

First Advisor
Alejandro Ribeiro

Subject Categories
Electrical and Electronics | Mathematics | Statistics and Probability

NETWORKED DATA ANALYTICS: NETWORK COMPARISON AND APPLIED GRAPH SIGNAL PROCESSING

Weiyu Huang

A DISSERTATION

in

Electrical and Systems Engineering

Presented to the Faculties of the University of Pennsylvania

in

Partial Fulfillment of the Requirements for the

Degree of Doctor of Philosophy

2018

Supervisor of Dissertation

Alejandro Ribeiro, Rosenbluth Associate Professor, Electrical and Systems Engineering

Graduate Group Chairperson

Alejandro Ribeiro, Rosenbluth Associate Professor, Electrical and Systems Engineering

Dissertation Committee

Robert Ghrist, Andrea Mitchell Penn Integrating Knowledge Professor of Mathematics and Electrical and Systems Engineering

Danielle S. Bassett, Eduardo D. Glandt Faculty Fellow and Associate Professor of Bioengineering and Electrical and Systems Engineering

Antonio Ortega, Professor of Electrical Engineering, University of Southern California

John D. Medaglia, Assistant Professor of Psychology, Drexel University

NETWORKED DATA ANALYTICS: NETWORK COMPARISON AND
APPLIED GRAPH SIGNAL PROCESSING

Acknowledgement

The past four and half years at Penn was full of pleasurable and memorable moments for me. As it says, “Life is a journey, not a destination”; I believe life is more of a journey with the moments together with the people we have met along. I’m very glad and grateful that I have met many great people on my journey so far, who have inspired, guided, mentored, and helped me, and whom I highly value and appreciate.

I would like to express my great gratitude to my advisor Prof. Alejandro Ribeiro. His great mentorship, patient guidance, sharpness, visionary planning, as well as eagerness to nurture students, has greatly impacted me in many aspects. I would like to thank him greatly for his great mentorship both professionally and personally, thank him for mentoring me on how to conduct research, how to apply academic research results into industry and practical challenges, and his constant nurturing and support throughout the past years.

Due thanks go to Profs. Robert Ghrist, Danielle S. Bassett, Antonio Ortega, and John D. Medaglia for agreeing to serve on my committee. I am doubly grateful to Prof. Antonio Ortega for traveling to attend my Ph. D. thesis defense. I want to thank Profs. Danielle S. Bassett and John D. Medaglia for our interdisciplinary collaboration in the past four years. I have great memories of our discussion on defining suitable domain-specific tools, interpretation of results from both areas, and brainstorming of names enjoyed by both research communities. I would like to specially thank Prof. John D. Medaglia for the support and guidance both professionally and personally. I also want to thank Prof. Robert Ghrist for the inspiring discussion on the core of persistent homology, which leads to results in several chapters in the first part of this thesis, and thank Prof. Antonio Ortega for our discussion during several conferences on finding “killer apps” for graph signal processing, which is the goal, irrespective of achieving or not, for the applications in the second part of this thesis.

In the past few years, I had the great pleasure to collaborate with some excellent researchers who contributed significantly to various parts of this thesis. I would like to thank Prof. John D. Medaglia, Prof. Danielle S. Bassett, Prof. Antonio G. Marques, Thomas A. W. Bolton, Prof. Dimitri Van De VilleProf. Scott T. Grafton, Dr. Nicholas F. Wymbs, Dr. Santiago Segarra, Leah Goldsberry, and Jeremy Ma. The work in this thesis would not have been possible without the productive collaboration with them. The material here has also benefited from the constant discussions with Dr. Brian Sadler, Prof. Geert Leus, Prof. Yuantao Gu, and Dr. Siheng Chen throughout the past three and half years, as well as the discussions with current and former lab-mates at the University of Pennsylvania: Ximing Chen, Prof. Zijian Guo, Dr. Aryan Mokhtari, Dr. Alec Koppel, Dr. Ceyhun Eksin, Dr. David Q. Sun, Meng Xu, Min Wen, Dr. Yichuan Hu, Thomas O. G. Bevelsborg, Santiago Paternain, Mark Eisen, Shih-Ling Phuong, Fernando Gama, Luiz Chamon, Mahyar Fazlyab, Markos Epitropou, Dr. Miguel Carvo, Dr. Soheil Eshgh, and Dr. James Stephan. Not only did the discussion spark multiple ideas in this thesis, it also generated many good memories in the last four and half years at Penn.

I would like to express my gratitude to my mentors and/or colleagues in Australia who made a great impact for me to decide to come to U.S. for doctoral studies: Dr. Zarko Krusevac, Ray Contreras, Mark Bidwell, Dr. Bradley E. Treeby, Prof. Rodney A. Kennedy, Prof. Zubiar Khalid, Prof. Parastoo Sadeghi, and Dr. Salman Durrani. Without their encouragement and support, I may have spent the past five years elsewhere, and this thesis would certainly not be possible. I would like to thank Dr. Baboo V. Gowreesunker, Dr. Jacob E. Mattingley, Dr. Behrooz Shahsavari, Sneha Kadetotad for their great support and for the exciting five months we experienced together. I would also like to thank Dr. Yunlong Wang, Dr. Yilian Yuan, Li Zhou, and Dr. Emily Zhao for the very enjoyable and productive collaboration.

Finally, my deepest gratitude goes to my family. To my parents, thank you for your constant encouragement and great help and support whenever I face difficulties. To my uncles and grandparents, thank you for your constant belief and support in me ever since I was young. To my parents in-law and uncles in-law, thank you for your love, support, and belief in me. And for you Lei, our journey just starts, what lies ahead is unimportant as long as we are holding the wheel together. The good thing is yet to happen.

Weiyu Huang
Philadelphia, April 2018.

ABSTRACT

NETWORKED DATA ANALYTICS: NETWORK COMPARISON AND APPLIED GRAPH SIGNAL PROCESSING

Weiyu Huang

Alejandro Ribeiro

Networked data structures has been getting big, ubiquitous, and pervasive. As our day-to-day activities become more incorporated with and influenced by the digital world, we rely more on our intuition to provide us a high-level idea and subconscious understanding of the encountered data. This thesis aims at translating the qualitative intuitions we have about networked data into quantitative and formal tools by designing rigorous yet reasonable algorithms. In a nutshell, this thesis constructs models to compare and cluster networked data, to simplify a complicated networked structure, and to formalize the notion of smoothness and variation for domain-specific signals on a network. This thesis consists of two interrelated thrusts which explore both the scenarios where networks have intrinsic value and are themselves the object of study, and where the interest is for signals defined on top of the networks, so we leverage the information in the network to analyze the signals. Our results suggest that the intuition we have in analyzing huge data can be transformed into rigorous algorithms, and often the intuition results in superior performance, new observations, better complexity, and/or bridging two commonly implemented methods. Even though different in the principles they investigate, both thrusts are constructed on what we think as a contemporary alternation in data analytics: from *building an algorithm then understanding it* to *having an intuition then building an algorithm around it*.

We show that in order to formalize the intuitive idea to measure the difference between a pair of networks of arbitrary sizes, we could design two algorithms based on the intuition to find mappings between the node sets or to map one network into the subset of another network. Such methods also lead to a clustering algorithm to categorize networked data structures. Besides, we could define the notion of frequencies of a given network by ordering features in the network according to how important they are to the overall information conveyed by the network. These proposed algorithms succeed in comparing collaboration histories of researchers, clustering research communities via their publication patterns, categorizing moving objects from uncertain measurmenets, and separating networks constructed from different processes.

In the context of data analytics on top of networks, we design domain-specific tools by leveraging the recent advances in graph signal processing, which formalizes the intuitive notion of smoothness and variation of signals defined on top of networked structures, and generalizes conventional Fourier analysis to the graph domain. In specific, we show how these tools can be used to better classify the cancer subtypes by considering genetic profiles as signals on top of gene-to-gene interaction networks, to gain new insights to explain the difference between human beings in learning new tasks and switching attentions by considering brain activities as signals on top of brain connectivity networks, as well as to demonstrate how common methods in rating prediction are special graph filters and to base on this observation to design novel recommendation system algorithms.

Contents

Notations

N_X	Pairwise network on the node set X
$r_X(x_0, x_1)$	Same as $r_X^1(x_0, x_1)$; relationship between pair of nodes
$r_X^k(x_{0:k})$	Relationship between tuples $x_0, x_1, \dots, x_k$; if necessary
$d_X^k(x_{0:k}),\ \ p_X^k(x_{0:k})$	Dissimilarity or proximity between tuples
$N_X^K,\ \ D_X^K,\ \ P_X^K$	High order network, dissimilarity network, proximity network
$\mathcal{N}^K,\ \ \mathcal{D}^K,\ \ \mathcal{P}^K$	Space of K-order network, dissimilarity network, and proximity network
$o(x_{0:k})$	Number of unique elements in the tuple
$\pi,\ \ \omega$	Maps between set of nodes
$\Gamma_{X,Y}^k(C),\ \ \Gamma_{X,Y}^k(\pi)$	Difference between the network measured by C or π
$d_{\mathcal{N}}^k,\ \ d_{\mathcal{D}}^k,\ \ d_{\mathcal{P}}^k$	k-order network distance by correspondence, default is $k = 1$
$d_{\mathcal{N},p},\ \ d_{\mathcal{D},p},\ \ d_{\mathcal{P},p}$	p-norm high order network distance
ϕ, ψ	Simplex in the form of $[x_{0:k}]$
$\Phi_k = \sum_i \beta_i \phi_i$	k-chain with coefficients β_i
$\partial_k : \Phi_k \mapsto \Psi_{k-1}$	Boundary operator
L and $\mathcal{L}$	Simplicial complex and filtration
$\mathcal{H}_k$	k-order homological features
Q_k and $\tilde{Q}_k$	k-order persistent diagram and extended persistent diagram
$\mathcal{Q} = \{\mathbf{q}\}$	Collection of points $\mathbf{q}$ in Q_k with birth time q_b and death time q_d
$b^k(\cdot, \cdot)$	Bottleneck distance between the corresponding persistent diagrams
$A_{X,C}^K$	Augmented network constructed from D_X^K and correspondence C
$\mathcal{G}_b$ and $\mathcal{G}_d$	Birth and death generators of homological features
$s_k(\mathbf{q})$	Homofrequency $q_d - q_b$ of the point $\mathbf{q}$ in the extended persistent diagram
$S_k = \{s_k\}$	Homospectrum: all homofrequencies of k-dimensional features
$S^K = \{S_k\}_{k=0}^{K-1}$	All homospectrum up to order $K - 1$
$\tilde{\mathcal{D}}^K$	Space of relaxed K-order dissimilarity network
$\mathcal{Z} : \tilde{\mathcal{D}}^K \to \mathcal{S}^K$	H-transform
$\mathbf{F}$ and f	H-filter applied on the homospectrum domain and network domain

M_X	Metric space on the node set X
$\bar{d}_X(x,x'),\ \underline{d}_X(x,x')$	Upper and lower bound on the unknown dissimilarity $d_X(x,x')$
$\mathcal{I}$	Space of $I_X = (X, \underline{d}_X, \bar{d}_X)$ of nodes and distance bounds
$O_X(\delta)$	Partitions of X at resolution parameter δ
$\mathcal{W} : \mathcal{I} \to \mathcal{O}$	Hierarchical clustering as a function from $\mathcal{I}$ to dendrograms
u_X	Ultrametric on the space X
$\bar{c}_X$ and $\underline{c}_X$	Minimum upper and minimum lower chain cost
sep^{α}	α-seperation, $\min_{x \neq x'}[\alpha \bar{c}_X(x,x') + (1-\alpha)\underline{c}_X(x,x')]$
$d_{\text{PE}},\ d_{\text{PE, V}},\ d_{\text{PE, Q}}$	Partial embedding distance, and w.r.t to the induced and sampled space
d_{E}	Embedding metric distance
(V_X, v_x)	Induced space of $N_X = (X, r_x)$ by adding interiors
$(Q_X, \hat{r}_X)$	Sampled space of $N_X = (X, r_x)$ by adding discrete interiors
w_{ab}	Length transversed in the direction parallel from node a to node b
$\mathbf{A},\ \mathbf{D},\ \mathbf{L}$	Graph adjacency, graph degree matrix, and graph Laplacian
$\mathbf{S},\ \mathbf{I}$	Graph shift operator, and identity matrix
$\mathbf{x},\ \mathbf{y},\ \mathbf{z}$	General graph signals
$\tilde{\mathbf{x}},\ \tilde{\mathbf{y}},\ \tilde{\mathbf{z}}$	Graph Fourier transform of the graph signals
$\mathbf{v}_k,\ \lambda_k$	Eigenvector and eigenvalue of the shift operator
$\mathbf{V},\ \mathbf{\Lambda} = \text{diag}(\boldsymbol{\lambda})$	Eigenvector and eigenvalue matrix
$\mathbf{H} = \sum_l h_l \mathbf{S}^l, \mathbf{G} = \sum_q g_q \mathbf{S}^q$	Graph filters
$\lVert \cdot \rVert,\ \lVert \cdot \rVert_p,\ \lVert \cdot \rVert_*$	General norm, p-norm, nuclear norm
$\mathbf{\Theta},\ \mathbf{V}_{\text{F}}$	Random phase factors, Fourier matrix
$\mathcal{S}_u,\ \mathcal{T}_i,\ \mathcal{R}$	Set of items rated by user u, set of users who rated item i, set of known ratings
$\mathbf{X},\ \hat{\mathbf{X}},\ \bar{\mathbf{X}}$	Observed ratings, estimated ratings, underlying ratings
$\otimes,\ \text{vec}$	Cartesian product and vector operation

Chapter 1

Introduction

Data is getting big, but more than big it is getting pervasive. As our lives become more dependent and integrated with the digital infrastructure, more aspects of our life get measured and recorded. This pervasiveness leads to the emergence of novel types of signals for which existing analytic tools cannot be applied easily. Networked data falls in this category. In the past two decades we have realized the importance of network models in understanding problems as disparate as knowledge bases in artificial intelligence, collaboration networks in the social sciences, and brain and gene networks in bioengineering. Yet, the analysis of these networks is largely based on heuristics and lags far behind the availability of tools to analyze more conventional signals such as time series and images. The goal of my thesis is to advance the field of network analytics which is concerned with the development of formal tools to analyze network data.

Often, networks have intrinsic value and are themselves the object of study. Other times, the network defines an underlying notion of proximity, but the object of interest is signals defined on top of this graph. In this thesis I propose to study both types of networked data. In the former space my goal is to develop tools for network discrimination based on the definition and evaluation of proper distances between networks. In the latter space my research is on exploring the application of graph signal processing tools to find insights that would be absent with other tools. My specific goal is to study the applicability to recommendation system and brain imaging analytics. A description of these two research directions is offered in the following.

1.1. Network Comparison

The purpose of this part of my thesis is to develop network discrimination tools that can be applied to network comparison problems that appear in neuroscience, biology, and the social sciences [1, 2, 3, 4]. As a prototypical example consider neurodegenerative diseases for which ultimate causes remain unknown but for which proximate causes are alterations in the pattern of brain connectivity. Memory, cognitive, coordination, and behavioral changes associated with

Parkinson's, Alzheimer's, and Huntington's diseases have all been associated with patterns of brain activity that have distinct markers when compared with the activity patterns of healthy individuals [5, 6]. Knowing these alterations in brain connectivity is not only useful to foster our understanding of these disorders but also as a diagnostic tool. Changes in connectivity allow differentiation of disorders that appear with similar symptoms – an example being the discrimination between Alzheimer's and frontotemporal dementia [7] – and can also be used for early diagnosis as patterns start to change before patients exhibit clear symptoms. The outcome of these research effort is a network discrimination tool that can solve this diagnostic question and other similar questions such as discerning collaboration mores of research communities [8] and predicting the mortality of an emergent virus by studying the shape of its evolutionary tree [9].

The problem of defining distances between networks is not complicated if nodes have equal labels in both networks [10, 11, 12]. The problem, however, becomes very challenging if a common labeling doesn't exist in both networks, as we need to consider all possible mappings between nodes of each network. This complexity has motivated the use of network features as alternatives to the use of distances. Examples of features that have proved useful in particular settings are clustering coefficients [13], neighborhood topology [14], betweenness [15], motifs [16], wavelets [17], as well as graphlet degree distributions or signatures [18, 19, 20]. Although feature analysis is often effective, it is application-dependent, utilizes only a small portion of the information conveyed by the networks, and networks not isomorphic may still have zero dissimilarity as measured by features. These drawbacks can be overcome with the definition of valid metric distances that are be universal, depend on all edge weights, and are null if and only if the networks are isomorphic [21]. We point out that one can think of defining distances between networks as a generalization of the graph isomorphism problem [22] where the question asked is whether two networks are the same or not. When defining network distances we also want a measure of how far the networks are and we want these measure to be symmetric and satisfy the triangle inequality [21]. In Chapter 2 we address the problem to construct metric distances in the space of network.

Moreover, in Chapter 2 we also consider high order networks that describe relationships between elements of tuples and address the problem of constructing valid metric distances between them. Most often, networks are defined as structures that describe interactions between pairs of nodes [23, 24]. This is an indisputable appropriate model for networks that describe binary relationships, such as communication or influence, but not so appropriate for problems in which binary, ternary, or n-ary relationships in general, have different implications. This is, e.g., true of coauthorship networks where we count the number of joint publications by groups of scholars. Papers written by pairs of authors capture information that can be used to identify important authors and study mores of research communities. However, there is extra information to be gleaned from collaborations between triplets of authors, or even single author publications. The importance of capturing tuple proximities between groups of nodes other than pairs has been recognized and exploited in multiple domains including coverage analysis in sensor networks [25, 26, 27], cognitive learning and memory [28], broadcasting in wireless networks [29], image ranking [30], three-dimensional object retrieval and recognition [31], and group relationship structure in social networks [32].

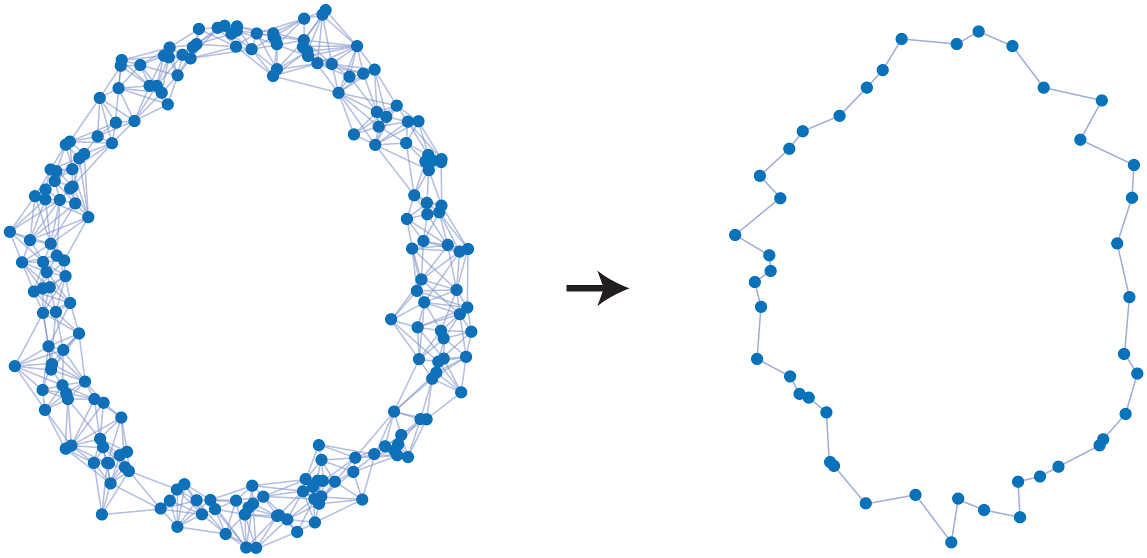


Figure 1: Network filtering based on persistent homology enables us to remove unimportant features to preserve the defining attributes represented by the network.

The metric distances between high order networks defined in Chapter 2 can be applied to compare networks with small number of nodes and succeed in identifying collaboration patterns of coauthorship networks. However, because they have to consider all possible node correspondences (Definition 1), network distances are difficult to compute when the number of nodes in the networks is large. The goal of Chapter 3 is to develop network discrimination methods that are computable in networks with large numbers of nodes. These discrimination methods are constructed by drawing a parallel between high order networks and algebraic topology filtrations. Homological features of filtrations are then used to compare high order networks and shown to provide *lower bounds* for the actual network distances. Although distance lower bounds suffer from some of the same problems associated with feature comparisons, they nonetheless have important properties. Among them, we know that a large lower bound entails a large distance and that we can use lower bounds to estimate distance intervals because upper bounds are easy to determine. The idea of using persistent homology to get lower bounds is related to the development of size theory – e.g. [33] – and the consideration of multidimensional persistent Betti numbers, see e.g. [34]. Besides, the discrete approach examined in this manuscript is not restrictive but closely related with continuous approaches considered in previous work such as [35].

In defining the network distance lower bounds via persistent homology, we realize that important features in a given network have a lasting homology while an unimportant feature tends to be more ephemeral in homology. Therefore, persistent homology introduces a natural notion of frequency for information in the network, and can be used to filter out irrelevant information to keep the most significant features. In Chapter 4 we explore how to *filter* a network to make it simpler while keeping its signatures. Intuitively, for the network describing the point cloud structure on the left of Figure 1, because most points form a ring structure, we would expect the *filtered* network to form a simpler and less twisting ring, like the right of Figure 1. In Chapter 4, we follow results in Chapter 3 to advocate the use of persistent homology [36, 37, 38] in such network filtering operation. It is known that more persistent homological features are more likely associated with true features, rather than artifacts of sampling or noise [38]. We leverage such charactersitic to define the frequency, spectrum, and the filter operation on networks. Our main contributions are: (i) We can define valid transforms from networks to homology frequency, and from homology frequency back to networks (Theorem 7), (ii) The difference between the original network and its filtered network is small (Theorem 8), and (iii) If we apply filter onto a pair of networks, the distance between the filtered networks well preserves the original distance between the pair (Theorem 9).

From the network distances defined in Chapter 2, we could evaluate network distance lower bound by using homological features of the corresponding network from results in Chapter 3. In the same time, network upper bounds can be easily obtained by using specific correspondence between the node spaces of the two networks. These two observations combined imply that we have both lower and upper bounds – computationally efficient – of the actual network distances, which are unknown due to computational intractability. In principal, we could cluster the space of networks using these bounds. Motivated by this, in Chapter 5 we study the clustering problem where distances between points are uncertain but known to be in an interval with lower and upper bounds. We use axiomatic approach by first defining reasonable requirements that any clustering methods in such problem should satisfy, and exploring which methods are reasonable. Clustering via distance intervals is a particular case of the problem of clustering with uncertain observations where the unpredictability is given by the distance intervals. Clustering methods that attempt to take uncertainty into consideration include the construction of models to replicate the properties of uncertainties in the data [39, 40, 41], the consideration of multiple observations of points given in a Euclidean space [42, 43, 44, 45], and the uncertainty exclusive clustering methods [46, 39, 47]. The work in Chapter 5 differs in that we investigate situations where the only available information are the upper and lower bounds of the dissimilarities. Since distance intervals can be constructed from partial information in the uncertain samples, distance intervals can be considered as a more crude observation.

It has been proved in [48] that single linkage [49, Ch. 4] is the unique hierarchical clustering method that satisfies three reasonable axioms. These results were later extended to asymmetric networks not necessarily metric, and the number of axioms required for unicity results reduced to only two [48, 21]. In the case of metric spaces the two properties that are imposed as axioms in [21] can be intuitively stated as:

(A1) Axiom of Value. Two nodes form a single cluster determined by their distance.

(A2) Axiom of Transformation. A metric space that is uniformly dominated by another metric space should have clusters that are uniformly dominated.

The goal of Chapter 5 is to extend the axiomatic construction of hierarchical clustering in [48, 21]

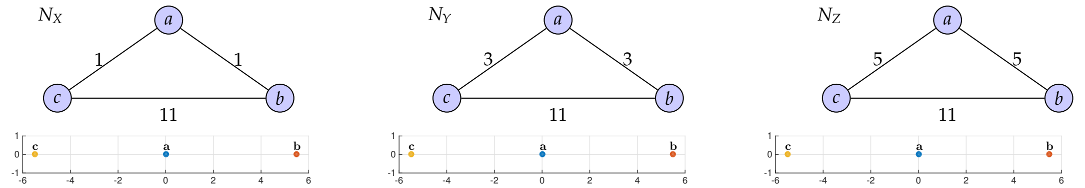


Figure 2: An example where different networks result in identical multi-dimensional scaling results. We emphasize that the number of dimension used in multi-dimensional scaling would not distinguish networks since the triangle inequality property for relationships between nodes in the networks is violated. Such a caveat would be solved by inducing semimetrics in the space defined by the given networks, as we develop throughout Section 6.2.

for clustering based on distance intervals. To adapt condition (A1) we introduce a confidence parameter which is intended to assign different relative trusts to lower and upper bounds and require that: (A1) The nodes in a network with two nodes are clustered at the convex combination of the interval extremes dictated by the confidence parameter. Condition (A2) are then adapted correspondingly to fit in the realm of confidence parameter for distance intervals. The contributions of Chapter 5 are: (i) To define the *combine-and-cluster* and *cluster-and-combine* methods that satisfy axioms (A1) and (A2). (ii) To prove that these methods are extremal across all methods that satisfy axioms (A1) and (A2). (iii) To demonstrate the practical applicability of the methods in the clustering of moving points via snapshots and the clustering of coauthorship networks denoting collaboration between researchers.

Finally, instead of searching mappings among nodes between networks as in the methods I developed, network discrimination problems can also be solved by considering that each network represents a space. Under such setup, network distances can be evaluated as the difference between the first network and the embedding of the second network onto the space defined by the first network. Prior work indicates that multi-dimensional scaling (MDS) yields good approximation of embedding distance when relationships satisfy triangle inequality [50, 51, 52]. However, we observed that MDS results are suboptimal since different networks may have identical MDS results, as illustrated in Figure 2. In Chapter 6 we propose to mitigate such issue by constructing interiors of networks, i.e. adding more points in the network that preserve the property of the original networks. Having the ability to extend networks into their interiors, we extend different networks and compute partial embedding distances between their extensions. In principle, distances between two networks and their respective extensions need not be related. In Section 6.2.1 we show that a restriction in the embedding of the extended networks renders them identical. The most important theoretical results is that we are able to show that the embedding distance that results from this restriction is the same embedding distance between the original networks. In Chapter 6 we exploit the interiors and embeddings to approximate the computation of embedding distances using the MDS techniques in [51] but applied to networks extended to their interiors. The definition of an interior improves the quality of MDS distance approximations.

1.2. Applied Graph Signal Processing

In the network discrimination part of my thesis, networks are themselves the interest of analysis. In this second thrust, the network serves as an underlying topology describing proximities between vertices and the subject of study is a signal supported on the network [53, 54]. The purpose of this part of my thesis is to develop domain specific network analysis tools by leveraging the information entailed by the network in order to make apparent the presence of features that would otherwise remain occult.

As an example, consider the core task of a recommendation system which is the estimation of the rating that a user would give to a certain item. Traditionally, ratings are considered as a two dimensional matrix and the main techniques used to solve the prediction problem are matrix completion [55] and collaborative filtering [56]. These methods rely on the fact that while the number of users of the system may be large there is a much smaller number of tastes. While empirical observations say that this is largely true, the inherent noise in available rankings limits the precision of these tools. My approach to this problem is to use ratings and meta-information to construct a network of similarities between items. The ratings given by each user can then be viewed as a graph signal. It further follows that the ratings given by the same user should be similar if the evaluated items are highly alike with respect to the similarity measure in the constructed network. The information can also be used to construct an underlying network entailing proximities between user tastes, and the ratings for each item could be considered as a graph signal defined on top of this network – the exact dual by switching the roles of items and users –, and the ratings given for the same item should be similar if the users who assess them possess similar tastes. In either case, the estimation of rankings can be formulated as a low pass filtering problem. The signal that we are given is highly variable because many ratings are missing. The signal that we want is a low frequency signal where similar nodes have similar ratings.

The emerging field of graph signal processing (GSP) addresses the problem of analyzing and extracting information from data defined not in regular domains such as time or space, but on more irregular domains that can be conveniently represented by a graph. The fundamental GSP concepts that we utilize to analyze brain signals are the graph Fourier transform (GFT) and the corresponding notions of graph frequency components and graph filters. These concepts are generalizations of the Fourier transform, frequency components, and filters that have been used in regular domains such as time and spatial grids [57, 58, 59]. As such, they permit the decomposition of a graph signal into pieces that represent different levels of variability. We can define low graph frequency components representing signals that change slowly with respect to brain networks, and high graph frequency components representing signals that change swiftly with respect to the connectivity networks. This is crucial because low and high *temporal* variability have proven to be important in the analysis of neurological disease and behavior [60, 61]. GFT-based decompositions permit a similar analysis of variability across regions of the brain for a fixed time – a sort of *spatial* variability measured with respect to the connectivity pattern. We review a recent study [62] that such a decomposition can be used to explain individual cognitive differences, as illustrated in Figures 76 and 78, and offer other perspectives to apply graph signal processing to functional brain analytics. The theory of GSP has been growing rapidly in recent years, with development in areas including sampling theory [63, 64], stationarity [65, 66] and uncertainty [67, 68, 69, 70], filtering [71, 72, 73], directed graphs [74], and dictionary learning [75]. Applications have been spanning many areas including neuroscience [76, 77], imaging [78, 79], medical imaging [80], video [81], online learning [82], and rating prediction [83, 84, 85, 86, 87].

After introducing notations and definitions of GSP in Chapter 7, we start the Part II in Chapter 8 by first considering signals supported on graphs and addressing the challenge of defining a notion of distance between these signals that incorporates the structure of the underlying network. We want these distances to be such that two signals are deemed close if they are themselves close – in the examples in the previous paragraph we have gene expression or brain activation patterns that are similar –, or if they have similar values in adjacent or nearby nodes – the expressed genes or the active areas of the brain are not similar but they effect similar changes in the gene network or represent activation of closely connected areas of the brain. We define here the diffusion and superposition distances and argue that they inherit this functionality through their connection to diffusion processes. Besides, in Chapter 8 we illustrate that both distances are well behaved with respect to small perturbations in the underlying network and demonstrate their value in two practical scenarios; the classification of ovarian cancer types from gene mutation profiles and the classification of handwritten arabic digits.

Advances in neuroimaging techniques such as magnetic resonance imaging (MRI) have provided opportunities to measure human brain structure and function in a non-invasive manner [88]. Diffusion-weighted MRI allows to measure major fiber tracts in white matter and thereby map the structural scaffold that supports neural communication. Functional MRI (fMRI) takes an indirect estimate of the brain approximately each second, in the form of blood oxygenation level-dependent (BOLD) signals. An emerging theme in computational neuroimaging is to study the brain at the systems level with such fundamental questions as how it supports coordinated cognition, learning, and consciousness.

Shaped by evolution, the brain has evolved connectivity patterns that often look haphazard yet are crucial in cognitive processes. The apparent importance of these *connectomes*, has motivated the emergence of network neuroscience as a clearly defined field to study the relevance of network structure for cognitive function [89, 90, 91]. The fundamental components in network neuroscience are graph models [92] where nodes are associated to brain regions and edge weights are associated with the strength of the respective connections. This connectivity structure can be measured directly by counting fiber tracts in diffusion weighted MRI or can be inferred from fMRI BOLD measurements. In the latter case, networks are said to be functional and represent a measure of co-activation, e.g., the pairwise Pearson correlation between the activation time series of nodes. Functional connectivity networks do not necessarily represent physical connections although it has been observed that there is a strong basis of anatomical support for functional networks [93].

Connectomes, structural and functional alike, have been successfully analyzed utilizing a variety of tools from graph theory and network science [92]. These analyses have uncovered a variety of measures that reflect organizational principles of brain networks such as the presence of communities where groups of regions are more strongly connected between each other than with other communities [93, 94]. Network analysis has also been related to behavioral and clinical measures by statistical methods or machine learning tools to study development, behavior, and ability [95, 96, 97].

As network neuroscience expands from understanding connectomes into understanding how connectomes and functional brain activity support behavior, the study of *dynamics* has taken center stage. In addition, there is a rise of interest in analyzing and understanding dynamics of functional signals and with them, network structure. Such changes happen at different timescales, from years – e.g., in developmental studies [98] – down to seconds within a single fMRI run of several minutes [99], or following tasks such as learning paradigms [97, 62, 76]. So far, common approaches include examining changes in network structure (e.g., reflecting segregation and integration) [100] or investigating time-resolved measures of the underlying functional signals [101, 102, 103]. In the case of developmental studies, the evolution of structural networks is important, but large-scale anatomical changes do not occur in the shorter time scales that are involved in behavior and ability studies. In the latter case, the notion of a dynamic network itself makes little sense and the more pertinent objects of interest are the dynamic changes in brain activity signals [62, 76]. Inasmuch as brain activity is mediated by physical connections, the underlying network structure must be taken into account when studying these signals. Tools from the emerging field of GSP are tailored for this purpose. We have applied such tools to analyze fMRI signals on top of the brain networks denoting functional connectivity (Chapter 9) as well as the brain networks quantifying structural connectivity (Chapter 10).

For brain networks denoting functional similarity between pairs of brain regions as in Chapter 9, we applied graph frequency analysis to a group of subjects try to learn a visual task. The most important observations are: (i) In terms of changes throughout the learning process, at the start of the task, BOLD signals concentrate on graph high frequency components; at the end of the task, BOLD signals concentrate on graph low frequency components; such change is significant. (ii) In terms of contribution to better learning performance, at the start of learning, BOLD signals concentrating on graph low frequency entail better learning performance; at the end of learning, BOLD signals concentrating on graph high frequency imply better learning performance; such change in association is significant as well.

For brain networks denoting structural connectivity between pairs of brain regions as in Chapter 10, I applied graph frequency analysis to a group of subjects in two tasks that require no attention switching, and active attention switching, respectively. We are interested in the additional time for subjects to switch their attention, and observed that: higher concentration in graph high frequency implies slower attention switching. These two results, from both the functional and structural brain perspectives, illustrate that when subjects are unfamiliar with an task, it is good to have BOLD measurements to be more concentrated on graph low frequency components.

Finally, GSP tools can also be used to predict unknown in modern recommendation systems. Widespread deployment of Internet technologies has generated massive enrollment of online customers in web services, propelling the emergence of recommendation systems to assist customers in making decisions [104, 105]. Recommendation systems use ratings that customers have given to specific products they have consumed to predict the ratings that similar users would give to similar products. In making these predictions, recommendation systems exploit product similarities and the closeness of user preferences, both of which can be inferred from information that is exogenous or endogenous to the system. The most popular exogenous information approach is content filtering, which starts by defining a set of features that characterize users and items and then uses those to perform predictions on the unrated items [104, 105]. The most popular endogenous information approach is collaborative filtering, which relies on past user behavior and carries out predictions without defining a priori set of features. Although collaborative filtering comes with certain disadvantages (in particular when rating new products or users), it typically requires less assumptions than content filtering and yields a superior performance in real datasets [105]. As a result, it has emerged as the central approach for recommendation systems.

The two main techniques to design collaborative filtering algorithms are nearest neighbors (NN) estimators and latent linear factor (LF) models. User-based NN schemes work under the assumption that users who are similar tend to give similar ratings for the same product and proceed in two phases. In the first phase they use a pre-specified similarity metric to compute a similarity score for each pair of users. To avoid over-fitting and simplify computations, only similarities that exceed a threshold are considered, thereby producing a sparse graph of user similarity scores [106]. In the second phase, the unknown ratings for a particular user are obtained by combining the ratings that similar users have given to the unrated items as dictated by the similarity graph. Likewise, item-based NN approaches work under the assumption that similar products receive similar ratings and create a product similarity graph to assign ratings to unrated products.

LF approaches bypass notions of user or product similarity by posing the existence of a number of latent factors that generate the user-item rating function. The main difference relative to content-filtering approaches is that here the factors are not defined a priori, but inferred from the available ratings. Although nonlinear latent factor models exist, the linear models based on matrix factorization (aka matrix completion methods [107]) combine tractability with good practical performance [105]. In particular, by arranging the available ratings in a matrix form with one of the dimensions corresponding to users and the other one to items, LF schemes typically carry out a low-rank singular value decomposition (SVD) that jointly maps users and items to a latent factor linear space of reduced dimensionality [105]. The rating user-item function is then modeled simply as inner products in the reduced subspace defined by the SVD.

The goal of Chapter 11 is to reinterpret collaborative filtering algorithms using tools from GSP and designs new and more general recommendation schemes, but equally relevant unveils important connections between collaborative filtering and GSP. More precisely, we show that NN can be viewed as algorithms that obtain the ratings by processing the available information with a graph filter. Moreover, matrix factorization methods can be reinterpreted as interpolation algorithms that, given a subset of signal observations (ratings), recover the full signal under the assumption that the ratings are bandlimited in a particular spectral domain. This interpretation not only provides a better understanding on the differences and similarities between the two approaches, but it also opens the door to the design of more advanced algorithms leading to a better recommendation accuracy. In a nutshell, the contributions of Chapter 11 are: (i) To demonstrate how the standard collaborative filtering approaches based on NN and LF can be interpreted as particular types of GSP algorithms that model the rating signal as bandlimited. (ii) To exploit this interpretation to design more general algorithms for NN and LF. (iii) To show that the proposed methods indeed produce significant improvement regarding rating prediction accuracy in the publicly available MovieLens 100k dataset [108]. (iv) To identify and discuss interesting observations found when using this GSP approach, which can be leveraged in the design of future recommendation system algorithms. Relative to existing contributions dealing with graph SP schemes for two-dimensional rating prediction [83, 109, 87], the work in this paper is more comprehensive, provides novel insights and puts forth new algorithms. More specifically, in the context of LF models, [83] formulates a low-rank matrix completion problem augmented with a regularizer term that promotes smoothness of the predicted ratings on the similarity graph. In comparison, our work proposes a bandlimited sampling implementation, where we separate the estimation of the frequency components (singular vectors) from the estimation of frequency coefficients (singular values) of the rating matrix; this separation allows for more general interpolators and reduces the computational complexity, see Section 11.4.1. In the context of NN prediction, [109] uses a pre-determined low-pass graph filter to predict ratings. In comparison, this work postulates more flexible graph filters, finds the optimal (band-stop) filter coefficients in the training phase, and introduces graph filters that operate in both the user and the item domain. Finally, [87], which was published after the submission of this manuscript, uses similarity graphs and graph SP to extract local spatial features from the observed ratings, and then feeds the extracted features into a recurrent neural network to diffuse entries to reconstruct the rating matrix. Our work uses graph SP to find the optimal higher order band-stop filter via training, and to separate the estimation of singular vectors from the estimation of singular values of the rating matrix.

1.3. Published Results

My Ph.D. work on network theory and applied graph signal processing has resulted in the publication of 10 journals papers in the Institute of Electrical and Electronic Engineers (IEEE) IEEE Transactions on Signal Processing [8, 76, 110, 111, 112, 113], IEEE Journal on Selected Topics in Signal Processing [114], IEEE Transaction on Signal and Information Processing over Networks [115], Nature Human Behavior [62], and Neurology [116]. A tutorial paper featuring work in this thesis appeared in the Proceedings of IEEE [117]. The work has also been disseminated at pertinent conferences where a total of 13 articles have been accepted for presentation [118, 119, 120, 121, 85, 122, 123, 84, 124, 125, 126, 127, 128].

Part I

Network Comparison

Chapter 2

Network Comparison via Correspondence

The main problem addressed in this chapter is the construction of metric distances between high order networks. Formal definitions of high order networks are presented (Section 2.2) as a generalization of pairwise networks (Section 2.1). Dissimilarity networks (Section 2.3) and proximity networks (Section 2.4) are specific high order networks where relationship functions are intended to encode dissimilarities or proximities between members of tuples. Dissimilarity networks are characterized by the order increasing property which states that tuples become more dissimilar when members are added to a group. Proximity networks abide to the order decreasing property which states that tuples becomes less similar when adding nodes to the group. Two families of proper metric distances are then defined in the respective space of dissimilarity (Section 2.3.1) and proximity (Section 2.4.1) networks modulo permutation isomorphisms. These distances are built as generalizations of the pairwise distances in [21], which are themselves generalizations of the Gromov-Hausdorff distance between metric spaces [129, 130]. We also establish a duality between dissimilarity and proximity networks and the different metrics (Section 2.4.2). We use the proximity network distances defined in the chapter to compare the coauthorship networks of two popular signal processing researchers and show that they succeed in discriminating their collaboration patterns (Section 2.5).

2.1. Pairwise Networks

Conventionally, a network is defined as a pair $N_X = (X, r_X^1)$, where X is a finite set of nodes and $r_X^1 : X^2 = X \times X \to \mathbb{R}_+$ is a function that may encode similarity or dissimilarity between elements. For points $x, x' \in X$, values of this function are denoted as $r_X^1(x, x')$. We assume that $r_X^1(x, x') = 0$ if and only if $x = x'$ and we further restrict attention to symmetric networks where $r_X^1(x, x') = r_X^1(x', x)$ for all pairs of nodes $x, x' \in X$. The set of all such networks is denoted as $\mathcal{N}$.

When defining a distance between networks we need to take into consideration that permutations of nodes amount to relabelling nodes and should be considered as same entities. We therefore say that two networks $N_X = (X, r_X^1)$ and $N_Y = (Y, r_Y^1)$ are isomorphic whenever there exists a bijection $\pi : X \to Y$ such that for all points $x, x' \in X$,

$$r_X^1(x, x') = r_Y^1(\pi(x), \pi(x')). \tag{2.1}$$

Such a map is called an isometry. Since the map π is bijective, (2.1) can only be satisfied when X is a permutation of Y. When networks are isomorphic we write $N_X \cong N_Y$. The space of networks where isomorphic networks $N_X \cong N_Y$ are represented by the same element is termed the set of networks modulo isomorphism and denoted by $\mathcal{N} \mod \cong$. The space $\mathcal{N} \mod \cong$ can be endowed with a valid metric [21]. The definition of this distance requires introducing the prerequisite notion of correspondence [131, Def. 7.3.17].

Definition 1 *A correspondence between two sets X and Y is a subset $C \subseteq X \times Y$ such that $\forall\, x \in X$, there exists $y \in Y$ such that $(x, y) \in C$ and $\forall\, y \in Y$ there exists $x \in X$ such that $(x, y) \in C$. The set of all correspondences between X and Y is denoted as $\mathcal{C}(X, Y)$.*

A correspondence in the sense of Definition 1 is a map between node sets X and Y so that every element of each set has at least one correspondent in the other set. Correspondences include permutations as particular cases but also allow for the mapping of a single point in X to multiple correspondents in Y or, vice versa. Most importantly, this allows definition of correspondences between networks with different numbers of elements. We can now define the distance between two networks by selecting the correspondence that makes them most similar as we formally define next.

Definition 2 *Given two networks $N_X = (X, r_X^1)$ and $N_Y = (Y, r_Y^1)$ and a correspondence C between the node spaces X and Y define the network difference with respect to C as*

$$\Gamma_{X,Y}^1(C) := \max_{(x_1,y_1),(x_2,y_2)\in C} \left| r_X^1(x_1, x_2) - r_Y^1(y_1, y_2) \right|. \tag{2.2}$$

The network distance between networks N_X and N_Y is then defined as

$$d_{\mathcal{N}}^1(N_X, N_Y) := \min_{C\in\mathcal{C}(X,Y)} \left\{ \Gamma_{X,Y}^1(C) \right\}. \tag{2.3}$$

For a given correspondence $C \in \mathcal{C}(X, Y)$ the network difference $\Gamma_{X,Y}^1(C)$ selects the maximum distance difference $|r_X^1(x_1, x_2) - r_Y^1(y_1, y_2)|$ among all pairs of correspondents – we compare $r_X^1(x_1, x_2)$ with $r_Y^1(y_1, y_2)$ when the points x_1 and y_1, as well as the points x_2 and y_2, are correspondents. The distance in (2.3) is defined by selecting the correspondence that minimizes these maximal differences. The distance in Definition 2 is a proper metric in the space of networks modulo isomorphism. It is nonnegative, symmetric, satisfies the triangle inequality, and is null if and only if the networks are isomorphic [21]. For future reference, the notions of metric and pseudometric are formally stated next.

Definition 3 *Given a space $\mathcal{S}$ and an isomorphism $\cong$, a function $d : \mathcal{S} \times \mathcal{S} \to \mathbb{R}$ is a metric in $\mathcal{S} \mod \cong$ if for any $a, b, c \in \mathcal{S}$ the function d satisfies:*

(i) **Nonnegativity.** $d(a, b) \geq 0$.

(ii) **Symmetry.** $d(a, b) = d(b, a)$.

(iii) **Identity.** $d(a, b) = 0$ *if and only if* $a \cong b$.

(iv) **Triangle inequality.** $d(a, b) \leq d(a, c) + d(c, b)$.

The function is a pseudometric in $\mathcal{S} \mod \cong$ if for any $a, b, c \in \mathcal{S}$ the function d satisfies (i), (ii), (iv), and

(iii′) **Relaxed identity.** $d(a, b) = 0$ *if* $a \cong b$.

A metric d in $\mathcal{S} \mod \cong$ gives a proper notion of distance. Since zero distances imply elements being isomorphic, the distance between elements reflects how far they are from being isomorphic. Pseudometrics are relaxed since elements not isomorphic may still have zero distance measured by the pseudometrics. The distance in Definition 2 is a metric in space $\mathcal{N} \mod \cong$. Observe that since correspondences may be between networks with different number of elements, Definition 2 defines a distance $d^1_{\mathcal{N}}(N_X, N_Y)$ when the node cardinalities $|X|$ and $|Y|$ are different. In the particular case when the functions r^1_X satisfy the triangle inequality, the set of networks $\mathcal{N}$ reduces to the set of metric spaces $\mathcal{M}$. In this case the metric in Definition 2 reduces to the Gromov-Hausdorff (GH) distance between metric spaces. The distances $d^1_{\mathcal{N}}(N_X, N_Y)$ in (2.3) are valid metrics even if the triangle inequalities are violated by r^1_X or r^1_Y [21].

In this chapter we consider high order networks where the specification of functions $r^k_X : X^{k+1} \to \mathbb{R}_+$ are meant to encode similarities or dissimilarities between node $(k+1)$-tuples. The goal of this chapter is to devise generalizations of Definition 2 to high order networks and to prove that they define valid metrics in the space of high order networks modulo isomorphism; see Definitions 11, 12, 14, and 15.

2.2. High Order Networks

A network of order K over the node space X is defined as a collection of $K+1$ relationship functions $\{r^k_X : X^{k+1} \to \mathbb{R}_+\}_{k=0}^{K}$ from the space X^{k+1} of $(k+1)$-tuples to the nonnegative reals,

$$N^K_X = \left(X, r^0_X, r^1_X, \ldots, r^K_X\right). \tag{2.4}$$

A network of order K can be considered as a weighted complete hypergraph [132, 133] whose weights for all hyperedges of elements of all $(k+1)$ tuples with $0 \leq k \leq K$ are defined.

When some nodes are repeated in the point collection $x_{0:k} := (x_0, x_1, \ldots, x_k) \in X^{k+1}$, the relationship function $r_X^k(x_{0:k})$ entails the same information as the relationship function between the largest non-repeating subtuple of $x_{0:k}$. In future definitions, it would be important to take the number of distinct elements of a tuple into consideration. We formalize this property by introducing the notion of the rank of tuples as we formally specify next.

Definition 4 *The rank $s(x_{0:k})$ of a given tuple $x_{0:k}$ is the number of unique elements in the tuple.*

It follows from Definition 4 that the rank $s(x, x) = 1$ and that the rank $s(x', x, x') = 2$. Moreover, the relationship function between a tuple $x_{0:k}$ is identical to the relationship functions of subtuples of $x_{0:k}$ that have same rank as $s(x_{0:k})$ since they imply same information. This remark along with a symmetry property makes up the formal definition of high order networks that we introduce next.

Definition 5 *$N_X^K = (X, r_X^0, r_X^1, \ldots, r_X^K)$ is a K-order network if the following two properties holds:*

Symmetry. *For any $0 \leq k \leq K$ and any point collections $x_{0:k}$, we have that*

$$r_X^k(x_{[0:k]}) = r_X^k(x_{0:k}), \tag{2.5}$$

where $x_{[0:k]} = ([x_0], [x_1], \ldots, [x_k])$ is a reordering of $x_{0:k} := (x_0, x_1, \ldots, x_k)$.

Identity. *For any $0 \leq k \leq K$ and tuple $x_{0:k}$, any of its subtuple $x_{l_0:l_{\tilde{k}}}$ with $s(x_{0:k}) = s(x_{l_0:l_{\tilde{k}}})$ satisfies*

$$r_X^k(x_{0:k}) = r_X^{\tilde{k}}(x_{l_0:l_{\tilde{k}}}). \tag{2.6}$$

The set of all high order networks of order K is denoted as $\mathcal{N}^K$.

For point collections $x_{0:k}$, values of their k-order relationship functions are denoted as $r_X^k(x_{0:k})$ and are intended to represent a measure of similarity or dissimilarity for members of the group. In particular, the zeroth order function r_X^0 encodes relative weights of different nodes and the first order function r_X^1 represents the pairwise information discussed in Section 2.1. Observe however that pairwise networks are not particular cases of networks of order 1 because a network of order K not only requires the definition of relationships between $(K+1)$-tuples but also of relationships between $(k+1)$-tuples for all integers $0 \leq k \leq K$. A network of order 0 is one in which only node weights are given, a network of order 1 is one in which weights and pairwise relationships are defined, a network of order 2 adds relationships between triplets and so on. Examples for the identity property includes $r_X^2(x, x) = r_X^1(x)$ and $r_X^3(x', x, x') = r_X^2(x, x')$. We assume that relationship values are normalized so that $0 \leq r_X^k(x_{0:k}) \leq 1$ for all k and $x_{0:k}$. As in the case of pairwise networks we consider K-order networks N_X^K and N_Y^K to be equivalent for their k-order relationship functions if r_X^k is a permutation of r_Y^k as we formally define next.

Definition 6 *We say that two networks N_X^K and N_Y^K are k-isomorphic if there exists a bijection $\pi : X \to Y$ such that for all $x_{0:k} \in X^{k+1}$ we have*

$$r_Y^k(\pi(x_{0:k})) = r_X^k(x_{0:k}), \tag{2.7}$$

where we use the shorthand notation $r_Y^k(\pi(x_{0:k})) := r_Y^k(\pi(x_0), \pi(x_1), \dots, \pi(x_k))$. The map π is called a k-isometry.

When networks N_X^K and N_Y^K are k-isomorphic we write $N_X^K \cong_k N_Y^K$. The space of K-order networks modulo k-isomorphism is denoted by $\mathcal{N}^K \mod \cong_k$. For each nonnegative integer $0 \leq k \leq K$, the space $\mathcal{N}^K \mod \cong_k$ of networks of order K modulo k-isomorphism can be endowed with a pseudometric. The definition of this family of pseudometrics is a generalization of Definition 2 as we formally state next.

Definition 7 *Given networks N_X^K and N_Y^K, a correspondence C between the node spaces X and Y, and an integer $0 \leq k \leq K$ define the k-order network difference with respect to C as*

$$\Gamma_{X,Y}^k(C) := \max_{(x_{0:k}, y_{0:k}) \in C} \left| r_X^k(x_{0:k}) - r_Y^k(y_{0:k}) \right|, \tag{2.8}$$

where the notation $(x_{0:k}, y_{0:k})$ stands for $(x_0, y_0), (x_1, y_1), \dots, (x_k, y_k)$. The k-order network distance between networks N_X^K and N_Y^K is then defined as

$$d_{\mathcal{N}}^k(N_X^K, N_Y^K) := \min_{C \in \mathcal{C}(X,Y)} \left\{ \Gamma_{X,Y}^k(C) \right\}. \tag{2.9}$$

We further define the K-order distance vector as the $K+1$ dimensional vector $\mathbf{d}_{\mathcal{N}}^K(N_X^K, N_Y^K) = \left[d_{\mathcal{N}}^0(N_X^K, N_Y^K), \dots, d_{\mathcal{N}}^K(N_X^K, N_Y^K)\right]^T$ that groups the k-order distances in (2.9).

Both, Definition 2 and Definition 7 consider correspondences C that map the node space X onto the node space Y, compare dissimilarities, and set the network distance to the comparison that yields the smallest value in terms of maximum differences. The distinction between them is that in (2.2) we compare the values in $r_X^1(x_1, x_2)$ and $r_Y^1(y_1, y_2)$, whereas in (2.8) we compare the values in each of the k-order relationships $r_X^k(x_{0:k})$ and $r_Y^k(y_{0:k})$ to compute the k-order distances $d_{\mathcal{N}}^k(N_X^K, N_Y^K)$ that we group in the vector $\mathbf{d}_{\mathcal{N}}^K(N_X^K, N_Y^K)$. Except for this distinction, Definition 2 and Definition 7 are analogous since $\Gamma_{X,Y}^k(C)$ selects the maximum k-order relationship difference $|r_X^k(x_{0:k}) - r_Y^k(y_{0:k})|$ among all tuples of correspondents – we compare $r_X^k(x_{0:k})$ with $r_Y^k(y_{0:k})$ when all the points $x_l \in x_{0:k}$ and $y_l \in y_{0:k}$ are correspondents. The distance $d_{\mathcal{N}}^k(N_X^K, N_Y^K)$ is defined by selecting the correspondence that minimizes these maximal differences.

Notice that, in general, the correspondence C minimizing $\Gamma_{X,Y}^k(C)$ is not necessarily identical to the correspondence C' minimizing $\Gamma_{X,Y}^l(C')$ for $k \neq l$. The distance vector $\mathbf{d}_{\mathcal{N}}^K$ is a vector with each element measuring the dissimilarity between relationship functions of a specific order, possibly using different minimizing correspondences. We emphasize that, as in the case of Definition 2,

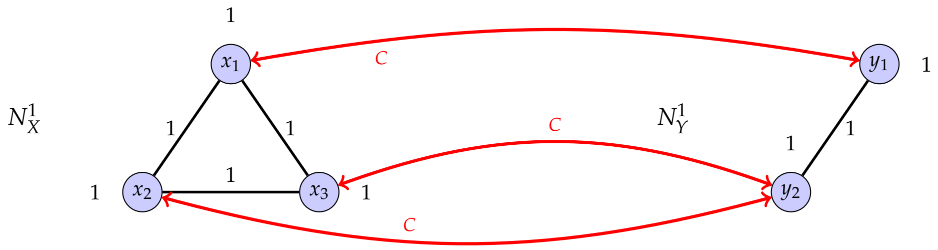


Figure 3: An example of two networks being not 1-isomorphic but having zero 1-order network distance between them. For the given correspondence C, $r_X^2(x_1, x_2) = r_Y^2(y_1, y_2)$, $r_X^2(x_1, x_3) = r_Y^2(y_1, y_2)$. $r_X^2(x_2, x_3) = r_Y^2(y_2, y_2) = r_Y^1(y_2)$ where the second equality follows from the identity property. Moreover, $r_X^2(x_1, x_1) = r_Y^2(y_1, y_1)$, $r_X^2(x_2, x_2) = r_Y^2(y_2, y_2)$, $r_X^2(x_3, x_3) = r_Y^2(y_2, y_2)$. $\Gamma_{X,Y}^1(C) = 0$ witnesses the zero 1-order network distance between N_X^1 and N_Y^1. However these networks cannot be 1-isomorphic since they possess different number of nodes.

$d_{\mathcal{N}}^k(N_X^K, N_Y^K)$ and $\mathbf{d}_{\mathcal{N}}^K(N_X^K, N_Y^K)$ are defined even if the numbers of nodes in X and Y are different. We show in the following proposition that the function $d_{\mathcal{N}}^k : \mathcal{N}^K \times \mathcal{N}^K \to \mathbb{R}_+$ is, indeed, a pseudometric in the space of K-order networks modulo k-isomorphism for any integer $0 \leq k \leq K$.

Proposition 1 *Given any nonnegative integer K, for any integers $0 \leq k \leq K$, the function $d_{\mathcal{N}}^k : \mathcal{N}^K \times \mathcal{N}^K \to \mathbb{R}_+$ defined in (2.9) is a pseudometric in the space $\mathcal{N}^K \mod \cong_k$.*

Proof: To prove that $d_{\mathcal{N}}^k$ for any integer $0 \leq k \leq K$ is a pseudometric in the space of K-order networks modulo k-isomorphism we prove the (i) nonnegativity, (ii) symmetry, (iii′) relaxed identity, and (iv) triangle inequality properties in Definition 3.

Proof of nonnegativity property: For any integers $0 \leq k \leq K$, since $|r_X^k(x_{0:k}) - r_Y^k(y_{0:k})|$ is nonnegative $\Gamma_{X,Y}^k(C)$ defined in (2.8) also is. The network distance must then satisfy $d_{\mathcal{N}}^k(N_X^K, N_Y^K) \geq 0$ because it is a minimum of nonnegative numbers. ■

Proof of symmetry property: A correspondence $C \subseteq X \times Y$ with elements $c_i = (x_i, y_i)$ results in the same associations as the correspondence $\tilde{C} \subseteq Y \times X$ with element $\tilde{c}_i = (y_i, x_i)$. Thus, for any correspondence C and integers $0 \leq k \leq K$, we have a correspondence $\tilde{C}$ such that $\Gamma_{X,Y}^k(C) = \Gamma_{Y,X}^k(\tilde{C})$. It follows that the minima in (2.9) must coincide from where it follows that $d_{\mathcal{N}}^k(N_X^K, N_Y^K) = d_{\mathcal{N}}^k(N_Y^K, N_X^K)$. ■

Proof of relaxed identity property: We need to show that for any integers $0 \leq k \leq K$ if N_X^K and N_Y^K are k-isomorphic we must have $d_{\mathcal{N}}^k(N_X^K, N_Y^K) = 0$. To see that this is true recall that for k-isomorphic networks there exists a bijection $\pi : X \to Y$ that preserves distance functions at order k [cf. (2.7)]. Consider then the particular correspondence $C_\pi = \{(x, \pi(x)), x \in X\}$. For all $x_0 \in X$ there is an element $c = (x_0, y) \in C_\pi$ and for all $y_0 \in Y$ there is an element $c' = (x, y_0) \in C_\pi$ since π is bijective. Thus C_π is a valid correspondence between X and Y for which (2.7) indicates that it must be

$$r_Y^k(y_{0:k}) = r_Y^k(\pi(x_{0:k})) = r_X^k(x_{0:k}), \tag{2.10}$$

for any $(x_{0:k}, y_{0:k}) \in C_\pi$. This implies $\Gamma_{X,Y}^k(C) = \left| r_X^k(x_{0:k}) - r_Y^k(y_{0:k}) \right| = 0$ for any $(x_{0:k}, y_{0:k}) \in C_\pi$. Since C_π is a particular correspondence, taking a minimum over all correspondences as in (2.9) yields

$$d_{\mathcal{N}}^k(N_X^K, N_Y^K) \leq \Gamma_{X,Y}^k(C) = 0. \tag{2.11}$$

Since $d_{\mathcal{N}}^k(N_X^K, N_Y^K) \geq 0$, as already shown, it must be that $d_{\mathcal{N}}^k(N_X^K, N_Y^K) = 0$ when N_X^K and N_Y^K are k-isomorphic. ■

Proof of triangle inequality: To show that the triangle inequality holds, let the correspondence C_1 between X and Z and the correspondence C_2 between Z and Y be the minimizing correspondences in (2.9). We can then write

$$d_{\mathcal{N}}^k(N_X^K, N_Z^K) = \Gamma_{X,Z}^k(C_1),\ d_{\mathcal{N}}^k(N_Z^K, N_Y^K) = \Gamma_{Z,Y}^k(C_2). \tag{2.12}$$

Define a correspondence C between X and Y as the one induced by pairs (x, z) and (z, y) sharing a common node $z \in Z$,

$$C := \left\{ (x, y) \mid \exists z \in Z \text{ with } (x, z) \in C_1, (z, y) \in C_2 \right\}. \tag{2.13}$$

To show that C is a well defined correspondence we need to show that for every $x \in X$ there exists $y_0 \in Y$ such that $(x, y_0) \in C$ and by symmetry for every $y \in Y$ there exists $x_0 \in Y$ such that $(x_0, y) \in C$. To see this, first pick an arbitrary $x \in X$. Because C_1 is a correspondence between X and Z there must exist $z_0 \in Z$ such that $(x, z_0) \in C_1$. There must exist $y_0 \in Y$ such that $(z_0, y_0) \in C_2$ since C_2 is also a correspondence between Y and Z. Therefore, there exists a pair $(x, y_0) \in T$ with $y_0 \in Y$ for any $x \in X$. The second part follows by symmetry and C is a well defined correspondence. The correspondence C may not be the minimizing correspondence for the distance $d_{\mathcal{N}}^k(N_X^K, N_Y^K)$. However since it is a valid correspondence with the definition in (2.9) we can write

$$d_{\mathcal{N}}^k(N_X^K, N_Y^K) \leq \Gamma_{X,Y}^k(C). \tag{2.14}$$

By the definition of C in (2.13), the requirement $(x_{0:k}, y_{0:k}) \in C$ is equivalent as $(x_{0:k}, z_{0:k}) \in C_1$ and $(z_{0:k}, y_{0:k}) \in C_2$ for any $0 \leq k \leq K$. Further adding and subtracting $r_Z^k(z_{0:k})$ in the absolute value of $\Gamma_{X,Y}^k(C) = \left| r_X^k(x_{0:k}) - r_Y^k(y_{0:k}) \right|$ and using the triangle inequality of the absolute value yields

$$\Gamma_{X,Y}^k(C) \leq \max_{\substack{(x_{0:k}, z_{0:k}) \in C_1 \\ (z_{0:k}, y_{0:k}) \in C_2}} \left\{ \left| r_X^k(x_{0:k}) - r_Z^k(z_{0:k}) \right| + \left| r_Z^k(z_{0:k}) - r_Y^k(y_{0:k}) \right| \right\}. \tag{2.15}$$

We can further bound (2.15) by taking maximum over each summand,

$$\Gamma^k_{X,Y}(C) \leq \max_{(x_{0:k},z_{0:k})\in C_1} \left|r^k_X(x_{0:k}) - r^k_Z(z_{0:k})\right| + \max_{(z_{0:k},y_{0:k})\in C_2} \left|r^k_Z(z_{0:k}) - r^k_Y(y_{0:k})\right| = \Gamma^k_{X,Z}(C_1) + \Gamma^k_{Z,Y}(C_2). \tag{2.16}$$

Substituting (2.14) and (2.12) into (2.16) yields triangle inequality. ■

Having proofs all statements, the global proof completes. ■

$d^k_{\mathcal{N}}$ being a pseudometric implies that two high order networks not k-isomorphic may still have zero k-order network distance between them. A specific example can be found in Figure 3 where two 1-order networks not 1-isomorphic have zero dissimilarity measured by the 1-order network distance. For each integer $0 \leq k \leq K$, the pseudometric $d^k_{\mathcal{N}}(N^K_X, N^K_Y)$ defined in Definition 7 in the space $\mathcal{N}^K \mod \cong_k$ measures dissimilarity between k-order functions r^k_X and r^k_Y. We can also ask the question of how different two networks are by considering *all* their order functions. To that end we consider K-order networks to be equivalent if r^k_X is a permutation of r^k_X for all integers $0 \leq k \leq K$ as we formally state next.

Definition 8 *We say that two networks of order K, N^K_X and N^K_Y, are isomorphic if there exists a bijection $\pi : X \to Y$ such that* (2.7) *holds for all* $0 \leq k \leq K$ *and* $x_{0:k} \in X^{k+1}$. *The map π is called an isometry.*

When networks N^K_X and N^K_Y are isomorphic we write $N^K_X \cong N^K_Y$. The difference between k-isomorphism and isomorphism is that the bijection in the latter case preserves relationship functions over all orders whereas only k-order relationship functions are preserved in the former case. That $N^K_X \cong N^K_Y$ implies that $N^K_X \cong_k N^K_Y$ for all integers $0 \leq k \leq K$, but the opposite is not necessarily true.

The space of K-order networks modulo isomorphism is denoted as $\mathcal{N}^K \mod \cong$. A family of pseudometrics measuring the difference between networks over all order functions as a whole can be endowed in the space $\mathcal{N}^K \mod \cong$. The definition of this family of distances can be considered as an extension of Definition 2 and an aggregation of Definition 7 as we formally state next.

Definition 9 *Given networks N^K_X and N^K_Y, a correspondence C between the node spaces X and Y, and some p-norm $\|\cdot\|_p$, define the network difference with respect to C as*

$$\left\|\mathbf{\Gamma}^K_{X,Y}(C)\right\|_p := \left\|\left(\Gamma^0_{X,Y}(C), \Gamma^1_{X,Y}(C), \ldots, \Gamma^K_{X,Y}(C)\right)^T\right\|_p, \tag{2.17}$$

where for each integer $0 \leq k \leq K$, *$\Gamma^k_{X,Y}(C)$ is the k-order network difference with respect to C defined in* (2.8). *The p-norm network distance between N^K_X and N^K_Y is then defined as*

$$d_{\mathcal{N},p}(N^K_X, N^K_Y) := \min_{C\in\mathcal{C}(X,Y)} \left\{\left\|\mathbf{\Gamma}^K_{X,Y}(C)\right\|_p\right\}. \tag{2.18}$$

The difference between Definition 2, Definition 7 and Definition 9 is that in the case of the network distance $d_{\mathcal{N},p}(N_X^K, N_Y^K)$, we compare not only relationship functions $r_X^k(x_{0:k})$ and $r_Y^k(y_{0:k})$ but also all the relationship functions of order not larger than K. The norm over the vector $\mathbf{\Gamma}_{X,Y}^K(C)$ formed by k-order network differences with respect to C for all integers $0 \leq k \leq K$ is assigned as the difference between N_X^K and N_Y^K measured by the correspondence C. The distance $d_{\mathcal{N},p}(N_X^K, N_Y^K)$ is then defined as the minimum of these differences achieved by some correspondence. As in the cases of Definition 2 and Definition 7, $d_{\mathcal{N},p}(N_X^K, N_Y^K)$ is defined even if the numbers of nodes in X and Y are different. The function $d_{\mathcal{N},p} : \mathcal{N}^K \times \mathcal{N}^K \to \mathbb{R}_+$ is a pseudometric in the space of K-order networks modulo isomorphism as we show in the following proposition.

Proposition 2 *Given some p-norm* $\| \cdot \|_p$, *for any nonnegative integer K the function* $d_{\mathcal{N},p} : \mathcal{N}^K \times \mathcal{N}^K \to \mathbb{R}_+$ *defined in (2.18) is a pseudometric in the space* $\mathcal{N}^K \mod \cong$.

Proof: To prove that $d_{\mathcal{N},p}$ is a distance in the space of K-order networks modulo isomorphism we prove the (i) nonnegativity, (ii) symmetry, (iii') relaxed identity, and (iv) triangle inequality properties in Definition 3.

Proof of nonnegativity property: Since $\|\mathbf{\Gamma}_{X,Y}^K(C)\|_p \geq 0$, the network distance must then satisfy $d_{\mathcal{N},p}(N_X^K, N_Y^K) \geq 0$ as it is a minimum of nonnegative numbers. ■

Proof of symmetry property: A correspondence $C \subseteq X \times Y$ with elements $c_i = (x_i, y_i)$ results in the same associations as the correspondence $\tilde{C} \subseteq Y \times X$ with element $\tilde{c}_i = (y_i, x_i)$. Thus, for any correspondence C we have a correspondence $\tilde{C}$ such that $\mathbf{\Gamma}_{X,Y}^K(C) = \mathbf{\Gamma}_{Y,X}^K(\tilde{C})$. This implies $\|\mathbf{\Gamma}_{X,Y}^K(C)\|_p = \|\mathbf{\Gamma}_{Y,X}^K(\tilde{C})\|_p$. It follows that the minima in (2.18) must coincide and therefore $d_{\mathcal{N},p}(N_X^K, N_Y^K) = d_{\mathcal{N},p}(N_Y^K, N_X^K)$. ■

Proof of relaxed identity property: We need to show that if N_X^K and N_Y^K are isomorphic we must have $d_{\mathcal{N},p}(N_X^K, N_Y^K) = 0$. To see that this is true recall that for isomorphic networks there exists a bijection $\pi : X \to Y$ that preserves distance functions at every order [cf. (2.7)]. Consider then the particular correspondence $C_\pi = \{(x, \pi(x)), x \in X\}$. We have demonstrated in the proof of Proposition 1 that C_π is a valid correspondence between X and Y. The definition of isomorphism indicates that it must be (2.10) holds true for all $0 \leq k \leq K$ and $(x_{0:k}, y_{0:k}) \in C_\pi$. Since C_π is a particular correspondence, from (2.18) it follows that

$$d_{\mathcal{N},p}(N_X^K, N_Y^K) \leq \left\| \mathbf{\Gamma}_{X,Y}^K(C) \right\|_p. \tag{2.19}$$

Because $r_X^k(x_{0:k}) - r_Y^k(y_{0:k}) = 0$ for any $0 \leq k \leq K$ and any $(x_{0:k}, y_{0:k}) \in C_\pi$ by (2.10), we have $\mathbf{\Gamma}_{X,Y}^K(C) = \mathbf{0}$. $\| \cdot \|_p$ being a proper norm implies $\|\mathbf{\Gamma}_{X,Y}^K(C)\|_p = 0$. Substituting this back into (2.19) shows $d_{\mathcal{N},p}(N_X^K, N_Y^K) \leq 0$. Since $d_{\mathcal{N},p}(N_X^K, N_Y^K) \geq 0$, as already shown, it must be that $d_{\mathcal{N},p}(N_X^K, N_Y^K) = 0$ when N_X^K and N_Y^K are isomorphic. ■

Proof of triangle inequality: To show that the triangle inequality holds, let the correspondence C_1

between X and Z and the correspondence C_2 between Z and Y be the minimizing correspondences in (2.18). We can then write

$$d_{\mathcal{N},p}(N_X^K, N_Z^K) = \left\|\mathbf{\Gamma}_{X,Z}^K(C_1)\right\|_p, \quad d_{\mathcal{N},p}(N_Z^K, N_Y^K) \quad = \left\|\mathbf{\Gamma}_{Z,Y}^K(C_2)\right\|_p. \tag{2.20}$$

Define a correspondence C between X and Y in the same way as (2.13). We have demonstrated in the proof of Proposition 1 that C is a well defined correspondence. Therefore with the definition in (2.18) we can write

$$d_{\mathcal{N},p}(N_X^K, N_Y^K) \leq \left\|\mathbf{\Gamma}_{X,Y}^K(C)\right\|_p. \tag{2.21}$$

Moreover, in the proof of Proposition 1 we also showed for any $0 \leq k \leq K$,

$$\Gamma_{X,Y}^k(C) \leq \Gamma_{X,Z}^k(C_1) + \Gamma_{Z,Y}^k(C_2). \tag{2.22}$$

This implies the vector $\mathbf{\Gamma}_{X,Z}^K(C_1) + \mathbf{\Gamma}_{Z,Y}^K(C_2)$ is elementwise no smaller than the vector $\mathbf{\Gamma}_{X,Y}^K(C)$. The definition of p-norm $\|\mathbf{x}\|_p = \left(\sum_{k=0}^K |x_i|^p\right)^{1/p}$ guarantees that the value of $\|\mathbf{x}\|_p$ is monotonically nondecreasing on each element x_i in $\mathbf{x} = (x_0, x_1, \ldots, x_n)^T$. Therefore,

$$\left\|\mathbf{\Gamma}_{X,Y}^k(C)\right\|_p \leq \left\|\mathbf{\Gamma}_{X,Z}^k(C_1) + \mathbf{\Gamma}_{Z,Y}^k(C_2)\right\|_p. \tag{2.23}$$

We can further bound (2.23) by using the triangle inequality of the p-norm,

$$\left\|\mathbf{\Gamma}_{X,Y}^k(C)\right\|_p \leq \left\|\mathbf{\Gamma}_{X,Z}^k(C_1)\right\|_p + \left\|\mathbf{\Gamma}_{Z,Y}^k(C_2)\right\|_p. \tag{2.24}$$

Substituting (2.21) and (2.20) back into (2.24) yields the triangle inequality. ■

Having demonstrated all statements, the global proof completes. ■

Observe that in (2.18) we are only allowed to pick one correspondence minimizing $\|\mathbf{\Gamma}_{X,Y}^K(C)\|_p$ whereas in (2.9) for each k we are able to pick one correspondence minimizing the order specific $\Gamma_{X,Y}^k(C)$. This establishes a relationship between $d_{\mathcal{N},p}$ and $\|\mathbf{d}_{\mathcal{N}}^K\|_p$ that we show next.

Proposition 3 *Given some p-norm* $\|\cdot\|_p$, *for any nonnegative integer K the function* $d_{\mathcal{N},p}$ *defined in* (2.18) *is no smaller than* $\|\mathbf{d}_{\mathcal{N}}^K\|_p$ *where* $\mathbf{d}_{\mathcal{N}}^K$ *is the vector of distances defined in Definition 7. I.e., for any pair of K-order networks* N_X^K, N_Y^K, *we have that*

$$d_{\mathcal{N},p}(N_X^K, N_Y^K) \geq \left\|\mathbf{d}_{\mathcal{N}}^K(N_X^K, N_Y^K)\right\|_p. \tag{2.25}$$

Proof: Given K-order networks N_X^K, N_Y^K, a correspondence C between the node spaces X and Y, and an integer $0 \leq k \leq K$, it follows from (2.9) that

$$\Gamma^{k}_{X,Y}(C) \geq d^{k}_{\mathcal{N}}(N^{K}_{X}, N^{K}_{Y}). \tag{2.26}$$

This implies that the vector $\mathbf{d}^{K}_{\mathcal{N}}(N^{K}_{X}, N^{K}_{Y})$ is element-wise no greater than $\mathbf{\Gamma}^{K}_{X,Y}(C)$ from where it follows that

$$\left\|\mathbf{\Gamma}^{K}_{X,Y}(C)\right\|_{p} \geq \left\|\mathbf{d}^{K}_{\mathcal{N}}(N^{K}_{X}, N^{K}_{Y})\right\|_{p}. \tag{2.27}$$

Since (2.27) applies for any correspondence C, the minimum of $\left\|\mathbf{\Gamma}^{K}_{X,Y}(C)\right\|_{p}$ achieved by some correspondence in the set of correspondence $\mathcal{C}(X,Y)$ is still no smaller than $\left\|\mathbf{d}^{K}_{\mathcal{N}}(N^{K}_{X}, N^{K}_{Y})\right\|_{p}$,

$$\min_{C \in \mathcal{C}(X,Y)} \left\{\left\|\mathbf{\Gamma}^{K}_{X,Y}(C)\right\|_{p}\right\} \geq \left\|\mathbf{d}^{K}_{\mathcal{N}}(N^{K}_{X}, N^{K}_{Y})\right\|_{p}. \tag{2.28}$$

The result in (2.25) follows after noting that the minimum in the left hand side of (2.28) is the distance $d_{\mathcal{N},p}(N^{K}_{X}, N^{K}_{Y})$ in (2.18). ■

Definitions 7 and 9 are *pseudometrics* in the space of high order networks modulo appropriate isomorphisms. To obtain proper *metrics*, we restrict attention to subclasses of networks having specific structures. To do so, observe that the k-order function r^{k}_{X} of a given network N^{K}_{X} does not impose constraints on the l-order function r^{l}_{X} of the same network except the identity property. In practical situations, however, it is common to observe that adding nodes to a tuple results in either increasing or decreasing relationships between elements of the extended tuple. This motivates the consideration of dissimilarity networks and proximity networks that we undertake in the next two sections.

2.3. Dissimilarity Networks

In dissimilarity networks the function $r^{k}_{X}(x_{0:k})$ encodes a level of dissimilarity between elements of the $x_{0:k}$ tuple. In this scenario it is reasonable to assume that adding elements to a tuple makes the group more dissimilar. This restriction along with a generalization of the requirement that $r^{1}_{X}(x, x') = 0$ if and only if $x = x'$ in pairwise network makes up the formal definition that we introducre next.

Definition 10 *We say that the K-order network $D^{K}_{X} = \left(X, r^{0}_{X}, r^{1}_{X}, \ldots, r^{K}_{X}\right)$ is a dissimilarity network if for any order $0 \leq k \leq K$ and tuples $x_{0:k} \in X^{k+1}$, its relationship function is the summation of a dissimilarity function and the multiplication of its rank with a small constant ϵ,*

$$r^{k}_{X}(x_{0:k}) = d^{k}_{X}(x_{0:k}) + \epsilon s(x_{0:k}) \tag{2.29}$$

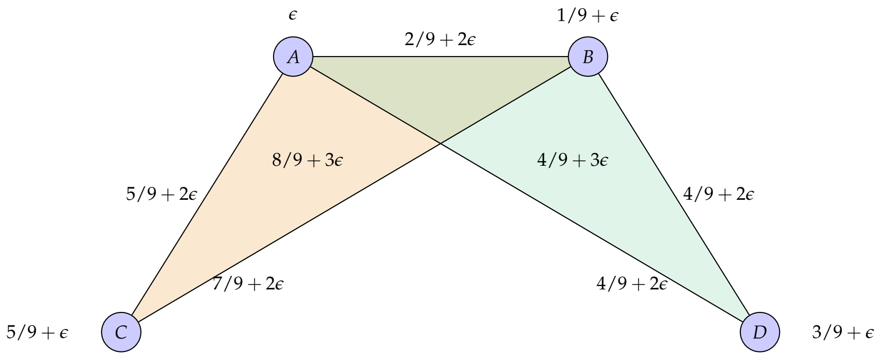


Figure 4: Temporal dynamics for the formation of a research community. The k-order relationship function in this 2-order dissimilarity network [cf. Definition 10] incorporates the dissimilarity function – the normalized time instant at which members of a given $(k+1)$-tuple write their first joint publication – and the multiplication of ϵ with the rank of the tuple. E.g., A writes her first publication at time 0, and coauthors with B, D, and C at times 2/9, 4/9, and 5/9. She also writes jointly with B and D at time 4/9.

The dissimilarity terms satisfy the order increasing property so that for any $1 \leq k \leq K$ *and* $x_{0:k}$,

$$d_X^k(x_{0:k}) \geq d_X^{k-1}(x_{0:k-1}), \tag{2.30}$$

and the constant $\epsilon > 0$ *is a strictly positive value that satisfies*

$$0 < \epsilon \leq 1 - \frac{1}{K} \max_{\tilde{x}_{0:K} \in X^{K+1}} d_X^K(\tilde{x}_{0:K}). \tag{2.31}$$

The set of all dissimilarity networks of order K *is denoted as* $\mathcal{D}^K$.

To see that the order increasing property (2.30) in Definition 10 is reasonable consider a network describing the temporal dynamics of the formation of a research community – see Figure 4. The dissimilarity term in the k-order relationship function in this network marks the normalized time instant at which members of a given $(k+1)$-tuple write their first joint publication. In particular, the zeroth order dissimilarities d_X^0 are the normalized time instants when authors publish their first publication. In Figure 4 authors A, B, C, and D publish their first publications at times 0, 1/9, 5/9, and 3/9. The first order dissimilarities d_X^1 between pairs denote the normalized times at which nodes become coauthors. Since authors can't become coauthors until after they write their first publication it is certain that $d_X^1(x,x') \geq d_X^0(x)$ and $d_X^1(x,x') \geq d_X^0(x')$ for all x and x'. In Figure 4, A and B become coauthors at time 2/9, which occurs after they publish their respective first publications at times 0 and 1/9. Authors A and D as well as B and D become coauthors at time 4/9, A and C become coauthors at time 5/9. Authors C and D never write a publication together.

Second order dissimilarities d_X^2 for triplets denote the normalized time at which a publication is coauthored by the three members of the triplet. Since a publication can't be coauthored by three people without being at the same time coauthored by each of the three possible pairs of authors we must have that $d_X^2(x,x',x'') \geq d_X^1(x,x')$, $d_X^2(x,x',x'') \geq d_X^1(x,x'')$, and $d_X^2(x,x',x'') \geq d_X^1(x',x'')$ for all x, x', and x''. In Figure 4, authors A, B, and D publish a joint publication at time 4/9, which is no smaller than the pairwise coauthorship times between each two of the individual authors. Authors A, B, and C publish a joint publication at time 8/9, which is a time that comes after the individual paired publications that occur at times 2/9, 5/9, and 7/9. Note that due to symmetry property a relationship as in (2.30) holds if we remove an arbitrary node from the tuple $x_{0:k}$, not necessarily the last.

In pairwise dissimilarity networks we required $d_X^1(x,x') = 0$ if and only if $x = x'$. Relationships between two different nodes are *strictly greater* than relationships between two nodes that are actually identical. The multiplication of ϵ and the rank of the tuples in (2.29) in Definition 10 can be considered as a generalization. Consider tuples $x_{0:k}$ and $(x_{0:k-1}, x_0)$ where every node in $x_{0:k}$ is unique, the identity property for high order networks forces $r_X^k(x_{0:k-1}, x_0) = r_X^{k-1}(x_{0:k-1})$. We must then have the relationship between $k+1$ different elements $r_X^k(x_{0:k})$ being strictly greater than the relationship between k different elements $r_X^k(x_{0:k-1}, x_0) = r_X^{k-1}(x_{0:k-1})$. This is because $d_X^k(x_{0:k}) \geq d_X^{k-1}(x_{0:k-1})$ follows from (2.30) and $\epsilon s(x_{0:k}) = (k+1)\epsilon > k\epsilon = \epsilon s(x_{0:k-1})$ follows from the definition of ranks. Therefore, the multiplication of ϵ and the rank of tuples in (2.29) in Definition 10 forces that adding a new element to a tuple makes the set *strictly more dissimilar* than it was. Or equivalently, removing an element from a tuple makes the set strictly less dissimilar than it was. The requirement for ϵ as in (2.31) ensures that the highest relationship in the network $\max_{\tilde{x}_{0:K} \in X^{K+1}} d_X^K(\tilde{x}_{0:K}) + \epsilon s(\tilde{x}_{0:K})$ is bounded above by 1. The rank correction term $\epsilon s(x_{0:k})$ is a technical modification to distinguish between full rank (proper) k-tuples and rank deficient (degenerate) tuples. In practice it can be set to a sufficiently small value compared to dissimilarities or completely ignored. Since distances up to order 2 are defined and relationship functions can be decomposed, the network in Figure 4 is a dissimilarity network of order 2.

2.3.1. Metrics in The Space of Dissimilarity Networks

When the input networks in Definition 7 are dissimilarity networks we refer to the k-order distance as the k-order dissimilarity network distance. We state this formally in the following definition for future reference.

Definition 11 *Given dissimilarity networks* $D_X^K, D_Y^K \in \mathcal{D}^K$ *we say that the k-order distance* $d_{\mathcal{N}}^k(D_X^K, D_Y^K) = d_{\mathcal{D}}^k(D_X^K, D_Y^K)$ *of Definition 7 is the k-order dissimilarity network distance between* D_X^K *and* D_Y^K*.*

Since $\mathcal{D}^K \subseteq \mathcal{N}^K$, the function $d_{\mathcal{D}}^k : \mathcal{D}^K \times \mathcal{D}^K \to \mathbb{R}_+$ is a pseudometric in the space of K-order dissimilarity networks modulo k-isomorphism. The restriction, however, makes $d_{\mathcal{D}}^k$ not only a pseudometric but a well-defined metric in the space $\mathcal{D}^K \mod \cong_k$ of dissimilarity networks of order K modulo k-isomorphism. We show this in the following theorem.

Theorem 1 *The k-order dissimilarity network distance function* $d_{\mathcal{D}}^k : \mathcal{D}^K \times \mathcal{D}^K \to \mathbb{R}_+$ *of Definition 11 is a metric in the space* $\mathcal{D}^K \mod \cong_k$ *for all* $1 \leq k \leq K$*.*

Proof: The proof of Proposition 1 has demonstrated $d^k_{\mathcal{D}}$ is a pseudometric in the space $\mathcal{D}^K$ mod $\cong_k$. To prove that $d^k_{\mathcal{D}}$ is a metric in the same space we need to show the missing part in the (iii) identity property in Definition 3.

Proof of the second part of the identity property: We want to prove $d^k_{\mathcal{D}}(D^K_X, D^K_Y) = 0$ must imply that D^K_X and D^K_Y are k-isomorphic. If $d^k_{\mathcal{D}}(D^K_X, D^K_Y) = 0$, there exists a correspondence C such that $r^k_X(x_{0:k}) = r^k_Y(y_{0:k})$ for any $(x_{0:k}, y_{0:k}) \in C$. Define a function $\pi : X \to Y$ that associates x with an arbitrary y chosen from the set that form a pair with x in C,

$$\pi : x \mapsto y_0 \in \{y \mid (x, y) \in C\}. \tag{2.32}$$

Since C is a correspondence the set $\{y \mid (x, y) \in C\}$ is nonempty for any x implying that π is well-defined for any $x \in X$. Therefore $r^k_X(x_{0:k}) = r^k_Y(\pi(x_{0:k}))$ for any $x_{0:k}$. This implies the function π must be injective. If it were not, there would be a pair of nodes $x \neq x'$ with $\pi(x) = \pi(x') = y$ for some $y \in Y$. Hence the k-order relationship function between $(x, \ldots, x, x')$ where the first $k-1$ nodes in the tuple are x and the last node is x' would satisfy

$$r^k_X(x \ldots, x, x') = r^k_Y(\pi(x, \ldots, x, x')) = r^k_Y(y, \ldots, y), \tag{2.33}$$

follows from the definition of π. The k-order relationship between the tuple $(x, \ldots, x)$ where all the k nodes are identical would also satisfy

$$r^k_X(x, \ldots, x) = r^k_Y(\pi(x, \ldots, x)) = r^k_Y(y, \ldots, y). \tag{2.34}$$

Combining (2.33) and (2.34) yields

$$r^k_X(x, \ldots, x, x') = r^k_X(x, \ldots, x). \tag{2.35}$$

Meanwhile, the identity property for high order networks [cf. Definition 5] implies

$$r^k_X(x, \ldots, x, x') = r^2_X(x, x'), \quad r^k_X(x, \ldots, x) = r^1_X(x). \tag{2.36}$$

Using the fact that for dissimilarity networks, relationship functions are the summations of dissimilarity functions and the multiplication of ϵ and ranks, we have that

$$r^2_X(x, x') = d^2_X(x, x') + 2\epsilon, \quad r^1_X(x) = d^1_X(x) + \epsilon. \tag{2.37}$$

Moreover, the order increasing property for dissimilarity functions implie

$$d^2_X(x, x') \geq d^1_X(x). \tag{2.38}$$

Substituting the decompositions (2.37) and (2.38) into (2.36) yields

$$r^k_X(x, \ldots, x, x') > r^k_X(x, \ldots, x). \tag{2.39}$$

which contradicts with (2.35) and shows that π must be injective.

Likewise, define the function $\omega : Y \to X$ that associates y with an arbitrary x chosen from the set that form a pair with y in C,

$$\omega : y \mapsto x_0 \in \{x | (x, y) \in C\}. \tag{2.40}$$

It follows by similar arguments that ω must be injective. By applying the Cantor-Bernstein-Schroeder theorem [134, Section 2.6] to the reciprocal injections $\pi : X \to Y$ and $\omega : Y \to X$, the existence of a bijection between X and Y is guaranteed. This forces X and Y to have same cardinality and π and ω being bijections. Pick the bijection π and it follows $r_X^k(x_{0:k}) = r_Y^k(\pi(x_{0:k}))$ for all nodes $(k+1)$-tuples $x_{0:k} \in X^{k+1}$. This shows that $D_X^K \cong_k D_Y^K$ and completes the proof of the identity statement. ■

Having demonstrated all four properties in Theorem 1, the global proof completes. ■

Observe that in Theorem 1 we have that $d_{\mathcal{D}}^k$ is a proper metric for all k other than 0. This caveat for $d_{\mathcal{D}}^0$ is because we may have two dissimilarity networks D_X^K and D_Y^K with different number of nodes but whose zeroth other relationships are equals for all pairs of nodes, i.e., $r_X^0(x) = r_Y^0(y)$ for all $x \in X$ and $y \in Y$. In this case we we would have $d_{\mathcal{D}}^0(D_X^K, D_Y^K) = 0$, however the two dissimilarity networks are not 0-isomorphic.

Restricting Definition 9 to dissimilarity networks also yields a family of dissimilarity network distances as next.

Definition 12 *Given dissimilarity networks* $D_X^K, D_Y^K \in \mathcal{D}^K$ *we say that the p-norm network distance* $d_{\mathcal{N},p}(D_X^K, D_Y^K) = d_{\mathcal{D},p}(D_X^K, D_Y^K)$ *of Definition 9 is the p-norm dissimilarity network distance between* D_X^K *and* D_Y^K*.*

By restricting our attention to dissimilarity networks instead of general high order networks, $d_{\mathcal{D},p}$ also becomes a valid metric in the space $\mathcal{D}^K \mod \cong$ of dissimilarity networks of order $K \geq 1$ modulo isomorphism as we state in the following theorem.

Theorem 2 *Given some p-norm* $\|\cdot\|_p$*, for any nonnegative integer* $K \geq 1$ *the function* $d_{\mathcal{D},p} : \mathcal{D}^K \times \mathcal{D}^K \to \mathbb{R}_+$ *in Definition 12 is a metric in the space* $\mathcal{D}^K \mod \cong$*.*

Proof: The proof of Proposition 2 has demonstrated that $d_{\mathcal{D},p}$ is a pseudometric in the space $\mathcal{D}^K \mod \cong$. To prove that $d_{\mathcal{D},p}$ is a metric in the same space we further demonstrate the missing part in the (iii) identity property in Definition 3.

Proof of the second part of the identity property: We want to show $d_{\mathcal{D},p}(D_X^K, D_Y^K) = 0$ implying D_X^K and D_Y^K being isomorphic. If $d_{\mathcal{D},p}(D_X^K, D_Y^K) = \min_{C \in \mathcal{C}(X,Y)} \|\mathbf{\Gamma}_{X,Y}^K(C)\|_p = 0$, there exists a correspondence C such that

$$\|\mathbf{\Gamma}^K_{X,Y}(C)\|_p = 0. \tag{2.41}$$

The property of p-norm implies that this correspondence C satisfies $\Gamma^k_{X,Y}(C) = 0$ for $0 \leq k \leq K$, i.e. $r^k_X(x_{0:k}) = r^k_Y(y_{0:k})$ for any $0 \leq k \leq K$ and $(x_{0:k}, y_{0:k}) \in C$. Define functions $\pi : X \to Y$ as in (2.32) and $\omega : Y \to X$ as in (2.40). The analysis in Proof of Theorem 1 has demonstrated that π and ω are bijections and that X and Y have same cardinality. Pick the bijection π and it follows $r^k_X(x_{0:k}) = r^k_Y(\pi(x_{0:k}))$ for any $0 \leq k \leq K$ and all $(k+1)$-tuples $x_{0:k} \in X$. This shows that $D^K_X \cong D^K_Y$ and completes the proof of the identity statement. ■ ■

Further note that since Proposition 3 holds for any pair of networks, the same relationship holds true for the dissimilarity network distances in Definitions 11 and 12. Observe, however, that the norm $\left\|\mathbf{d}^K_{\mathcal{D}}(D^K_X, D^K_Y)\right\|_p$ is not a valid metric because we can have instances in which two dissimilarity networks are k-isomorphic for all integers $0 \leq k \leq K$ without being isomorphic.

2.4. Proximity Networks

In proximity networks the relationship functions $r^k_X(x_{0:k})$ denote similarity or proximity between elements of a tuple. Thus, large values of the proximity function $r^k_X(x_{0:k})$ represent strong relationship whereas small values denote weak relationships – the exact opposite is true of dissimilarity networks. In this framework it is reasonable to assume that adding elements to a tuple forces the group to be less similar. This constraint makes up the formal definition we introduce as follows.

Definition 13 *We say that the K-order network $P^K_X = \left(X, r^0_X, r^1_X, \ldots, r^K_X\right)$ is a proximity network if for any order $0 \leq k \leq K$ and tuples $x_{0:k} \in X^{k+1}$, its relationship function is the summation of a proximity term and the multiplication of its rank with $-\epsilon$,*

$$r^k_X(x_{0:k}) = d^k_X(x_{0:k}) - \epsilon s(x_{0:k}), \tag{2.42}$$

The proximity terms satisfy the order increasing property that for any $1 \leq k \leq K$ and $x_{0:k}$,

$$p^k_X(x_{0:k}) \leq p^{k-1}_X(x_{0:k-1}), \tag{2.43}$$

and the constant $\epsilon > 0$ is a strictly positive value that satisfies

$$0 < \epsilon \leq \frac{1}{K} \min_{\tilde{x} \in X^{K+1}} p^K_X(\tilde{x}_{0:K}). \tag{2.44}$$

The set of all proximity networks of order K is denoted as $\mathcal{P}^K$.

To see that the order decreasing property (2.43) in Definition 13 is reasonable, consider a network illustrating the collaborations between authors in a research community – See Figure 5. The k-order proximity function in this network labels the number of publications between members of

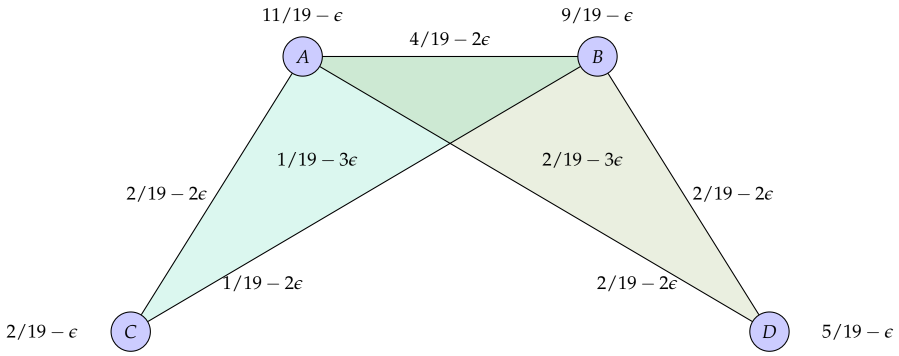


Figure 5: Collaborations between authors in a research community. The k-order relationship function in this 2-order network [cf. Definition 13] incorporates the proximity function – the number of publications between members of a given $(k+1)$-tuples normalized by the total number of publications – and the multiplication of $-\epsilon$ with the rank of the tuple.

a given $(k+1)$-tuple. In specific, the zeroth order proximities p_X^0 are the numbers of publications published by authors normalized by the total number of publications. In Figure 5 authors A, B, C, D publish $11, 9, 2, 5$ publications respectively and there are 19 publications in total which implies $p_X^0(A) = 11/19$, $p_X^0(B) = 9/19$, $p_X^0(C) = 2/19$, $p_X^0(D) = 5/19$. The first order proximities p_X^1 represent the number of publications co-published by nodes. Since collaboration for a pair of authors is also a publication for each of the individuals it is certain that $p_X^1(x, x') \leq p_X^0(x)$ and $p_X^1(x, x') \leq p_X^0(x')$ for all x and x'. In Figure 5, A and B collaborate on 4 publications, which is less than the 11 and 9 publications written by each of the individuals. Authors A and C as well as A and D coauthor 2 publications in total. Authors C and D never write a publication together.

Second order proximities p_X^2 for triplets indicate the normalized number of publications coauthored by the three members of the triplet. Since a publication with three authors is also a collaboration for the three pairs of authors we must have $p_X^2(x, x', x'') \leq p_X^1(x, x')$, $p_X^2(x, x', x'') \leq p_X^1(x, x'')$, and $p_X^2(x, x', x'') \leq p_X^1(x', x'')$ for all x, x', and x''. In Figure 5, authors A, B, and D cowrite 2 publications, which is no more than the number of pairwise collaborations between each pair of the authors. Remark that symmetry property inherited from high order networks [cf. Definition 5] implies (2.43) if we remove an arbitrary node from the tuple $x_{0:k}$, not necessarily the last.

In dissimilarity networks we required the relationship within tuple $x_{0:k}$ of unique elements to be strictly greater than the relationship between the point collection $(x_{0:k-1}, x_0)$ where some nodes are repeating. The multiplication of $-\epsilon$ and ranks in (2.42) in Definition 13 can also be considered as a generalization. Following the identity property of high order networks, $r_X^k(x_{0:k-1}, x_0) = r_X^{k-1}(x_{0:k-1})$. We must then have the function between $k+1$ different elements $r_X^k(x_{0:k})$ being strictly smaller than the function between k different elements $r_X^k(x_{0:k-1}, x_0) = r_X^{k-1}(x_{0:k-1})$. This is because in the decomposition $p_X^k(x_{0:k}) \leq p_X^{k-1}(x_{0:k-1})$ follows from (2.43) and $-\epsilon s(x_{0:k}) = -(k+1)\epsilon < -k\epsilon = -\epsilon s(x_{0:k-1})$ follows from the definition of ranks. Therefore, the multiplication of $-\epsilon$ and rank of tuples in (2.42) in Definition 10 forces that adding a new element to a tuple makes the set *strictly less similar* than it was. Or equivalently, removing an element from a tuple makes the set strictly more similar than it was. The requirement for ϵ as in (2.44) ensures that the lowest relationship function in the network $\min_{\tilde{x}_{0:K} \in X^{K+1}} d_X^K(\tilde{x}_{0:K}) - \epsilon s(\tilde{x}_{0:k})$ is nonnegative. Again the rank correction term $\epsilon s(x_{0:k})$ is a technical modification and in practice it can be set to sufficiently small compared to proximities or completely ignored. Since relationships up to order 2 are defined and can be decomposed, the network in Figure 5 is a proximity network of order 2.

2.4.1. Metrics in The Space of Proximity Networks

In the same way that restricting attention to dissimilarity networks transforms the pseudometrics in Definitions 7 and 9 into metrics, restricting attention to proximity networks also results in the definitions of proper metrics. We state the restrictions of Definitions 7 and 9 in the following two definitions.

Definition 14 *Given proximity networks* $P_X^K, P_Y^K \in \mathcal{P}^K$ *we say that the k-order distance* $d_{\mathcal{N}}^k(P_X^K, P_Y^K) = d_{\mathcal{P}}^k(P_X^K, P_Y^K)$ *of Definition 7 is the k-order proximity network distance between* P_X^K *and* P_Y^K.

Definition 15 *Given proximity networks* $P_X^K, P_Y^K \in \mathcal{P}^K$ *we say that the p-norm network distance* $d_{\mathcal{N},p}(P_X^K, P_Y^K) = d_{\mathcal{P},p}(P_X^K, P_Y^K)$ *of Definition 9 is the p-norm proximity network distance between* P_X^K *and* P_Y^K.

Analogously to the definition of the dissimilarity network distance $d_{\mathcal{D}}^k$ of Definition 11, the function $d_{\mathcal{P}}^k : \mathcal{P}^K \times \mathcal{P}^K \to \mathbb{R}_+$ is a proper metric in the space $\mathcal{P}^K \mod \cong_k$ of proximity networks of order K modulo k-isomorphism for all integers $1 \leq k \leq K$. Likewise, restricting the function $d_{\mathcal{N},p}$ of Definition 9 to proximity networks as Definition 15 results in $d_{\mathcal{P},p}$ being a proper metric. We state these facts in the following theorems.

Theorem 3 *The k-order proximity network distance function* $d_{\mathcal{P}}^k : \mathcal{P}^K \times \mathcal{P}^K \to \mathbb{R}_+$ *of Definition 14 is a metric in the space* $\mathcal{P}^K \mod \cong_k$ *for all* $k \geq 1$.

Proof: The proof of Proposition 1 has demonstrated that $d_{\mathcal{P}}^k$ is a pseudometric in the space $\mathcal{P}^K \mod \cong_k$. To prove that $d_{\mathcal{P}}^k$ is a metric in the same space we need to show the missing part in the (iii) identity property in Definition 3.

Proof of the second part of the identity property: Most parts of the proof follow from the proof of the second part of the identity property for Theorem 1 in the proof of Proposition 1. The only difference is in demonstrating the function π constructed in (2.32) is injective. Under the same setup where there exist a pair of nodes $x \neq x'$ such that $\pi(x) = \pi(x') = y$ for some $y \in Y$, the k-order relationship between $(x, \ldots, x, x')$ would satisfy

$$r_X^k(x \ldots, x, x') = r_Y^k(y, \ldots, y) = r_X^k(x \ldots, x). \tag{2.45}$$

Meanwhile, the facts of proximities in proximity networks follow order decreasing property

$p_X^2(x,x') \leq p_X^1(x)$ and $r_X^2(x,x') = p_X^2(x,x') - 2\epsilon, r_X^1(x) = p_X^1(x) - \epsilon$ from (2.42) implies

$$r_X^2(x,x') < r_X^1(x). \tag{2.46}$$

Combining (2.46) with the identity property inherited from high order networks [cf. Definition 5] $r_X^k(x,\ldots,x,x') = r_X^2(x,x'), r_X^k(x,\ldots,x) = r_X^1(x)$ gives us

$$r_X^k(x,\ldots,x,x') < r_X^k(x,\ldots,x), \tag{2.47}$$

which contradicts with (2.45) and shows that π must be injective. The rest of the proof follows. ■

■

Theorem 4 *Given some p-norm $\|\cdot\|_p$, for any nonnegative integer $K \geq 1$ the function $d_{\mathcal{P},p} : \mathcal{P}^K \times \mathcal{P}^K \to \mathbb{R}_+$ in Definition 15 is a metric in the space $\mathcal{P}^K \mod \cong$.*

Proof: The proof of Proposition 2 has demonstrated that $d_{\mathcal{P},p}$ is a pseudometric in the space $\mathcal{P}^K \mod \cong$. To prove that $d_{\mathcal{P},p}$ is a metric in the same space we further demonstrate the missing part in the (iii) identity property in Definition 3.

Proof of the second part of the identity property: We want to show that having $d_{\mathcal{P},p}(P_X^K, P_Y^K) = 0$ must imply that P_X^K being isomorphic to P_Y^K. If $d_{\mathcal{D},p}(P_X^K, P_Y^K) = 0$, there exists a correspondence C such that $\|\mathbf{\Gamma}_{X,Y}^K(C)\|_p = 0$. The property of p-norm implies that this correspondence C satisfies $r_X^k(x_{0:k}) = r_Y^k(y_{0:k})$ for any $0 \leq k \leq K$ and any $(x_{0:k}, y_{0:k}) \in C$. Define functions $\pi : X \to Y$ as in (2.32) and $\omega : Y \to X$ as in (2.40), the analysis in the proof of Theorem 3 has demonstrated that π and ω are bijections and that X and Y have same cardinality. Pick the bijection π and it follows $r_X^k(x_{0:k}) = r_Y^k(\pi(x_{0:k}))$ for any $0 \leq k \leq K$ and $x_{0:k} \in X$. This shows that $P_X^K \cong P_Y^K$ and completes the proof of the identity statement. ■ ■

In Theorem 3 we require $k \geq 1$ for the same reason as in Theorem 1. We emphasize that $d_{\mathcal{P}}^k$ is a metric in the space of proximity network modulo k-isomorphisms, whereas $d_{\mathcal{P},p}$ is a metric in the space of networks modulo isomorphism. Also note that we must have $d_{\mathcal{P},p}(P_X^K, P_Y^K) \geq \left\|\mathbf{d}_{\mathcal{P}}^K(P_X^K, P_Y^K)\right\|_p$ as per Proposition 3 but $\left\|\mathbf{d}_{\mathcal{P}}^K(P_X^K, P_Y^K)\right\|_p$ is not necessarily a metric.

Remark 1 GH distance is the minimum across correspondences of the maximum difference in distances between pairs of nodes for a given correspondence. The metric definitions as in Definitions 11, 12, 14, and 15 inherit this property, which means that network distances can be dominated by a small portion of the networks. Put differently, the proposed distances are more sensitive to a few large differences in a few edges than to a large number of small differences in a large number of edges. Analogous consideration can be found in signal processing theory of the tradeoffs between comparing signals with averages – such as 2-norm comparisons – and comparing signals with max-min differences – the ∞-norm comparison. When compare networks with different number

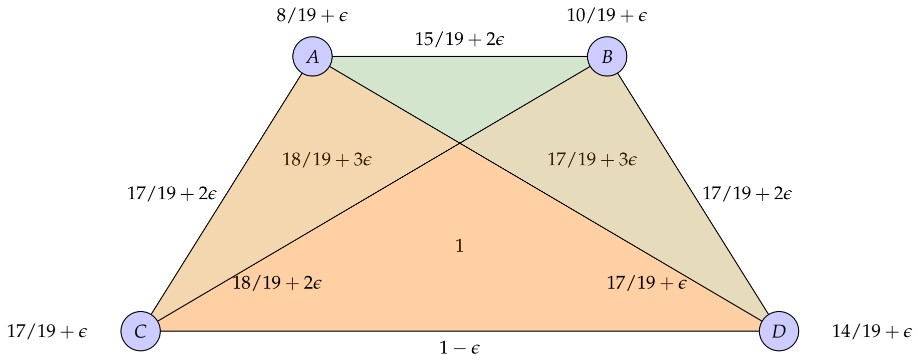


Figure 6: Relationships between authors expressed in terms of dissimilarities constructed from the proximity network in Figure 5. The k-order relationship function in this 2-order network denotes the level of dissimilarities between members of a given $(k+1)$-tuples. This is a dissimilarity network that has same order and identical node sets as the proximity network.

of nodes, a max-min comparison is reasonable because it focuses attention in the bottleneck tuple that makes it impossible to match smaller network onto the larger.

Remark 2 Once endowed with the proposed valid metrics as in Definitions 11, 12, 14, and 15, the space of dissimilarity networks and the space of proximity networks become metric spaces. This implies that a number of algorithms that are used to analyze metric spaces can now be used to analyze high order networks.

2.4.2. Duality between Dissimilarity and Proximity Networks

Proximity and dissimilarity networks have been defined separately for simplicity of presentation, but they are actually related entities. For any proximity network P_X^K with relationship functions $\hat{p}_X^k(x_{0:k})$, we can construct a dissimilarity network D_X^K on the same node space by defining relationships as $\hat{d}_X^k(x_{0:k}) = 1 - \hat{p}_X^k(x_{0:k})$ for all orders k and tuples $x_{0:k}$. Likewise given a dissimilarity network D_X^K with relationship functions $\hat{d}_X^k(x_{0:k})$ we can construct a proximity network P_X^K by defining relationships $\hat{p}_X^k(x_{0:k}) = 1 - \hat{d}_X^k(x_{0:k})$. We formalize this equivalence through the introduction of dual networks in the following definition.

Definition 16 *Given a node space X, the K-order proximity and dissimilarity networks $P_X^K = \left(X, \hat{p}_X^0, \hat{p}_X^1, \ldots, \hat{p}_X^K\right)$ and $D_X^K = \left(X, \hat{d}_X^0, \hat{d}_X^1, \ldots, \hat{d}_X^K\right)$ are said duals if and only if*

$$\hat{p}_X^k(x_{0:k}) = 1 - \hat{d}_X^k(x_{0:k}), \tag{2.48}$$

for all orders $0 \le k \le K$ and tuples $x_{0:k}$.

It is ready to see that all proximity networks have a dual dissimilarity network and that, conversely, all dissimilarity networks have a dual proximity network. To do so we just reinterpret

(2.48) as a definition and observe that: (i) The decomposition of relationships in the proximity network implies the valid decomposition of relationships in the dual dissimilarity network, and vice versa. (ii) The order decreasing property of the proximities in the proximity network implies the order increasing property of the dissimilarities in the dual dissimilarity network, and vice versa. An illustration for the construction of a dual dissimilarity network is presented in Figure 6, where we construct the corresponding dual dissimilarity network for the coauthorship network considered in Figure 5.

Given dual networks we can compute the distances in definitions 14 and 15 for proximity networks and the distances in definitions 11 and 12 for the dual dissimilarity networks. These definitions have been constructed so that the resulting distances are the same, as we formally state in the following proposition.

Proposition 4 *Consider two proximity networks P_X^K and P_Y^K and their corresponding dual dissimilarity networks D_X^K and D_Y^K. The k-order proximity distances $d_{\mathcal{P}}^k(P_X^K, P_Y^K)$ [cf. Definition 14] and k-order dissimilarity distances $d_{\mathcal{D}}^k(D_X^K, D_Y^K)$ [cf. Definition 11] coincide for all $0 \leq k \leq K$,*

$$d_{\mathcal{P}}^k(P_X^K, P_Y^K) = d_{\mathcal{D}}^k(D_X^K, D_Y^K). \tag{2.49}$$

Likewise, the p-norm proximity distance $d_{\mathcal{P},p}(P_X^K, P_Y^K)$ [cf. Definition 15] and p-norm dissimilarity distance $d_{\mathcal{D},p}(D_X^K, D_Y^K)$ [cf. Definition 12] coincide,

$$d_{\mathcal{P},p}(P_X^K, P_Y^K) = d_{\mathcal{D},p}(D_X^K, D_Y^K). \tag{2.50}$$

Proof: We first prove (2.49) by considering proximity networks P_X^K and P_Y^K and their corresponding dual dissimilarity networks D_X^K and D_Y^K. Let the correspondence C between X and Y be the minimizing correspondence in $d_{\mathcal{P}}^k(P_X^K, P_Y^K)$ [cf. Definition 14] so that we can write

$$d_{\mathcal{P}}^k(P_X^K, P_Y^K) = \Gamma_{P_X,P_Y}^k(C). \tag{2.51}$$

C may not be the minimizing correspondence for the distance $d_{\mathcal{D}}^k(D_X^K, D_Y^K)$ [cf. Definition 11], but since it is a valid correspondence, it holds true that

$$d_{\mathcal{D}}^k(D_X^K, D_Y^K) \leq \Gamma_{D_X,D_Y}^k(C). \tag{2.52}$$

From the definition of duality [cf. (2.48)], we may write

$$\Gamma_{D_X,D_Y}^k(C) = \max_{(x_{0:k}, y_{0:k}) \in C} \Big| \big(1 - \hat{d}_X^k(x_{0:k})\big) - \big(1 - \hat{d}_Y^k(y_{0:k})\big) \Big|. \tag{2.53}$$

The ones in (2.53) cancel out and therefore,

$$\Gamma_{D_X,D_Y}^k(C) = \Gamma_{P_X,P_Y}^k(C). \tag{2.54}$$

Substituting (2.51) and (2.52) back to (2.54) implies

$$d^k_{\mathcal{P}}(P^K_X, P^K_Y) \geq d^k_{\mathcal{D}}(D^K_X, D^K_Y). \tag{2.55}$$

Let the correspondence C' between X and Y be the minimizing correspondence in $d^k_{\mathcal{D}}(D^K_X, D^K_Y)$. Then C' is also a valid correspondence for the distance $d^K_{\mathcal{P}}(P^K_X, P^K_Y)$. By symmetry, we have

$$d^k_{\mathcal{D}}(D^K_X, D^K_Y) \geq d^k_{\mathcal{P}}(P^K_X, P^K_Y). \tag{2.56}$$

Combining (2.55) and (2.56) yields the desired result in (2.49).

Next we prove (2.50) by considering P^K_X and P^K_Y and their corresponding duals D^K_X and D^K_Y. Let the correspondence C between X and Y be the minimizing correspondence in $d_{\mathcal{P},p}(P^K_X, P^K_Y)$ [cf. Definition 12] so that we can write

$$d_{\mathcal{P},p}(P^K_X, P^K_Y) = \left\| \mathbf{\Gamma}^K_{P_X,P_Y}(C) \right\|_p. \tag{2.57}$$

C may not be the minimizing correspondence for the distance $d_{\mathcal{D},p}(D^K_X, D^K_Y)$ [cf. Definition 12], but again since it is a valid correspondence, we may write

$$d_{\mathcal{D},p}(D^K_X, D^K_Y) \leq \left\| \mathbf{\Gamma}^K_{D_X,D_Y}(C) \right\|_p. \tag{2.58}$$

We have demonstrated in proving (2.49) that for any integers $0 \leq k \leq K$, $\Gamma^k_{D_X,D_Y}(C) = \Gamma^k_{P_X,P_Y}(C)$. In vector form, this is $\mathbf{\Gamma}^K_{D_X,D_Y}(C) = \mathbf{\Gamma}^K_{P_X,P_Y}(C)$. Therefore, the property of p-norm implies that

$$\left\| \mathbf{\Gamma}^K_{D_X,D_Y}(C) \right\|_p = \left\| \mathbf{\Gamma}^K_{P_X,P_Y}(C) \right\|_p. \tag{2.59}$$

Substituting (2.57) and (2.58) back to (2.59) yields

$$d_{\mathcal{P},p}(P^K_X, P^K_Y) \geq d_{\mathcal{D},p}(D^K_X, D^K_Y). \tag{2.60}$$

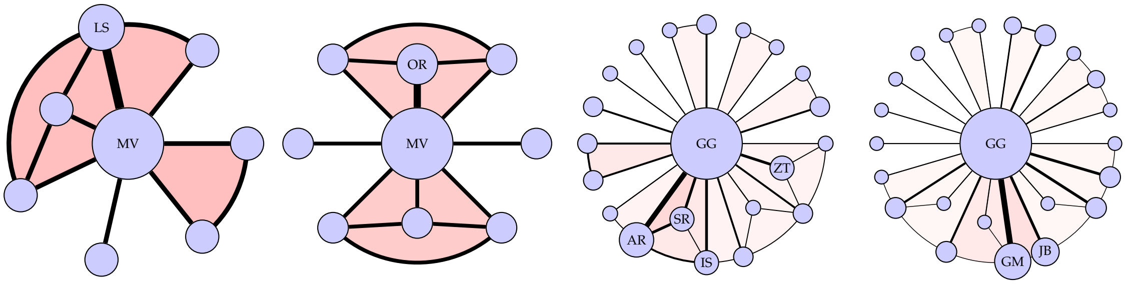
Let the correspondence C' between X and Y be the minimizing correspondence in $d_{\mathcal{D},p}(D^K_X, D^K_Y)$. Then C' is also a valid correspondence for $d_{\mathcal{P},p}(P^K_X, P^K_Y)$. By symmetry, we have

$$d_{\mathcal{D},p}(D^K_X, D^K_Y) \geq d_{\mathcal{P},p}(P^K_X, P^K_Y). \tag{2.61}$$

Combining (2.60) and (2.61) yields the desired result in (2.50). ■

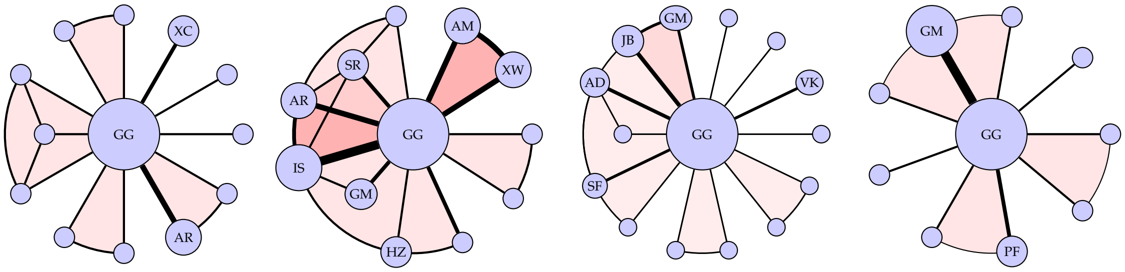
2.5. Comparison of Coauthorship Networks

We apply the metrics defined in Section 2.4.1 to compare second order coauthorship networks where relationship functions denote the number of publications of single authors, pairs of au-



MV, 2004 - 2008 MV, 2009 - 2013 GG, 2004 - 2008 GG, 2009 - 2013

Figure 7: Quinquennial coauthorship networks representing research communities centered at Prof. Georgios Giannakis (GG) or Prof. Martin Vetterli (MV). The size of the nodes is proportional to the zeroth order proximities, and the width of the links to the first order proximities. Second order proximities are represented by shading the triangle enclosed by the coauthor triplet. Color intensity is proportional to the second order proximities.



GG, 2006 - 2007 GG, 2008 - 2009 GG, 2010 - 2011 GG, 2012 - 2013

Figure 8: Biennial coauthorship networks representing research communities centered at Prof. Georgios Giannakis (GG).

thors, and triplets. These coauthorship networks are proximity networks because they satisfy the order decreasing property in Definition 13. Since both, Definition 14 and Definition 15, require searching over all possible correspondences between the node spaces, we can compute exact distances for networks with a small number of nodes only. Thus, we consider publications in the IEEE Transactions on Signal Processing (TSP) in the last decade but restrict attention to the collaboration networks of Prof. Georgios B. Giannakis (GG) of the University of Minnesota and Prof. Martin Vetterli (MV) of the École Polytechnique Fédérale de Lausanne. We choose these authors because their collaboration traits are more developed and stable and we expect their respective collaboration pattern to be steady over the past decade. The goal of the simulation is to illustrate that network metrics are able to distinguish discernible collaboration patterns. For each of the authors, GG and MV, we construct networks for the 2004-2008 and 2009-2013 quinquennia. These networks are referred as GG0408, GG0913, MV0408, and MV0913. For GG we also define networks for each of the biennia 2004-2005, 2006-2007, 2008-2009, 2010-2011, and 2012-2013. We denote these networks as GG0405, GG0607, GG0809, GG1011, and GG1213. Lists of publications are queried from [135].

For each of these authors we consider all of their TSP publications in the period of interest and construct proximity networks where the node space X is formed by the author and the respective set of coauthors. Zeroth order proximities are defined as the total number of publications of each member of the network, first order proximities as the number of publications coauthored by pairs, and second order proximities as the number of publications coauthored by triplets. The constant ϵ as in Definition 13 is for technical purpose. It can be chosen sufficiently small and for this reason we ignore it in this section. To make networks with different numbers of publications comparable we normalize all distances by the total number of publications in the network. With this construction we have that the zeroth order proximity of GG or MV are 1 in all of their respective networks. There are publications with more than three coauthors but we don't record proximities of order higher than 2.

The quenquennial networks GG0408, GG0913, MV0408, and MV0913 are shown in Figure 7 and the biennial networks GG0607, GG0809, GG1011, and GG1213 in Figure 8. The size of the nodes is proportional to the zeroth order distances, and the width of the links to the first order distances. Second order proximities are represented by shading the triangle enclosed by the coauthor triplet and the color intensity is proportional to the second order proximities. There are clear differences in the collaboration patterns. We show here that proximity network distances succeed in identifying these patterns and distinguish between the coauthorship networks of GG and MV.

2.5.1. Quinquennial Networks

Two dimensional Euclidean embeddings (respect to minimizing the sum of squares of the interpoint distances) of the k-order proximity network distances $d_{\mathcal{P}}^k$ for $k \in \{0,1,2\}$ and the proximity network distance with respect to the 1-norm, $d_{\mathcal{P},1}$ are shown in Figure 9. The two GG networks (diamonds) separate clearly from the two MV networks (circles) either by considering the individual k-order distances $d_{\mathcal{P}}^k$ or the aggregate distance $d_{\mathcal{P},1}$. The distances between the two MV networks are high but still smaller than the distances between GG networks and MV networks. An unsupervised classification run across all four distances would assign all four networks correctly.

The k-order network distance $d_{\mathcal{P}}^k$ is defined by searching for the correspondence such that the maximum k-order proximity difference $|r_X^k(x_{0:k}) - r_Y^k(y_{0:k})|$ among all tuples of correspondents is minimized [cf. (2.8) and (2.9)]. For the optimal correspondence $C^\star = \operatorname{argmin}_{C \in \mathcal{C}(X,Y)} \Gamma_{X,Y}^k(C)$, define the pair of correspondent tuples that achieve the maximum k-order difference as

$$(x_{0:k}^\star, y_{0:k}^\star) = \operatorname*{argmax}_{(x_{0:k}, y_{0:k}) \in C^\star} \left| r_X^k(x_{0:k}) - r_Y^k(y_{0:k}) \right|. \tag{2.62}$$

The tuple pair $(x_{0:k}^\star, y_{0:k}^\star)$ is the bottleneck that prevents making the networks closer to each other. Examining these bottleneck pairs for each k-order distance reveals what are the differences between proximity networks to which $d_{\mathcal{P}}^k$ is most sensitive about. In general, k-order bottleneck pairs tend to be pairs of tuples with high proximity values in their respective networks. The optimal correspondence $C^\star$ map tuples with high proximity as closely as possible. Therefore, network distances are typically determined by large proximity values in one of the networks that can't be matched closely to proximity values in the other network.

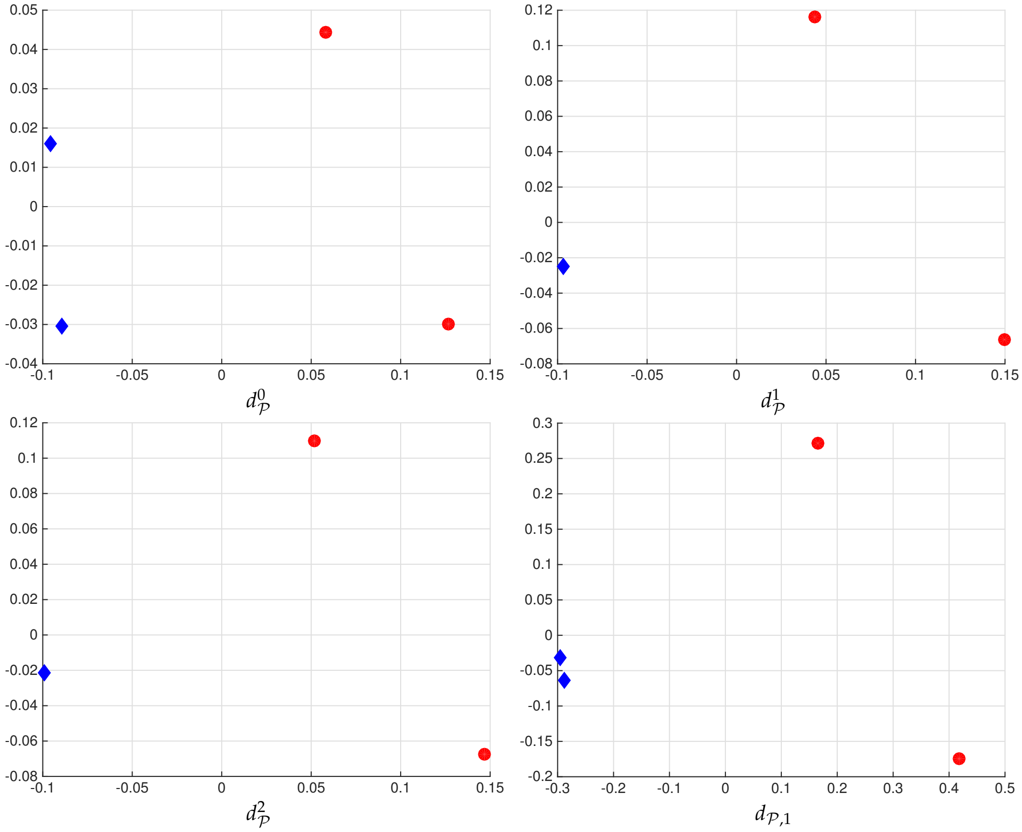


Figure 9: Two dimensional Euclidean embeddings of the k-order proximity network distances $d_{\mathcal{P}}^0, d_{\mathcal{P}}^1, d_{\mathcal{P}}^2$ and the proximity network distance with respect to the 1-norm, $d_{\mathcal{P},1}$, between the quinquennial networks. In the embeddings, denote MV0408, MV0913 as circles, GG0408, GG0913 as diamonds. GG0408 and GG0913 are colocated regarding $d_{\mathcal{P}}^1, d_{\mathcal{P}}^2$.

In the quinquennial coauthorship networks of Figure 7 the bottleneck pair for 0-order distances $d_{\mathcal{P}}^0$, is formed by nodes with high zero order proximities and $d_{\mathcal{P}}^0$ reflects the difference between their zero order proximities. Since the networks are normalized so that the lead nodes have size 1, $d_{\mathcal{P}}^0$ is determined by their predominant coauthors, i.e., the scholars that collaborated most prolifically with GG or VM during the period of interest. The distances $d_{\mathcal{P}}^0$ between GG and VM networks are large because these predominant collaborations are different. In GG networks there are usually groups of 3 to 5 predominant collaborators, whereas in MV networks there are usually one or two that concentrate a larger fraction of the total number of publications.

Similarly, high first order proximity distances are likely due to one of the following situations: (i) Large differences between the numbers of publications authored by the predominant collaborators. (ii) Different patterns in the formation of communities – defined here as clusters of pairwise collaboration. In the latter case large distances arise because it is impossible to match the communities in one network to communities in the other. The distances $d_{\mathcal{P}}^1$ between GG and MV networks are large because the latter contain a smaller number of communities, which are also more strongly connected than the communities in GG networks.

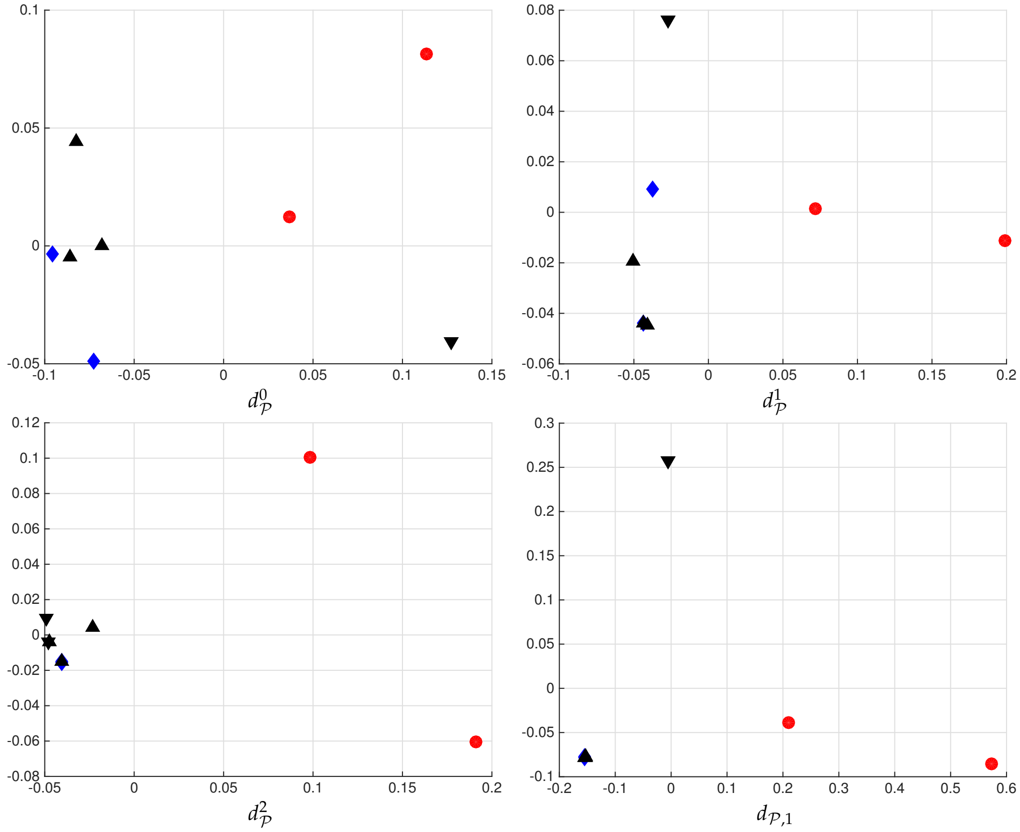


Figure 10: Two dimensional Euclidean embeddings of the distances $d_{\mathcal{P}}^0, d_{\mathcal{P}}^1, d_{\mathcal{P}}^2, d_{\mathcal{P},1}$ between all quinquennial and biennial networks. In the embeddings, denote MV0408, MV0913 as circles, GG0408, GG0913 as diamonds, GG0405, GG0607, GG1011 as up triangles and GG0809, GG1213 as down triangles. GG0809, GG1213 are colocated regarding $d_{\mathcal{P}}^0, d_{\mathcal{P}}^1, d_{\mathcal{P},1}$. GG0408 and GG0913 have identical coordinates in $d_{\mathcal{P}}^1$.

In second order distances the bottleneck pair of triplets may reflect one of the following scenarios: (i) One network has collaboration between four or more authors while the other doesn't. (ii) There exist three authors with a strong collaboration between them in one network whereas in the other network there does not exist collaboration between three authors or, if such collaboration exists, it is weak. Many publications written by MV are collaborations of three or four scholars and the predominant coauthor in MV networks appears in at least one collaboration of four scholars. For GG, his 2004-2008 network has a few collaborations consisting of four scholars however all such collaborations are weak. His 2009-2013 network has no publications written by four authors.

2.5.2. *Biennial Networks*

The networks GG0408 and GG0913 have more nodes than the networks MV0408 and MV0913 prompting the possibility that the differences in distances discussed in Section 2.5.2 are just due to their different number of publications. This is part of the reason, but not all. To see that this is true we consider the biennial GG collaboration networks. Each of these networks contain numbers of publications that are comparable to the number of publications in the quinquennial MV networks.

Two dimensional Euclidean embeddings of the individual k-order distances $d_{\mathcal{P}}^{k}$ for $k \in \{0,1,2\}$ and the aggregate distance $d_{\mathcal{P},1}$ between the 4 quinquennial networks and the 5 biennial networks are shown in Figure 10. An unsupervised classification run across four distances would assign all nine networks correctly ($d_{\mathcal{P}}^{1}, d_{\mathcal{P}}^{2}$) or two of them incorrectly ($d_{\mathcal{P}}^{0}, d_{\mathcal{P},1}$).

We expect more variation in biennial networks because the time for averaging behavior is reduced. E.g., we may see deviations from usual collaboration patterns due to the presence of exceptional doctoral students. Still, three of the biennial networks, GG0405, GG0607, GG1011, (up triangles) and the two quinquennial networks GG0408, GG0913 (diamonds) are close to each other in every metric used and form a cluster clearly separate from the two five-year networks MV0408 and MV0913 (circles). This is due to the fact that the distinctive features of GG coauthorship are well reflected in GG0405, GG0607, GG1011. These features include: (i) Multiple predominant coauthors, each of whose collaboration with GG does not comprise a dominant portion of GG's scholarship during the period. (ii) Multiple small coauthorship communities in which strong collaborations within each community are rare. (iii) The number of publications with four or more authors is low. These features contrast with the rather opposite properties of the MV networks.

The networks GG0809 and GG1213 (down triangles) do not cluster nicely with the other five GG networks. Depending on which distance we consider they may be closest to some of the other GG networks or to one of the two MV networks. This is because, likely due to random variation, GG0809 and GG1213 have some features that resemble GG networks and some other features that resemble MV networks. Fundamentally this happens because of the exceptionally prolific collaborations with Ioannis Schizas (IS) in the 2008-2009 period and Gonzalo Mateos (GM) in the 2012-2013 period. In the network GG0809 the IS node commands a significant fraction of GG publications and creates strong links between collaboration clusters that would be otherwise separate. Both of these features are more characteristic of MV networks. In GG1213 network the GM node accounts for half of the publications in which GG is an author. This is, also, a feature more representative of MV networks than of GG networks.

In summary, proximity network distances capture features of scholar collaboration that permit discerning networks of different authors even when we consider networks that have very different numbers of nodes. The zeroth order distance $d_{\mathcal{P}}^{0}$ responds primarily to the number of predominant coauthors and the proportion of collaboration between predominant coauthors and the central scholar. The first order distance $d_{\mathcal{P}}^{1}$ is mostly determined by the fraction of collaborations that involve predominant coauthors and the central scholar as well as the level and number of strong collaborations within each community in the group. The second order distance $d_{\mathcal{P}}^{2}$ is largely given by the existence, level, and number of collaborations between four or more scholars and the appearance of predominant coauthors in a collaboration between four or more scholars.

Remark 3 The proposed metrics successfully identify the distinct collaborative behaviors of Prof. G.

B. Giannakis and Prof. M. Vetterli from incomplete subsets of their publication datasets. The distances between Giannkis's networks (either quinquennial or biennial) are smaller than the distances between Giannakis's networks and Vetterli's networks. This proximity can be used in author name disambiguration or related problems, e.g., adjudicate the biennial networks to their rightful author if only the authors of the quinquennial networks are known.

Remark 4 As a comparison, we applied some simple and reasonable methods to compare the corresponding pairwise networks of the coauthorship networks considered in this section. Motifs have been shown effective in distinguishing coauthorship networks from different scientific fields [16]. To compare high order coauthorship networks by motifs, we restrict attention to pairwise relationships. The dissimilarities between coauthorship networks are assigned as the differences between the summations of the weighted motifs in their corresponding pairwise networks. Analysis based on triangle motifs (weighted) results in MV0408, MV0913, GG0408, and GG0809 being closer to each other and GG0913, GG0405, GG0607, GG1011, and GG1213 being more proximate. Tetrahedron motif analysis (weighted) results in MV0408, MV0913, GG0408, GG0405, GG0607, and GG0809 being closer to each other and GG0913, GG1011, and GG1213 being more proximate. Other simple and common methods to compare pairwise networks yield similar results. Methods to compare pairwise networks via features give us similar observations as those based on the metric distances proposed in the chapter. Notice that GG0408 and GG0913 are highly similar regarding the proposed network distances however their differences are relatively large in terms of feature comparisons.

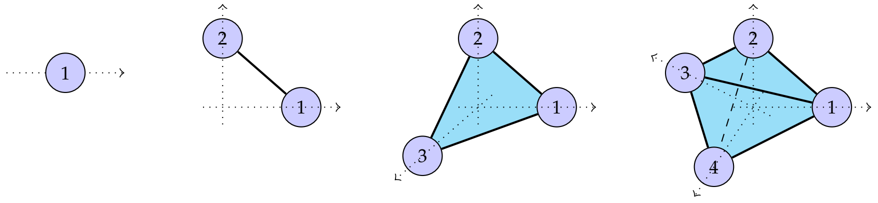
Chapter 3

Persistent Homology Lower Bounds on Network Distances

In this chapter, we propose to compare networks by considering their respective filtrations and using the difference between the persistence diagrams of the respective filtrations as a proxy for their distance; this is related with the universality property studied in [136]. (Section 3.2). This proposed methodology is substantiated by the fact that we can lower bound the computationally intractable distance between two high order networks with a tractable distance between their respective persistence diagrams (Theorems 5 and 6 in Section 3.2). These lower bounds are tight such that there are examples in which the lower bound and the actual distances coincide. Since persistent homologies can be computed efficiently for large networks (Section 3.3 and [137]), we can use these lower bounds in, e.g., network classification problems. We do so for artificial networks created with different models – random networks, Gaussian kernel proximity networks, and Euclidean feature networks – and for coauthorship networks constructed from the publications of a number of journals from engineering and mathematics communities. The proposed methods succeed in distinguishing networks with different generative models (Section 3.4.1) and are also effective in discriminating engineering journals from mathematics journals (Section 3.4.2). We also attempt a more challenging classification problem of three different engineering communities where we achieve a moderate success (Section 3.4.3).

3.1. Networks and Simplicial Complexes

We introduce elemental notions of computational topology as they apply to the study of high order networks. We refer authors to [138] for more background and more formal notions on computational homology. Begin by defining a k-simplex $\phi = [x_{0:k}]$ as the convex hull of the set of points $x_{0:k}$ (see Figure 11) and a simplicial complex L as the collection of simplices such that for any simplex $[x_{0:k}]$, the convex hull of any subset of $x_{0:k}$ also belongs to L. Figure 12 (a) exemplifies



0-simplex 1-simplex 2-simplex 3-simplex

Figure 11: Elementary k-simplices for $0 \leq k \leq 3$.

two triangles connected together as a simplicial complex of dimension 2. It is a simplicial complex because for any of its simplices, say, $[a, b, c]$, the convex hulls of the subsets of $\{a, b, c\}$ – which include the points $[a]$, $[b]$, and $[c]$ as well as the edges $[a, b]$, $[b, c]$, and $[a, c]$ – all belong to the simplicial complex as well.

An important concept in simplicial complexes is that of a hole without interior. For the simplicial complex in Figure 12 (a), the area enclosed by $[b, d], [d, c], [c, b]$ is a hole without interior because the area is not filled. The area enclosed by $[a, b], [b, d], [d, a]$ is not because the interior is filled by the 2-simplex $[a, b, d]$. Homologies are defined to formalize this intuition and rely on the definitions of chains, cycles, and boundaries. The *k-chain* $\Phi_k = \sum_i \beta_i \phi_i$ is a summation of k-simplices ϕ_i modulated by coefficients β_i whose signs denote orientation. This definition is a generalization of the familiar definition of chains in graphs. E.g., in Figure 12, $[a, b] + [b, d]$ is a 1-chain that we can equivalently represent as $[a, b] - [d, b]$. Further consider a given k-simplex $\phi = [x_{0:k}]$ and its border $(k-1)$-simplices defined as the ordered set of elements $[x_{0:\widehat{l}:k}] = \text{conv}\{x_{0:k} \backslash x_l\}$, in which each of the elements is removed in order. The *boundary* $\partial_k \phi$ of the simplex ϕ is the chain formed by its borders using alternating orientations,

$$\partial_k \phi = \sum_{l=0}^{k} (-1)^l [x_{0:\widehat{l}:k}]. \tag{3.1}$$

According to (3.1), the boundary $\partial_k \phi$ of a k-simplex is the collection of $(k-1)$-simplices. For the simplices in Figure 11, the boundaries are $\partial_0[a] = 0, \partial_1[a, b] = [b] - [a], \partial_2[a, b, c] = [b, c] - [a, c] + [a, b]$ and $\partial_3[a, b, c, d] = [b, c, d] - [a, c, d] + [a, b, d] - [a, b, c]$.

Having defined chains and boundaries we consider a k-chain $\Phi_k = \sum_i \beta_i \phi_i$ and define the *chain boundary* $\partial_k \Phi_k$ as the summation of the boundaries of its component simplices,

$$\partial_k \Phi_k = \sum_i \beta_i (\partial_k \phi_i). \tag{3.2}$$

As per this definition, the boundary of the chain $\Phi = [a, b] + [b, d]$ in Figure 12 is $\partial_k \Phi = [a] - [b] + [b] - [d] = [a] - [d]$, which matches our intuition of what the chain's boundary should be. We can now formally define a *k-cycle* Ψ_k as a chain whose boundary is null, i.e., a k-chain for which $\partial_k \Psi = 0$. In Figure 12 the chain $\Psi = [a, b] + [b, d] + [d, a]$ is a cycle because its boundary is

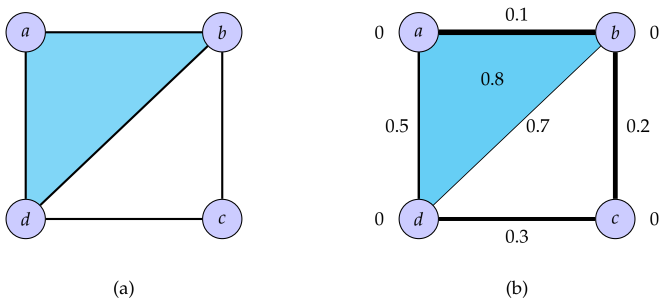


Figure 12: (a) An example of simplicial complex L which consists of four 0-simplices, five 1-simplices, and one 2-simplex. (b) A dissimilarity network can be represented as a simplicial complexes with weights. When the weight of a simplex denotes the time instant the simplex appears in the nested sequence of simplicial complexes, it yields a valid filtration $\mathcal{L}$.

$\partial_k \Psi = [a] - [b] + [b] - [d] + [d] - [a] = 0.$

A k-cycle Ψ_k is a *k-homological feature* if it cannot be represented as the boundary of a $(k+1)$-chain Φ_{k+1}; otherwise Ψ_k is a *k-boundary*. We refer readers to [138] for the more formal definition. The group $\mathcal{H}_k(L)$ of k-homological features of a simplicial complex L represents all k-cycles that are not k-boundaries. I.e., a homological feature appears when a k-cycle Ψ_k is not among the group of boundaries $\partial_{k+1}\Phi_{k+1}$ of $(k+1)$-chains.

For the simplicial complex in Figure 12 (a), the chain $\Psi_1 = [a,b] + [b,d] + [d,a]$ is a 1-cycle because $\partial_1 \Psi_1 = 0$; simultaneously, Ψ_1 is also a 1-boundary because $\partial_2[a,b,d] = [b,d] - [a,d] + [a,b]$, which is identical to Ψ_1. Therefore, Ψ_1 does not represent a homological feature. On the other hand, the chain $\Psi'_1 = [b,c] + [c,d] + [d,b]$ describes a homological feature because it is a cycle with $\partial_1 \Psi'_1 = 0$, and not a boundary due to the fact that the 2-simplex $[b,c,d]$ is absent from the complex. One could also say that $\Psi''_1 = [a,b] + [b,c] + [c,d] + [d,a]$ describes a 1-cycle that is not a 1-boundary. Nonetheless, because the difference between Ψ''_1 and Ψ'_1 is $[a,b] + [b,d] + [d,a]$ which is a 1-boundary, the cycles Ψ''_1 and Ψ'_1 are essentially denote the same homological feature. Therefore, there only exists one 1-homological feature, which may be represented by chain Ψ'_1 or Ψ''_1.

Other illustrative examples are shown in Figure 13. The complex (a) has one 1-homological feature whose one representation is the chain $[a,b] + [b,c] + [c,a]$ and the complex (b) has 1 feature which can be represented by the chain $[a,b] + [b,c] + [c,d] + [d,a]$. Complex (c) has 2 features that may be denoted by chains $[a,b] + [b,d] + [d,a]$ and $[b,c] + [c,d] + [d,b]$ but complex (d) has one feature because the chain $[a,b] + [b,d] + [d,a]$ is the boundary of the simplex $[a,b,d]$.

3.1.1. Representation of High Order Networks as Filtrations

In the same manner in which a graph represents an unweighted network, a simplicial complex represents a high order unweighted network. To represent weighted high order networks we assign a weight to each simplex, but instead of thinking the network as a weighted simplicial

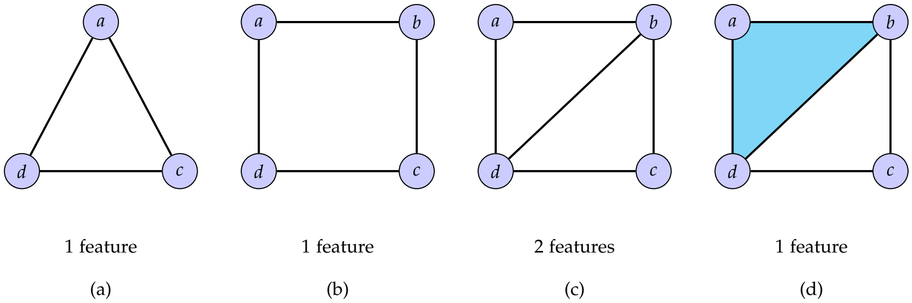


Figure 13: More examples of simplicial complexes and homologies. The number of 2-homological features are described below each of the simplicial complex.

complex we think of weights as parameters that indicates the time at which the simplex comes into existence. Formally, for parameters $\alpha \in [0,1]$ we define a filtration $\mathcal{L}$ as a collection of simplicial complexes L_α such that for any ordered sequence $0 = \alpha_0 < \alpha_1 < \ldots < \alpha_m = 1$ it holds $\emptyset = L_{\alpha_0} \subseteq L_{\alpha_1} \subseteq \ldots \subseteq L_{\alpha_m} = L$. The minimum time α at which a simplex becomes an element of L_α is the birth time of the simplex.

Given a dissimilarity network D_X^K, we construct $\mathcal{L}(D_X^K)$ by assigning the appearing time of simplex $[x_{0:k}]$ as the relationship $r_X^k(x_{0:k})$ of the corresponding tuple,

$$[x_{0:k}] \in L_\alpha \iff r_X^k(x_{0:k}) \leq \alpha. \tag{3.3}$$

We show next that (3.3) defines a valid filtration.

Proposition 5 *The construction* $\mathcal{L}(D_X^K)$ *with* L_α *established from a dissimilarity network* D_X^K *via the relationship in* (3.3) *is a well defined filtration.*

Proof: For a given α, (3.3) defines a set of simplices L_α. It is clear that the collection of sets L_α is nested, i.e., that $\emptyset = L_{\alpha_0} \subseteq \ldots \subseteq L_{\alpha_m} = L$ holds for any set of birth times $0 = \alpha_0 < \ldots < \alpha_m = 1$. To prove the statement we need to show that the set of simplices L_α is a valid simplicial complex. To show this it suffices to verify that for any α, all faces of each simplex in L_α also appear no later than α. Suppose simplex $[x_{0:k}]$ appears before time α for some chosen α, then it must be true that $r_X^k(x_{0:k}) \leq \alpha$. For any faces of $[x_{0:k}]$, say $[x_{0:\tilde{k}}]$ with $\tilde{k} < k$, the order increasing property implies $r_X^{\tilde{k}}(x_{0:\tilde{k}}) \leq r_X^k(x_{0:k})$. Therefore,

$$r_X^{\tilde{k}}(x_{0:\tilde{k}}) \leq r_X^k(x_{0:k}) \leq \alpha, \tag{3.4}$$

which shows that the face $[x_{0:\tilde{k}}]$ appears before time or on time α. This means that L_α is a valid simplicial complex. ■

We emphasize that filtrations can't be defined for an arbitrary high order networks because the order increasing property is necessary for the proof. For dissimilarity networks that do satisfy the

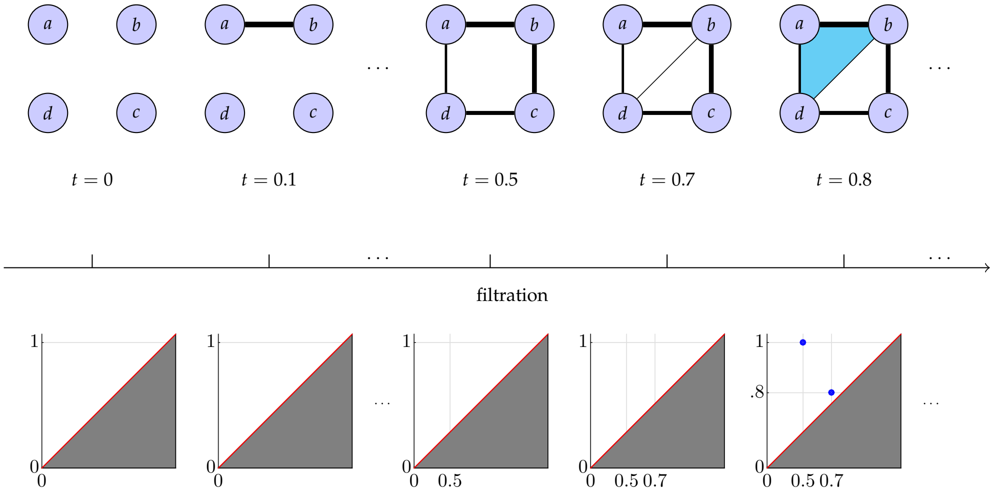


Figure 14: Top: filtration of nested simplicial complexes of the dissimilarity network exhibited in Figure 12 (b). The simplicial complex at each time instant are the collections of all vertices appearing before or on that time. Bottom: 1-persistence diagrams describing 1-persistent homologies for each of the simplicial complexes detailed in Top. The horizontal axis denotes the birth time of homological features and the vertical axis represents the death time.

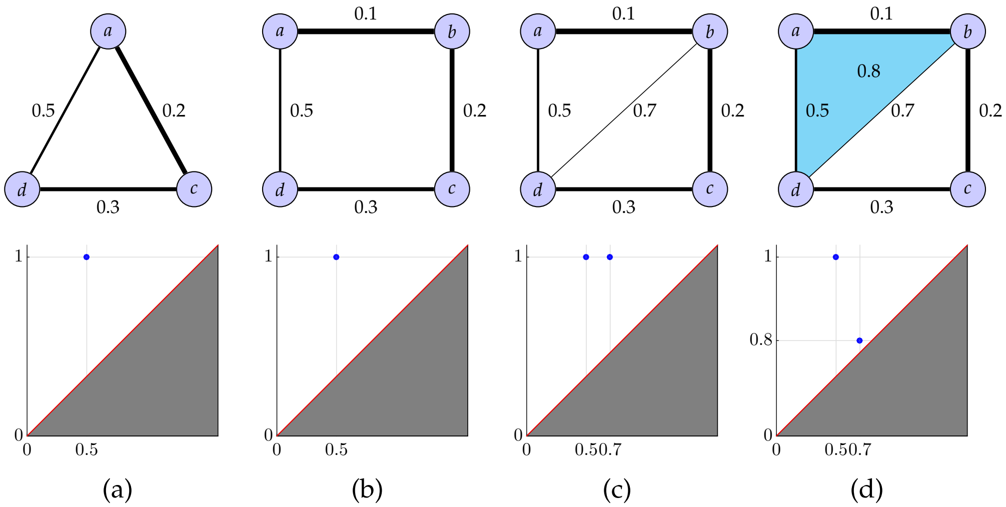


Figure 15: The 1-persistence diagrams for the filtration induced by each of the dissimilarity networks. The maximum death time of any homological features is 1 because hyperedges with undefined relationships take value 1 implicitly.

order increasing property, Proposition 5 establishes the existence of an equivalent filtration.

Figure 14 (top) shows the construction of the filtration associated with the dissimilarity network in Figure 12 (b). At time $\alpha = 0$, there are four nodes because $r_X^0(a) = r_X^0(b) = r_X^0(c) = r_X^0(d) = 0$. At time $\alpha = 0.1$, the edge $[a, b]$ starts to appear in the simplicial complex because the relationship $r_X^1(a, b) = 0.1$. As time gradually increases, more simplices get included in the simplicial complex at each time instant. At the end of the filtration with $\alpha = 1$, all simplices in the simplicial complex, whose corresponding relationships in the dissimilarity networks are either defined or undefined, are involved.

3.1.2. Persistent Homologies and Persistence Diagrams

As time advances in a filtration, holes and interiors appear and disappear. Persistent homologies examine when these homological features appear for the sequence of simplicial complexes in the filtration. Formally, given a filtration $\mathcal{L}$, its k-dimensional persistence diagram $\mathcal{P}_k\mathcal{L}$ is a collection of points of the form $\mathbf{q} = [q_b, q_d]$ where q_b and $q_d > q_b$ represent the birth and death time of a homological feature. I.e., the times q_b represent resolutions $\alpha = q_b$ at which a feature is added to the group of k-homological features $\mathcal{H}_k(L_\alpha) = \mathcal{H}_k(L_{q_b})$ and the times q_d are resolutions $\alpha = q_d$ at which a feature is removed from the group of k-homological features $\mathcal{H}_k(L_\alpha) = \mathcal{H}_k(L_{q_d})$. We refer authors to [138] for more background and more formal definitions on persistent homology, and refer to [139] for a more comprehensive treatment of persistence and its connection to application.

The persistence diagram for the filtration in Figure 14-(top) is shown in Figure 14-(bottom). In this diagram the horizontal axis denotes the birth time of 1-homological features (when holes appear) and the vertical axis represents the death time of 1-homological features (when holes are filled). At time $\alpha = 0$, there are no 1-simplices and consequently no 1-holes. The first 1-cycle appears at time $\alpha = 0.5$ when the 1-chain $[a, b] + [b, c] + [c, d] + [d, a]$ appears. We mark this event by the addition of a vertical line in the persistence diagram. A second homological feature appears in the form of the cycle $[a, b] + [b, d] + [d, a]$ at time $\alpha = 0.7$. This event is marked by the addition of a second vertical line in the persistence diagram. This homological feature dies at time $\alpha = 0.8$ when the 2-simplex $[a, b, d]$ appears and makes the 1-cycle $[a, b] + [b, d] + [d, a]$ a 1-boundary and therefore no longer a homological feature. This is marked by the addition of the *point* $\mathbf{q}_1 = (0.7, 0.8)$ to the persistence diagram. At this time all simplices with defined relationship functions have been added to the filtration. This means that the homological feature $[a, b] + [b, c] + [c, d] + [d, a]$ never becomes a boundary of a 2-simplex. This is marked by adding the point $\mathbf{q}_2 = (0.5, 1)$ to the persistence diagram – recall that dissimilarities must be between 0 and 1 by definition. All undefined simplifies, e.g. $[a, c], [a, c, d], [a, b, c, d]$, are considered to appear at 1 and would not affect the persistence diagram.

A few more examples are shown in Figure 15. Network (a) contains one feature that appears at $\alpha = 0.5$ and stays alive until $\alpha = 1$. Network (b) also has one feature born at $\alpha = 0.5$ that never gets trivialized until $\alpha = 1$. Network (c) involves two features, born at $\alpha = 0.5$ and $\alpha = 0.7$ that also stay alive until $\alpha = 1$. Network (d) is the same as the filtration in Figure 14-(top).

We close this section by noting all dissimilarity values between tuples with non-repeating elements of any dissimilarity network appear in the homological features of the induced filtration.

Proposition 6 *Given a dissimilarity network D_X^K, any of its k-order dissimilarities between tuples with unique elements appear either in the death time of the $(k-1)$-th dimensional homological features or the birth time of the k-th dimensional features.*

Proof: Given any tuple $x_{0:k}$ with non-duplicating nodes, (3.4) indicates that the k-simplex ϕ^k defined by the convex hull $\text{conv}\{x_{0:k}\}$ appears strictly after any of its faces $\text{conv}\{x_{0:\widehat{s}:k}\}$ in the filtration. Suppose ϕ^k appears at time α and denote $\partial_k \phi^k = \sum_i \beta_i \psi_i^{k-1}$ with β_i the coefficients, then each ψ_i^{k-1} appears strictly before time α.

Now suppose that the appearance of ϕ^k trivializes a $(k-1)$-th dimensional homological feature. This means that ϕ^k is the boundary to trivialize the $(k-1)$-th dimensional cycle $\partial_k \phi^k$. Since each face ψ_i^{k-1} of ϕ^k appears strictly before time α, the cycle $\partial_k \phi^k$ results in a homological feature. The death time of this homological feature is α, or equivalently, the time represented by the relationship $r_X^k(x_{0:k})$.

On the other hand, if the appearance of ϕ^k does not trivialize a $(k-1)$-th dimensional homological feature, then the $(k-1)$-cycle $\partial_k \phi^k$ is in the collection of simplices appearing before or on time α. This means that $\partial_k \phi^k$ can be represented by a sum of the boundaries of some k-chains Φ_i^k,

$$\partial_k \phi^k = \sum_i \beta_i \partial_k \Phi_i^k \tag{3.5}$$

with coefficients β_i and k-chains Φ_i^k appearing before or on time α. By the definition of k-chains, $\Phi_i = \sum_j \beta_j' \psi_j^k$ with coefficients β_j' and k-simplices ψ_j^k appears before or on time α. Therefore, (3.5) can be written as $\partial_k \phi^k = \sum_j \beta_j'' \partial_k \psi_j^k$. Rearranging terms,

$$\partial_k \Big(\sum_j \beta_j'' \psi_j^k - \phi^k \Big) = 0. \tag{3.6}$$

This implies that $\sum_i \beta_i'' \psi_i^k - \phi^k$ is a k-cycle. There must be a new cycle formed since ϕ^k just appears. The cycle cannot be trivialized immediately since any $(k+1)$-chain Ψ^{k+1} with $\partial_{k+1} \Psi^{k+1} = \sum_i \beta_i'' \psi_i^k - \phi^k$ would involve a simplex $[x_{0:k,l}]$ for some node x_l with tuple $x_{0:k,l}$ consisted of non-repeating elements where this simplex $[x_{0:k,l}]$ appears strictly after α. Therefore we have a k-th dimensional homological feature with birth time α, or equivalently, the time denoted by the relationship $r_X^k(x_{0:k})$. This concludes the proof. ■

Proposition 6 is specific to filtrations induced from dissimilarity networks since it follows from the fact that adding elements to a tuple results in *strictly increasing* dissimilarities [cf. Definition 10]. The proposition implies that different networks result in different persistence diagrams except in the rare cases when a single point in the diagram represents multiple homological features. Thus, persistence diagrams retain almost all of the dissimilarity values of a given network and are therefore not unreasonable proxies for network discrimination. We cement these observations in the next section by proving that differences between persistence diagrams yield lower bounds on the network distances defined in Section 2.2. We do so after an important remark.

Remark 5 The maximum death time of any homological features in filtrations induced from dissimilarity networks is 1. This is specific for filtrations constructed from dissimilarity networks because any hyperedges with undefined relationships in dissimilarity networks all take value 1 implicitly. In computing the persistence diagram, we can either (i) set the appearance time of all hyperedges with undefined relationships as 1 before evaluating the persistence diagrams, or (ii) determine the persistence diagram with no additional care of undefined edges, and then set the death time of any undead features to 1. These two procedures yield the same persistence diagrams, except for the first 0-dimensional persistent interval which dies at ∞ and represents the connected component. We use the latter method in practice because it is computational simpler.

3.2. Persistence Bounds on Network Distances

In this section we use differences between persistence diagrams to compute lower bounds of network distances. Recall that a persistence diagram $\mathcal{P}_k\mathcal{L}(D_X^K)$ is a collection of points of the form $\mathbf{q} = [q_b, q_d]$ where q_b and $q_d > q_b$ represent the birth and death time of a k-homological feature [cf. Figure 15]. To compare persistence diagrams $\mathcal{P}_k\mathcal{L}(D_X^K)$ and $\mathcal{P}_k\tilde{\mathcal{L}}(D_Y^K)$ of networks D_X^K and D_Y^K we begin by defining the cost of matching features $\mathbf{q} \in \mathcal{Q}$ and $\tilde{\mathbf{q}} \in \tilde{\mathcal{Q}}$ through the infinity norm of their difference,

$$\|\mathbf{q} - \tilde{\mathbf{q}}\|_\infty := \max\left[|q_b - \tilde{q}_b|, |q_d - \tilde{q}_d|\right]. \tag{3.7}$$

Further observe that the diagonal of a persistence diagram can be construed to represent an uncountable number of features with equal birth and death times. Thus, any feature $\mathbf{q} \in \mathcal{Q}$ can be compared to any of the artificial features of the form $\tilde{\mathbf{q}} = [\tilde{q}, \tilde{q}]$. Since this feature $\tilde{\mathbf{q}}$ can be placed anywhere in the diagonal of the persistence diagram, we choose to place it in the point that makes the infinity norm difference smallest. This point is $\tilde{\mathbf{q}} = [(q_b + q_d)/2, (q_b + q_d)/2]$ which yields the difference $\|\mathbf{q} - \tilde{\mathbf{q}}\|_\infty = |q_d - q_b|/2$. Likewise, artificial features $\mathbf{q} = [(\tilde{q}_b + \tilde{q}_d)/2, (\tilde{q}_b + \tilde{q}_d)/2]$ can be added for any point $\tilde{\mathbf{q}} \in \tilde{\mathcal{Q}}$. We can then rephrase the comparison in (3.7) so that if it is more advantageous to compare with artificial diagonal features. This is formally accomplished by defining the matching cost $c(\mathbf{q}, \tilde{\mathbf{q}})$ between $\mathbf{q}$ and $\tilde{\mathbf{q}}$ as

$$c(\mathbf{q}, \tilde{\mathbf{q}}) := \min\Big[\,\|\mathbf{q} - \tilde{\mathbf{q}}\|_\infty\,, (1/2)\max\big[\,|q_d - q_b|\,, |\tilde{q}_d - \tilde{q}_b|\,\big]\Big], \tag{3.8}$$

The norm $\|\mathbf{q} - \tilde{\mathbf{q}}\|_\infty$ is the cost of directly matching features $\mathbf{q}$ and $\tilde{\mathbf{q}}$. The term $(1/2)\max\big[\,|q_d - q_b|\,, |\tilde{q}_d - \tilde{q}_b|$ is the cost of matching both, $\mathbf{q}$ and $\tilde{\mathbf{q}}$ to artificial diagonal features. The cost $c(\mathbf{q}, \tilde{\mathbf{q}})$ of matching $\mathbf{q}$ and $\tilde{\mathbf{q}}$ is the smaller of these two.

If the respective collections of features $\mathcal{Q}$ and $\tilde{\mathcal{Q}}$ contain the same number of elements $m = \tilde{m}$ we can consider bijections $\pi : \mathcal{Q} \to \tilde{\mathcal{Q}}$ from $\mathcal{Q}$ to $\tilde{\mathcal{Q}}$. The bottleneck distance between the persistence diagrams $\mathcal{P}_k\mathcal{L}(D_X^K)$ and $\mathcal{P}_k\tilde{\mathcal{L}}(D_Y^K)$ of networks D_X^K and D_Y^K is then defined as

$$b^k(D_X^K, D_Y^K) := \min_{\pi:\mathcal{Q}\to\tilde{\mathcal{Q}}}\ \max_{\mathbf{q}\in\mathcal{Q}} c(\mathbf{q}, \pi(\mathbf{q})). \tag{3.9}$$

The distance in (3.9) is set to the pair of features $\mathbf{q}$ and $\tilde{\mathbf{q}}$ that are most difficult to match across all possible bijections π.

When the number of points in the persistence diagrams are different we assume without loss of generality that $\tilde{m} < m$. In this case we extend $\tilde{\mathcal{Q}}$ by adding $\tilde{m} - m$ diagonal features to define the set $\tilde{\mathcal{Q}}_e := \tilde{\mathcal{Q}} \cup \{(\tilde{q}_i, \tilde{q}_i)\}_{i=1}^{\tilde{m}-m}$. Since the sets $\mathcal{Q}$ and $\tilde{\mathcal{Q}}_e$ contain the same number of elements bijections $\pi : \mathcal{Q} \to \tilde{\mathcal{Q}}$ are well defined. We can then modify (3.9) to define a valid comparison between $\mathcal{P}_k\mathcal{L}$ and $\mathcal{P}_k\tilde{\mathcal{L}}$ when the number of points in the diagrams are possibly different. We do so in the following formal definition.

Definition 17 *Given persistence diagrams $\mathcal{P}_k\mathcal{L}(D_X^K)$ and $\mathcal{P}_k\tilde{\mathcal{L}}(D_Y^K)$ with sets of points $\mathcal{Q}$ and $\tilde{\mathcal{Q}}$ having cardinalities $\tilde{m} < m$, define the extended set $\tilde{\mathcal{Q}}_e := \tilde{\mathcal{Q}} \cup \{(\tilde{q}_i, \tilde{q}_i)\}_{i=1}^{\tilde{m}-m}$ by adding $\tilde{m} - m$ artificial diagonal features. The bottleneck distance between the persistence diagrams $\mathcal{P}_k\mathcal{L}(D_X^K)$ and $\mathcal{P}_k\tilde{\mathcal{L}}(D_Y^K)$ of networks D_X^K and D_Y^K is defined as*

$$b^k(D_X^K, D_Y^K) := \min_{\pi:\mathcal{Q}\to\tilde{\mathcal{Q}}_e} \max_{\mathbf{q}\in\mathcal{Q}} c(\mathbf{q}, \pi(\mathbf{q})), \tag{3.10}$$

where π ranges over all bijections from $\mathcal{Q}$ to $\tilde{\mathcal{Q}}_e$ and the cost $c(\mathbf{q}, \pi(\mathbf{q}))$ is defined in (3.8).

The number of bijections between two sets of points is factorial and it appears that the problem as in (3.10) is as difficult as the problem of finding the correspondence in evaluating the network distance. However, the problem in (3.10) is a instantiation of the Linear Bottleneck Assignment Problem (LBAP) that can be solved efficiently – see Section 3.3 and [140, Algorithm 6.1]. We emphasize that $c(\mathbf{q}, \tilde{\mathbf{q}}) = c(\mathbf{q}, \tilde{\mathbf{q}}')$ for any $\mathbf{q}$ whenever $\tilde{\mathbf{q}}$ and $\tilde{\mathbf{q}}'$ are on the diagonal. Therefore, the locations of diagonal points added to construct $\mathcal{Q}_e$ are unsubstantial as per their cost. The definition of b^k has been used in [141] to compare three-dimensional surfaces using persistent homology.

We prove now that the bottleneck distance between the persistence diagrams of the filtrations induced by two dissimilarity networks is a lower bound of their dissimilarity network distance.

Theorem 5 *Let D_X^K and D_Y^K be two K-order dissimilarity networks. The bottleneck distance between the k-th dimensional persistence diagrams of the filtrations $\mathcal{L}(D_X^K)$ and $\mathcal{L}(D_Y^K)$ is at most $d_{\mathcal{D},\infty}(D_X^K, D_Y^K)$ for any $0 \leq k \leq K$, i.e.*

$$b^k(D_X^K, D_Y^K) \leq d_{\mathcal{D},\infty}(D_X^K, D_Y^K). \tag{3.11}$$

Proof: The proof of Theorem 5 relies on the stability theorem for persistence diagrams (also termed Interleaving Theorem) [142], which state that if the birth times of each simplex in two filtrations differ by no more than δ, the bottleneck distance between the corresponding persistence diagrams is no greater than δ as well. In order to prove Theorem 5, we introduce the following notion of augmented networks. This definition solves the issue when a single node in one network has multiple correspondents in the other network. In such cases, say, if a in X, has both c and d in Y as correspondents, we examine the difference between $r_X^1(a,a)$ and $r_Y^1(c,d)$ in evaluating the network difference. However $r_X^1(a,a) = r_X^0(a)$ represents the birth time of $[a]$ while $r_Y^1(c,d)$ denotes the

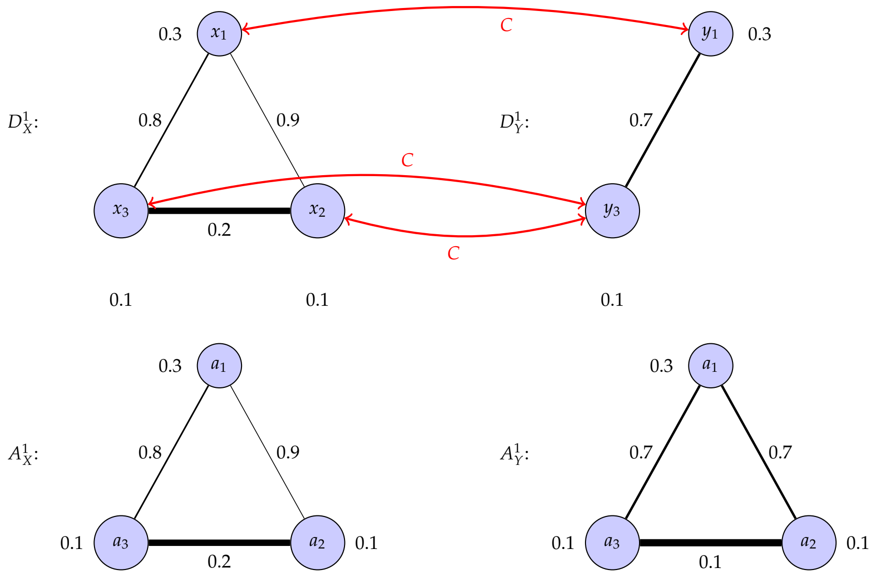


Figure 16: An example to construct augmented networks. D_X^1 and D_Y^1 are 1-order dissimilarity networks. The augmented network using correspondence C induces $A_X^1 \cong D_X^1$. An additional node a_2 and two edges are augmented in A_Y^1.

time of appearance of an edge $[c, d]$. The augmented networks are introduced to resolve this discrepancy.

Definition 18 *Given two K-order dissimilarity networks D_X^K and D_Y^K and a correspondence C between their node sets X and Y, the augmented networks $A_{X,C}^K$ and $A_{Y,C}^K$ are a pair of K-order networks defined on C. Each node a_i in $A_{X,C}^K$ and $A_{Y,C}^K$ represents a correspondent pair (x_{C_i}, y_{C_i}) in C. Relationship functions for $a_{0:k}$ with $0 \leq k \leq K$ are defined as*

$$r_{A_X}^k(a_{0:k}) = r_X^k(x_{C_0:C_k}), \;\; r_{A_Y}^k(a_{0:k}) = r_Y^k(y_{C_0:C_k}). \tag{3.12}$$

When the underlying correspondence is clear, C is omitted in the subscripts. For a pair of dissimilarity networks D_X^K and D_Y^K and a correspondence C between X and Y, the augmented networks A_X^K and A_Y^K have identical $|C|$ nodes where $|C|$ denotes the number of correspondence pairs in C. Each node a_i in both A_X^K and A_Y^K represents a correspondent pair (x_{C_i}, y_{C_i}) in C. For each tuple $a_{0:k}$, its relationship $r_{A_X}^k(a_{0:k})$ for the network A_X^K is the same as the relationship $r_X^k(x_{C_0:C_k})$ between the tuple $x_{C_0:C_k}$ in D_X^K. An example to construct augmented networks is illustrated in Figure 16. Augmented networks satisfy the following properties.

Fact 1 *Given a K-order dissimilarity network D_X^K and a correspondence C between the node sets X and Y, the augmented network A_X^K constructed by Definition 18 is a valid K-order network. The induced filtration*

$\mathcal{L}(A_X^K)$ is a valid filtration and has identical persistence diagrams as $\mathcal{L}(D_X^K)$, i.e. for any $0 \leq k \leq K$,

$$\mathcal{P}_k\mathcal{L}(A_X^K) = \mathcal{P}_k\mathcal{L}(D_X^K). \tag{3.13}$$

Proof: We need to prove the (i) symmetry property and (ii) identity property of high order networks, (iii) $\mathcal{L}(A_X^K)$ being a valid filtration, and (iv) $\mathcal{L}(A_X^K)$ and $\mathcal{L}(D_X^K)$ having identical persistent homology.

Proof of the symmetry property: For any tuples $a_{0:k}$, the symmetry property of dissimilarity network D_X^K and the definition of augmented networks imply $r_{A_X}^k(a_{[0:k]}) = r_X^k(x_{[C_0:C_k]}) = r_X^k(x_{C_0:C_k}) = r_{A_X}^k(a_{0:k})$ for any reordering $a_{[0:k]}$. This shows the symmetry property of A_X^K. ■

Proof of the identity property: Given a tuple $a_{0:k}$, if its subtuple $a_{l_0:l_{\tilde{k}}}$ have same set of unique elements as that of $a_{0:k}$, according to the Definition of augmented networks, we would have that $x_{C_0:C_k}$ and $x_{C_{l_0}:C_{l_{\tilde{k}}}}$ also possess identical set of unique elements. The identity property of dissimilarity network D_X^K and the definition of augmented networks yield $r_{A_X}^k(a_{0:k}) = r_X^k(x_{C_0:C_k}) = r_X^{\tilde{k}}(x_{C_{l_0}:C_{l_{\tilde{k}}}}) = r_{A_X}^{\tilde{k}}(a_{l_0:l_{\tilde{k}}})$. This shows the identity property of A_X^K. ■

Proof of $\mathcal{L}(A_X^K)$ being a valid filtration: The order increasing property of relationship function in augmented networks holds true due to the fact that

$$r_A^k(a_{0:k}) = r_X^k(x_{C_0:C_k}) \geq r_X^{k-1}(x_{C_0:C_{k-1}}) = r_A^{k-1}(a_{0:k-1}). \tag{3.14}$$

The remaining proof is identical to the proof of Proposition 5. ■

Proof of $\mathcal{L}(A_X^K)$ and $\mathcal{L}(D_X^K)$ having identical persistence diagrams: First, for each point $x \in X$, pick one pair (x, y) from the correspondence C to construct C_0 that is a subset of C. If we define a map that maps each pair (x, y) to its first element x, this gives a bijective projection from C_0 to X. Construct $A_{C_0,X}^K$ and $A_{C,X}^K$ as the augmented networks using the respective correspondence. It then follows naturally that $A_{C_0,X}^K$ is isomorphic to the original dissimilarity network D_X^K and so do the corresponding filtrations. Denote $\mathcal{L} := \mathcal{L}(A_{C,X}^K)$ and $\mathcal{L}_0 := \mathcal{L}(A_{C_0,X}^K) = \mathcal{L}(D_X^K)$.

Next, consider the projection of the filtration $\mathcal{L}$ onto $\mathcal{L}_0$, where each vertex (x, y) in C is mapped to (x, y_0) in C_0 who share the first element in the pair. Denote L^α and L_0^α as the simplicial complexes that collect simplices appearing before or prior to α in the respective filtration. For each value of α, the projection defines a retraction of simplicial complexes $L^\alpha \to L_0^\alpha$. Since the relationship functions are the same in the augmented network $A_{C,X}^K$ as in the original network $A_{C_0,X}^K$, whenever a simplex ϕ appears in L^α, not only its projection appears in L_0^α at the same time, but all the simplices connecting vertices of ϕ and vertices of its projection are also already in L_0^α. Hence, as a simplicial map, the projection $L^\alpha \to L_0^\alpha$ is contiguous to the identity of L^α. As a result, the section $L_0^\alpha \to L^\alpha$ is a homotopy equivalence. Its induced homomorphism at the homology level is therefore an isomorphism, and consequently the two persistence diagrams are isomorphic. ■

Having demonstrated all statements, the proof completes. ■

Back to the proof of Theorem 5. Let $\delta = d_{\mathcal{D},\infty}(D_X^K, D_Y^K)$. From Definitions 9, there exists a correspondence C between X and Y such that $|r_X^k(x_{0:k}) - r_Y^k(y_{0:k})| \leq \delta$ for any k and any pairs of correspondents $(x_{0:k}, y_{0:k}) = (x_0, y_0), \ldots, (x_k, y_k) \in C$. Construct the pair of augmented networks A_X^K and A_Y^K defined in Definition 18 using this correspondence. It follows from (3.12) that given a tuple $a_{0:k} \in C$ with each node a_i representing the correspondent pair (x_{C_i}, y_{C_i}) in C,

$$\left| r_{A_X}^k(a_{0:k}) - r_{A_Y}^k(a_{0:k}) \right| = \left| r_X^k(x_{C_0:C_k}) - r_Y^k(y_{C_0:C_k}) \right| \leq \delta. \tag{3.15}$$

Since A_X^K and A_Y^K have identical nodes, (3.15) implies that any simplices that appear at time α in the induced filtration $\mathcal{L}(A_X^K)$ will appear no earlier than $\alpha - \delta$ and no later than $\alpha + \delta$ in the induced filtration $\mathcal{L}(A_Y^K)$. Using the Interleaving Theorem, this yields the bound

$$b^k(A_X^K, A_Y^K) \leq \delta = d_{\mathcal{D},\infty}(D_X^K, D_Y^K). \tag{3.16}$$

From Fact 1, $b^k(A_X^K, A_Y^K) = b^k(D_X^K, D_Y^K)$. Substituting this into (3.16) concludes the proof. ■

When comparing high order networks via their induced persistence diagrams, Theorem 5 provides justification that the dissimilarity obtained from persistent homologies is a lower bound on their infinity norm network distance. In particular, Theorem 5 implies that: (i) A large difference in persistent homologies imply the networks to be highly different. (ii) Lower bounds can be used to estimate distance intervals because upper bounds are easy to determine using specific correspondences as per Definition 7.

We note that the differences between persistence diagrams do capture important differences between networks. E.g., in Figure 15 we can consider network (b) as a modification of network (a) in which we separate node a into the closely related nodes a and b that have a dissimilarity $r_X^1(a, b) = 0.1$. The persistence diagrams of these two networks are identical, which is consistent with the relative proximity of these two networks. Network (c) is more different, because the addition of the dissimilarity $r_X^1(b, d) = 0.7$ complicates the argument that nodes a and b can be well represented by a single node as in network (a). Network (d) can be argued to be in between networks (b) and (c) since the dissimilarity $r_X^1(a, b, d) = 0.8$ implies a sense of added proximity between nodes a, b, and d. The difference between the persistence diagrams of networks (c) and (b) is indeed larger than the difference between the diagrams of networks (d) and (b).

Further evidence for the usefulness of the bound in (3.11) follows from the fact that the bound is tight in some cases. An example of a tight bound is shown in Figure 17. The optimal correspondence C, shown in the figure, yields a network distance $d_{\mathcal{D},\infty}(D_X^1, D_Y^1) = 0.1$. The coordinates of the points in the 0-dimension persistence diagrams for $\mathcal{L}(D_X^1)$ are $(0, \infty), (0.12, 0.42), (0.2, 0.32)$ and for $\mathcal{L}(D_Y^1)$ are $(0.1, \infty), (0.21, 0.51), (0.25, 0.39)$. The coordinates of the point in the 1-dimension persistence diagrams for $\mathcal{L}(D_X^1)$ are $(0.6, 1)$ and for $\mathcal{L}(D_Y^1)$ is $(0.5, 1)$. The bottleneck distances be-

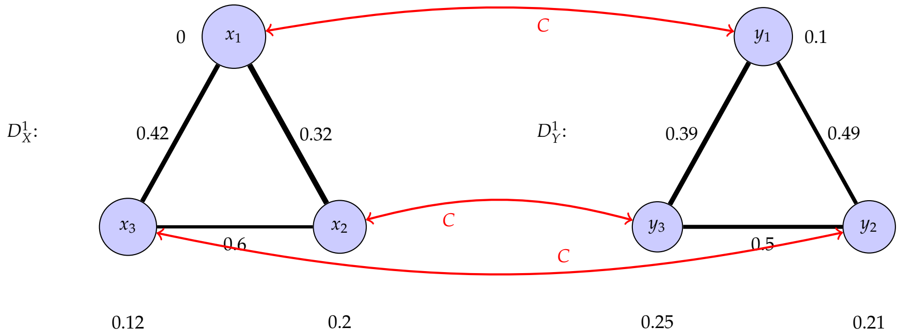


Figure 17: An example where the bottleneck distance between the k-th dimensional persistence diagrams of the filtrations $\mathcal{L}(D_X^K)$ and $\mathcal{L}(D_Y^K)$ is the same as their ∞-norm network distance for $k \in \{0, 1\}$. The optimal correspondence C yields $d_{\mathcal{D},\infty}(D_X^1, D_Y^1) = 0.1$. The points in the 0-dimension persistence diagrams for $\mathcal{L}(D_X^1)$ are with coordinates $(0, \infty), (0.12, 0.42), (0.2, 0.32)$ and for $\mathcal{L}(D_Y^1)$ are $(0.1, \infty), (0.21, 0.51), (0.25, 0.39)$. The point in the 1-dimension persistence diagrams for $\mathcal{L}(D_X^1)$ is with coordinate $(0.6, 1)$ and for $\mathcal{L}(D_Y^1)$ is $(0.5, 1)$. The bottleneck distances between the 0- as well as the 1- dimensional persistence diagrams of the filtrations induced from the two networks are 0.1.

tween the 0- and 1-dimensional persistence diagrams of the induced filtrations are 0.1; same as $d_{\mathcal{D},\infty}(D_X^K, D_Y^K)$.

3.2.1. *Persistence Bounds on k-order Distances*

Theorem 5 guarantees that the ∞-norm dissimilarity network distance $d_{\mathcal{D},\infty}$ can be tightly bounded by persistence diagrams. In this section we build on Theorem 5 to show that the k-order network distance $d_{\mathcal{D}}^k$ can also be bounded by persistent methods.

Theorem 6 *Given two dissimilarity networks D_X^K and D_Y^K and an integer $1 \leq k \leq K$, the bottleneck distance between the k'-th dimensional persistence diagrams of the filtrations $\mathcal{L}(D_X^K)$ and $\mathcal{L}(D_Y^K)$ is at most $d_{\mathcal{D}}^k(D_X^K, D_Y^K)$ for any $0 \leq k' < k$, i.e.*

$$b^{k'}(D_X^K, D_Y^K) \leq d_{\mathcal{D}}^k(D_X^K, D_Y^K). \tag{3.17}$$

Proof: We prove Theorem 6 from Theorem 5. To do that, we leverage the relationship for network distances as we present next.

Fact 2 *Given high order networks N_X^K and N_Y^K, we have*

$$d_{\mathcal{N}}^{k'}(N_X^K, N_Y^K) \leq d_{\mathcal{N}}^k(N_X^K, N_Y^K), \tag{3.18}$$

for $0 \leq k' \leq k-1 \leq K$, and

$$d_{\mathcal{N},\infty}(N_X^K, N_Y^K) = d_{\mathcal{N}}^K(N_X^K, N_Y^K). \tag{3.19}$$

Proof: We need to prove the two statements in (3.18) and (3.19). Given any $0 \le k' \le k \le K$, it follows from (2.8) that for any correspondence C between X and Y,

$$\Gamma^k_{X,Y}(C) = \max_{(x_{0:k}, y_{0:k}) \in C} \left| r^k_X(x_{0:k}) - r^k_Y(y_{0:k}) \right|, \tag{3.20}$$

$$\Gamma^{k'}_{X,Y}(C) = \max_{(x_{0:k'}, y_{0:k'}) \in C} \left| r^{k'}_X(x_{0:k'}) - r^{k'}_Y(y_{0:k'}) \right|. \tag{3.21}$$

For the $(x^\star_{0:k'}, y^\star_{0:k'}) \in C$ achieving the maximum difference $|r^{k'}_X(x^\star_{0:k'}) - r^{k'}_Y(y^\star_{0:k'})|$ in $\Gamma^{k'}_{X,Y}(C)$, we can construct another correspondent pair $(x^\star_{0:k}, y^\star_{0:k})$ such that $x^\star_{0:k'}$ is a sub-tuple of $x^\star_{0:k}$ with identical set of unique elements and $y^\star_{0:k'}$ is a sub-tuple of $y^\star_{0:k}$ with identical set of unique elements. It follows from the identity property of high order networks that $r^{k'}_X(x^\star_{0:k'}) = r^k_X(x^\star_{0:k})$ and $r^{k'}_Y(y^\star_{0:k'}) = r^k_Y(y^\star_{0:k})$. This implies that taking the maximum $|r^k_X(x_{0:k}) - r^k_Y(y_{0:k})|$ over $(x_{0:k}, y_{0:k}) \in C$ cannot yield a lower difference than $|r^{k'}_X(x^\star_{0:k'}) - r^{k'}_Y(y^\star_{0:k'})|$, i.e.

$$\Gamma^k_{X,Y}(C) \ge \left| r^{k'}_X(x^\star_{0:k'}) - r^{k'}_Y(y^\star_{0:k'}) \right| = \Gamma^{k'}_{X,Y}(C). \tag{3.22}$$

Since (3.22) holds true for any correspondence $C \in \mathcal{C}(X,Y)$, the inequality must hold true when we take the minimum over all correspondences $\mathcal{C}(X,Y)$ between X and Y,

$$\min_{C \in \mathcal{C}(X,Y)} \Gamma^k_{X,Y}(C) \ge \min_{C' \in \mathcal{C}(X,Y)} \Gamma^{k'}_{X,Y}(C'). \tag{3.23}$$

Substituting the definition of k-order and k'-order network distances into (3.23) yields $d^k_{\mathcal{D}} \ge d^{k'}_{\mathcal{D}}$, concluding the proof of (3.18).

Also, it follows from (2.17) that for any correspondence C between the node sets X and Y, the network difference between N^K_X and N^K_Y measured by C is

$$\left\| \mathbf{\Gamma}^K_{X,Y}(C) \right\|_\infty = \max_{k=0,1,\dots,K} \left\{ \Gamma^k_{X,Y}(C) \right\}. \tag{3.24}$$

From (3.22) we know that for any k', $\Gamma^k_{X,Y}(C) \ge \Gamma^{k'}_{X,Y}(C)$, and therefore for any correspondence C, $\max_{k=0}^K \left\{ \Gamma^k_{X,Y}(C) \right\} = \Gamma^K_{X,Y}(C)$. Substituting this into (3.24) and taking a minimum over all correspondences concludes the proof of (3.19).

Having proven the two statements, the proof completes. ■

Fact 2 implies that $d^k_{\mathcal{N}}$ increases as the order k becomes higher, and the ∞-norm distance $d_{\mathcal{N},\infty}$ is the same as the K-order network distance. The result in Theorem 6 only considers relationship functions of order k, and to leverage the connection as in Fact 2, we introduce the following notion of truncated networks.

Definition 19 *Given K-order network $N^K_X = (X, r^0_X, \dots, r^K_X)$, its k-order truncated network N^k_X is defined as $N^k_X = (X, r^0_X, \dots, r^k_X)$.*

A k-order truncated network N_X^k has the same node set as its parent network N_X^K and collects the lowest $k+1$ order relationship functions of N_X^K. Back to the proof of Theorem 6, construct the k-order truncated networks D_X^k and D_Y^k from D_X^K and D_Y^K. It follows directly from Theorem 5 that for any $0 \leq k' \leq k-1$,

$$b^{k'}(D_X^K, D_Y^K) \leq d_{\mathcal{D},\infty}(D_X^k, D_Y^k). \tag{3.25}$$

Since D_X^k and D_Y^k are valid high order networks, Fact 2 implies that $d_{\mathcal{D},\infty}(D_X^k, D_Y^k) = d_{\mathcal{D}}^k(D_X^k, D_Y^k)$. Meanwhile, $d_{\mathcal{D}}^k(D_X^k, D_Y^k) = d_{\mathcal{D}}^k(D_X^K, D_Y^K)$ follows from the facts that D_X^k and D_X^K have identical k-order relationship functions and D_Y^k and D_Y^K have same k-order relationship. Finally, for any $k' \leq k-1$, the k'-th dimensional persistence diagram of $\mathcal{L}(D_X^k)$ is identical to that of $\mathcal{L}(D_X^K)$ and the k'-th dimensional persistence diagram of $\mathcal{L}(D_Y^k)$ is identical to that of $\mathcal{L}(D_Y^K)$, therefore $b^{k'}(D_X^k, D_Y^k) = b^{k'}(D_X^K, D_Y^K)$. Combining these observations, the proof concludes. ■

Theorem 6 bounds the k-order network distance by the bottleneck distance between persistence diagrams of any order $k' < k$. We note that best lower bound is not necessarily $k' = k-1$. Observe that the result in Theorem 6 does not apply for $k = 0$. This is not a problem because the 0-order network distances $d_{\mathcal{D}}^0$ examine relationships between individual nodes only. Thus, we can compute the optimal correspondence in Definition 7 by matching each of the relationships $r_X^0(x)$ in the network $\mathcal{D}_X^K$ to the closest relationship $r_Y^0(y)$ in the network $\mathcal{D}_X^K$. We emphasize that the lower bound described in Theorem 6 is also tight since we can find dissimilarity networks D_X^K and D_Y^K such that $d_{\mathcal{D}}^k(D_X^K, D_Y^K)$ equals the bottleneck distance between the k'-th dimensional persistence diagrams of the filtration $\mathcal{L}(D_X^k)$ and $\mathcal{L}(D_Y^k)$ for any $0 \leq k' \leq k-1 \leq K$. See Figure 17 for an illustration where bottleneck distances between the k-dimensional persistence diagrams coincide with $d_{\mathcal{D}}^k$.

One of the key arguments in the proof of Theorem 2 is the fact that the infinity norm network distance $d_{\mathcal{N},\infty}(N_X^K, N_Y^K)$ and the K-order network distance $d_{\mathcal{N}}^K(N_X^K, N_Y^K)$ coincide [cf. Fact 2]. This equality implies that Theorem 5 follows as a particular case of Theorem 6 by setting $k = K$ in (3.17) and observing that $d_{\mathcal{N},\infty}(N_X^K, N_Y^K) = d_{\mathcal{N}}^K(N_X^K, N_Y^K)$. Do notice that this doesn't mean that the proof of Theorem 5 is redundant because Theorem 5 is leveraged in the proof of Theorem 6. However, the result does imply that the infinity norm distance $d_{\mathcal{N},\infty}(N_X^K, N_Y^K)$ does not contain information beside the one that is contained in the collection of k-order distance $d_{\mathcal{N}}^k(N_X^K, N_Y^K)$ for $k = 0, \ldots, K$.

Another consequence of Theorem 6 is that we can bound the p-norm network distance $d_{\mathcal{D},p}(D_X^K, D_Y^K)$ [cf. Definition 9] for arbitrary p. This can be done using the individual bounds for $d_{\mathcal{D}}^k(D_X^K, D_Y^K)$ established in (3.17) as we show next.

Corollary 1 *Group the k-order dissimilarity network distances in the vector* $\mathbf{d}_{\mathcal{D}}^K(D_X^K, D_Y^K) := \big[d_{\mathcal{D}}^0(D_X^K, D_Y^K), d_{\mathcal{D}}^1(D_X^K, D_Y^K), \ldots, d_{\mathcal{D}}^K(D_X^K, D_Y^K)\big]^T$. *Further define the vector* $\mathbf{b}^K(D_X^K, D_Y^K) := \big[d_{\mathcal{D}}^0(D_X^K, D_Y^K), b^{k_1}(D_X^K, D_Y^K), \ldots, b^{k_K}(D_X^K, D_Y^K)\big]^T$ *whose first component is the zero order distance* $d_{\mathcal{D}}^0(D_X^K, D_Y^K)$ *and whose other components are bottleneck lower bounds with* $k_l < l$. *The p-norm network distance can be lower bounded* *as*

$$\left\| \mathbf{b}^K(D_X^K, D_Y^K) \right\|_p \leq \left\| \mathbf{d}_{\mathcal{D}}^K(D_X^K, D_Y^K) \right\|_p \leq d_{\mathcal{D},p}(D_X^K, D_Y^K). \tag{3.26}$$

Proof: The second inequality follows from the fact that a single correspondence is used in $d_{\mathcal{D},p}$ whereas an order-specific correspondence may be utilized in each k-order distance $d_{\mathcal{D}}^k$ [8]. For the first inequality, $b^{k_l}(D_X^K, D_Y^K) \leq d_{\mathcal{D}}^l(D_X^K, D_Y^K)$ comes from Theorem (6). This implies that the vector $\mathbf{b}^K(D_X^K, D_Y^K)$ is element-wise smaller or equal to the vector $\mathbf{d}_{\mathcal{D}}^K(D_X^K, D_Y^K)$. ■

Corollary 1 shows that the p-norm network distance $d_{\mathcal{D},p}$ can be lower bounded using: (i) The exact value of the 0-order network distance. (ii) The persistence homology lower bounds in (3.17) of Theorem 6. The bounds in Corollary 1 are expected to be the most useful when they combine information gleaned from persistence diagrams of different orders.

Remark 6 The requirement of having the inequality in (2.30) be strict unless all the elements of $x_{0:k}$ appear in $x_{0:k-1}$ for the network to be a dissimilarity network [cf. Definition 10] is not always naturally satisfied in practice. As we have seen, its satisfaction may necessitate the addition of a small but arbitrary constant to some tuple relationships [cf. Figure 4]. This is not a problem in practice because Theorems 5 and 6 still holds if this technical condition is violated. The only modification in the statements is that the respective network distances are not metrics but pseudometrics. What this means is that it is possible to construct pathological examples of networks that satisfy the order decreasing property *not* strictly and that have null network distance while *not* being isomorphic. These networks are unlikely to appear in practice and, even if they do, the bounds in Theorems 5 and 6 hold, albeit with a different interpretation.

3.3. Implementation Details

We discuss implementation issues regarding the procedure for evaluating lower bounds, its computational cost, and some heuristic simplifications.

Procedure. To compute distance lower bounds for a dissimilarity network we construct filtrations and evaluate persistence diagrams. Algorithmic procedures to compute filtrations and persistence diagrams are discussed in [37, 143]. The bottleneck distances $b^k(D_X^K, D_Y^K)$ between the persistence diagrams of different orders $0 \leq k \leq K$ are then evaluated. These distances follow from Definition 17 and are found as solutions of LBAP with costs given by (3.8). If the interest is in the infinity norm network distances $d_{\mathcal{D},\infty}$, any of the bottlenecks distances is a lower bound [cf. Theorem 5]. If we are interested in the k-order network distance $d_{\mathcal{D}}^k$, the bottleneck distance between persistence diagrams of dimension $k' \leq k-1$ is a lower bound [cf. Theorem 6]. When given a proximity network we construct the dual network with relationships $r_X^k(x_{0:k}) = 1 - r_X^k(x_{0:k})$ and proceed as before.

Computational cost. The process above involves computation of the persistent homology and evaluation of the bottleneck distance. Given a K-order network with n nodes, denote as $\tilde{n}$ the total number of simplices and m as the number of points in the persistence diagram. The complexity of computing a persistence diagram is akin to matrix multiplication of matrices with size $\tilde{n}$, i.e. the number of simplices considered [37]. Because the complexity for matrix completion is $O(\tilde{n}^{2.373})$ [143], computational complexity of evaluating a persistence diagram is also $O(\tilde{n}^{2.373})$. The complexity of evaluating the bottleneck distance using LBAP [cf. (3.10)] is of order $O(m^{2.5}\log m)$ [140, Algorithm 6.1]. The number of simplices of order k can be as large n^k and the number of points in a persistence diagram as large as the number of simplices. This means that we can have $m = \tilde{n} = n^K$, which yields impractical computational times for orders larger than $K = 2$. In the practical examples we consider in Sections 3.4.2 and 3.4.3, the number of simplices $\tilde{n}$ is much smaller than the maximum possible and scales linearly with the number of nodes. This yields complexity $O(n^{2.373})$ for computing the persistence diagram and $O(n^{2.5}\log n)$ for the bottleneck distance.

Undefined tuple relationships. Formally, dissimilarity networks require that all undefined tuple relationships take value 1. As defined above this is impractical because it would scale the total number of simplices as $\tilde{n} = n^k$. In computations we leave these simplices undefined and let features die at infinity. As it follows from Remark 5, the death time of these features can be set to $\alpha = 1$ to yield the same result that would be obtained from setting undefined tuple relationships to 1 before computing the persistence diagram.

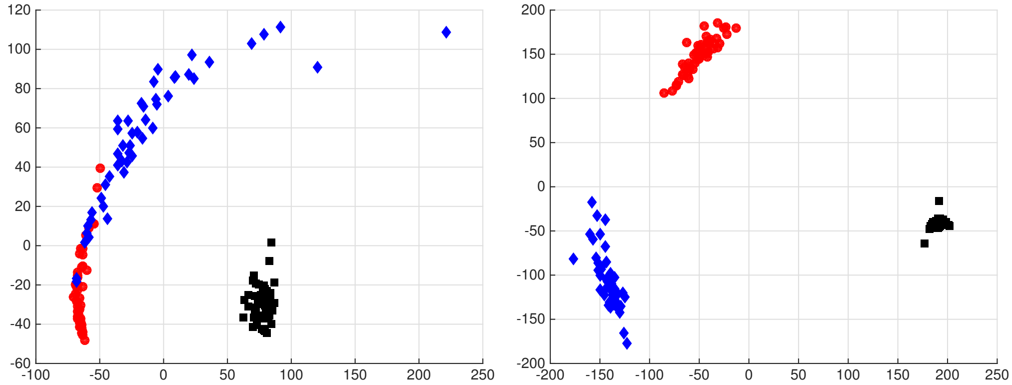
Elimination of small homological features. We can reduce the number of points in a persistence diagram by removing points $\mathbf{q}$ close to the diagonal. This is justified because: (i) These points represent ephemeral homological features and are likely to be generated by noise in observations. (ii) They contribute a small value to the cost in (3.8) and won't improve the lower bound much from the trivial nonnegative lower bounds $0 \leq d_{\mathcal{D},\infty}(D_X^K, D_Y^K)$ or $0 \leq d_{\mathcal{D}}^k(D_X^K, D_Y^K)$ that hold for any distance.

3.4. Applications

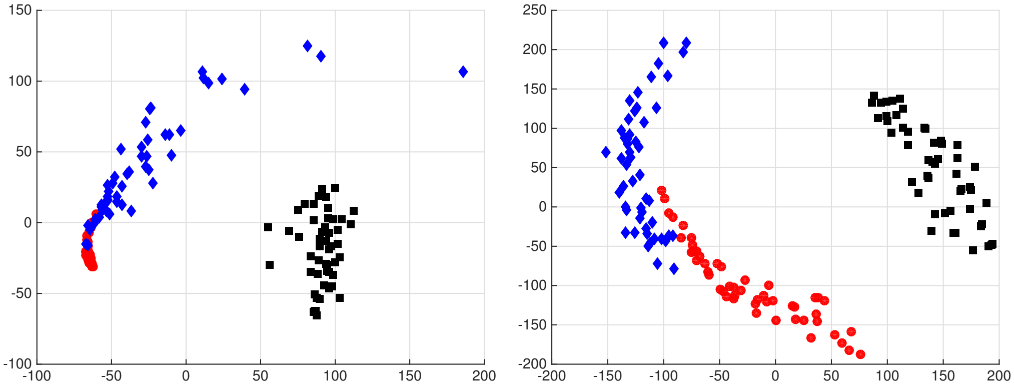
We illustrate the usefulness of homology methods through experiments in both synthetic (Section 3.4.1) and real-world data (Sections 3.4.2 and 3.4.3). The objective is to demonstrate that networks with similar structures and should be alike to each other are indeed similar in their persistent homologies.

3.4.1. Classification of Synthetic Networks

We consider three types of synthetic weighted pairwise networks. Edge weights in all three types of networks encode proximities. The first type of networks are with weighted Erdős-Rényi model [144], where the edge weight between any pair of nodes is a random number uniformly selected from the unit interval $[0, 1]$. In the second type of networks, the coordinates of the vertices



(a) b^0, 30 nodes (b) b^1, 30 nodes

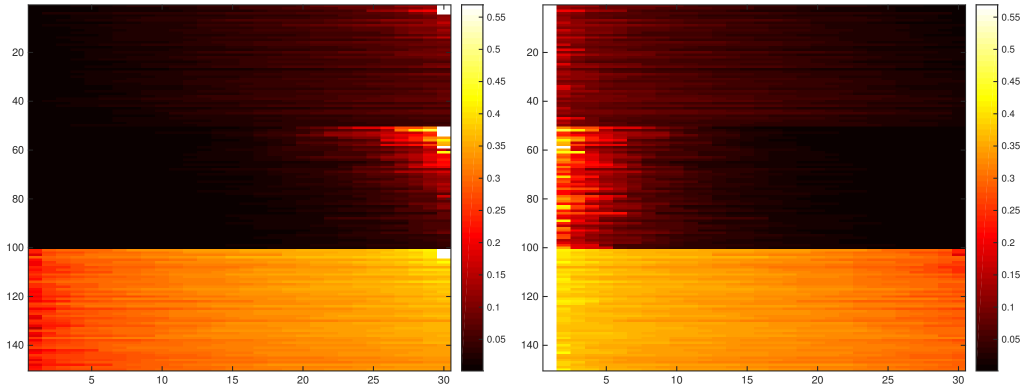


(c) b^0, 30 to 39 nodes (d) b^1, 30 to 39 nodes

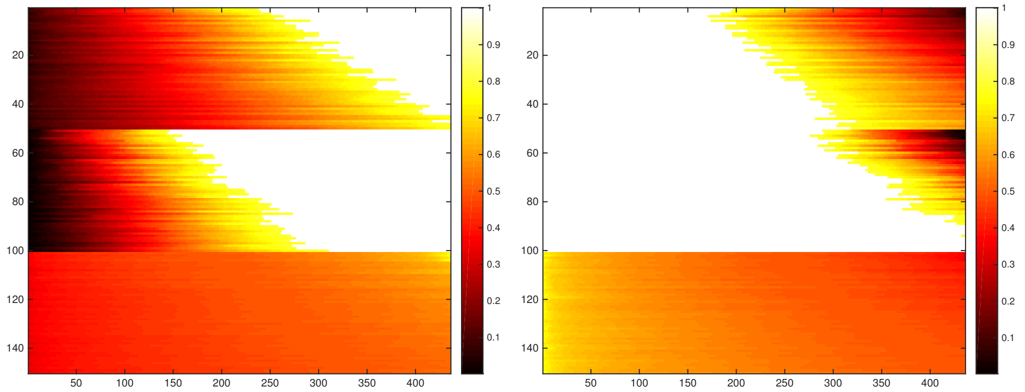
Figure 18: Two dimensional Euclidean embeddings of the networks constructed from three different models with different number of nodes with respect to the network metric lower bounds b^0 and b^1. In the embeddings, red circles denote networks constructed from the Erdős-Rényi model, blue diamonds represent networks constructed from the unit circle model, and black squares the networks from the correlation model.

are generated uniformly and randomly in the unit circle, and the edge weights are evaluated with the Gaussian radial basis function $\exp(-d(i,j)^2/2\sigma^2)$ where $d(i,j)$ is the distance between vertices i and j in the unit circle and σ is a kernel width parameter. In all simulations, we set σ to 0.5. The edge weight measures the proximity between the pair of vertices and takes value in the unit interval. In the third type of networks, we consider that each vertex i represents an underlying feature $\mathbf{u}_i \in \mathbb{R}^d$ of dimension d, and examine the Pearson's linear correlation coefficient ρ_{ij} between the corresponding features $\mathbf{u}_i$ and $\mathbf{u}_j$ for a given pair of nodes i and j. The weight for the edge connecting the pair is then set as $\rho_{ij}/2 + 0.5$, a proximity measure in the unit interval. The feature space dimension d is set as 30 in all simulations.

We start with networks of equal size $|X| = 30$ and construct 50 random networks for each aforementioned type. The edges with weights no greater than a threshold τ are removed to create sparsity. We set $\tau = 0.2$ in all simulations. In order to transform the constructed pairwise networks into high order proximity networks, the 0-order proximity for any node is set to 0, i.e.



(a) death time in $\mathcal{P}_0\mathcal{L}$, ascending order (b) death time in $\mathcal{P}_0\mathcal{L}$, descending order



(c) birth time in $\mathcal{P}_1\mathcal{L}$, ascending order (d) birth time in $\mathcal{P}_1\mathcal{L}$, descending order

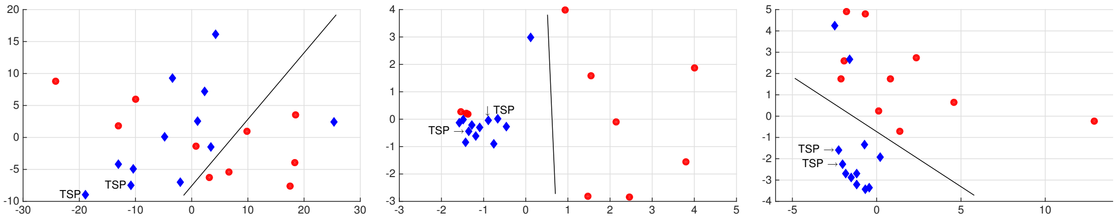
Figure 19: Illustrations of the set of death time in 0-dimensional persistence diagrams $\mathcal{P}_0\mathcal{L}$ and the set of birth time in 1-dimensional persistence diagrams $\mathcal{P}_1\mathcal{L}$ for the filtrations constructed from the randomly generated networks with number of nodes ranging between 30 and 39. The top 50 networks represent those constructed from the Erdős-Rényi model, the middle 50 networks are with the unit circle model, and the bottom 50 are with the correlation model.

$r_X^0(x) := 0$ for any $x \in X$. We then use the persistent homology method described in Section 3.3 to evaluate the dissimilarities between all networks. The persistent homologies are computed using JavaPlex [145]. Figure 18 (a) and (b) plot the two dimensional Euclidean embeddings [146] of the network metric lower bounds b^0 and b^1 between the 0- and 1-dimensional persistence diagrams respectively. Networks constructed with different models form clear separate clusters with respect to b^1 where networks with Erdős-Rényi model are denoted by red circles, networks with unit circle model are described by blue diamonds, and correlation model represented as black squares. The clustering structure is not that clear in terms of b^0 but the dissimilarities between networks constructed from different models are in general much higher than that between networks from the same model, and an unsupervised classification with one linear boundary in the embedded space would yield 10 out of 150 errors (6.7%).

Next we consider networks with number of nodes ranging between 30 and 39. Ten networks are randomly generated for each network type and each number of nodes, resulting in 150 networks in total. Figure 18 (c) and (d) illustrate the two dimensional Euclidean embeddings of the network metric lower bounds b^0 and b^1 for these networks. Despite the fact that networks with same model have different number of nodes, dissimilarities between persistent homologies are smaller when their underlying networks are from the same process. Besides, networks with the correlation model are highly similar to each other regarding their corresponding persistent homologies, irrespective of the sizes of the networks nor the dimension of persistence diagrams. An unsupervised classification with one linear boundary would yield 6 out of 150 errors (4.0%) for b^0 and 7 errors (4.7%) for b^1. The performance of b^0 stays relatively unchanged when the number of nodes in the networks considered reside in a range of nodes. The simulations illustrate the applicability of persistent homology in identifying the patterns of networks of different processes. Similar results are obtained when different parameters are used in generating the networks.

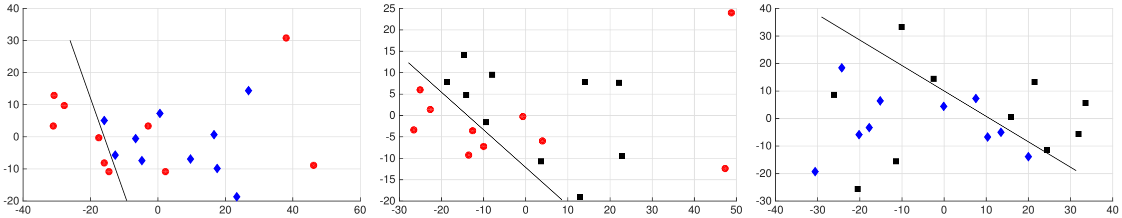
Here we give interpretations of why persistent homologies succeed in network discrimination. Since only pairwise relationships are examined, the bottleneck distance b^0 is determined by the death time of the 0-dimensional persistent homologies. If we consider that nodes connected by an edge of weight w become members of the same community at time w, the death time in $\mathcal{P}_0\mathcal{L}$ can be interpreted as the time instant when isolated nodes join the main community of the network. We focus our attention on the 150 networks with size ranging between 30 and 39. Each row in Figure 19 (a) represents a network, and plots the death time in the 0-dimensional persistence diagram $\mathcal{P}_0\mathcal{L}$ of the filtration induced from that network in ascending order from left to right. The top 50 networks represent those with Erdős-Rényi model, the middle 50 are with the unit circle model, and the bottom 50 are with the correlation model. Each row in Figure 19 (b) plots the death time in $\mathcal{P}_0\mathcal{L}$, in descending order from left to right, of the induced filtration. It is not a direct mirror of Figure 19 (a) because the size of the networks ranges from 30 to 39 and so the number of points in the persistence diagrams are different. The time instants when isolated nodes join the main community in the networks with the correlation model are concentrated in the interval $[0.25, 0.4]$. This is due to the fact that the linear correlation coefficient between two randomly generated feature vectors cannot be too positive nor too negative. Also, networks with unit circle model are different from those constructed with the Erdős-Rényi model because in the latter case, the distribution of death time of points in $\mathcal{P}_0\mathcal{L}$ has heavy tail towards 1, which results from the fact that some points in the unit circle may be far away from the main component and it requires larger distance for them to join the main component.

For higher order persistent homologies, b^1 is only affected by the birth time of the 1-dimensional persistent homologies, which can be interpreted as the maximum of the three pairwise connections between three nodes and therefore the time instant when a 'closely-connected' community is formed by the three nodes. Each row in Figure 19 (c) and (d) exhibits the birth time in $\mathcal{P}_1\mathcal{L}$, in descending and ascending order from left to right respectively, of the filtration induced from the corresponding network. The time instants when three nodes in the networks with correlation model form a closely connected triplet are highly focused in the interval of $[0.4, 0.6]$. Recall that edge weight is defined as $\rho_{ij}/2 + 0.5$, therefore a edge weight in the interval $[0.4, 0.6]$ implies very weak correlation ($|\rho_{ij}| \leq 0.1$). This is due to the fact that when the correlation coefficient ρ_{ij}



(a) Embedding of lower bound $b^k(\mathcal{P}_0\mathcal{L})$ (b) Embedding of lower bound $b^k(\mathcal{P}_1\mathcal{L})$ (c) Embedding of lower bound $b^k(\mathcal{P}_2\mathcal{L})$

Figure 20: Two dimensional Euclidean embeddings of the networks constructed from quinquennial publications in engineering and mathematics journals with respect to the network metric lower bounds $b^k(\mathcal{P}_0\mathcal{L})$, $b^k(\mathcal{P}_1\mathcal{L})$, and $b^k(\mathcal{P}_2\mathcal{L})$. In the embeddings, red circles denote networks constructed from mathematics journals and blue diamonds represent networks from engineering journals. Networks constructed from publications of TSP are labeled.



(a) TAC - TSP (b) TAC - TWC (c) TSP - TWC

Figure 21: Two dimensional Euclidean embeddings of the networks constructed from annual publications in TAC, TSP, and TWC with respect to the summation of the metric lower bounds $b^k(\mathcal{P}_0\mathcal{L})$, $b^k(\mathcal{P}_1\mathcal{L})$, and $b^k(\mathcal{P}_2\mathcal{L})$. In the embeddings, red circles represent TAC, blue diamonds TSP, and black squares TWC.

between features $\mathbf{u}_i$ and $\mathbf{u}_j$ are small and ρ_{ik} between $\mathbf{u}_i$ and $\mathbf{u}_k$ is small, ρ_{jk} between $\mathbf{u}_j$ and $\mathbf{u}_k$ also tends to be small. For networks with unit circle model, the time points for many nodes in the network forming a closely connected triplet are either smaller or larger compared to the Erdős-Rényi model. This is because if three points in the unit circle are close to each other, all the pairwise distances between them would not be high; otherwise the minimum inscribed circle of the three points would possess large radius, resulting in a high value on the maximum of the pairwise distances. Admittedly, other methods to compare networks may also succeed in distinguishing networks, after some proper treatment towards the issue of different sizes. Nonetheless, persistent homology method would be more universal, not only for the reason that it establishes a lower bound to the actual network metrics, but also since it provides a systematic way to analyze the formation of communities in a given network.

3.4.2. *Comparison of Coauthorship Networks*

We apply the lower bounds to compare 2-order coauthorship networks where relationship functions denote the number of publications of single authors, pairs of authors, and triplets. These coauthorship networks are proximity networks because they satisfy the order decreasing property. We consider publications in 5 journals from mathematics community: Computational Geometry

(CG), Discrete Computational Geometry (DCG), J. of Applied Probability, (JAP) J. of Mathematical Analysis and Applications (JMAA), SIAM J. on Numerical Analysis (SJNA), and 6 journals from engineering community, all from IEEE: Signal Processing Magazine (SPM), Trans. Automatic Control (TAC), Trans. Pattern Analysis and Machine Intelligence (TPAMI), Trans. Information Theory, Trans. Signal Processing (TSP), Trans. Wireless Communication (TWC). For each journal, we construct networks for the 2004-2008 and 2009-2013 quinquennia. For TAC, TSP, and TWC, we also construct networks for each annual from 2004 to 2013. Lists of publications are queried from [135].

For each of these journals we consider all publications in the period of interest and construct proximity networks where the node set X is formed by all authors of the publications. Zeroth order proximities are defined as the total number of publications of each member of the network, first order proximities as the number of publications coauthored by pairs, and second order proximities as the number of publications coauthored by triplets. To make networks with different numbers of publications comparable we normalize all relationships by the total number of publications in the network. The positive constant ϵ as in Figure 5 is set to $1/1000$. There are publications with more than three coauthors but we don't record proximities of order higher than 2. By assuming that networks from the same community or constructed from the same journal have similar collaboration patterns, we show here that network metric lower bounds succeed in identifying these patterns and in distinguishing coauthorship networks from communities with different interests.

Figure 20 shows the two dimensional Euclidean embeddings of the network metric lower bounds b^0, b^1 and b^2. The 12 engineering networks (blue diamonds) separate clearly from the 10 mathematics networks (red circles) in b^1 and b^2. The clustering is not that clear in b^0 but still networks from same community tend to be similar to each other. An unsupervised classification with one linear boundary running across the embeddings would generate errors of 2 (9.09%) to 5 (22.73%) out of 22 networks. Networks constructed from the same journal tend to be close in the lower bounds. As an example, the networks of TSP with different quinquennia are marked in the embeddings and it is clear that their differences in homologies are considerably low. Such scenarios are observed for several other journals as well.

We analyze the persistent homology of each of the coauthorship networks to investigate the reason why persistent homologies succeed in network discrimination. Compared to networks from mathematics communities, networks from engineering communities in general would yield 0-dimensional persistent homologies with smaller birth time but larger death time, 1-dimensional homologies with larger birth and death time, and 2-dimensional homologies with larger birth time. An interpretation of such observations would be that in engineering, there exist more small communities that never collaborate with each other and it is uncommon to have a "club" of 3 to 5 authors in engineering that a strong collaboration exists between any pairs of the authors in the "club"; such scenarios are absent for mathematicians.

As a comparison, we applied some simple and reasonable methods to compare the coauthorship networks considered in this section. Motifs have been shown effective in distinguishing coauthorship networks from different scientific fields [16]. To compare high order coauthorship networks by motifs, we restrict attention to pairwise relationships. The dissimilarities between coauthorship networks are assigned as the differences between the summations of the weighted motifs in their corresponding pairwise networks. Analysis based on triangle motifs (weighted) results in a clear cluster between networks from CG, DCG, JAP and another cluster between networks from TSP, TWC, but cannot distinguish other networks very well. Tetrahedron motif analysis (weighted) results in three clear clusters: networks from CG, DCG, JAP, networks from JMAA, SJNA, SPM, TPAMI, and networks from TSP, TWC. Other simple and common methods to compare pairwise networks yield similar results. Methods to compare networks via features give us similar observations as those based on the persistent methods; feature methods would generate 6 to 8 errors in classifications.

3.4.3. Engineering Communities with Different Research Interests

The network metric lower bounds succeed in distinguishing the different collaboration patterns in engineering and mathematics communities. We now illustrate that the lower bounds are also able to identify distinctive features of engineering communities with different research interests. To see this we consider the networks constructed from annual publications of TAC, TSP, and TWC.

Figure 21 shows the two dimensional Euclidean embeddings of the networks with respect to the summation of the lower bounds b^0, b^1, and b^2. We expect more variations in annual networks because the time for averaging behavior is reduced. Besides, it is hard to argue that intrinsic and obvious differences exist in the collaboration patterns in automatic control, signal processing, and wireless communication communities. Still, networks constructed from the same journal but different annuals tend to be close to each other and form clustering structures. An unsupervised classification with one linear boundary in the embeddings run across the summation of lower bounds would generate 4 (20%) errors out of 20 networks in all three classification problems considered. The less obvious clustering structure formed by networks from different journals in Figure 21 (c) compared to (a) and (b) also suggests that the collaboration patterns in research communities of signal processing and wireless communication are more similar compared to that of automatic control.

Chapter 4

Frequency Representation of Networks by Persistent Homology

Leveraging on the association between networks and persistent homology in Section 3, in this chapter we propose to define *homofrequency* of tuples in the network as the duration of the homological feature generated by the tuples (Definition 21 in Section 4.1). Tuples forming a long-lasting homological feature – a high homofrequency – represent a core feature in the network that should not be considered as noise. On the other hand, tuples that generate a short-lasting homological feature – a low homofrequency – denote an unimportant feature in the network that are more likely to be noise. The *homospectrum* of a network collects all of its homofrequencies and the corresponding homological generators. This definition of homospectrums enable us to easily distinguish networks generated from different processes (Section 4.1.1). The first key theoretical result is that we can recover a network from its homospectrum (Definition 22 and Theorem 7). In other words, homospectrums offer a different representation of the same information represented in the network space. The dual space, formed by homospectrums and networks, behaves like frequency domain and time domain in classical signal processing. With such definition of homofrequencies, we can then define *homofilters* to remove undesired homofrequencies and keep the homofrequencies of our interest (Section 4.2). The filtering can be applied in the homospectrum domain (Definition 23) as well as directly in the network domain (Definition 24). We illustrate that as expected, h-lowpass filter enables us to remove the unimportant features and keep the core representation in a point cloud data (Section 4.2.1). Another important aspect about defining frequency via homological features is that the difference between the original network and the filtered network using h-highpass filter is small (Theorem 8 in Section 4.3). In fact, such difference is no larger than the duration of the longest-lasting homological features removed during filtering. In time-series filtering, Parseval's theorem in Fourier transform guarantees that the energy in the filtered signal is mostly preserved, if we only remove frequencies with small absolute frequency coefficients. Theorem 8 has similar interpretation, with the duration of the homological features

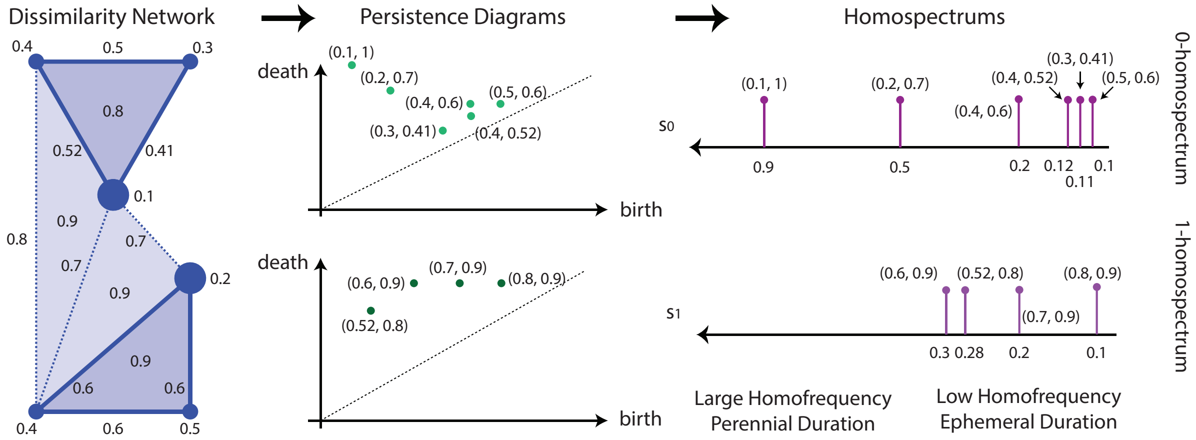


Figure 22: From dissimilarity networks to homospectrums through persistence diagrams. Persistence diagrams are constructed from the dissimilarity networks, with generators for each homological feature being recorded. For each order k, homofrequency s_k denotes the duration of the homological feature. The aggregation of homofrequencies for all homological features forms the homospectrum.

denote the "energy". Besides, if we apply filter onto a pair of networks, the difference between the original network distance and the distance between the filtered networks is no greater than the longest duration removed (Theorem 9).

4.1. Homospectrum of Networks

The appearance and disappearance of homological features induce a natural notion of duration, i.e. the difference between the disappearance time of a homological feature and the corresponding appearance time of the feature. This observation motivates us to represent the unstructured information encoded by the network by the more organized information of durations of homological features. In classical signal processing, a time-series can be represented by coefficients in Fourier frequency domain, where low frequencies denote small temporal variation and high frequencies represent large variation. In homological signal processing, a network can be represented by its *homospectrums*, in which high homofrequencies indicate perennial homological features and high homofrequencies represent ephemeral homological features.

To formalize such intuition, we need to slightly adapt the definition of persistent homology as in classical topology. The reason for this is because the birth and death time of persistent homology only capture the relationship values of the nodes, but do not capture which nodes are included in the relationship. For that purpose, we include the generators of each homological feature as we formally state the next.

Definition 20 *Given a filtration L, its k-dimensional extended persistence diagram Q_k is a collection of points of the form $\mathbf{q} = [q_b, q_d]$, each associated with a tuple $(\mathcal{G}_b, \mathcal{G}_d)$ where q_b and $q_d > q_b$ represent the birth and death time of a homological feature, $\mathcal{G}_b = \{x_{0:k}\}$ denotes the collection of simplices that exhibit the birth of the homological feature, and $\mathcal{G}_d = x_{0:k+1}$ denotes the simplex which kills the homological feature.*

As per Definition 20, for each point in the persistence diagram, the time q_b represents the resolution $\alpha = q_b$ at which the k-homological features generated by the simplices in the collection $\mathcal{G}_b$ start to appear, and the time q_d describes resolution $\alpha = q_d$ at which the feature is no longer a homological feature due to the addition of the simplex $\mathcal{G}_d = x_{0:k+1}$. For the same example as in Figure 14, there are still two points, $[0.5, 1]$ and $[0.7, 0.8]$ in the 1-persistence diagram, but each of them is associated with a tuple denoting their generators. In specific, the point $[0.5, 1]$ is associated with $(\{[a, b], [b, c], [c, d], [d, a]\}, [b, c, d])$ denoting its birth at resolution 0.5 is due to the cycle formed by $[a, b]$, $[b, c]$, $[c, d]$, and $[d, a]$ and its death at resolution 1 is due to the simplex $[b, c, d]$. In fact, the actual death generator in this case is inessential as it represents the maximum dissimilarity, i.e. the simplex actually does not exist. Similarly, the point $[0.7, 0.8]$ is associated with $(\{[a, b], [b, d], [d, a\}, [a, b, d])$. Compared to persistence diagram, extended diagram also records the generators for each of the homological features.

With the extended persistence diagram, we would be able to define the homofrequencies as the difference between the death time and birth time of each homological feature, as we state the next.

Definition 21 *Given a point* $\mathbf{q}$ *in the k-dimensional extended persistence diagram* Q_k*, its homofrequency* $s_k(\mathbf{q})$ *is* $q_d - q_b$*. The k-homospectrum* $S_k = \{s_k\}$ *collects the homofrequency of all k-dimensional homological feature. The K-order homospectra* $S^K = \{S_k\}_{k=0}^{K-1}$ *collects all k-homospectrums up to order* $K - 1$*. Denote the space of all K-order homospectras as* $\mathcal{S}^K$ *and the h-transform from networks to homospectrums as* $\mathcal{Z} : \tilde{\mathcal{D}}^K \to \mathcal{S}^K$*.*

The homofrequency for a certain homological feature represents how short the homological feature exists in the filtration. The lower the homofrequency, the shorter the duration where the homological feature exists. On the other hand, the higher the homofrequency, the longer the existence. As for networks representing the temporal dynamics for the formation of a research community as in Figure 4, a homological feature denoting high 1-homofrequency describes a number of authors, who each pair started collaboration a while ago but only recently start to collaborate together or never collaborate together; in other words, it describes a “hole” in the community that last a relatively long time. On the other hand, a homological feature denoting low 1-homofrequency describes a group of researchers, who start to write publications together shortly after their pairwise initial collaboration; in other words, it describes a “hole” in the community that gets filled very soon. It can be seen that a network up to order K can have $K + 1$ different notions of homofrequency: starting from 0-homofrequency, describing when different communities join together, then 1-homofrequency, quantifying holes within communities. An illustrating example describing how we construct the homofrequency of networks, via persistence diagram, is demonstrated in Figure 22. Inherited from the extended persistence diagram, each homofrequency records the associated generator tuple. This enables the recovery of the original network from the homospectrums, as we state next.

Definition 22 *Given a homospectra* $S^K = \{S_k\}_{k=0}^{K-1}$*, its corresponding dissimilarity network* D_X^K *is recovered as follows. Starting with* F_0*, for each homofrequency, build the dissimilarity in* D_X^K *using the birth and* *death time $[q_b, q_d]$ as well as the generators $(\mathcal{G}_b, \mathcal{G}_d)$ recorded, where the dissimilarities for all simplices in $\mathcal{G}_b$ not defined yet are set to q_b, and the dissimilarity for the simplex in $\mathcal{G}_d$ is set to q_d. Iterate this process over all s_0 and then iterate over all S_k up to $K-1$. If a tuple $x_{0:k}$ appears in the birth generators of multiple k-homological features, its dissimilarity $d^k_X(x_{0:k})$ is set as the smallest of the birth times of the features.*

The inverse h-spectrum from homospectrums to networks is denoted as $\mathcal{Z}^{-1} : \mathcal{S}^K \to \tilde{\mathcal{D}}^K$.

The network is recovered in a bottom-up fashion. For each homological feature, there is only one simplex in the death generator $\mathcal{G}_d$, and the dissimilarity for that simplex is set to the death time of the homological feature. Each homological feature may have multiple simplices in the birth generator $\mathcal{G}_b$, and those that have not been defined are set to the birth time of the homological feature. The condition for a tuple appearing in multiple birth generators ensures order increasing property to be satisfied. The main result in this section is that it is able to prove that $\mathcal{Z}^{-1}$ is well defined for dissimilarity networks, as we state next.

Theorem 7 *The transform $\mathcal{Z}^{-1}$ defined in Definition 22 is a well-defined inverse map of $\mathcal{Z}$ defined in Definition 21, i.e. for any dissimilarity network $D^K_X \in \mathcal{D}^K$, we have $\mathcal{Z}^{-1} \circ \mathcal{Z}(D^K_X) = D^K_X$.*

Proof: To prove Theorem 7, we first demonstrate that all tuples with recorded dissimilarity exist in the generators of extended persistence diagram, as we state next.

Lemma 1 *Given a dissimilarity network D^K_X, any of its tuples with recorded dissimilarity and with unique elements appear either in the death generator $\mathcal{G}_d$ of a $(k-1)$-th dimensional homological feature or the birth generator $\mathcal{G}_b$ of the k-th dimensional features.*

Proof: Given any tuple $x_{0:k}$ with unique nodes, Definition 10 indicates that the k-simplex ϕ^k defined by the convex hull $\operatorname{conv}\{x_{0:k}\}$ appears strictly after any of its faces $\operatorname{conv}\{x_{0:\hat{s}:k}\}$ in the filtration. Suppose ϕ^k appears at time α and denote $\partial_k \phi^k = \sum_i \beta_i \psi^{k-1}_i$ with β_i the coefficients, then each ψ^{k-1}_i appears strictly before time α.

Now suppose that the appearance of ϕ^k trivializes a $(k-1)$-th dimensional homological feature. This means that ϕ^k is the boundary to trivialize the $(k-1)$-th dimensional cycle $\partial_k \phi^k$. Since each face ψ^{k-1}_i of ϕ^k appears strictly before time α, the cycle $\partial_k \phi^k$ results in a homological feature. The death time of this homological feature is α, or equivalently, the time represented by the relationship $r^k_X(x_{0:k})$. This indicates that $x_{0:k}$ is the death generator of the homological feature.

On the other hand, if the appearance of ϕ^k does not trivialize a $(k-1)$-th dimensional homological feature, then the $(k-1)$-cycle $\partial_k \phi^k$ is in the collection of simplices appearing before or on time α. This means that $\partial_k \phi^k$ can be represented by a sum of the boundaries of some k-chains Φ^k_i,

$$\partial_k \phi^k = \sum_i \beta_i \partial_k \Phi^k_i \tag{4.1}$$

with coefficients β_i and k-chains Φ_i^k appearing before or on time α. By the definition of k-chains, $\Phi_i = \sum_j \beta_j' \psi_j^k$ with coefficients β_j' and k-simplices ψ_j^k appears before or on time α. Therefore, (4.1) can be written as $\partial_k \phi^k = \sum_j \beta_j'' \partial_k \psi_j^k$. Rearranging terms,

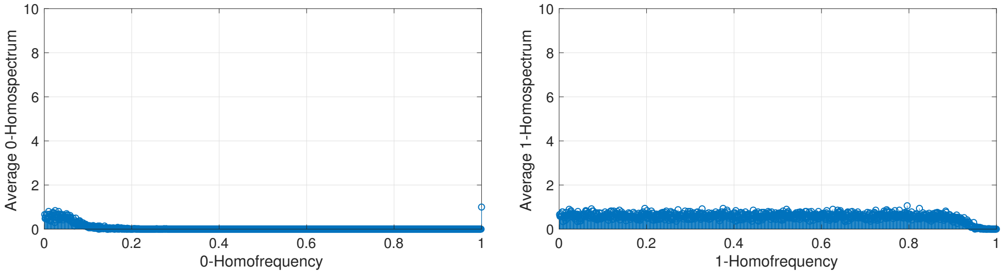
$$\partial_k \Big(\sum_j \beta_j'' \psi_j^k - \phi^k \Big) = 0. \tag{4.2}$$

This implies that $\sum_i \beta_i'' \psi_i^k - \phi^k$ is a k-cycle. There must be a new cycle formed since ϕ^k just appears. The cycle cannot be trivialized immediately since any $(k+1)$-chain Ψ^{k+1} with $\partial_{k+1}\Psi^{k+1} = \sum_i \beta_i'' \psi_i^k - \phi^k$ would involve a simplex $[x_{0:k,l}]$ for some node x_l with tuple $x_{0:k,l}$ consisted of non-repeating elements where this simplex $[x_{0:k,l}]$ appears strictly after α. Therefore we have a k-th dimensional homological feature with birth time α, or equivalently, the time denoted by the relationship $r_X^k(x_{0:k})$. Consequently, $x_{0:k}$ is in the birth generator of the homological feature formed.

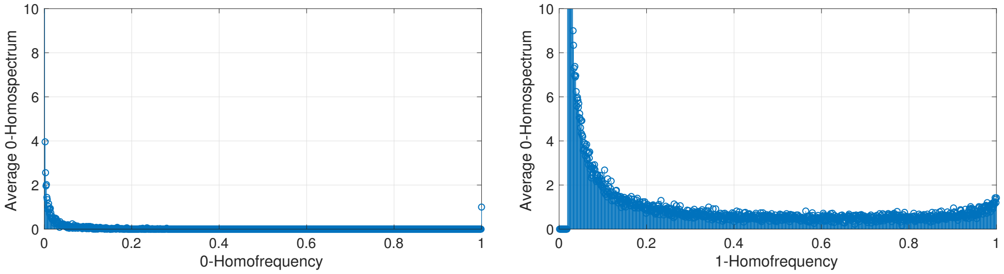
Finally, combining the observations with the fact that the death and birth generators of all homological features are recorded in the extended persistence diagram concludes the proof. ■

Back to the proof of Theorem 7, since network homofrequency also records the birth and death generators, all non-trivial tuples can be found in the birth and/or the death generators of network homofrequency as well. We reconstruct the network starting from the 0-homofrequency. Since the birth generators of 0-persistent homology is just a 0-simplex, i.e. a single node, and from Lemma 1, all 0-simplices can be found in generators, we could perfectly recover all single nodes and their corresponding dissimilarities in the form of $d_X^0(x)$. Besides dissimilarities of all single nodes coming from the birth generators of 0-persistent homology, the death generators of 0-persistent homology unveils information about some of the edges. Consequently, we could recover some edges and their corresponding dissimilarities in the form of $d_X^1(x, x')$. Only examining the generators of 0-persistent homology could not recover all edges. Nonetheless, from Lemma 1, all 1-simplices can be found in generators. Those edges do not appear in the death generators of 0-persistent homology must appear in the birth generators of some 1-persistent homological features. As a result, we could recover the remaining edges uncovered during investigating 0-persistent homology, and their corresponding dissimilarities. After examining the death generators of 0-persistent homological features and and the birth generators of 1-persistent homological features, we will recover all edges and their corresponding dissimilarities. Following the same approach, after examining the death generators of 1-persistent homological features and and the birth generators of 2-persistent homological features, we will perfectly recover all triangles and their corresponding dissimilarities in the form of $d_X^2(x, x', x'')$. Iterating by increasing order of homological features would eventually recover all dissimilarities defined in the original network. ■

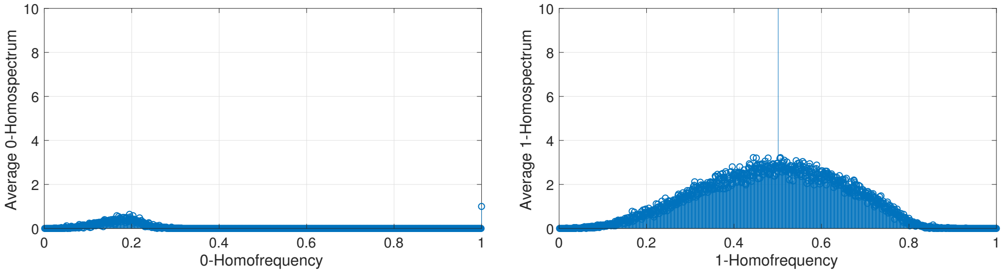
The result as in Theorem 7 justifies the definition of homofrequency and homospectrum, i.e. a network could be equally represented in the network domain as well as in homospectrum. Compared to networks which depend on the underlying node space and labelling, homospectrum provides a universal description and also has the advantage that there is an implied ordering



(a) 50 nodes, Erdős-Rényi model (d) 50 nodes, Erdős-Rényi model



(b) 50 nodes, unit circle model (e) 50 nodes, unit circle model



(c) 50 nodes, correlation model (f) 0 nodes, correlation model

Figure 23: The average homospectrums of the networks constructed from three different models with same number of nodes. For each network model, we generate 50 random networks with 50 nodes. We evaluate the 0-frequency and 1-frequency for all networks, where homofrequency coefficients are computed up to resolution of 0.001. The average of such homofrequency coefficients across all networks is examined and visualized. Networks generated from different models exhibit highly distinguishable patterns in their homospectrums.

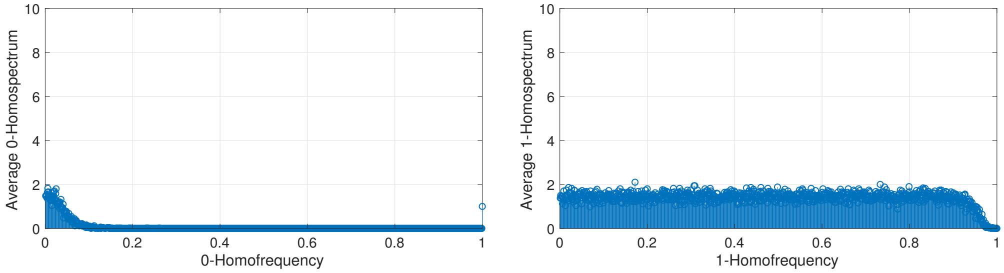
of frequencies. Theorem 7 only applies for dissimilarity networks, because simplices in relaxed dissimilarity network may generate homological features that are killed in the same time, resulting in information lost in the homospectrums. We will illustrate how the homospectrum looks like for exemplifying networks as next.

4.1.1. *Case Study: Homospectrums of Exemplifying Networks*

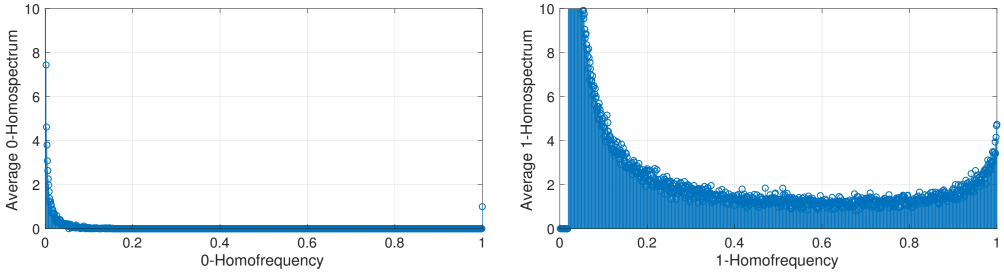
In this section, we illustrate the homospectrums of networks generated from difference processes. To that end we consider: (i) Weighted Erdős-Rényi random networks [144] with connection probability $p = 0.5$ and edge weights random and uniformly chosen from the unit interval $[0, 1]$. (ii) Random geometric networks where nodes are placed at random in the unit circle and edge weights are of the form $\exp(-d(i,j)^2/2\sigma^2)$, where $d(i,j)$ is the distance between vertices i and j in the unit circle and ϕ^2 is a kernel width parameter. We set $\sigma^2 = 0.5$. (iii) Random feature networks where edge weights are determined by the Pearson correlation coefficient ρ_{ij} between a pair of corresponding features $\mathbf{u}_i, \mathbf{u}_j \in \mathbb{R}^d$. Features are randomly chosen standard white Gaussian vectors in a space of dimension $d = 10$ and edge weights are chosen as $d_X(i,j) = (1 + \rho_{ij})/2$. Observe that in all three cases edge weights measure the relationship between pairs of vertices and take values in the unit interval. In order to transform the constructed pairwise networks into high order dissimilarity networks, the 0-order relationship for any node is set to 0, i.e. $d_X^0(x) := 0$ for any $x \in X$.

We start by considering random networks with same number of nodes. For each network model, we generate 50 random networks with 50 nodes. For each network, we evaluate its 0-homospectrum and 1-homospectrum according to Definition 21. In theory, homospectrum is a collection of points in the unit interval. In practice, we break the interval $[0,1]$ into τ equally-sized bins, and count the number of homological features with duration in that bin. This yields homofrequency coefficients up to resolution of $1/\tau$. We used $\tau = 1000$ in this chapter. We then evaluate the average of such homofrequency coefficients across all networks, illustrated in Figure 23. It can be seen that networks generated from different models exhibit highly distinguishable patterns in their homospectrums. For 0-homofrequency, most 0-frequency for networks generated from Erdős-Rényi models lie in the interval between $[0, 0.15]$. Networks generated from unit the circle model tend to have more short-lived homofrequency, as it is more likely for nodes to represent points that are very close on the unit circle and form highly proximate relationships. Networks generated from correlation model, on the other hand, have most of their 0-frequency in the interval $[0.1, 0.3]$; this is likely because edge weights denote correlations between features and therefore tend to be neither too high nor too low. For 1-frequency, Erdős-Rényi networks yield an almost flat homospectrum, likely due to the fact that edge weights are uniformly distributed in the unit interval. Networks generated from the unit circle model have more short-lived homofrequency and some long-lived homofrequency. This is likely due to the reason that the points on the unit circle can form clusters that are either very close or very far. The 1-frequency for networks generated from correlation model forms a bell shape. The main takeaway message here is that networks constructed from different models yield highly distinguishable patterns in both of their 0-frequency spectrum and 1-frequency spectrum. Consequently, network homofrequency captures distinguishing features from the underlying networks.

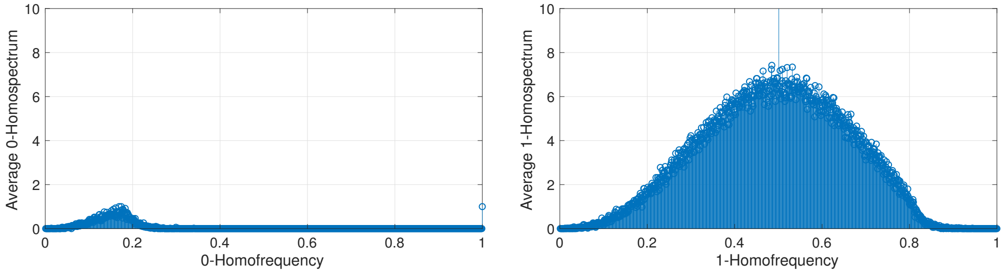
Next we consider networks constructed according to the models with different number of nodes. For each network model, we generate 50 random networks with the number of nodes ranging from 51 to 100. Similar as in the previous case, we evaluate the 0-homofrequency and 1-homofrequency with homofrequency coefficients computed up to resolution of 0.001. We compute the average homospectrum across networks generated from the same model – although different number of nodes – and illustrate them in Figure 24. Compared to networks with same number of nodes as in Figure 23, homofrequency coefficients here tend to be larger and more varied. This is unexpected as we consider different number of nodes and more nodes. Still, networks generated from different



(a) 50 to 100 nodes, Erdős-Rényi model (d) 50 to 100 nodes, Erdős-Rényi model



(b) 50 to 100 nodes, unit circle model (e) 50 to 100 nodes, unit circle model



(c) 50 to 100 nodes, correlation model (f) 50 to 100 nodes, correlation model

Figure 24: The average homospectrums of the networks constructed from three different models with different number of nodes. For each network model, we generate 50 random networks with the number of nodes ranging from 51 to 100. We evaluate the 0-frequency and 1-frequency for all networks, where homofrequency coefficients are computed up to resolution of 0.001. Compared to networks with same number of nodes as in Figure 23, homofrequency coefficients here tend to be larger and more varied, due to the fact that different number of nodes and more nodes are considered here. Still, networks generated from different models yield very distinguishable homospectrums, and such pattern is robust with the number of nodes in the networks.

models yield very distinguishable homospectrum, and such pattern is not altered by the number of nodes in the networks. We emphasize that this observation also indicates that networks – defined potentially on different number of nodes – have underlying features preserved in their homospectrums – defined always on the interval $[0, 1]$. Therefore, when we study networks of different sizes, instead of examining their nodal domains which are not comparable, we can study their homospectrum, which provides a universal notion to compare networks of arbitrary sizes.

$$\begin{tikzcd}
\tilde{\mathcal{D}}^K \arrow[rr, "\mathcal{T}"] \arrow[dr, dashed, "f"'] & & \mathcal{S}^K \arrow[dr, "\mathbf{F}"] & \\
& \tilde{\mathcal{D}}^K & & \mathcal{S}^K \arrow[ll, "\mathcal{T}^{-1}"]
\end{tikzcd}$$

Figure 25: Filter f on the space of dissimilarity networks is constructed by first establishing the homospectrums via $\mathcal{Z}$, then applying filtering $\mathbf{F}$ on the homospectrums, and finally recovering to the network space via $\mathcal{Z}^{-1}$.

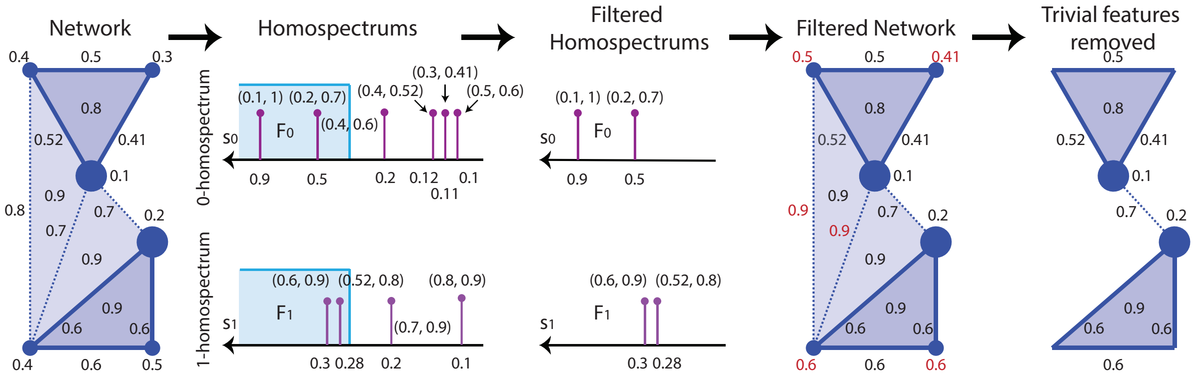


Figure 26: The resulting network filtering applied towards the dissimilarity network in Figure 22 with $F_0(s_0) = 0$ for any $s_0 \leq 2/9$ and $F_0(s_0) = 1$ otherwise and $F_1 = F_0$. In filtered homospectrums, lasting homological features with duration greater than $2/9$ are kept; ephemeral homological features with duration smaller or equal to $2/9$ are all removed. In the network space, this corresponds to remove everything except the two nodes with small dissimilarities on them respectively as well as the triangle which generates a 1-order homological feature at $5/9$.

4.2. Filtering of Networks

Since homospectrums enable frequencies to be defined for networks, it is natural to explore if we can define operations on the frequency domain so that important features can be extracted while unimportant characteristics removed. We do so through network filtering, and we define such operation first in the homospectrum space as next.

Definition 23 *Given a homospectra* $S^K = \{S_k\}_{k=0}^{K-1}$, *a filter in the homospectrum space* $\mathbf{F} : \mathcal{S}^K \to \mathcal{S}^K$ *is a collection of transfer function is a mapping* $\mathbf{F} = [F_0, F_1, \ldots, F_{K-1}]$ *where each transfer function* $F_k : [0,1] \to \{0,1\}$, *such that in the filtered homospectrum* $F_k(S_k)$, *the birth time* q_b *of the homological features with homofrequency* s_k *satisfying* $F_k(s_k) = 0$ *are set to the corresponding death time* q_d.

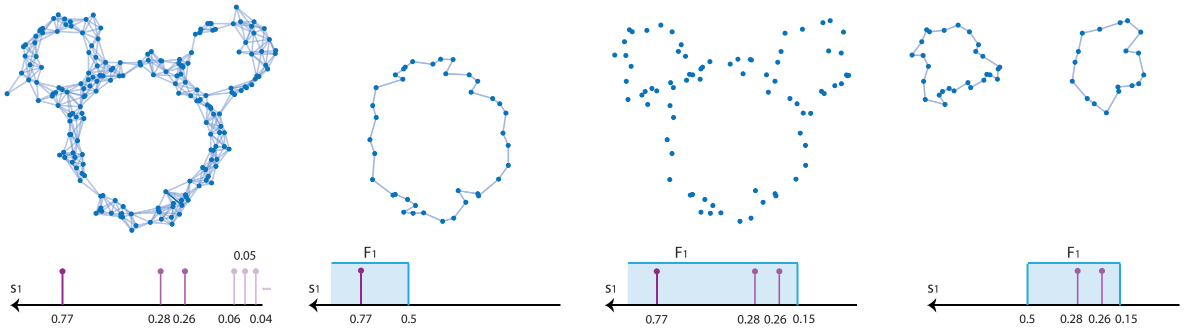
Because each F^K collects $K-1$ homospectrums, filtering applied upon F^K has $K-1$ different transfer functions, each F_k corresponding to the one to be applied to the k-homospectrum F_k. Unlike conventional filtering in signal processing where the transfer function can be of arbitrary shape, the output of the transfer function F_k is binary. As a result, $F_k(s_k) = 1$ implies the homological feature with homofrequency s_k will be preserved. On the other hand, $F_k(s_k) = 0$ implies the homological feature with the homofrequency s_k will die the same time as the homological feature gets generated. Note that despite the homological feature has duration 0, having the generators and the birth and death time of the feature enables the corresponding tuples to be recovered. Using the map $\mathcal{Z}$ and its inverse map $\mathcal{Z}^{-1}$, we can define the filtering applied onto networks directly.

Definition 24 *Given a relaxed dissimilarity network* D_X^K *and a filter in the homospectrum space* $\mathbf{F}$, *the filtered network is* $f(D_X^K) = \mathcal{Z}^{-1} \circ \mathbf{F} \circ \mathcal{Z}(D_X^K)$, *where maps* $\mathcal{Z}$ *and* $\mathcal{Z}^{-1}$ *are defined in Definitions 21 and 22, respectively.*

Networks can be filtered directly in the node space by f via the homospectrums and the pre-specified transfer functions $\mathbf{F}$, as illustrated in Figure 25. We emphasize that the transfer functions $\mathbf{F}$ are agnostic to the node set X to be specified and can be set without any knowledge about the network. An illustrating example of network filtering applied towards the dissimilarity network in Figure 22 with $F_0(s_0) = 0$ for any $s_0 \leq 0.25$ and $F_0(s_0) = 1$ otherwise and $F_1 = F_0$ is shown in Figure 26. In the filtered homospectrums, homological features with duration greater than 0.25 are kept; ephemeral homological features are removed. In the network space, this corresponds to set the dissimilarities of the four periphery nodes to the smaller of the edges joining each of them, and to set the dissimilarity of the edges between top and bottom parts – except the one joining nodes with dissimilarities 0.1 and 0.2 – to the dissimilarity of their enclosing triangles. What remains unchanged can be interpreted as the core features of the original network: (i) two nodes with small dissimilarities on themselves but large dissimilarity between them, and (ii) two communities each composed of three nodes with relatively large difference between their pairwise dissimilarities and the dissimilarity between the three of them. Other information in the original network can be treated as unimportant detailed features and are removed in the filtering process.

Still, the filtered network has all tuples in the original network. We could remove unimportant tuple $x_{0:k}$ if (i) it does not appear in the $(k-1)$-order death generators of any homological features, and (ii) the k-order birth generators in which it appears get their birth time set to the death time of the respective homological features. These two conditions imply that the homological features in which the tuple contribute were removed in the network filtering process as in Definition 23. The result of such trivial-feature-removal applied onto the filtered network in the previous case is presented as the rightmost presentation in Figure 26, where the remaining structure leaves only the two nodes and the two triangles.

Remark 7 The extended persistent diagram and the recording of generators in the homospectrums are for network filtering. None of these are required in the sole presentation of homospectrums. Therefore, if we are not interested in applying network filtering, homospectrums can be generated via persistent diagrams P_k without recording the generators.



(a) Point cloud (b) Homo-highpass filtered 1 (c) Homo-highpass filtered 2 (d) Homo-bandpass filtered

Figure 27: Homofrequency filters can be applied towards the point cloud to extract the points contributing to the most important feature of the point cloud. (a) Top: Initial point cloud. Bottom: 1-homospectrum of the dissimilarity network formed from the point cloud. (b) Resultant point cloud and 1-homospectrum after filtering with a h-highpass filter with cut-off homofrequency 0.5. (c) Point cloud and 1-homospectrum after filtering with a h-highpass filter with cut-off homofrequency 0.15. (d) Point cloud and 1-homospectrum after a h-bandpass filter.

4.2.1. Case Study: Filtering of Exemplifying Networks

In this section, we illustrate the network filtering upon a network representing distance between point clouds. Given X as a point cloud in a plane, for a tuple $x_{0:k} \in X$, its relationship $d_X^k(x_{0:k})$ represents the smallest d such that the $k+1$ circles centered at points $x_0, \ldots, x_k$ with radius $d/2$ have pairwise intersections between all pairs. Note that it does require all $k+1$ circles have a common intersection. This is the so-called Vietoris-Rips complex in algebraic topology [147, 148]. We then normalize relationships such that they lie in the unit interval.

As the example illustrated in Figure 27 (a), the point cloud forms the shape of a Mickey Mouse face, consisted of a large circle at the bottom and two smaller circles at the top, and many points around the face. The 1-homospectrum of the point cloud, as presented in the lower part of Figure 27 (a), composes of five 1-homological features. The generators for the features are plotted on the original point cloud with different colors denoting homofrequencies: three of the circles correspond to the circles of the Mickey Mouse face, two of the circles denote lower homofrequencies. Each point in the point cloud will generate a 0-homofrequency, and therefore we do not present 0-homospetrum for simplicity. Recall 1-homofrequency indicates the length of an 1-homological feature in the filtration, and a higher homofrequency implies that the homological feature exists for a longer duration. It can be seen that the more important 1-homological features in this example are the three circles form the Mickey Mouse face.

If we apply a *h-highpass filter* $\mathbf{F}$ with $F_0(s_0) = 0$ for all s_0, $F_1(s_1) = 1$ if $s_1 \geq 0.5$, and $F_1(s_1) = 0$ otherwise, we will only keep the most lasting homofrequency and result in only the large bottom circle in the original point cloud, as in Figure 27 (b). In this example, we remove points that contribute to trivial homological features in all orders. Notice h-highpass filter preserves the core feature and removes noise. Similarly, If we apply a h-highpass filter $\mathbf{F}'$ with $F_0'(s_0) = 0$ for all s_0, $F_1'(s_1) = 1$ if $s_1 \geq 0.15$, and $F_1'(s_1) = 0$ otherwise, we will remove the unimportant 1-homological features and nodes not contributing to the lasting 1-homofrequencies. This yields the point cloud which forms the Mickey Mouse face, as in Figure 27 (c). Finally, we can also apply a *h-bandpass filter* $\mathbf{F}''$ with $F_0''(s_0) = 0$ for all s_0, $F_1''(s_1) = 1$ if $0.15 \leq s_1 \leq 0.5$, and $F_1''(s_1) = 0$ otherwise, to screen out points contributing to the two smaller circles consisting the ears of the Mickey Mouse, as in Figure 27 (d). The results are robust to the exact cut-off homofrequencies applied as well as perturbation to the original point cloud. The takeaway message from the case study is that the network filtering developed in Definitions 23 and 24 can be used in practical scenarios to extract the features of our interest, by using different types of homofrequency filter and choosing different cut-off homofrequencies.

4.3. Stability of Network Filtering

We have been considering the h-highpass filter to filter out features in the network that are unimportant while preserves the core features. In this section, we formalize such intuition by specifying that the difference between the original network and the filtered network is small. To do that, we need a measure of closeness of networks. In Section 2, we have defined a valid metric in the space of networks. Our main result in this section is that the network distance in terms of $d_{\mathcal{D}}^k$ between the original network and the filtered network defined in Definition 24 is small for all k, as we state next.

Theorem 8 *Given a dissimilarity network* D_X^K*, the filtered network* $f(D_X^K)$ *using the filtering* $\mathbf{F}$ *on the homospectrums with* $F_k(s_k) = 1$ *for any* $s_k \geq \delta$ *and any* $0 \leq k \leq K-1$ *satisfies*

$$d_{\mathcal{D}}^k(D_X^K, f(D_X^K)) \leq \delta, \tag{4.3}$$

for any $0 \leq k \leq K-1$.

Proof: Denote the filtered network $f(D_X^K)$ as the tuple $(X', d_{X'}^0, \dots, d_{X'}^K)$ and the homospectrum of $D_{X'}^K$ as S_K'. We use $\mathbf{q} = [q_b, q_d]$ to denote birth and death time of homological features of D_X^K, and $\tilde{\mathbf{q}} = [\tilde{q}_b, \tilde{q}_d]$ to denote birth and death time of $D_{X'}^K$. Since $X' = X$, It suffices to show that

$$|d_{X'}^k(x_{0:k}) - d_X^k(x_{0:k})| \leq \delta \tag{4.4}$$

for any $x_{0:k}$. Once that is established, we can construct a specific correspondence $C^* = \{(x, x) \mid x \in X\}$, and demonstrate $\Gamma_{X,X'}^k(C^*) \leq \delta$ for any k. The specific C^* may not be the optimal correspondence, but it yields $d_{\mathcal{D}}^k(D_X^K, D_{X'}^K) \leq \Gamma_{X,X'}^k(C^*) \leq \delta$ for any k and completes the proof.

To show (4.4), consider an arbitrary tuple $x_{0:k}$. From Lemma 1, it will appear in the birth generators of some $k-1$-homological features or the death generator of a k-homological feature in the homospectrum S_K. The filtering applied will not remove any simplices from any homological feature, and as a result $x_{0:k}$ will continue to appear in the birth generators of some $k-1$-homological features or the death generator of a k-homological feature in the homospectrum S_K'. First suppose $x_{0:k}$ appears in the death generator of a k-homological feature with death time q_d. One tuple can only kill one homological feature, so there will be only one such q_d. Filtering applied will not change the death time of the feature. Moreover, the recovery step from S'_K to $D^K_{X'}$ will just construct $d^k_{X'}(x_{0:k}) = q_d$, and therefore $d^k_{X'}(x_{0:k}) = d^k_X(x_{0:k})$ satisfies the condition.

Now consider that $x_{0:k}$ appears in the birth generators of some $k-1$-homological features. In the recovery step from S'_K to $D^K_{X'}$, if the dissimilarity $d^k_{X'}(x_{0:k})$ has been set, then $x_{0:k}$ must also appear in the death generator of some k-homological feature. This means turns the situation back to the case considered before and will yield $d^k_{X'}(x_{0:k}) = d^k_X(x_{0:k})$. What remains to show are scenarios where $x_{0:k}$ appears in the birth generators of some $k-1$-homological features and does not appear in the death generators of any k-homological features. Without loss of generality, suppose $x_{0:k}$ appears in the birth generator $\mathcal{G}_b$ corresponding to a $k-1$-homological feature with birth time q_b and death time q_d, as well as the birth generator $\mathcal{G}'_b$ corresponding to a $k-1$-homological feature with birth time q'_b and death time q'_d, and nothing else. This implies the dissimilarity of the tuple can be written as

$$d^k_X(x_{0:k}) = \min\{q_b, q'_b\}. \tag{4.5}$$

We show that by contradiction. Suppose $d^k_X(x_{0:k}) < \min\{q_b, q'_b\}$. This implies that $d^k_X(x_{0:k})$ does not appear in q_b nor q'_b, and from Lemma 1, it must appear in other generators, which contradicts our assumption. On the other hand, suppose $d^k_X(x_{0:k}) > \min\{q_b, q'_b\}$. Without loss of generality, suppose $\min\{q_b, q'_b\} = q_b$, this means that the homological feature with birth time at q_b in fact has been generated before the appearance of $x_{0:k}$ in the filtration, and therefore $x_{0:k}$ should not be in $\mathcal{G}_b$; again, it contradicts the assumption and completes (4.5).

Now, if the duration of the first feature $q_d - q_b \leq \delta$ in S_K, the filtering process will set the birth time in S'_K as $\tilde{q}_b := q_d$ and $\tilde{q}_d = q_d$, which yields $\tilde{q}_d - \tilde{q}_b = 0 \leq \delta$. If the duration of the first feature $q_d - q_b > \delta$ in S_K, nothing will happen in the filtering process, which leaves $\tilde{q}_b = q_b$. In any case, which can write

$$q_b \leq \tilde{q}_b \leq q_b + \delta. \tag{4.6}$$

Similarly, examining the second feature yields us

$$q'_b \leq \tilde{q}'_b \leq q'_b + \delta. \tag{4.7}$$

The recovery process from G_K to $D^K_{X'}$ as done in Definition 22 constructs the dissimilarity $d^k_{X'}(x_{0:k})$ as the smallest of the birth times of the features, i.e.

$$d^k_{X'}(x_{0:k}) = \min\{\tilde{q}_b, \tilde{q}'_b\}. \tag{4.8}$$

Now, combining (4.5), the first inequality of (4.6) and (4.7), and (4.8) together, we have

$$d^k_X(x_{0:k}) = \min\{q_b, q'_b\} \leq \min\{\tilde{q}_b, \tilde{q}'_b\} = d^k_{X'}(x_{0:k}). \tag{4.9}$$

Simultaneously, combining (4.5), the second inequality of (4.6) and (4.7), and (4.8) yields

$$d^k_{X'}(x_{0:k}) = \min\{\tilde{q}_b, \tilde{q}'_b\} \leq \min\{q_b + \delta, q'_b + \delta\} \leq \min\{q_b, q'_b\} + \delta = d^k_X(x_{0:k}) + \delta. \tag{4.10}$$

Finally, combining (4.9) and (4.10) implies the interleaving inequality

$$d^k_X(x_{0:k}) \leq d^k_{X'}(x_{0:k}) \leq d^k_X(x_{0:k}) + \delta. \tag{4.11}$$

This interleaving inequality proves (4.4). Since (4.4) holds for all simplices, the proof completes. ■

The result in Theorem 8 states that if the homological features we filter out are all short-lasting, i.e. low h-frequencies, than the difference between the filtered network and the original network is small. More specifically, the difference is no greater than the maximum duration of all removed homological features. This result justifies the consideration of short-lasting homological features as unimportant, because they generate small changes if removed from the network, and the consideration of long-lasting features as the core, due to the great change in the network if they are removed. In time-series filtering, Parseval's theorem in Fourier transform guarantees that the energy in the filtered signal is mostly preserved, if we only remove frequencies with small absolute frequency coefficients. Here, in network filtering, Theorem 8 has similar interpretation, with the duration of the homological features represents the "energy" of the tuples generating the features. We could also demonstrate that if we only remove short-lasting features, the distance between a pair of networks is not quiet different from the distance between the pair of their filtered networks, as we state the next.

Theorem 9 *Given a pair of dissimilarity networks D^K_X and D^K_Y with their distance $d^k_{\mathcal{D}}(D^K_X, D^K_Y)$, the filtered network $f(D^K_X)$ of D^K_X using the filtering $\mathbf{F}$ on the homospectrums with $F_k(s_k) = 1$ for any $s_k \geq \delta$ and the filtered network $f'(D^K_Y)$ of D^K_Y using the filtering $\mathbf{F}'$ on the homospectrums with $F'_k(s_k) = 1$ for any $s_k \geq \delta'$ satisfies*

$$d^k_{\mathcal{D}}(f(D^K_X), f'(D^K_Y)) \leq d^k_{\mathcal{D}}(D^K_X, D^K_Y) + \max\{\delta, \delta'\}. \tag{4.12}$$

for any $0 \leq k \leq K-1$.

Proof: Denote the filtered networks $f(D^K_X)$ and $f(D^K_Y)$ as the tuples $(X', d^0_{X'}, \ldots, d^K_{X'})$ and $(Y', d^0_{Y'}, \ldots, d^K_{Y'})$. For a specific k, from Definition 7, there exists a valid correspondence C such that for any pairs of tuples $x_{0:k}$ and $y_{0:k}$ with $(x_{0:k}, y_{0:k}) \in C$, we can write

$$d^k_Y(y_{0:k}) - d^k_{\mathcal{D}}(D^K_X, D^K_Y) \leq d^k_X(x_{0:k}) \leq d^k_Y(y_{0:k}) + d^k_{\mathcal{D}}(D^K_X, D^K_Y). \tag{4.13}$$

From the interleaving inequalities as in (4.11), we can write

$$d_X^k(x_{0:k}) \leq d_{X'}^k(x_{0:k}) \leq d_X^k(x_{0:k}) + \delta, \tag{4.14}$$

$$d_Y^k(y_{0:k}) \leq d_{Y'}^k(y_{0:k}) \leq d_Y^k(y_{0:k}) + \delta'. \tag{4.15}$$

Combining the first inequality in (4.13), the first inequality in (4.14), and the second inequality in (4.15) yields us

$$d_{Y'}^k(y_{0:k}) - \delta' - d_{\mathcal{D}}^k(D_X^K, D_Y^K) \leq d_Y^k(y_{0:k}) - d_{\mathcal{D}}^k(D_X^K, D_Y^K) \leq d_X^k(x_{0:k}) \leq d_{X'}^k(x_{0:k}). \tag{4.16}$$

Besides, combining the second inequality in (4.13), the second inequality in (4.14), and the first inequality in (4.15) yields us

$$d_{X'}^k(x_{0:k}) - \delta \leq d_X^k(x_{0:k}) \leq d_Y^k(y_{0:k}) + d_{\mathcal{D}}^k(D_X^K, D_Y^K) \leq d_{Y'}^k(y_{0:k}) + d_{\mathcal{D}}^k(D_X^K, D_Y^K). \tag{4.17}$$

Therefore, combining the observations in (4.16) and (4.17) yields us

$$\left| d_{X'}^k(x_{0:k}) - d_{Y'}^k(y_{0:k}) \right| \leq d_{\mathcal{D}}^k(D_X^K, D_Y^k) + \max\{\delta, \delta'\}. \tag{4.18}$$

Since (4.18) holds true for any pairs of tuples $x_{0:k}$ and $y_{0:k}$ with $(x_{0:k}, y_{0:k}) \in C$, $X = X'$, and $Y = Y'$, it holds for the pair of tuples that cannot be matched well. From (2.8), we have

$$\Gamma_{X',Y'}^k(C) \leq d_{\mathcal{D}}^k(D_X^K, D_Y^k) + \max\{\delta, \delta'\}. \tag{4.19}$$

The correspondence C is a valid one between X' and Y'. Although it may not be the most optimal one, we can still write $d_{\mathcal{D}}^k(D_{X'}^K, D_{Y'}^K) \leq d_{\mathcal{D}}^k(D_X^K, D_Y^k) + \max\{\delta, \delta'\}$. The inequality holds for all k and completes the proof. ■

Theorem 9 guarantees that in comparing a pair of networks, if we remove the short-lasting homological features in each of the networks, the distance between the filtered networks is close to the distance between the original networks. In specific, the distance between the pair of filtered networks is no greater than the original distance, plus the duration of the most long-lasting features that get removed. This provides us a way to simplify the networks before comparing the networks. We note that the result presented in Theorem 9 is not a direct consequence from Theorem 8, because the direct application of Theorem 8 could only give us

$$d_{\mathcal{D}}^k(f(D_X^K), f'(D_Y^K)) \leq d_{\mathcal{D}}^k(D_X^K, D_Y^K) + \delta + \delta'. \tag{4.20}$$

Chapter 5

Clustering of Networks based on Distance Bounds

In this chapter, we begin by visiting necessary definition of hierarchical clustering, dendrograms, ultrametrics, and chains (Section 5.1). We then state formally the axioms of value and transformation (Section 5.2). We further demonstrate that the two axioms combined yield another intuitive property that no pairs should be clustered together at a resolution smaller than a given threshold (Section 5.2.1). Within this axiomatic framework we construct the *combine-and-cluster* and *cluster-and-combine* methods (Section 5.4). Both of these methods rely on single linkage but differ on the chain cost that is measured to determine if nodes are clustered or not. In combine-and-cluster pairwise distances are estimated by the convex combination of lower and upper bounds and the cost of a chain is the maximum resulting distance. In cluster-and-combine we compute separate chain costs for the lower and upper bounds that are then reduced to their convex combination. We then introduced our main theoretical contribution of the chapter by showing that combine-and-cluster and cluster-and-combine provide bounds on all methods that are admissible with respect to the axioms of value and transformation (Section 5.4). This enables us to characterize the space of admissible methods for metric spaces with dissimilarities specified by intervals and draw connections with admissible methods for metric spaces (Section 5.4.1). Practical values of the methods in synthetic scenarios (Section 5.5.1) and real world settings (Section 5.5.2) are presented.

5.1. Preliminaries

We consider a space M_X quantifying dissimilarity to be a pair (X, d_X) where X is a finite set of nodes and $d_X : X \times X \to \mathbb{R}_+$ measures dissimilarities between pairs. In specific, $d_X(x, x')$ between nodes $x \in X$ and $x' \in X$ is assumed to be nonnegative for all pairs x, x', is symmetric such that $d_X(x, x') = d_X(x', x)$. As a common consideration for clustering problem, we consider $d_X(x, x') = 0$ if and only if the nodes coincide with $x = x'$. The interest of study in this chapter

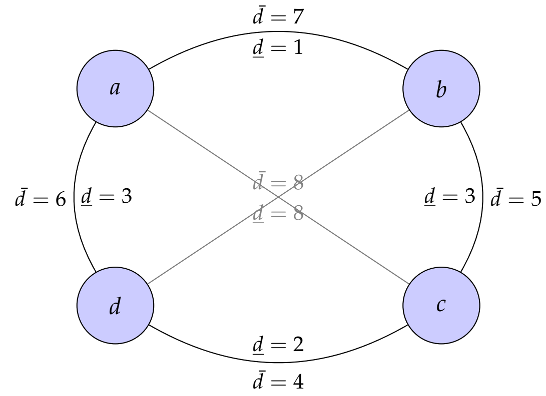


Figure 28: An example of metric space where dissimilarities between pairs of nodes are given in lower and upper bounds. The intuition of clustering is ambiguous because the notion of proximity is no longer clear, e.g. the pair a, b has the smallest distance lower bound whereas the pair c, d has the smallest average of their distance lower and upper bounds. It is not clear which of the two pairs is more proximate.

is not on the dissimilarity space M_X, but in scenarios where observation of $d_X(x, x')$ is not exact but given in a confidence interval. Formally, we consider I_X as the triplet $(X, \underline{d}_X, \bar{d}_X)$ where $\bar{d}_X : X \times X \to \mathbb{R}_+$ is an upper bound of the original dissimilarity and $\underline{d}_X : X \times X \to \mathbb{R}_+$ is a lower bound of the dissimilarity. Given a pair of nodes $x, x' \in X$, we therefore have the relationship $0 < \underline{d}_X(x, x') \leq d_X(x, x') \leq \bar{d}_X(x, x')$. The bounds $\underline{d}(x, x')$ as well as $\bar{d}(x, x')$ are symmetric, i.e. $\underline{d}(x, x')$ is the same as $\underline{d}(x', x)$ and similarly for $\bar{d}(x, x')$.

An example dissimilarity space with distance given by intervals is shown in Figure 28. The set of nodes is $X = \{a, b, c, d\}$ with distance upper and lower bounds represented by values adjacent to each edge. The lower bound $\underline{d}_X(a, b)$ of distance, e.g. from a to b is 1, is smaller than the distance upper bound $\bar{d}_X(a, b) = 7$. The smallest nontrivial case contains two nodes p and q with distance lower bound $\underline{d}(p, q) = \underline{d}$ as well as upper bound $\bar{d}(p, q) = \bar{d} \geq \underline{d} > 0$ is described in Figure 29. This special space appears often in the proceeding discussion of the chapter, and we define the two-node space $\Delta_2(\underline{d}, \bar{d})$ with parameters $\underline{d}$ and $\bar{d}$ as

$$\Delta_2(\underline{d}, \bar{d}) := (\{p, q\}, \underline{d}, \bar{d}). \tag{5.1}$$

A clustering of the set X denotes a partition O_X of X, i.e. a collection of pairwise disjoint sets $O_X = \{T_1, T_2, \ldots, T_J\}$ with $T_i \cap T_j = \emptyset$ for any $i \neq j$ are required to cover X, $\cup_{j=1}^{J} T_j = X$. The sets $T_1, \ldots, T_J$ are named the clusters of O_X. An equivalence relation $\sim$ on X is a binary relation such that for all $x, x', x'' \in X$ we have that (1) $x \sim x$, (2) $x \sim x'$ if and only if $x' \sim x$, and (3) $x \sim x'$ and $x' \sim x''$ would imply $x \sim x''$. A partition $O_X = \{T_1, T_2, \ldots, T_J\}$ of X always induces and is induced by an equivalence relation $\sim_{O_X}$ on X where for all $x, x' \in X$ we have that $x \sim_{O_X} x'$ if and only if x and x' is clustered to the same set S_j for some j.

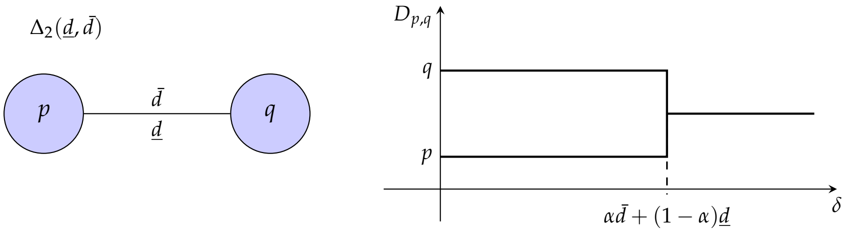


Figure 29: Two-node space $\Delta_2(\bar{d},\underline{d})$ and the Axiom of Value: nodes in a two-node space cluster at the convex combination of the distance upper and lower bounds.

In this chapter we focus on hierarchical clustering methods [48, 21]. The output of hierarchical clustering methods is not a single partition O_X but a nested collection $\mathcal{O}_X$ of partitions $O_X(\delta)$ of X indexed by the resolution parameter $\delta \geq 0$. In the language of equivalence relation defined previously, for a given $\mathcal{O}_X$, we say that two nodes x and x' are equivalent at resolution δ with notation $x \sim_{O_X(\delta)} x'$ if and only if nodes x and x' are in the same cluster of $O_X(\delta)$. The nested collection $\mathcal{O}_X$ is named a *dendrogram* and is required to satisfy the following property [48]:

(D1) Boundary conditions. For $\delta = 0$ the partition $O_X(0)$ clusters each $x \in X$ into a separate singleton and for some δ_∞ sufficiently large $O_X(\delta_\infty)$ clusters all elements into a single set.

(D2) Hierarchy. As δ increases clusters can be combined but not separated. I.e., for any $\delta < \delta'$, any given pair of nodes x, x' with $x \sim_{O_X(\delta)} x'$ would satisfy $x \sim_{O_X(\delta')} x'$.

(D3) Right continuity. For all $\delta \geq 0$, there exists an $\tau > 0$ such that $O_X(\delta') = O_X(\delta)$ for any $\delta' \in [\delta, \delta + \tau]$.

The interpretation of a dendrogram is that of a structure which yields different clustering results at different resolutions. At resolution $\delta = 0$ each node is in a cluster of its own. As the resolution parameter δ increases, nodes start forming clusters. Based on the condition (D2), nodes become more clustered since once they join together in a cluster at some resolution, they stay together in the same cluster for all larger resolutions. Denote $\mathcal{D}$ as the space of all dendrograms, hierarchical clustering method upon distance intervals is defined as a function $\mathcal{W} : \mathcal{I} \to \mathcal{O}$ from the space $\mathcal{I}$ to the space of dendrograms $\mathcal{O}$ such that the underlying space X is preserved. For the triplet $I_X = (X, \underline{d}_X, \bar{d}_X)$, we denote $O_X = \mathcal{W}(X, \underline{d}_X, \bar{d}_X)$ as the output of $\mathcal{W}$.

5.1.1. Dendrograms as Ultrametrics

Dendrograms are difficult to analyze. A more convenient representation is acquired when dendrograms are identified with finite ultrametric spaces. An ultrametric on the space X is a metric $u_X : X \times X \to \mathbb{R}_+$ satisfying the strong triangle inequality such that any points $x, x', x'' \in X$, the ultrametrics $u_X(x, x'')$, $u_X(x, x')$, and $u_X(x', x'')$ satisfy the relationship

$$u_X(x, x'') \leq \max\left(u_X(x, x'), u_X(x', x'')\right). \tag{5.2}$$

Ultrametric spaces are particular cases of metric spaces since (5.2) would imply the usual triangle inequality $u_X(x, x'') \leq u_X(x, x') + u_X(x', x'')$. We investigate ultrametrics because a structure preserving bijective mapping between dendrograms and ultrametrics can be established [48]. Consider the map $\Pi : \mathcal{O} \to \mathcal{U}$ from the space of dendrograms to the space of ultrametrics: given a dendrogram $\mathcal{O}_X$ over a finite set X, the output $\Pi(O_X) = (X, u_X)$ with $u_X(x, x')$ for any pair of nodes $x, x' \in X$ is defined as the smallest resolution at which x and x' are clustered together

$$u_X(x, x') := \min \left\{ \delta > 0 : x \sim_{O_X(\delta)} x' \right\}. \tag{5.3}$$

The map $\Omega : \mathcal{U} \to \mathcal{O}$ is constructed such that for a given ultrametric space (X, u_X) and any resolution $\delta \geq 0$, the equivalence relationship $\sim_{u_X(\delta)}$ is defined as

$$x \sim_{u_X(\delta)} x' \Leftrightarrow u_X(x, x') \leq \delta. \tag{5.4}$$

Denote the cluster result at δ as $O_X(\delta) := \{X \mod \sim_{u_X(\delta)}\}$ where nodes belonging to the same equivalence class is clustered together. The output of the map is then $\Omega(X, u_X) := \mathcal{O}_X$. It is shown [48] that the maps defined above preserve structures in the respective space as we state in the following theorem.

Theorem 10 *The maps* $\Pi : \mathcal{O} \to \mathcal{U}$ *and* $\Omega : \mathcal{U} \to \mathcal{O}$ *are both well defined. Moreover,* $\Pi \circ \Omega$ *is the identity on* $\mathcal{U}$ *and* $\Omega \circ \Pi$ *is the identity on* $\mathcal{O}$.

Given the equivalence between dendrograms and ultrametrics demonstrated by Theorem 10 we can consider hierarchical clustering methods $\mathcal{W}$ as inducing ultrametrics in node spaces X based on distance intervals $\underline{d}_X$ and $\bar{d}_X$ and reinterpret the method $\mathcal{W}$ as a map $\mathcal{W} : \mathcal{I} \to \mathcal{U}$ from the space of dissimilarity spaces given confidence intervals to the space of ultrametrics. The outcome of a hierarchical clustering method constructs an ultrametric in the same space X even when the original observation is given as distance intervals of dissimilarity. We say that two clustering methods $\mathcal{W}_1$ and $\mathcal{W}_2$ being equivalent with notation $\mathcal{W}_1 \equiv \mathcal{W}_2$ if and only if $\mathcal{W}_1(I) = \mathcal{W}_2(I)$ for any $I \in \mathcal{I}$.

5.1.2. Chain, Upper and Lower Chain Costs

The notions of chain and chain cost are substantial in hierarchical clustering. Given a dissimilarity space with distance intervals $(X, \underline{d}, \bar{d})$ and a pair of nodes $x, x' \in X$, a *chain* from x to x' is any ordered sequence of nodes in X,

$$[x = x_0, x_1, \ldots, x_{l-1}, x_l = x'], \tag{5.5}$$

which begins with x and ends at x'. We denote $C(x, x')$ as one such chain and say $C(x, x')$ connects x to x'. Given two chains $C(x, x')$ and $C(x', x'')$ such that the end point x' of the first chain is the

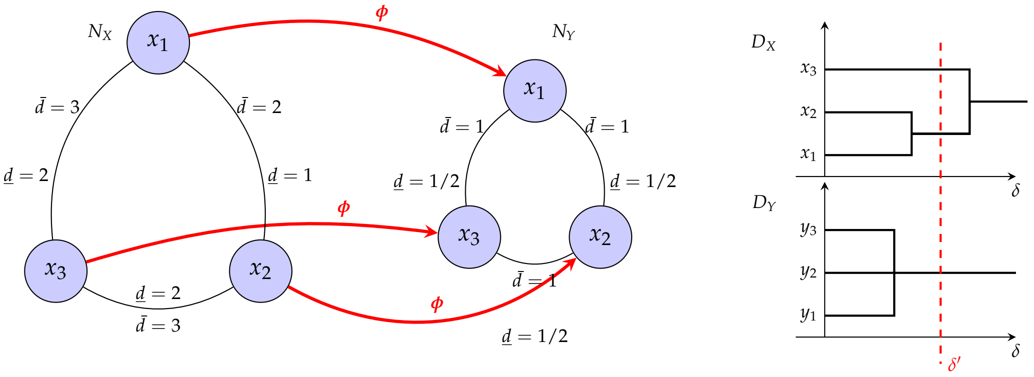


Figure 30: Axiom of Transformation. If I_X can be mapped to I_Y using a α-distance-reducing map π [cf. (5.13) and (5.14)], then for every resolution δ nodes clustered together in $O_X(\delta)$ must also be clustered in $D_Y(\delta)$.

same as the starting point of the second, we define the *concatenated chain* $C(x, x') \uplus C(x', x)$ as

$$[x = x_0, x_1, \dots, x_{l-1}, x_l = x' = x'_0, x'_1, \dots, x'_{l'} = x'']. \tag{5.6}$$

It follows from (5.6) that the concatenation operation $\uplus$ is associative such that $\big(C(x, x') \uplus C(x', x'')\big) \uplus C(x'', x''') = C(x, x') \uplus \big(C(x', x'') \uplus C(x'', x''')\big)$. For the chain $C(x, x')$, we define its *upper cost* and *lower cost* respectively as

$$\max_{i | x_i \in C(x,x')} \bar{d}_X(x_i, x_{i+1}), \qquad \max_{i | x_i \in C(x,x')} \underline{d}_X(x_i, x_{i+1}), \tag{5.7}$$

i.e. the maximum distance upper or lower bounds encountered as traversing the chain in order. The *minimum upper chain cost* $\bar{c}(x, x')$ and the *minimum lower chain cost* $\underline{c}(x, x')$ between x and x' is then defined respectively as the minimum upper and lower cost among all chains connecting x to x',

$$\bar{c}_X(x, x') := \min_{C(x,x')} \max_{i | x_i \in C(x,x')} \bar{d}_X(x_i, x_{i+1}), \tag{5.8}$$

$$\underline{c}_X(x, x') := \min_{C(x,x')} \max_{i | x_i \in C(x,x')} \underline{d}_X(x_i, x_{i+1}). \tag{5.9}$$

Here, we note that the distance upper bound $\bar{d}(x, x')$ may also be unknown for some pairs of nodes. Under that scenario, one can handle unknown upper bounds with a preset global upper bound $\bar{d}$ to represent them. If this upper bound is large, it will likely have a minimal effect in the clusters that are found by the algorithm because these links are unlikely to be part of the chains of minimum cost that we define in (5.8). The minimum upper chain cost $\bar{c}_X(x, x')$ and lower chain cost $\underline{c}_X(x, x')$ are different in general, however they are equal in the degenerate case where distance lower bounds and upper bounds coincide with $\underline{d}_X(x, x') = \bar{d}_X(x, x') := d_X(x, x')$ for any $x, x' \in X$. In this case, the minimum cost $\bar{c}_X(x, x') = \underline{c}_X(x, x')$ are important in the construction of the single linkage [48]. In specific, single linkage ultrametric $u_X^{\mathrm{SL}}(x,x')$ between x and x' is

$$u_X^{\mathrm{SL}}(x,x') = \bar{c}_X(x,x') = \underline{c}_X(x,x') = \min_{C(x,x')} \max_{i|x_i \in C(x,x')} d_X(x_i, x_{i+1}). \tag{5.10}$$

In terms of single linkage dendrogram SL_X, for a given resolution δ, the equivalence classes at resolution δ is

$$x \sim_{\mathrm{SL}_X(\delta)} x' \Leftrightarrow \bar{c}_X(x,x') = \underline{c}_X(x,x') \le \delta. \tag{5.11}$$

It can be seen that $\bar{c}_X$ is the result of applying single linkage towards the node set X equipped with dissimilarity $\bar{d}_X$ despite the fact that $\bar{d}_X$ may not be a valid metric; similar result holds for $\underline{c}_X$. In the degenerative case where distance lower bounds and upper bounds coincide, it is equivalent to consider metric spaces (X, d_X). It has been shown [48] that single linkage is the unique hierarchical clustering method fulfilling axioms (A1) and (A2) discussed in Section 5.2 plus a third axiom stating that the clusters cannot be formed at resolutions smaller than the minimum distance in the space. In the case when the dissimilarity $d_X(x,x')$ are only given in an interval $[\underline{d}_X(x,x'), \bar{d}_X(x,x')]$, the space of methods satisfying axioms (A1) and (A2) and their analogous ones becomes richer, as we explain throughout the chapter.

5.2. Axioms of Value and Transformation

To study hierarchical clustering methods in metric spaces where observations of dissimilarities between pairs are given in distance intervals, we translate natural concepts into the axioms of value and transformation, described in this section. We say a hierarchical clustering method $\mathcal{W}$ is *admissible* if and only if it satisfies both the the axioms of transformation and value.

The Axiom of Value is achieved by considering the two-node space $\Delta_2(\underline{d}, \bar{d})$ defined in (5.1) and described in Figure 29. In the degenerate special case where $\underline{d} = \bar{d} := d(p,q)$, it is apparent that the resolution at which nodes p and q are first clustered together should be $d(p,q)$. In general scenarios where the dissimilarity $d(p,q)$ is given in an interval $[\underline{d}, \bar{d}]$ with $\underline{d} < \bar{d}$, it is reasonable to consider different resolutions at which nodes p and q start to be in the same cluster. In specific, we say that nodes p and q form a single cluster first at resolution $\delta := \alpha\bar{d} + (1-\alpha)\underline{d}$, the convex combination of the upper and lower bounds $\bar{d}$ and $\underline{d}$. Property of hierarchical clustering then indicates nodes p and q are clustered together at any resolution $\delta \ge \alpha\bar{d} + (1-\alpha)\underline{d}$. The parameter α controls the level of confidence in examining the distance intervals. A higher value of α implies a more conservative consideration, where in the extreme case with $\alpha = 1$, nodes p and q are clustered together at the distance upper bound $\bar{d}$; a lower value of α suggests a more liberal examination, and in the other extremal scenario with $\alpha = 0$, nodes p and q considered to be in the same cluster as long as the resolution is no smaller than their distance lower bound $\underline{d}$. Since a hierarchical clustering method is a map $\mathcal{W}$ from metric distance intervals to dendrograms, we formalize this intuition as the following requirement.

(A1) Axiom of Value. Given a value $0 \le \alpha \le 1$, the dendrogram $O_{p,q} = \mathcal{W}(\Delta_2(\underline{d}, \bar{d}))$ produced

by applying $\mathcal{W}$ to the two-node space $\Delta_2(\underline{d}, \bar{d})$ is such that $O_{p,q}(\delta) = \{\{p\}, \{q\}\}$ for $0 \leq \delta < \alpha\bar{d} + (1-\alpha)\underline{d}$ and $O_{p,q}(\delta) = \{\{p, q\}\}$ otherwise.

One may argue that clustering nodes p and q at any monotone increasing function of $\alpha\bar{d} + (1-\alpha)\underline{d}$ would be admissible. Nonetheless, the current formulation implies that the clustering resolution parameter δ is expressed in the same units as the distance intervals. From Theorem 10, we can rewrite the Axiom of Value by referring to properties of the output ultrametrics.

(A1) Axiom of Value. Given a value $0 \leq \alpha \leq 1$, the ultrametric output $(\{p, q\}, u_{p,q}) = \mathcal{W}(\Delta_2(\underline{d}, \bar{d}))$ resulted from applying $\mathcal{W}$ upon the two-node space $\Delta_2(\underline{d}, \bar{d})$ satisfies that

$$u_{p,q}(p, q) = \alpha\bar{d} + (1-\alpha)\underline{d}. \tag{5.12}$$

The second requirement on the space of desired methods $\mathcal{W}$ formalizes the intuition for the behavior of $\mathcal{W}$ when considering a transformation w.r.t. the distance upper and lower bounds on the underlying space X; see Figure 30. Consider two dissimilarity spaces with observations given by distance intervals $I_X = (X, \underline{d}_X, \bar{d}_X)$ and $I_Y = (Y, \underline{d}_Y, \bar{d}_Y)$ and denote $O_X = \mathcal{W}(X, \underline{d}_X, \bar{d}_X)$ and $D_Y = \mathcal{W}(Y, \underline{d}_Y, \bar{d}_Y)$ as the corresponding dendrogram outputs. If we can map all the nodes of the triplet $(X, \underline{d}_X, \bar{d}_X)$ into nodes of $(Y, \underline{d}_Y, \bar{d}_Y)$ such that the combination of lower and upper bounds for any pair of nodes is not increased, we expect the latter distance intervals to be more clustered than the former one at any given resolution. Intuitively, nodes in I_Y are less dissimilar with respect to each other, and therefore at any resolution δ in the respective dendrograms, we expect that for nodes that are clustered in I_X, their corresponding nodes in Y are also clustered in I_Y. In order to formalize this intuition, we introduce the following notion that given two dissimilarity spaces with observations given by distance intervals $I_X = (X, \underline{d}_X, \bar{d}_X)$ and $I_Y = (Y, \underline{d}_Y, \bar{d}_Y)$ and a value $0 \leq \alpha \leq 1$, the map $\pi : X \to Y$ is called *α-distance-reducing* if for any $x, x' \in X$, it holds that

$$\alpha\bar{d}_X(x, x') + (1-\alpha)\underline{d}_X(x, x') \geq \alpha\bar{d}_Y(\pi(x), \pi(x')) + (1-\alpha)\underline{d}_Y(\pi(x), \pi(x')); \tag{5.13}$$

$$\alpha\bar{c}_X(x, x') + (1-\alpha)\underline{c}_X(x, x') \geq \alpha\bar{c}_Y(\pi(x), \pi(x')) + (1-\alpha)\underline{c}_Y(\pi(x), \pi(x')). \tag{5.14}$$

A mapping is α-distance-reducing if both the combinations of distance bounds and chain costs is non-increasing. Notice that, in the degenerate case where distance lower and upper bounds coincide, $u_X^{\mathrm{SL}}(x, x') := \bar{c}_X(x, x') = \underline{c}_X(x, x')$ is the output of applying single linkage upon the dissimilarity space. Therefore (5.13) becomes identical as the requirement $d_X(x, x') \geq d_Y(\pi(x), \pi(x'))$, from which the condition in (5.14) that $c_X(x, x') \geq c_Y(\pi(x), \pi(x'))$ follows directly. In general cases where distance bounds do not coincide, (5.14) does not follow from (5.13) and therefore we need to state both of them. The Axiom of Transformation introduced next is a formal statement of the intuition above.

(A2) Axiom of Transformation. Consider $I_X = (X, \underline{d}_X, \bar{d}_X)$ and $I_Y = (Y, \underline{d}_Y, \bar{d}_Y)$ and a α-dissimilarity-reducing map $\pi : X \to Y$. The method $\mathcal{W}$ satisfies the axiom of transformation if the dendrograms $O_X = \mathcal{W}(X, \underline{d}_X, \bar{d}_X)$ and $O_Y = \mathcal{W}(Y, \underline{d}_Y, \bar{d}_Y)$ satisfy for any $\delta \geq 0$, $x \sim_{O_X(\delta)} x'$ implies

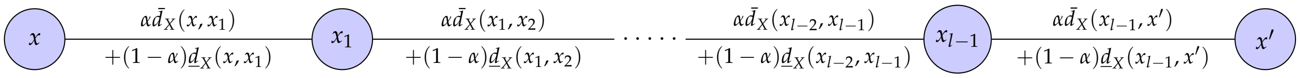


Figure 31: Combine-and-cluster clustering. Nodes x and x' are clustered together at resolution δ if there exists a chain such that the maximum convex combination of distance bounds $\hat{d}_X(x_i, x_{i+1}) = \alpha\bar{d}_X(x_i, x_{i+1}) + (1-\alpha)\underline{d}_X(x_i, x_{i+1})$ is no greater than δ [cf. (5.32)]. Of all methods that satisfy the Axioms of Value and Transformation, combine-and-cluster clustering yields the largest ultrametric between any pair of nodes.

$\pi(x) \sim_{O_Y(\delta)} \pi(x')$.

Rewrite the Axiom of Transformation as in the properties of the output ultrametrics yields the following statement.

(A2) Axiom of Transformation. Consider $I_X = (X, \underline{d}_X, \bar{d}_X)$ and $I_Y = (Y, \underline{d}_Y, \bar{d}_Y)$ and a given α-distance-reducing map $\pi : X \to Y$. For any pair of nodes $x, x' \in X$, the output ultrametrics $u_X = \mathcal{W}(X, \underline{d}_X, \bar{d}_X)$ and $u_Y = \mathcal{W}(Y, \underline{d}_Y, \bar{d}_Y)$ satisfy

$$u_X(x, x') \geq u_Y(\pi(x), \pi(x')). \tag{5.15}$$

In summary, Axiom (A1) states that the units of the resolution parameter δ are the same as that of the distance intervals and specifics our tendency in believing lower or upper bounds. Axiom (A2) states that if we reduce both the distance lower and upper bounds, clusters may be combined but cannot be separated. These axioms are an adaption of the axioms proposed in [48, 149] for the degenerate case of $\underline{d}_X = \bar{d}_X$ which is equivalent to finite metric spaces, and the axioms proposed in [21] for asymmetric networks.

5.2.1. *Minimum Separation*

In this subsection we build another intuition on clustering. In the degenerate case where distance lower and upper bounds coincide, it is intuitive that no clusters should be formed at resolutions smaller than the smallest dissimilarity in the metric space. To formalize such intuitive idea, defining separation of a given metric space (X, d_X) as the minimum positive distance,

$$\text{sep}(X, d_X) := \min_{x \neq x'} d_X(x, x'), \tag{5.16}$$

the ultrametrics resulting from any reasonable hierarchical clustering then need to satisfy $u_X(x, x') \geq \text{sep}(X, d_X)$ for any pair of nodes $x \neq x' \in X$. This requirement is stated as an axiom in consideration of clustering methods for metric spaces in [48, 149]. The separation can also be represented in terms of chain costs

$$\text{sep}(X, d_X) := \min_{x \neq x'} \min_{C(x,x')} \max_{i | x_i \in C(x,x')} d_X(x_i, x_{i+1}). \tag{5.17}$$

Eq. (5.16) and (5.17) are equivalent because for the optimal pair of nodes $\dot{x}$ and $\dot{x}'$, the optimal chain $C^\star(\dot{x}, \dot{x}')$ would just be the connection $[\dot{x}, \dot{x}']$ between them. However, they are different when distance are given in an interval. For general scenarios where the distance upper and lower bounds differ, the α investigated in the Axiom of Value states when nodes in a two-node space should be clustered together. It provides a way to combine the distance bounds and represents where our belief lies in the distance interval. We would expect a measure defined using α carry an analogous notion of separation in metric spaces. To do that, we define α-separation $t^\alpha_X(x, x')$ between two different nodes $x, x' \in X$ in a metric space with distances given by intervals $(X, \underline{d}_X, \bar{d}_X)$ as

$$t^\alpha_X(x, x') = \alpha \bar{c}_X(x, x') + (1-\alpha)\underline{c}_X(x, x'). \tag{5.18}$$

In words, we search for the optimal chain $C(x, x')$ minimizing the upper chain cost, look for the optimal chain $C'(x, x')$ minimizing the lower chain cost, and take the convex combination of these chain costs. The *α-separation* for $(X, \underline{d}_X, \bar{d}_X)$ is then defined as the minimum of $t^\alpha_X(x, x')$ for all nodes $x \neq x'$

$$\operatorname{sep}^\alpha(X, \underline{d}_X, \bar{d}_X) := \min_{x \neq x'} t^\alpha_X(x, x'). \tag{5.19}$$

In the degenerate case we would have $\operatorname{sep}^\alpha(X, \underline{d}_X, \bar{d}_X) = \operatorname{sep}(X, d_X)$ for any α. Following the notion of separation, for resolutions $0 \leq \delta < \operatorname{sep}^\alpha(X, \underline{d}_X, \bar{d}_X)$, no nodes should be clustered together. In the language of ultrametrics, this implies that we must have $u_X(x, x') \geq \operatorname{sep}^\alpha(X, \underline{d}_X, \bar{d}_X)$ for any pair of different nodes $x \neq x' \in X$ as we state in the next property.

(P1) Property of Minimum Separation. For $(X, \underline{d}_X, \bar{d}_X)$, the output ultrametric $(X, u_X) = \mathcal{W}(X, \underline{d}_X, \bar{d}_X)$ of the hierarchical clustering method $\mathcal{W}$ needs to satisfy that the ultrametric $u_X(x, x')$ between any two different points x and x' cannot be smaller than the α-separation $\operatorname{sep}^\alpha(X, \underline{d}_X, \bar{d}_X)$, i.e.

$$u_X(x, x') \geq \operatorname{sep}^\alpha(X, \underline{d}_X, \bar{d}_X) \qquad \forall x \neq x'. \tag{5.20}$$

Equivalently, the output dendrogram is such that for resolutions $\delta < \operatorname{sep}^\alpha(X, \underline{d}_X, \bar{d}_X)$, each node is in its own block. We note that (P1) does not requires that a cluster with more than one node is formed at resolution $\operatorname{sep}^\alpha(X, \underline{d}_X, \bar{d}_X)$ but states that achieving this minimum resolution is a prerequisite condition for the emergence of clusters. Property of Minimum Separation does not only provide intuition in more complicated scenarios than two-node spaces, but is also substantial for later developments in the chapter; see, e.g. the proof of Theorem 12.

Notice that if we apply the Property of Minimum Separation (P1) onto the two-node space $\Delta_2(\underline{d}, \bar{d})$, we must have $u_{p,q}(p, q) \geq \operatorname{sep}_\alpha(\{p, q\}, \underline{d}, \bar{d}) = \alpha \bar{d} + (1-\alpha)\underline{d}$, which means that (P1) and the Axiom of Value (A1) are compatible requirements. We can therefore construct two alternative axiomatic formulations where admissible methods are required to satisfy the Axiom of Transformation (A2) as well as (P1), or (A2) as well as (A1). As we demonstrate in the following theorem that (P1) is implied by (A2) and (A1). Therefore, both two formulations are equivalent to requiring the fulfillment of axioms (A1) and (A2).

Theorem 11 *If a hierarchical clustering method satisfies the Axiom of Value (A1) and Axiom of Transformation (A2), it satisfies the Property of Minimum Separation (P1).*

Proof: To prove Theorem 11, for any $(X, \underline{d}_X, \bar{d}_X)$ and $0 \leq \alpha \leq 1$, we would like to define a two-node space $(\{p, q\}, \bar{d}, \underline{d})$. Moreover, given a pair of distinct nodes $x \neq x' \in X$, we would like to find a α-distance-reducing map $\pi_{x,x'} : X \to \{p, q\}$ from X to the space of two nodes. In order to achieve that, denote $\dot{x}$ and $\dot{x}'$ as the pair achieving $\mathrm{sep}^{\alpha}(X, \underline{d}_X, \bar{d}_X)$. For this pair of nodes, define

$$\underline{\eta} := \min_{C(\dot{x},\dot{x}')} \max_{i | x_i \in C(\dot{x},\dot{x}')} \underline{d}_X(x_i, x_{i+1}), \tag{5.21}$$

and similarly for $\bar{\eta}$ such that $\alpha\bar{\eta} + (1-\alpha)\underline{\eta}$ is the same as $\mathrm{sep}^{\alpha}(X, \underline{d}_X, \bar{d}_X)$. To construct a α-distance-reducing map $\pi_{x,x'}$, first define $\pi_{x,x'}(x) = p$ and $\pi_{x,x'}(x') = q$; then their α-separation in the respective node space satisfies

$$t^{\alpha}_X(x, x') \geq t^{\alpha}_X(\dot{x}, \dot{x}') =: t^{\alpha}_{p,q}(\pi_{x,x'}(x), \pi_{x,x'}(x')). \tag{5.22}$$

Now, for $\tilde{x}$ different from $\dot{x}$ and $\dot{x}'$, $\pi_{x,x'}(\tilde{x})$ can take p or q arbitrarily. To see why this is valid, consider $\tilde{x} \neq \tilde{x}'$, where at least one of the nodes is neither x nor x', if $\pi_{x,x'}(\tilde{x}) = \pi_{x,x'}(\tilde{x}')$, then $\pi_{x,x'}$ is α-distance-reducing. This follows because $t^{\alpha}_X(\tilde{x}, \tilde{x}') \geq 0 =: t^{\alpha}_{p,q}(\pi_{x,x'}(\tilde{x}), \pi_{x,x'}(\tilde{x}'))$, which is (5.14). Moreover,

$$\alpha\bar{d}_X(\tilde{x}, \tilde{x}') + (1-\alpha)\underline{d}_X(\tilde{x}, \tilde{x}') \geq t^{\alpha}_X(\tilde{x}, \tilde{x}'), \tag{5.23}$$

$$\underline{d}_{p,q}(\pi_{x,x'}(\tilde{x}), \pi_{x,x'}(\tilde{x}')) = \bar{d}_{p,q}(\pi_{x,x'}(\tilde{x}), \pi_{x,x'}(\tilde{x}')) = 0, \tag{5.24}$$

would yield us $\hat{d}_X(\tilde{x}, \tilde{x}') \geq 0 =: \hat{d}_{p,q}(\pi_{x,x'}(\tilde{x}), \pi_{x,x'}(\tilde{x}'))$, the condition in (5.13). On the other hand, if $\pi_{x,x'}(\tilde{x}) \neq \pi_{x,x'}(\tilde{x}')$,

$$t^{\alpha}_X(\tilde{x}, \tilde{x}') \geq t^{\alpha}_X(\dot{x}, \dot{x}') =: t^{\alpha}_X(\pi_{x,x'}(\tilde{x}), \pi_{x,x'}(\tilde{x}')), \tag{5.25}$$

which follows from the definition of $\dot{x}$ and $\dot{x}'$ as well as the construction of the two-node space $\Delta_2(\bar{\eta}, \underline{\eta})$. This is the requirement in (5.14). Besides, notice

$$\alpha\bar{d} + (1-\alpha)\underline{d} = s^{\alpha}_{p,q}(\pi_{x,x'}(\tilde{x}), \pi_{x,x'}(\tilde{x}')). \tag{5.26}$$

Combining (5.23), (5.25) and (5.26) yields the condition in (5.13),

$$\bar{d}_X(\tilde{x}, \tilde{x}') + (1-\alpha)\underline{d}_X(\tilde{x}, \tilde{x}') \geq \alpha\bar{d} + (1-\alpha)\underline{d}. \tag{5.27}$$

This shows that $\pi_{x,x'}$ is always a α-distance-reducing map.

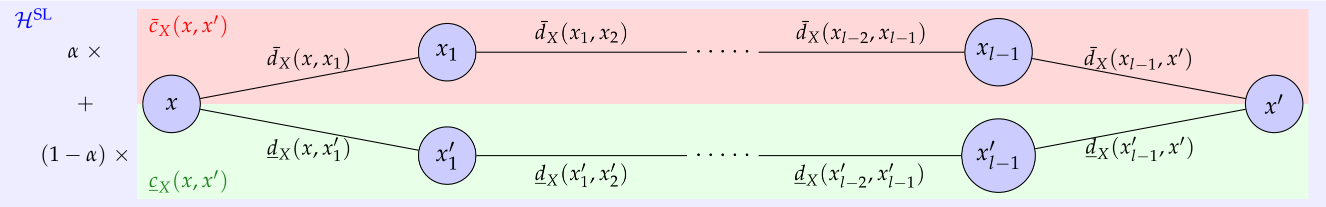


Figure 32: Cluster-and-combine clustering. Nodes x and x' are clustered together at resolution δ if there exists a chain such that the maximum convex combination $\alpha\bar{c}_X(x_i,x_{i+1}) + (1-\alpha)\underline{c}_X(x_i,x_{i+1})$ of minimum upper and lower chain costs is no greater than δ [cf. (5.38)]. Of all methods that satisfy the Axioms of Value and Transformation, cluster-and-combine clustering yields the smallest ultrametric between any pair of nodes.

Denote $(\{p,q\}, u_{p,q}) = \mathcal{W}(\Delta_2(\bar{\eta},\underline{\eta}))$ as the ultrametric space obtained when apply the $\mathcal{W}$ to the two-node space $\Delta_2(\bar{\eta},\underline{\eta})$. Since $\mathcal{W}$ satisfies the Axiom of Value (A1), we must have

$$u_{p,q}(p,q) = \text{sep}^{\alpha}(\{p,q\},\underline{\eta},\bar{\eta}) = \text{sep}^{\alpha}(X,\underline{d}_X,\bar{d}_X). \tag{5.28}$$

Meanwhile, consider the α-distance-reducing map constructed above and observe that $\mathcal{W}$ satisfies the Axiom of Transformation (A2), and therefore for the given pair of distinct nodes $x, x' \in X$,

$$u_X(x,x') \geq u_{p,q}(\pi_{x,x'}(x), \pi_{x,x'}(x')) = u_{p,q}(p,q). \tag{5.29}$$

Since we can construct a α-distance-reducing mapping $\pi_{x,x'}$ for any pair of nodes $x \neq x' \in X$, combining (5.28) and (5.29) yields

$$u_X(x,x') \geq u_{p,q}(p,q) = \text{sep}^{\alpha}(X,\underline{d}_X,\bar{d}_X), \forall x,x' \in X. \tag{5.30}$$

This is the definition of the Property of Minimum Separation (P1). ■

5.3. Admissible Ultrametrics

Consider a specific dissimilarity space with distances given by intervals $I_X = (X, \underline{d}_X, \bar{d}_X) \in \mathcal{I}$. Given a value $0 \leq \alpha \leq 1$, one particular clustering method satisfying axioms (A1) and (A2) can be established by examining the *α-combined* dissimilarity

$$\hat{d}_X(x,x') := \alpha\bar{d}_X(x,x') + (1-\alpha)\underline{d}_X(x,x'), \tag{5.31}$$

for any pair of nodes $x, x' \in X$. Though $\hat{d}_X$ does not necessarily satisfy the triangle inequality as the original dissimilarity distance d_X, it is symmetric; therefore the α-combined dissimilarity effectively reduces the problem to clustering of symmetric data, a case where the single linkage method defined in (5.10) is shown to satisfy axioms analogous to (A1) and (A2) [48]. Based on this observation, we define the *combine-and-cluster* method $\mathcal{W}^{\text{CO}}$ with output $(X, u_X^{\text{CO}}) = \mathcal{W}^{\text{CO}}(X, A_X)$

between a pair x and x' as

$$u_X^{\mathrm{CO}}(x, x') := \min_{C(x,x')} \max_{i | x_i \in C(x,x')} \hat{d}_X(x_i, x_{i+1}). \tag{5.32}$$

An illustration of the combine-and-cluster clustering method is shown in Figure 31. For a given pair of nodes x and x', we look for chains $C(x, x')$ connecting them. For a considered chain we examine each of its link, connecting say x_i with x_{i+1}, and investigate the convex combination of the distance bounds, i.e. the value of $\hat{d}_X(x_i, x_{i+1}) = \alpha \bar{d}_X(x_i, x_{i+1}) + (1-\alpha)\underline{d}_X(x_i, x_{i+1})$. The maximum value across all links in this chain is then recorded. The combine-and-cluster ultrametric $u_X^{\mathrm{CO}}(x, x')$ between points x and x' is the minimum of this value across all possible chains connecting x and x'. We prove that the output u_X^{CO} is a valid ultrametric and the method $\mathcal{W}^{\mathrm{CO}}$ satisfies axioms (A1) and (A2) as next.

Proposition 7 *Given any value of* $0 \leq \alpha \leq 1$, *the combine-and-cluster method* $\mathcal{W}^{\mathrm{CO}}$ *is valid and admissible. I.e.,* u_X^{CO} *defined by* (5.32) *is an ultrametric for all* $I_X = (X, \underline{d}_X, \bar{d}_X)$ *and* $\mathcal{W}^{\mathrm{CO}}$ *satisfies axioms (A1) and (A2).*

Proof: One way to see the validity of the ultrametric $u_X^{\mathrm{CO}}(x, x')$ is because that it is the result of applying single linkage hierarchical clustering on the symmetric dissimilarity $\hat{d}_X(x, x')$. Nonetheless, here we give a direct verification. The fact that $u_X^{\mathrm{CO}}(x, x') = 0$ if and only if $x = x'$ follows directly from that $\hat{d}_X(\tilde{x}, \tilde{x}') > 0$ for any distinct nodes which results from the requirements on the bounds $0 < \bar{d}_X(\tilde{x}, \tilde{x}')$ and $\alpha > 0$. The symmetry property $u_X^{\mathrm{CO}}(x, x') = u_X^{\mathrm{CO}}(x', x)$ is because the definition only depends on $\hat{d}_X(\tilde{x}, \tilde{x}')$ which is symmetric on $\tilde{x}$ and $\tilde{x}'$. To verify the strong triangle inequality in (5.2), let $C'(x, x')$ and $C''(x', x'')$ be the chains that achieve the minimum in (5.32) for $u_X^{\mathrm{CO}}(x, x')$ and $u_X^{\mathrm{CO}}(x', x'')$, respectively. The maximum convex combination in the concatenated chain $C(x, x'') = C'(x, x') \uplus C''(x', x'')$ does not exceed the maximum cost in each of the individual chains. Therefore, despite that $C(x, x'')$ may not be the optimal chain in (5.32), it suffices to bound $u_X^{\mathrm{CO}}(x, x'') \leq \max\left(u_X^{\mathrm{CO}}(x, x'), u_X^{\mathrm{CO}}(x', x'')\right)$ as the strong triangle inequality (5.2).

To see that the Axiom of Value (A1) is satisfied for any considered value $0 \leq \alpha \leq 1$, pick an arbitrary two-node space $\Delta_2(\underline{d}, \bar{d})$ and denote $(\{p, q\}, u_{p,q}^{\mathrm{CO}}) = \mathcal{W}^{\mathrm{CO}}(\Delta_2(\underline{d}, \bar{d}))$ as the output of applying cluster-and-combine clustering method to $\Delta_2(\underline{d}, \bar{d})$. Since every possible chain from p to q must include a link from p to q, applying the definition in (5.32) implies

$$u_{p,q}^{\mathrm{CO}}(p, q) = \alpha \bar{d} + (1-\alpha)\underline{d}, \tag{5.33}$$

from which axiom (A1) is satisfied.

To verify the fulfillment of axiom (A2), consider $(X, \underline{d}_X, \bar{d}_X)$, $(Y, \underline{d}_Y, \bar{d}_Y)$, a given value $0 \leq \alpha \leq 1$, and a α-distance-reducing map $\pi : X \to Y$. Let $(X, u_X^{\mathrm{CO}}) = \mathcal{W}^{\mathrm{CO}}(X, \underline{d}_X, \bar{d}_X)$ and $(Y, u_Y^{\mathrm{CO}}) = \mathcal{W}^{\mathrm{CO}}(Y, \underline{d}_Y, \bar{d}_Y)$ be the outputs of applying the combine-and-cluster clustering methods onto them. For any pair of nodes $x, x' \in X$, denote $C_X^{\star}(x, x') = [x = x_0, x_1, \ldots, x_{l-1}, x_l = x']$ as the optimal chain in (5.32) and therefore we can write

$$u_X^{\mathrm{CO}}(x, x') = \max_{i | x_i \in C_X^\star(x,x')} \hat{d}_X(x_i, x_{i+1}). \tag{5.34}$$

Consider the mapped chain $C_Y(\pi(x), \pi(x')) = [\pi(x) = \pi(x_0), \ldots, \pi(x_l) = \pi(x')]$ in the node space Y under the map π. Since π is α-distance-reducing, we have

$$\hat{d}_Y(\pi(x_i), \pi(x_{i+1})) \le \hat{d}_X(x_i, x_{i+1}), \tag{5.35}$$

for any $x_i \in C^\star(x, x')$. Combining (5.34) and (5.35) yields

$$\max_{\pi(x_i) \in C_Y(\pi(x), \pi(x'))} \hat{d}_Y(\pi(x_i), \pi(x_{i+1})) \le u_X^{\mathrm{CO}}(x, x'). \tag{5.36}$$

Since $C_Y(\pi(x), \pi(x'))$ is a particular chain connecting $\pi(x)$ and $\pi(x')$, the optimal chain cost can only be smaller. Hence,

$$u_Y^{\mathrm{CO}}(\pi(x), \pi(x')) \le \max_{\pi(x_i) \in C_Y(\pi(x), \pi(x'))} \hat{d}_Y(\pi(x_i), \pi(x_{i+1})). \tag{5.37}$$

Finally, substituting (5.37) into (5.36) demonstrates that $u_Y^{\mathrm{CO}}(\pi(x), \pi(x')) \le u_X^{\mathrm{CO}}(x, x')$, which is the requirement (5.15) in the statement of Axiom of Transformation (A2). ■

In combine-and-cluster clustering, nodes x and x' belong to the same cluster at resolution δ whenever we can find a single chain such that the maximum convex combination of distance bounds is no greater than δ. In *cluster-and-combine* clustering, we switch the order of operations and investigate chains, potentially different, connecting x and x', with one chain focusing on the distance upper bounds and the other chain examining the distance lower bounds, before combining the upper and lower estimations. To state this definition regarding ultrametrics, consider $I_X = (X, \underline{d}_X, \bar{d}_X)$ and $0 \le \alpha \le 1$. We define the cluster-and-combine method $\mathcal{W}^{\mathrm{CL}}$ with output $(X, u_X^{\mathrm{CL}}) = \mathcal{W}^{\mathrm{CL}}(X, \underline{d}_X, \bar{d}_X)$ as

$$u_X^{\mathrm{CL}}(x, x') := \min_{C(x,x')} \max_{i | x_i \in C(x,x')} \Big(\alpha \bar{c}_X(x_i, x_{i+1}) + (1-\alpha) \underline{c}_X(x_i, x_{i+1}) \Big), \tag{5.38}$$

where recall $\bar{c}_X$ and $\underline{c}_X$ is the minimum upper and lower chain costs defined in (5.8) and (5.9). An illustration of the cluster-and-combine clustering method is described in Figure 32. For any pair of nodes, we consider the minimum upper chain cost $\bar{c}_X(x, x')$ as the value $\max_{i | x_i \in C(x,x')} \bar{d}_X(x_i, x_{i+1})$ fulfilled by the chain $C'(x, x')$ and the minimum lower chain cost $\underline{c}_X(x, x')$ achieved using the chain $C''(x, x')$. The convex combination $\alpha \bar{c}_X(x, x') + (1-\alpha)\underline{c}_X(x, x')$ is then recorded and the output of the cluster-and-combine clustering method is the result by applying single linkage $\mathcal{W}^{\mathrm{SL}}$ [cf. (5.10)]. The single linkage is applied towards $\alpha \bar{c}_X(x, x') + (1-\alpha)\underline{c}_X(x, x')$ because convex combination of ultrametrics is a metric but not necessarily an ultrametric. Using the shorthand notation $\hat{c}_X(x, x') = \alpha \bar{c}_X(x, x') + (1-\alpha)\underline{c}_X(x, x')$, the output ultrametric of cluster-and-combine clustering is

$$u_X^{\text{CL}}(x,x') := \min_{C(x,x')} \max_{i | x_i \in C(x,x')} \hat{c}_X(x_i, x_{i+1}). \tag{5.39}$$

As the case for combine-and-cluster clustering method, we demonstrate that the output u_X^{CL} is a valid ultrametric and that the method $\mathcal{W}^{\text{CL}}$ satisfies axioms (A1) and (A2) next.

Proposition 8 *The combine-and-cluster method $\mathcal{W}^{CL}$ is valid and admissible given $0 \leq \alpha \leq 1$. I.e., u_X^{CL} defined by (5.38) is an ultrametric for all $I_X = (X, \underline{d}_X, \bar{d}_X)$ and $\mathcal{W}^{CL}$ satisfies axioms (A1) and (A2).*

Proof: Because u_X^{CL} is the output of single linkage to the symmetric dissimilarity $\alpha\bar{c}_X(x,x') + (1-\alpha)\underline{c}_X(x,x')$, u_X^{CL} is a ultrametric.

To see that axiom (A1) is fulfilled, pick an arbitrary two node space $\Delta_2(\underline{d}, \bar{d})$ and denote $(\{p,q\}, u_{p,q}^{\text{CO}}) = \mathcal{W}^{\text{CO}}(\Delta_2(\underline{d}, \bar{d}))$ as the output of applying combine-and-cluster clustering method to $\Delta_2(\underline{d}, \bar{d})$. It then follows that $\bar{c}_{p,q}(p,q) = \bar{d}$ and $\underline{c}_{p,q}(p,q) = \underline{d}$. Also, because every possible chain from p to q must include a link from p to q, the definition in (5.38) becomes $u_{p,q}^{\text{CL}}(p,q) = \alpha\bar{d} + (1-\alpha)\underline{d}$, which shows that axiom (A1) is satisfied.

To verify axiom (A2), consider arbitrary points $x, x' \in X$ and denote $C^\star(x,x')$ the chain achieving minimum cost in (5.39),

$$u_X^{\text{CL}}(x,x') = \max_{i | x_i \in C^\star(x,x')} \hat{c}_X(x_i, x_{i+1}), \tag{5.40}$$

Examine the transformed chain $C_Y(\pi(x), \pi(x'))$; since the map π is α-distance-reducing, it satisfies $\hat{c}_Y(\pi(x_i), \pi(x_{i+1})) \leq \hat{c}_X(x_i, x_{i+1})$ [cf. (5.14)] for any link. Therefore, we can write

$$\begin{aligned} &\max_{i | \pi(x_i) \in C_Y(\pi(x), \pi(x'))} \hat{c}_Y(\pi(x_i), \pi(x_{i+1})) \\ &\qquad\qquad \leq \max_{i | x_i \in C^\star(x,x')} \hat{c}_X(x_i, x_{i+1}). \end{aligned} \tag{5.41}$$

Further observe that $u_Y^{\text{CL}}(\pi(x), \pi(x'))$ cannot exceed the cost in the given chain $C_Y(\pi(x), \pi(x'))$. Hence,

$$u_Y^{\text{CL}}(\pi(x), \pi(x')) \leq \max_{i | x_i \in C^\star(x,x')} \hat{c}_X(x_i, x_{i+1}) = u_X^{\text{CL}}(x,x'), \tag{5.42}$$

where the equality follows from (5.40). This shows u_X^{CL} satisfies axiom (A2) as in (5.15) and concludes the proof. ■

5.4. Extremal Ultrametrics

Given that we have constructed two admissible methods satisfying axioms (A1)-(A2), it is natural to ask whether these two constructions are the only possible ones, and if not, whether they are special with respect to other satisfying methods. We prove in this section the important characterization that any method $\mathcal{W}$ satisfying axioms (A1)-(A2) yields ultrametrics that lie between u_X^{CL} and u_X^{CO}. The characterization can be considered as a generalization of Theorem 18 in [48] for

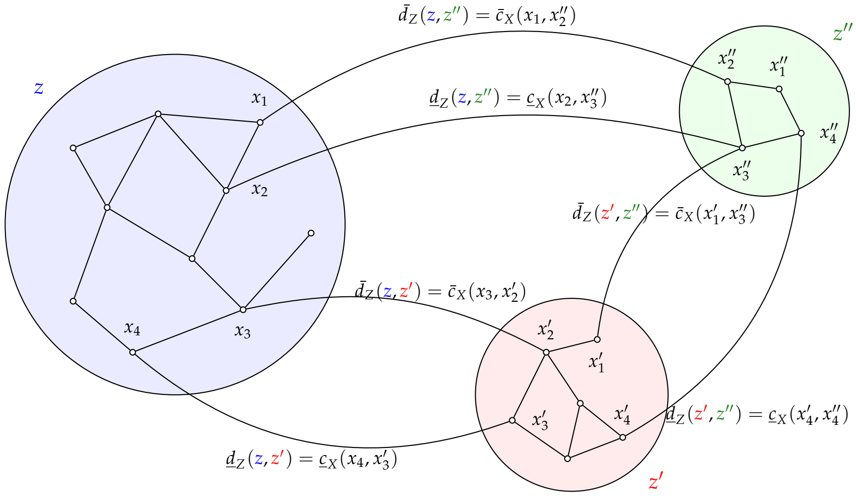


Figure 33: Network of equivalence classes at a given resolution. Each shaded subset of nodes represent an equivalence class. The Axiom of Transformation establishes the relationship between the clustering of nodes in the original network and the clustering of nodes in the network of equivalence classes.

metric spaces.

Theorem 12 *Consider an admissible clustering method $\mathcal{W}$ satisfying axioms (A1)-(A2). For an arbitrary $I_X = (X, \bar{d}_X, \underline{d}_X)$ and $0 \leq \alpha \leq 1$, denote $(X, u_X) = \mathcal{W}(I_X)$ the output of applying $\mathcal{W}$ onto I_X. Then for any pair of nodes $x, x' \in X$,*

$$u_X^{CL}(x, x') \leq u_X(x, x') \leq u_X^{CO}(x, x'), \tag{5.43}$$

where $u_X^{CL}(x, x')$ and $u_X^{CO}(x, x')$ denote the cluster-and-combine and combine-and-cluster ultrametrics defined in (5.38) *and* (5.32).

Proof of $\mathbf{u}_{\mathbf{X}}^{\mathrm{CL}}(\mathbf{x}, \mathbf{x}') \leq \mathbf{u}_{\mathbf{X}}(\mathbf{x}, \mathbf{x}')$: By Theorem 11, $\mathcal{W}$ satisfying (A1)-(A2) implies that it also satisfies (P1). To show the first inequality in (5.43), consider the cluster-and-combine clustering equivalence relation $\sim_{\mathrm{CL}_X(\delta)}$ at resolution δ using $x \sim_{\mathrm{CL}_X(\delta)} x'$ if and only if $u_X^{\mathrm{CL}}(x, x') \leq \delta$. Define the space $Z := X \mod \sim_{\mathrm{CL}_X(\delta)}$ where points in X belonging to the same equivalence class are represented by a single node in Z and the map $\pi_\delta : X \to Z$ that maps each point of X to its equivalence class. Points x and x' are mapped to the same point under π_δ if and only if they belong to the same equivalence class at δ, i.e.

$$\pi_\delta(x) = \pi_\delta(x') \iff u_X^{\mathrm{CL}}(x, x') \leq \delta. \tag{5.44}$$

We define the dissimilarity space with distances given by intervals $I_Z := (Z, \bar{d}_Z, \underline{d}_Z)$ by equipping

Z with distance bounds as

$$\underline{d}_Z(z,z') := \min_{x\in\pi_\delta^{-1}(z),x'\in\pi_\delta^{-1}(z')} \underline{c}_X(x,x'), \tag{5.45}$$

and similarly for $\bar{d}_Z(z,z')$. The distance lower bounds $\underline{d}_Z(z,z')$ compares all the minimum lower chain costs $\underline{c}_X(x,x')$ between a member of the equivalence class z and a member of the equivalence class z' and sets $\underline{d}_Z(z,z')$ to the value corresponding to the pair yielding the lowest minimum lower chain cost. The distance upper bounds $\bar{d}_Z(z,z')$ are constructed similarly; see Figure 33. Observe that follows from the construction, the map π_δ is α-distance-reducing such that for any $x,x'\in X$

$$\hat{d}_X(x,x') \geq \hat{d}_Z(\pi_\delta(x),\pi_\delta(x')), \tag{5.46}$$

$$\hat{c}_X(x,x') \geq \hat{c}_Z(\pi_\delta(x),\pi_\delta(x')). \tag{5.47}$$

To see this, when x and x' are co-clustered at resolution δ, $\hat{d}_Z(\pi_\delta(x),\pi_\delta(x')) = \hat{c}_Z(\pi_\delta(x),\pi_\delta(x')) = 0$. Otherwise, if they are mapped to different equivalent classes, we can write

$$\begin{aligned}\underline{d}_X(x,x') \geq \underline{c}_X(x,x') &\geq \min_{x\in\pi_\delta^{-1}(z),x'\in\pi_\delta^{-1}(z')} \underline{c}_X(x,x')\\ &= \underline{d}_Z(\pi_\delta(x),\pi_\delta(x')),\end{aligned} \tag{5.48}$$

and similarly $\bar{d}_X(x,x') \geq \bar{d}_Z(\pi_\delta(x),\pi_\delta(x'))$. Eq. (5.46) then follows from these two inequalities. Besides, we can also write

$$\begin{aligned}\underline{c}_X(x,x') &\geq \min_{x\in\pi_\delta^{-1}(z),x'\in\pi_\delta^{-1}(z')} \underline{c}_X(x,x')\\ &= \underline{d}_Z(\pi_\delta(x),\pi_\delta(x')) \geq \underline{c}_Z(\pi_\delta(x),\pi_\delta(x')),\end{aligned} \tag{5.49}$$

and similarly $\bar{c}_X(x,x') \geq \bar{c}_Z(\pi_\delta(x),\pi_\delta(x'))$. The convex combination of these two inequalities is identical to Eq. (5.47). This completes the proof that π_δ is α-distance-reducing.

Consider a clustering method $\mathcal{W}$ satisfying axioms (A1)-(A2) and write $(Z,u_Z) = \mathcal{W}(I_Z)$ as the output of applying $\mathcal{W}$ upon I_Z. To apply (P1) we investigate the α-separation of I_Z as next.

Fact 3 *The α-separation of I_Z is*

$$sep^\alpha(I_Z) > \delta. \tag{5.50}$$

Proof: Suppose the contrary is true, i.e. $\text{sep}^\alpha(I_Z) \leq \delta$, then there exists a pair of distinct nodes $z \neq z' \in Z$ such that the convex combination of their distance bounds satisfies

$$\alpha \min_{C(z,z')} \max_{i|z_i\in C(z,z')} \bar{d}_Z(z_i,z_{i+1}) + (1-\alpha) \min_{\tilde{C}(z,z')} \max_{i|z_i\in\tilde{C}(z,z')} \underline{d}_Z(z_i,z_{i+1}) \leq \delta. \tag{5.51}$$

Denote $C^\star$ as the optimal chain in minimizing $\min_{C(z,z')} \max_{i|z_i\in C(z,z')} \bar{d}_Z(z_i,z_{i+1})$ and $\tilde{C}^\star$ as the

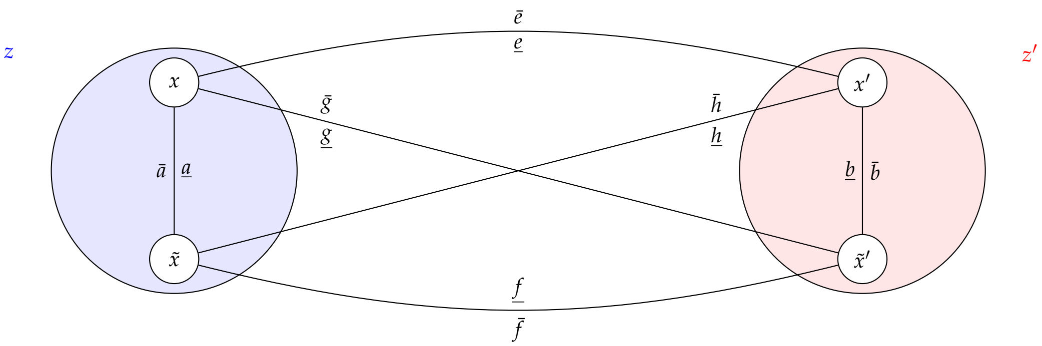


Figure 34: The illustration in the proof of Fact 4. The alphabets adjacent to edges denote the corresponding minimum upper and lower chain costs. Given that $\alpha\bar{a} + (1-\alpha)\underline{a}\delta$, $\alpha\bar{b} + (1-\alpha)\underline{b} \leq \delta$, and $\alpha\bar{e} + (1-\alpha)\underline{f} \leq \delta$, there exists a pair of nodes $\dot{x} \in \{x, \tilde{x}\}$ and $\dot{x}' \in \{x', \tilde{x}'\}$ such that $\alpha\bar{c}_X(\dot{x}, \dot{x}') + (1-\alpha)\underline{c}_X(\dot{x}, \dot{x}') \leq \delta$.

chain in $\min_{\tilde{C}(z,z')} \max_{i|z_i \in \tilde{C}(z,z')} \underline{d}_Z(z_i, z_{i+1})$, (5.51) then becomes

$$\alpha \max_{i|z_i \in C^\star(z,z')} \bar{d}_Z(z_i, z_{i+1}) + (1-\alpha) \max_{i|z_i \in \tilde{C}^\star(z,z')} \underline{d}_Z(z_i, z_{i+1}) \leq \delta. \tag{5.52}$$

From the definitions of $\underline{d}_Z$ given by (5.45) and $\bar{d}_Z$, we can find four nodes $x, \tilde{x}, x', \tilde{x}'$ with $\pi_\delta(x) = \pi_\delta(\tilde{x}) = z$, $\pi_\delta(x') = \pi_\delta(\tilde{x}') = z'$, and two chains $C^\star(x, x')$ and $\tilde{C}^\star(\tilde{x}, \tilde{x}')$ which are mapped to $C^\star(z, z')$ and $\tilde{C}^\star(z, z')$ under π_δ such that

$$\alpha \max_{i|x_i \in C^\star(x,x')} \bar{c}_X(x_i, x_{i+1}) + (1-\alpha) \max_{i|x_i \in \tilde{C}^\star(\tilde{x},\tilde{x}')} \underline{c}_X(x_i, x_{i+1}) \leq \delta. \tag{5.53}$$

Because $\bar{c}_X$ is a valid ultrametric, we can write

$$\begin{aligned} \bar{c}_X(x, x') \leq \max\left\{\bar{c}_X(x, x_1), \bar{c}_X(x_1, x')\right\} &\leq \max\left\{\bar{c}_X(x, x_1), \bar{c}_X(x_1, x_2), \bar{c}_X(x_2, x')\right\} \\ &\leq \cdots \leq \max_{i|x_i \in C^\star(x,x')} \bar{c}_X(x_i, x_{i+1}). \end{aligned} \tag{5.54}$$

Similarly $\underline{c}_X(\tilde{x}, \tilde{x}') \leq \max_{i|x_i \in \tilde{C}^\star(\tilde{x},\tilde{x}')} \underline{c}_X(x_i, x_{i+1})$. Substituting these two bounds into (5.53) implies

$$\alpha\bar{c}_X(x, x') + (1-\alpha)\underline{c}_X(\tilde{x}, \tilde{x}') \leq \delta. \tag{5.55}$$

Further observe that because x and $\tilde{x}$ belong to the same cluster (z) as well as x' and $\tilde{x}'$ belong to the same cluster (z') at resolution δ, we know that $\alpha\bar{c}_X(x, x') + (1-\alpha)\underline{c}_X(x, x') \leq \delta$ and $\alpha\bar{c}_X(\tilde{x}, \tilde{x}') + (1-\alpha)\underline{c}_X(\tilde{x}, \tilde{x}') \leq \delta$. To reach a contradiction we use the following fact.

Fact 4 *There exists a pair* $\dot{x} \in \{x, \tilde{x}\}$ *and* $\dot{x}' \in \{x', \tilde{x}'\}$ *such that*

$$\alpha\bar{c}_X(\dot{x}, \dot{x}') + (1-\alpha)\underline{c}_X(\dot{x}, \dot{x}') \leq \delta. \tag{5.56}$$

Proof: Define the following shorthand notations, $\bar{a} := \bar{c}_X(x,\tilde{x})$, $\bar{b} := \bar{c}_X(x',\tilde{x}')$, $\bar{e} := \bar{c}_X(x,x')$, $\bar{f} := \bar{c}_X(\tilde{x},\tilde{x}')$, $\bar{g} := \bar{c}_X(x,\tilde{x}')$, $\bar{h} := \bar{c}_X(x',\tilde{x})$. Similarly define $\underline{a}, \underline{b}, \underline{e}, \underline{f}, \underline{g}, \underline{h}$; see Figure 34. The problem then becomes: given $\alpha\bar{a} + (1-\alpha)\underline{a} \le \delta$, $\alpha\bar{b} + (1-\alpha)\underline{b} \le \delta$, and $\alpha\bar{e} + (1-\alpha)\underline{f} \le \delta$, we would like to prove that at least one of the following holds: $\alpha\bar{e} + (1-\alpha)\underline{e} \le \delta$, $\alpha\bar{f} + (1-\alpha)\underline{f} \le \delta$, $\alpha\bar{g} + (1-\alpha)\underline{g} \le \delta$, or $\alpha\bar{h} + (1-\alpha)\underline{h} \le \delta$. We show the fact by examining which is the maximum one out of $\bar{a}, \bar{b}, \bar{e}$ and which is the maximum one out of $\underline{a}, \underline{b}, \underline{f}$.

Firstly, in scenarios where $\bar{e} = \max\{\bar{a}, \bar{b}, \bar{e}\}$, it follows from the strong triangle inequality of $\bar{c}_X$ that $\bar{f} \le \bar{e}$. Therefore,

$$\alpha\bar{f} + (1-\alpha)\underline{f} \le \alpha\bar{e} + (1-\alpha)\underline{f} \le \delta, \tag{5.57}$$

which shows the desired result. The proof for cases with $\underline{f} = \max\{\underline{a}, \underline{b}, \underline{f}\}$ follows by symmetry. Therefore, what remain are scenarios where neither $\bar{e} = \max\{\bar{a}, \bar{b}, \bar{e}\}$ nor $\underline{f} = \max\{\underline{a}, \underline{b}, \underline{f}\}$.

Secondly, in scenarios where $\bar{a} = \max\{\bar{a}, \bar{b}, \bar{e}\}$ and $\underline{a} = \max\{\underline{a}, \underline{b}, \underline{f}\}$, we have $\bar{e} \le \bar{a}$ and $\underline{e} \le \underline{a}$ where the latter follows from the strong triangle inequality of $\bar{c}_X$. Consequently

$$\alpha\bar{e} + (1-\alpha)\underline{e} \le \alpha\bar{a} + (1-\alpha)\underline{a} \le \delta, \tag{5.58}$$

which shows the desired result. The proof for cases with $\bar{b} = \max\{\bar{a}, \bar{b}, \bar{e}\}$ and $\underline{b} = \max\{\underline{a}, \underline{b}, \underline{f}\}$ follows by symmetry.

Thirdly, consider $\bar{a} = \max\{\bar{a}, \bar{b}, \bar{e}\}$ and $\underline{b} = \max\{\underline{a}, \underline{b}, \underline{f}\}$. If $\bar{e} \le \bar{b}$, because $\underline{e} \le \underline{b}$ by the strong inequality of $\underline{c}_X$, we have

$$\alpha\bar{e} + (1-\alpha)\underline{e} \le \alpha\bar{b} + (1-\alpha)\underline{b} \le \delta, \tag{5.59}$$

which is the desired result. Otherwise, if $\bar{b} \le \bar{e}$, we can write $\bar{g} \le \max\{\bar{b}, \bar{e}\} = \bar{e}$ and $\underline{g} \le \max\{\underline{a}, \underline{f}\}$. Therefore,

$$\alpha\bar{g} + (1-\alpha)\underline{g} \le \alpha\bar{e} + (1-\alpha)\max\{\underline{a}, \underline{f}\}. \tag{5.60}$$

Utilizing the fact $\bar{e} \le \bar{a}$ in (5.60) yields

$$\alpha\bar{g} + (1-\alpha)\underline{g} \le \max\left\{\alpha\bar{a} + (1-\alpha)\underline{a}, \alpha\bar{e} + (1-\alpha)\underline{f}\right\} \le \delta, \tag{5.61}$$

which shows the desired result. The proof for $\bar{b} = \max\{\bar{a}, \bar{b}, \bar{e}\}$ and $\underline{a} = \max\{\underline{a}, \underline{b}, \underline{f}\}$ follows by symmetry. We have proven the statement under all cases, and the proof of Fact 4 is complete. ■

Continuing with the proof of Fact 3, since there exists a pair of nodes $\dot{x} \in \{x, \tilde{x}\}$ and $\dot{x}' \in \{x', \tilde{x}'\}$ with $\alpha\bar{c}_X(\dot{x}, \dot{x}') + (1-\alpha)\underline{c}_X(\dot{x}, \dot{x}') \le \delta$, the fact $u_X^{\text{CL}}(x, x') \le \delta$ contradicts the assumption $\pi(\dot{x}) = z \neq z' = \pi(\dot{x}')$. Therefore, the assumption that (5.50) is false cannot hold. The opposite must be true. ■

Back to the main proof of $u_X^{\rm CL}(x,x') \leq u_X(x,x')$, recall that $(Z, u_Z) = \mathcal{W}(Z, \underline{d}_Z, \bar{d}_Z)$. Since the α-separation of Z satisfies (5.50), (P1) implies for any pair of nodes $z \neq z'$, $u_Z(z,z') > \delta$. Also observe that because π is α-distance-reducing and $\mathcal{W}$ satisfies (A2), we must have $u_X(x,x') \geq u_Z(z,z')$. This inequality, combined with $u_Z(z,z') > \delta$ enables us to conclude that when x and x' are mapped to different equivalence classes,

$$u_X(x,x') \geq u_Z(z,z') > \delta. \tag{5.62}$$

Notice that from (5.44), x and x' are mapped to different equivalence classes if and only if $u_X^{\rm CL}(x,x') > \delta$. Therefore, we can claim that $u_X^{\rm CL}(x,x') > \delta$ implies $u_X(x,x') > \delta$. Because this statement is true for any $\delta > 0$, it induces that $u_X^{\rm CL}(x,x') \leq u_X(x,x')$ for any $x \neq x' \in X$ as the first inequality in (5.43). ■

Proof of $\mathbf{u_X(x,x') \leq u_X^{CO}(x,x')}$: To show the second inequality in (5.43), first notice that for any distinct nodes $x_i \neq x_j \in X$, we can construct a two-node space $\Delta_{i,j} = \big(\{p,q\}, \underline{d}_X(x_i,x_j), \bar{d}_X(x_i,x_j)\big)$ and a mapping $\pi_{i,j} : \{p,q\} \to X$ with $\pi_{i,j}(p) = x_i$ and $\pi_{i,j}(q) = x_j$ such that $\pi_{i,j}$ is α-distance-reducing. To demonstrate this, we need to verify conditions (5.13) and (5.14). Eq. (5.13) follows because $\bar{d}_X(\pi_{i,j}(p), \pi_{i,j}(q)) = \bar{d}_{p,q}(p,q), \underline{d}_X(\pi_{i,j}(p), \pi_{i,j}(q)) = \underline{d}_{p,q}(p,q)$ and therefore the convex combination of the distance bounds also coincide. To see (5.14), using the relationships between distance bounds and minimum chain costs, we can write

$$\begin{aligned}
\bar{c}_X(\pi_{i,j}(p), \pi_{i,j}(q)) &\leq \bar{d}_X(\pi_{i,j}(p), \pi_{i,j}(q)),\\
\underline{c}_X(\pi_{i,j}(p), \pi_{i,j}(q)) &\leq \underline{d}_X(\pi_{i,j}(p), \pi_{i,j}(q)),\\
\underline{d}_{p,q}(p,q) = \underline{c}_{p,q}(p,q), &\quad \bar{d}_{p,q}(p,q) = \bar{c}_{p,q}(p,q).
\end{aligned} \tag{5.63}$$

Therefore, have

$$\bar{c}_X(\pi_{i,j}(p), \pi_{i,j}(q)) \leq \bar{c}_{p,q}(p,q),\ \underline{c}_X(\pi_{i,j}(p), \pi_{i,j}(q)) \leq \underline{c}_{p,q}(p,q), \tag{5.64}$$

from which the requirement of convex combination in (5.14) follows directly. Because $\mathcal{W}$ satisfies (A1), the output ultrametric $(\{p,q\}, u_{p,q})$ of applying $\mathcal{W}$ onto $\Delta_{i,j}$ implies

$$u_{p,q}(p,q) = \alpha \bar{d}_{p,q}(p,q) + (1-\alpha)\underline{d}_{p,q}(p,q) = \hat{d}_X(x_i,x_j), \tag{5.65}$$

Moreover, $\mathcal{W}$ satisfies (A2), and therefore

$$u_X(x_i,x_j) \leq u_{p,q}(p,q) = \hat{d}_X(x_i,x_j). \tag{5.66}$$

Observe that when $x_i = x_j$, (5.66) also holds because both sides on the inequality is zero. Consequently, (5.66) holds true for any points $x_i, x_j \in X$. Now, consider the nodes x and x' and denote $C^\star(x,x')$ as the chain yielding the minimum cost in (5.32),

$$u_X^{\rm CO}(x,x') = \max_{i | x_i \in C^\star(x,x')} \hat{d}_X(x_i, x_{i+1}). \tag{5.67}$$

Substituting the inequality (5.66) in (5.67) yields

$$u_X^{\mathrm{CO}}(x, x') \geq \max_{i | x_i \in C^\star(x,x')} u_X(x_i, x_{i+1}). \tag{5.68}$$

Finally, because u_X is a valid ultrametric, as in (5.54), we can bound $u_X(x, x') \leq \max_{i|x_i \in C^\star(x,x')} u_X(x_i, x_{i+1})$. Combining with (5.68) yields $u_X^{\mathrm{CO}}(x, x') \geq u_X(x, x')$ as the second inequality in (5.43). ■

From Theorem 12, cluster-and-combine clustering u_X^{CL} applied to $I_X = (X, \underline{d}_X, \bar{d}_X)$ yields a minimal ultrametric among outputs by all methods satisfying axioms (A1)-(A2). Combine-and-cluster clustering u_X^{CO} yields a uniformly maximal ultrametric.

Remark 8 Theorem 12 resembles the results obtained for asymmetric clustering in [21] where two methods are obtained and shown to be extremal with respect to similar axioms. The difference is that in Theorem 12, dissimilarity bounds represent the uncertain but symmetric relationship between the pair. In asymmetric networks, all observations are certain but the relationship from node x to x' is asymmetric and may not be the same as the relationship from node x' to x. These differences manifest on the selection of a different axiom of value where instead of clustering at the larger of the two relationships, we cluster at the convex combination of distance bounds $\alpha \bar{d} + (1 - \alpha)\underline{d}$ [cf. (5.12)]

Remark 9 Theorem 12 states that the extremal clustering methods satisfying the axioms are constructed via single linkage clustering. However, this does not necessarily imply that *all* methods satisfying the axioms need to be constructed via single linkage clustering. Single linkage clustering has a tendency toward what is called "chaining", and can produce irregularly shaped clusters. But, it is also good at combining close observations that other methods might leave separate [150], as we illustrate in Section 5.5.2 using coauthorship networks. Besides, in the chapter we do not set out to design methods based on single linkage. Rather, we define reasonable axioms with respect to which single linkage based clustering methods are extremal.

Remark 10 In the problem formulation we assume $0 < \underline{d}_X(x, x') \leq \bar{d}_X(x, x')$ for any pair $x \neq x'$, because such consideration ensures that valid ultrametrics can be established when $\alpha = 0$ and therefore the validity of Propositions 7 and 8. We note that we can relax the constraint to only assume $\underline{d}_X(x, x') \geq 0$. Under such considerations, the main result in Theorem 12 holds; besides, same results as in Propositions 7 and 8 can be established for $0 < \alpha \leq 1$.

Remark 11 We note that different from the dissimilarity bounds $\underline{d}_X$ and $\bar{d}_X$, α is a global parameter preset by the user to the algorithm. It is possible to consider the framework where a different confidence parameter $\alpha_X(x, x')$ is used for each edge. In specific, the clustering algorithm is then constructed for the tuple $(X, \underline{d}_X, \bar{d}_X, \alpha_X)$ where $\alpha_X(x, x')$ is the confidence parameter for the edge joining x and x'. If we modify the definition of distance-reducing mappings in (5.13) and (5.14) to replace α by $\alpha_X(x, x')$ and $\alpha_Y(\pi(x), \pi(x'))$, we can consider the clustering problem by building axioms almost identical as those presented in the chapter. It can be shown that most results, including Theorem 11, Propositions 7 and 8, and the second part $u_X(x, x') \leq u_X^{\mathrm{CO}}(x, x')$ in Theorem

12 follow using the same proof technique. The first part $u_X^{\mathrm{CL}}(x,x') \leq u_X(x,x')$ in Theorem 12 cannot be derived using the same proof technique. We anticipate similar result to hold as well, but leave detailed consideration to future works.

5.4.1. Hierarchical Clustering given Extremal Confidence Level

In the previous section, we consider admissible clustering methods given as an arbitrary value $0 \leq \alpha \leq 1$. In this subsection, we investigate the special cases given α at the extreme points, i.e. $\alpha \in \{0, 1\}$. Starting with $\alpha = 1$, this means we are the most conservative and believe the distance between two points x and x' being their distance upper bound $\bar{d}_X(x,x')$. The output of the combine-and-cluster clustering methods can then be written as

$$u_X^{\mathrm{CO}}(x,x') = \min_{C(x,x')} \max_{i|x_i \in C(x,x')} \bar{d}_X(x_i, x_{i+1}), \tag{5.69}$$

which is the same as applying single linkage clustering $\mathcal{W}^{\mathrm{SL}}$ [cf. (5.10)] onto distance upper bounds $\bar{d}_X$. On the other hand, the output of the cluster-and-combine clustering methods is

$$u_X^{\mathrm{CL}}(x,x') = \min_{C(x,x')} \max_{i|x_i \in C(x,x')} \bar{c}_X(x_i, x_{i+1}), \tag{5.70}$$

with $\bar{c}_X$ the minimum upper chain costs defined in (5.8). Notice that $\bar{c}_X$ is also the output of applying single linkage clustering $\mathcal{W}^{\mathrm{SL}}$ onto the distance upper bounds. Moreover, because $\bar{c}_X$ is a valid ultrametric, $\min_{C(x,x')} \max_{i|x_i \in C(x,x')} \bar{c}_X(x_i, x_{i+1})$ is the same as $\bar{c}_X(x,x')$. Combining these observations, it follows that

$$u_X^{\mathrm{CL}}(x,x') = u_X^{\mathrm{SL}}(x,x') = u_X^{\mathrm{CO}}(x,x'). \tag{5.71}$$

When $\alpha = 0$, meaning that we are the most liberate and believe the distance between two points x and x' being their distance lower bound $\underline{d}_X(x,x')$, a similar analysis would follow. We can now utilize Theorem 12 and (5.71) to prove the uniqueness of admissible hierarchical clustering methods abiding (A1)-(A2), given that the confidence level α is at the extremes, i.e. $\alpha \in \{0, 1\}$.

Corollary 2 *Consider a clustering method $\mathcal{W}$ satisfying axioms (A1)-(A2). For arbitrary $I_X = (X, \bar{d}_X, \underline{d}_X)$, denote $(X, u_X) = \mathcal{W}(I_X)$ the output of applying $\mathcal{W}$ onto I_X. When $\alpha = 1$, $\mathcal{W} \equiv \mathcal{W}^{SL}(\bar{d}_X)$ is the same as the singe linkage clustering [cf.* (5.10)*] onto the distance upper bounds; when $\alpha = 0$, $\mathcal{W} \equiv \mathcal{W}^{SL}(\underline{d}_X)$.*

Proof: When $\alpha = 1$, because $\mathcal{W}$ satisfies the hypotheses of Theorem 12, (5.43) is true for any distinct nodes $x, x' \in X$. But by (5.71), cluster-and-combine and combine-and-cluster ultrametrics coincide; as a result (5.43) can be written as

$$u_X^{\mathrm{SL}}(x,x') \leq u_X(x,x') \leq u_X^{\mathrm{SL}}(x,x'). \tag{5.72}$$

It follows that $u_X(x,x') = u_X^{\mathrm{SL}}(x,x')$ for any pair of nodes x, x'. Therefore $\mathcal{W} \equiv \mathcal{W}^{\mathrm{SL}}(\bar{d}_X)$. Similar

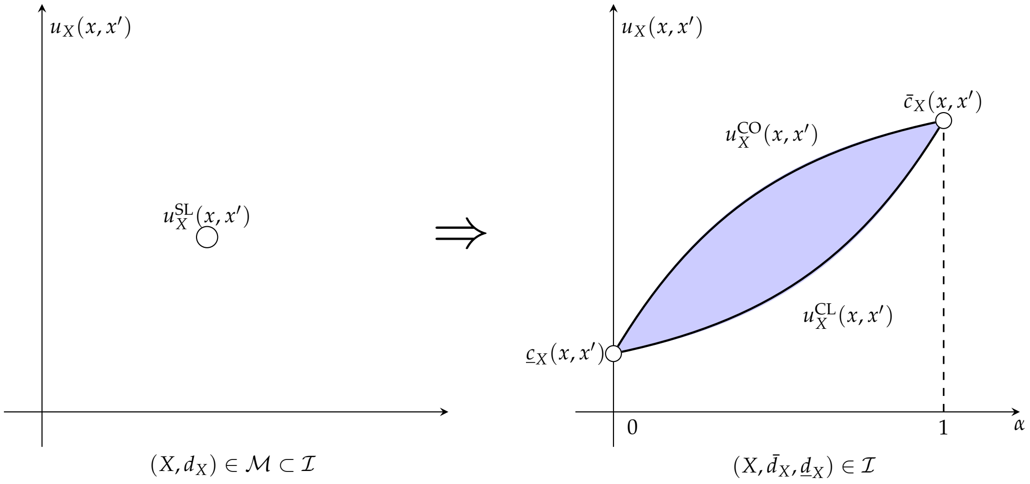


Figure 35: Summary of admissible hierarchical clustering methods for dissimilarity spaces with distances given by intervals. Given a pair of nodes $x \neq x' \in X$, in dissimilarity space, single linkage clustering ultrametric $u_X^{\mathrm{SL}}(x,x')$ is the unique output satisfying (A1)-(A2). In general scenarios, there exists a family of admissible clustering methods (the blue region on the right). When the confidence level is at the extreme points ($\alpha \in \{0,1\}$), singe linkage clustering ultrametric $\bar{c}_X(x,x')$ and $\underline{c}_X(x,x')$ are the respective unique method (red and green points). For a given confidence level $0 < \alpha < 1$, cluster-and-combine output $u_X^{\mathrm{CO}}(x,x')$ and combine-and-clustering output $u_X^{\mathrm{CL}}(x,x')$ are the extremal on all admissible methods.

derivation holds for $\alpha = 0$. ■

Restrict attention to the metric space $\mathcal{M} \subset \mathcal{I}$ of the form (X, d_X). A further application of Corollary 2 implies singe linkage clustering is the unique admissible methods as next.

Corollary 3 *Let $\mathcal{W} : \mathcal{M} \to \mathcal{U}$ be a hierarchical clustering method. If $\mathcal{W}$ satisfies axioms (A1) and (A2) then $\mathcal{W} \equiv \mathcal{W}^{SL}$.*

The uniqueness results claimed by Corollaries 2 and 3 can be considered as generalization of the uniqueness statement of single linkage clustering for metric space in [48, Theorem 18]. When we take the most conservative belief and consider distance between points as their distance upper bounds $\bar{d}_X$, the only admissible method is the single linkage clustering applied onto the upper bounds $\bar{d}_X$. On the other hand, when we are the most liberate and trust the information conveyed in the distance lower bounds $\underline{d}_X$, single linkage clustering applied onto $\underline{d}_X$ is the unique admissible method. In metric space (X, d_X) with $d_X := \bar{d}_X = \underline{d}_X$, irrespective of our belief of α, the unique clustering method is the single linkage clustering applied onto d_X. Therefore, we can summarize the space of admissible hierarchical clustering in Figure 35. The unique clustering method $\mathcal{W}^{\mathrm{SL}}$ in metric spaces becomes a space of admissible methods when distances are given by intervals. When the confidence interval is at the extreme points ($\alpha \in \{0,1\}$), the uniqueness of admissible methods is provided by Corollary 2. For general confidence level $0 < \alpha < 1$, the admissible methods are not unique; cluster-and-combine as well as combine-and-clustering methods provide

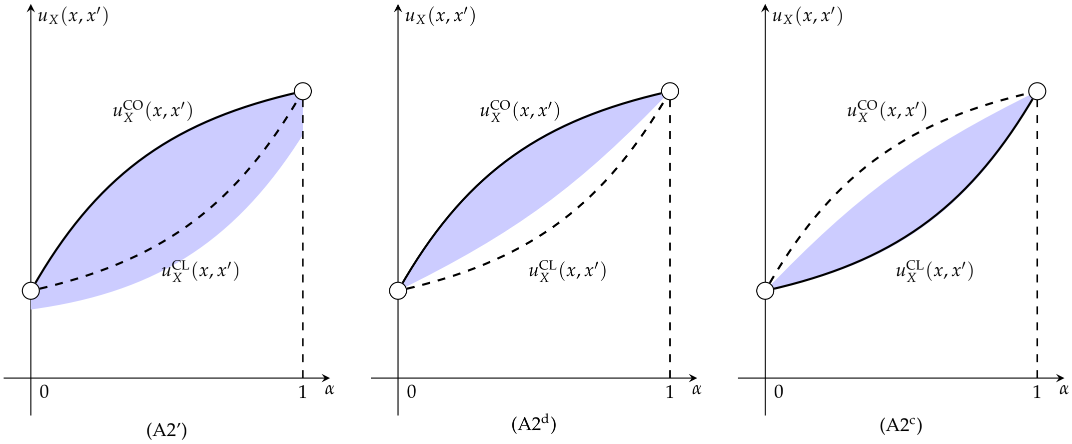


Figure 36: Admissible hierarchical clustering methods given other possible construction of Axiom of Transformation discussed in Section 5.4.2. On the left, axiom (A2′) is a weaker requirement compared to (A2) and therefore (A1)-(A2′) yields a larger set of admissible clustering methods. In specific, combine-and-cluster output $u_X^{\mathrm{CO}}(x,x')$ is still a global maximum but $u_X^{\mathrm{CL}}(x,x')$ may no longer be a global minimum. In the middle and on the right, axioms (A2$^{\mathrm{d}}$) as well as (A2$^{\mathrm{c}}$) are a stronger requirements compared to (A2) and therefore their respective combination with (A1) gives a smaller set of admissible clustering methods.

uniformly minimal and maximal bounds, which is established in Theorem 12. We note that, given a specific confidence level α, the output of the admissible methods do not differ much – see examples in Section 5.5. This indicates that seemingly weak conditions established by the intuitive axioms in fact well describe the space of admissible hierarchical clustering algorithms. We emphasize, though, that the focus of the chapter is to study the space of admissible methods under the axioms, and not to state that other methods not satisfying the axioms are unreasonable. Other clustering algorithms, e.g. dbscan [151] and hierarchical dbscan [152], are very useful and should be considered as reasonable as well. We expect that the generalization of these density based clustering algorithms to distance bounds would satisfy the described axioms, but leave it for future work.

5.4.2. *Other Constructions of Axiom of Transformation*

In Axiom of Transformation (A2), we require the output ultrametric to satisfy $u_X(x,x') \geq u_Y(\pi(x),\pi(x'))$ when the map π is α-distance-reducing, i.e. satisfying (5.13) and (5.14). Even though we justify that Eqs. (5.13) and (5.14) are equivalent to the natural condition on the map π such that $d_X(x,x') \geq d_Y(\pi(x),\pi(x'))$ when restrict attention onto metric spaces $\mathcal{M}$, some readers may find such requirement on $\mathcal{I}$ is not highly intuitive and are curious to see what would work for other constructions of axiom of transformation. In this section, we consider other generalizations of axiom of transformation and the admissible clustering methods induced by them. We focus on presenting results and omit proofs.

We start by considering the following construction.

(A2′) Axiom of Transformation (Alternative). Consider $I_X = (X, \underline{d}_X, \bar{d}_X)$ and $I_Y = (Y, \underline{d}_Y, \bar{d}_Y)$ and a given map $\pi : X \to Y$ such that

$$\underline{d}_X(x, x') \geq \underline{d}_Y(\pi(x), \pi(x')), \bar{d}_X(x, x') \geq \bar{d}_Y(\pi(x), \pi(x')), \tag{5.73}$$

for any nodes $x \neq x'$. The ultrametrics $u_X = \mathcal{W}(X, \underline{d}_X, \bar{d}_X)$ and $u_Y = \mathcal{W}(Y, \underline{d}_Y, \bar{d}_Y)$ are said to satisfy the axiom of transformation (alternative) if $u_X(x, x') \geq u_Y(\pi(x), \pi(x'))$.

Note that the requirement on the map (5.73) would imply π is a α-distance-reducing map. Hence, because compared to (A2), (A2′) implies the same output $u_X(x, x') \geq u_Y(\pi(x), \pi(x'))$ under a stricter requirement on π, (A2′) is a weaker condition than (A2) and therefore the admissible clustering methods satisfying (A1)-(A2′) would be richer. In specific, as illustrated on the left of Figure 36, combine-and-cluster output $u_X^{\mathrm{CO}}(x, x')$ and cluster-and-combine output $u_X^{\mathrm{CL}}(x, x')$ are still admissible; $u_X^{\mathrm{CO}}(x, x')$ is a global maximum but we could not verify that $u_X^{\mathrm{CL}}(x, x')$ is still a global minimum. There might be other admissible methods yielding output $u_X(x, x')$ which is smaller than $u_X^{\mathrm{CL}}(x, x')$.

In (A2), we say a map is α-distance-reducing if it satisfies both (5.13) and (5.14). Investigate the construction for maps that satisfy only a single requirement of them yields the two possible ways to construct different axioms of transformation as we state next.

(A2^d) Axiom of Transformation (Distance). Consider I_X, I_Y, $0 \leq \alpha \leq 1$, and a map $\pi : X \to Y$ such that

$$\hat{d}_X(x, x') \geq \hat{d}_Y(\pi(x), \pi(x')) \tag{5.74}$$

for any $x \neq x'$. The outputs u_X and u_Y are said to satisfy the axiom of transformation (distance) if $u_X(x, x') \geq u_Y(\pi(x), \pi(x'))$.

(A2^c) Axiom of Transformation (Chain Costs). Consider I_X, I_Y, $0 \leq \alpha \leq 1$, and a map $\pi : X \to Y$ such that

$$\hat{c}_X(x, x') \geq \hat{c}_Y(\pi(x), \pi(x')), \forall x \neq x' \in X \tag{5.75}$$

The outputs u_X and u_Y are said to satisfy the axiom of transformation (chain costs) if $u_X(x, x') \geq u_Y(\pi(x), \pi(x'))$.

For (A2$^{\mathrm{c}}$), even though the requirement in (5.75) is with respect to the combination of minimum chain costs and is different from distance bounds as in (5.74), it can be shown that single linkage clustering $\mathcal{W}^{\mathrm{SL}}$ is still the only admissible method satisfying (A1)-(A2$^{\mathrm{c}}$) when we restrict attention onto metric spaces.

Compared to (A2), both (A2$^{\mathrm{d}}$) and (A2$^{\mathrm{c}}$) induce the same output $u_X(x, x') \geq u_Y(\pi(x), \pi(x'))$ under weaker requirements on π. Consequently, both (A2$^{\mathrm{d}}$) and (A2$^{\mathrm{c}}$) are more stringent conditions than (A2). This implies that, compared to the the admissible clustering methods satisfying (A1)-(A2), the admissible methods satisfying (A1)-(A2$^{\mathrm{d}}$) as well as methods satisfying (A1)-(A2$^{\mathrm{c}}$)

would be smaller. Indeed, as illustrated in the middle of Figure 36, for the axioms (A1)-($\mathrm{A2}^{\mathrm{d}}$), combine-and-cluster output $u_X^{\mathrm{CO}}(x, x')$ is admissible but cluster-and-combine output $u_X^{\mathrm{CL}}(x, x')$ is not. In analogy, as on the right of Figure 36, for the axioms (A1)-($\mathrm{A2}^{\mathrm{c}}$), $u_X^{\mathrm{CL}}(x, x')$ is admissible but $u_X^{\mathrm{CO}}(x, x')$ is not.

We focus our analysis on axioms (A1)-(A2) because we believe $u_X^{\mathrm{CL}}(x, x')$ and $u_X^{\mathrm{CO}}(x, x')$ are reasonable clustering methods in dissimilarity space with distance given by intervals and should be included. Besides, we would like to have a statement in Theorem 11 for minimum separation and an observation as provided in Corollary 2 that when $\alpha \in \{0, 1\}$, the admissible methods would be unique given by the single linkage clustering methods applied onto the distance upper or lower bounds.

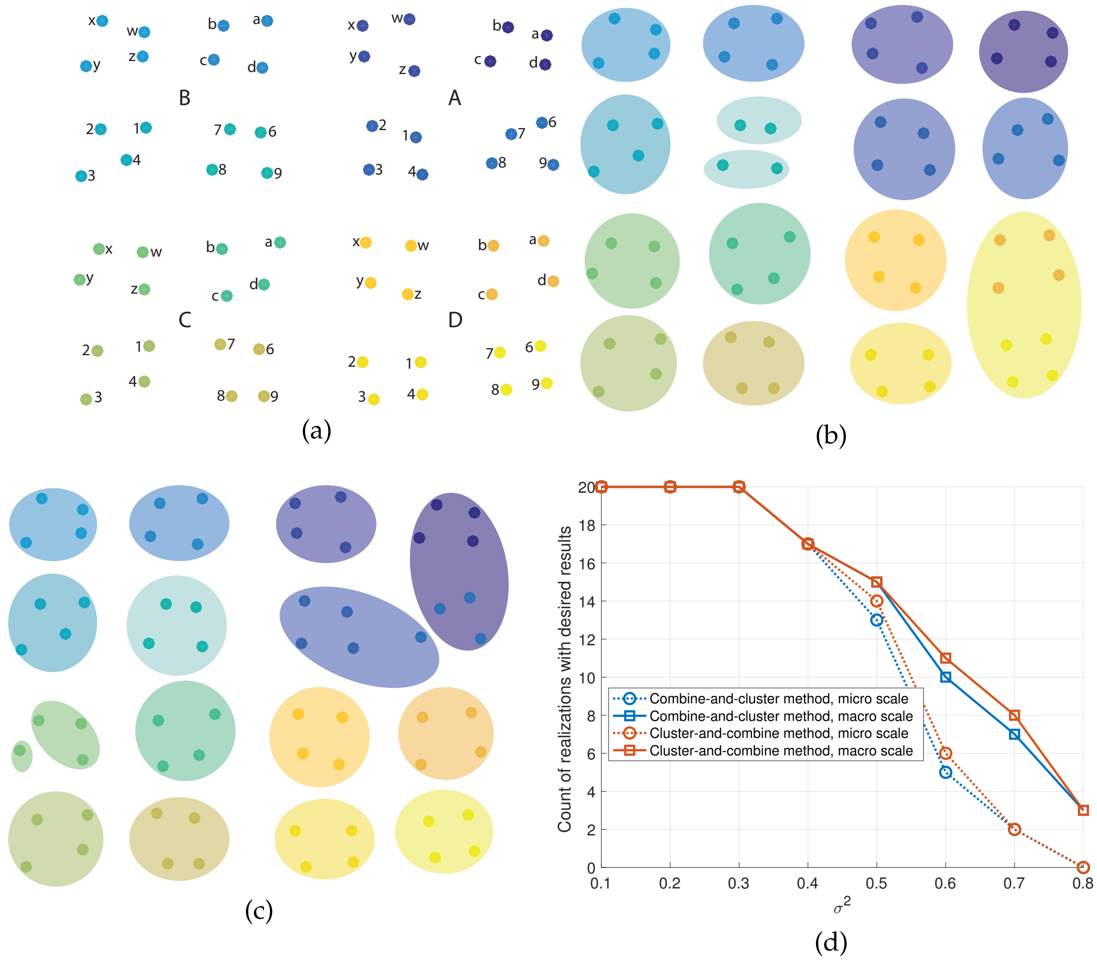
5.5. Applications

We illustrate the usefulness of clustering theory developed in previous sections through numerical experiments in both synthetic scenario (Section 5.5.1) and real world dataset (Section 5.5.2).

5.5.1. Clustering of Moving Points by Snapshots

We consider the clustering of n hierarchically clustered points moving in a two-dimensional plane with the initial coordinate of the i-th point represented by $\mathbf{p}_i^0 \in \mathbb{R}^2$. Points are moving in the plane and we have T snapshots with $\mathbf{p}_i^t \in \mathbb{R}^2$ denoting the coordinate of the i-th point at the t-th snapshot. We assume that the directions of movement of points are completely random and therefore model the observation as $\mathbf{p}_i^t := \mathbf{p}_i^{t-1} + \boldsymbol{\epsilon}$ for any i and time index $1 \leq t \leq T$, where $\boldsymbol{\epsilon} \in \mathbb{R}^2$ is a two-dimensional independent zero-mean Gaussian random variable with covariance matrix $\phi^2\mathbf{I}$. Having no knowledge about the starting coordinates, we would like to cluster points based on observations $\{\mathbf{p}_i^t\}_{i=1,\ldots,n, t=1,\ldots,T}$. To do so, we consider the node set X where $x_i \in X$ denotes the i-th point $\mathbf{p}_i$, and use $d_X^t(x_i, x_j) = \|\mathbf{p}_i^t - \mathbf{p}_j^t\|_2$ to represent the distance between the i-th and the j-th points at the t-th snapshot. Then we define dissimilarity space with distances given by intervals $(X, \underline{d}_X, \bar{d}_X)$ such that given a pair of nodes $x_i \neq x_j$, we set the distance lower bound $\underline{d}_X(x_i, x_j) = \min_{1 \leq t \leq T} d_X^t(x_i, x_j)$ as the minimum distance between the pair at all snapshots. Similarly, we define the distance upper bound $\bar{d}_X(x_i, x_j) = \max_{1 \leq t \leq T} d_X^t(x_i, x_j)$ as the maximum distance between the pair among all snapshots. Clustering methods are then applied upon the triplet $(X, \underline{d}_X, \bar{d}_X)$.

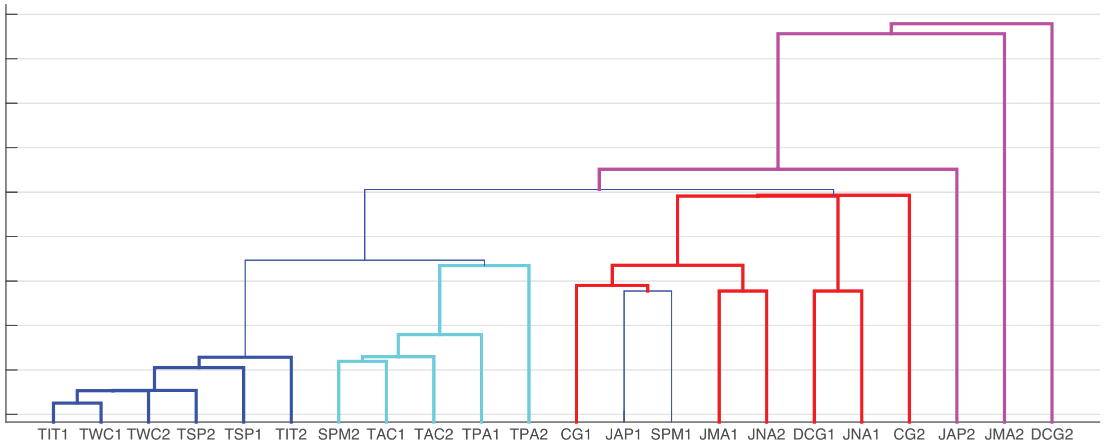
As an example, we consider $n = 64$ points whose initial coordinates form a hierarchically clustered structure as in Figure 37 (a), and investigate $T = 10$ snapshots of these moving points. We apply cluster-and-combine clustering $\mathcal{W}^{\mathrm{CL}}$ and combine-and-cluster clustering $\mathcal{W}^{\mathrm{CO}}$ onto the distance bounds $(X, \underline{d}_X, \bar{d}_X)$. Figure 39 (a) and (b) show the output dendrograms of cluster-and-combine and combine-and-cluster methods, respectively. The variance parameter ϕ^2 of movement $\boldsymbol{\epsilon}$ is set as 0.45 and the confidence level α as 0.5. We use two-digit to represent each node, as exemplified in Figure 37 (a). The first digit represent the quadrant where the point locates: A denotes the 16 points in the upper right quadrant, B denotes the 16 points in the upper left quadrant, C denotes



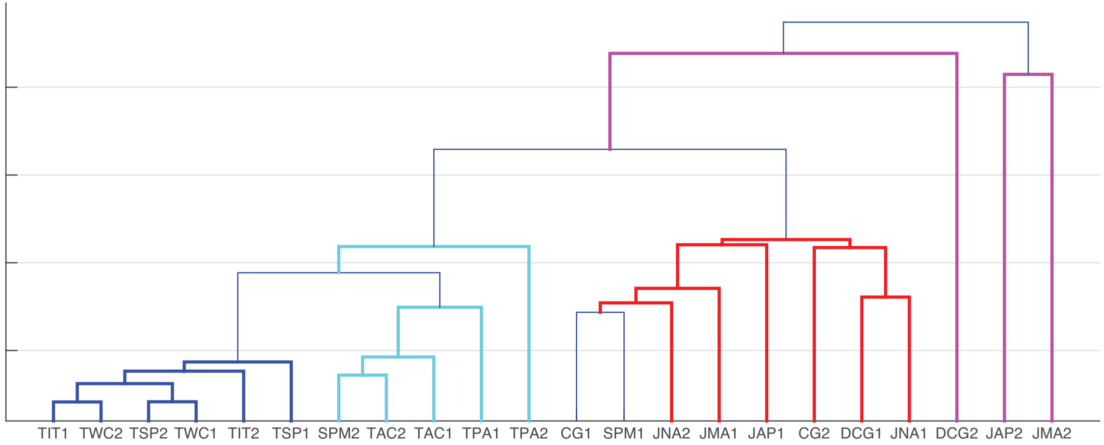
(a) (b) (c) (d)

Figure 37: Synthetic experiment of clustering of moving points based on multiple snapshots. (a) Initial positions of points, which correspond to a hierarchically clustered structure. Points are labeled using two digits, where the first digit denotes the quadrant and the second digit denotes the relative location within each quadrant. (b) A typical classification result of applying uncertainty *k*-means onto the dataset by considering each snapshot as a sample for that uncertain point. (c) A typical classification result of applying uncertainty *k*-medians onto the dataset by considering each snapshot as a sample for that uncertain point. (d) Out of 20 random realizations, count of realizations with desired results produced by the proposed clustering methods, with respect to different level of perturbation.

the 16 points in the lower left quadrant, and D denotes the 16 points in the lower right quadrant. For each of the 16 points in a given quadrant, points in the four clusters are represented by the second digit as a, b, c, d, or w, x, y, z, or 1, 2, 3, 4, or 6, 7, 8, 9. Points coming from the same cluster are plotted with the same color in Figure 39 (a) and (b). It can be seen that both $\mathcal{W}^{\text{CL}}$ and $\mathcal{W}^{\text{CO}}$ yield the desired output: (i) at macro scale, there are four major clusters A, B, C, D, and (ii) at micro scale, each major cluster consists of four clusters of points represented by a, b, c, d, and w, x, y, z, and 1, 2, 3, 4, and 6, 7, 8, 9, respectively. The fact that limited differences exist between the two dendrograms mean that the space of all hierarchical cluster methods is not rich (Theorem



(a) Cluster-and-combine dendrogram



(b) Combine-and-cluster dendrogram

Figure 38: Resulting dendrograms of cluster-and-combine method (a) and combine-and-cluster method (b) applied upon the triplet $(X, \underline{d}_X, \bar{d}_X)$ in the coauthorship network clustering. Each node in the dendrograms represents a coauthorship network constructed from a journal during a quinquennium. The difference between the output dendrograms of the two methods is small. Besides, cooperation pattern in engineering community is different from that in math community; within engineering community, Signal Processing, Information Theory, and Wireless Communication have different collaboration traits from Automatic Control and Pattern Recognition.

12).

As a benchmark for hierarchical clustering results, we consider the mean distance between any pair of nodes $\tilde{d}_X(x,x') = \frac{1}{T}\sum_{t=1}^{T} d_X^t(x,x')$ and apply single linkage upon $(X, \tilde{d}_X)$. Figure 39 (c) shows the resulting dendrogram, which fails to identify the clustering structure correctly: (i) at micro scale, the small cluster Ca, Cb, Cc, Cd and the cluster C6, C7, C8, C9 are not identified properly, (ii) at macro scale, the larger clusters C and D are not classified as expected. This indicates that synthesizing distance bounds, as in the proposed methods, improves the clustering result compared to only using mean distance. Besides hierarchical clustering, the problem of clustering moving points can also be casted as clustering points with uncertainty where the location at each snapshot can be considered as a sample for that point. Despite the fact that clustering points with distance intervals use a subset of information in the location samples, we evaluate the clustering results of previously proposed methods: uncertainty k-means and uncertainty k-medians [46, 39, 47]. For each random initialization in k-means and k-medians, we tune the number of clusters k to make the final clustering output have as close to 16 clusters as possible. We tried 30 different random initializations for each method. For uncertainty k-means, no initialization gives perfect classification of the 16 clusters. The percentage of points misclassified averaged across the realizations is 10.83%, while the median misclassification percentage across the realizations is 10.94%; a typical classification result is shown in Figure 37 (b). For uncertainty k-median, 1 out of 30 initializations gives perfect classification of the 16 clusters. The average percentage of points misclassified across the realizations is 8.33%, while the median across the realizations is 7.81%; a typical classification result is shown in Figure 37 (c).

Three supplementary tests are designed to strengthen the experiment. Firstly, to test the stability of the methods with respect to the parameter α. We examine the resulting dendrograms of the two methods by varying α from 0 to 1. For this specific example, parameter with $0 \leq \alpha \leq 0.75$ yields desired major clusters A, B, C, D at the macro scale; parameter with $0 \leq \alpha \leq 0.65$ makes major cluster consisting of four clusters of points represented by a, b, c, d, and w, x, y, z, and 1, 2, 3, 4, and 6, 7, 8, 9, respectively. This indicates that the results are stable within a relatively wide range of α. The actual stable range may change with respect to specific scenarios, but we expect stability to the parameter to hold in general. We expect theoretical result regarding stability can be established, and leave it for future work. Secondly, to examine the robustness of the clustering methods to the initial position of points as well as movement, we generate 50 sets of initial coordinates and the movement perturbations of these points. Out of the 50 sets, combine-and-cluster method yields 36 (72%) desired macro scale results – four major clusters A, B, C, D, and 33 (66%) desired micro scale results – each major cluster consisting of four clusters of points; cluster-and-combine method yields 37 (74%) desired macro scale results and 35 (70%) desired micro scale results. The benchmark method, single linkage applied upon the mean distance, yield 27 (54%) desired macro scale results and 25 (50%) desired micro scale results. This shows that the clustering results as well as the comparison illustrated in the figures are generalizable. Thirdly, to study the impact of noise on the performance of the clustering results, we apply both the methods to random realizations of perturbations with different variance ϕ^2. For each ϕ^2, we count the number of realizations such that the resulting dendrograms are as expected, and plot the counts with respect to ϕ^2 in Figure 37 (d). For reference, the average distance between adjacent points coming from same cluster in Figure 37 (a) is 2. Both proposed methods yield desired results in most cases for $\phi^2 \leq 0.5$, in both macro and micro scales. Performance begins to deteriorate with higher variance ϕ^2. If we use the ratio between the distance of adjacent points and the variance ϕ^2 of perturbation as a rough estimate of signal-to-noise ratio, the methods yield good results with higher than $2/0.4 = 5$ signal-to-noise ratio. In the experiment studied, the proposed methods exhibit some robustness to uncertainty in measurements.

5.5.2. Clustering of Networks via Distance Bounds

In this section, we go back to the motivating starting point of this work – clustering of networks. We achieve that by estimating the lower and upper bound of valid distance metrics in the space of networks, and apply the clustering methods proposed in this chapter. The problem of comparing and clustering networks is interesting on its own [153, 16]. In Section 2, we have defined network distances. The metric distances defined here have been applied to compare networks with small number of nodes and have succeeded in identifying collaboration patterns of different researchers [8]. However, because they have to consider all possible node correspondences, network distances are difficult to compute when the number of nodes in the networks is large. To resolve such problem, we mapped networks to filtrations of simplicial complexes and demonstrated that the difference between the homological features of their respective filtration can be used as a lower bound of $d_{\mathcal{N}}$ [121]. Computational of homological features is fast [137], which enables efficient estimation of the network distance lower bound.

On the other hand, $\Gamma_{Z,W}(C)$ in (2.8) for any correspondence C witnesses an upper bound on the distance $d_{\mathcal{N}}$. Therefore, given a set of networks X where the i-th element denotes a network N_i, we can evaluate the upper and lower bounds of network distance $d_{\mathcal{N}}(N_i, N_j)$ for any pair of networks N_i and N_j in X to yield a metric in the space of networks where distances are given by intervals $(X, \underline{d}_X, \bar{d}_X)$. Clustering methods examined in the chapter can then be applied towards the triplet to categorize networks.

As an example of network clustering, we consider coauthorship networks, where dissimilarity between nodes denote a preset number (the number of publication by the most prolific author in the network) minus the number of publications between pairs of authors. We consider publications in 5 journals from mathematics community: Computational Geometry (CG), Discrete Computational Geometry (DCG), J. of Applied Probability, (JAP) J. of Mathematical Analysis and Applications (JMA), SIAM J. on Numerical Analysis (JNA), and 6 journals from engineering community, all from IEEE: Signal Processing Magazine (SPM), Trans. Automatic Control (TAC), Trans. Pattern Analysis and Machine Intelligence (TPA), Trans. Information Theory, Trans. Signal Processing (TSP), Trans. Wireless Communication (TWC). For each journal, we construct two networks for the 2004-2008 and 2009-2013 quinquennia. Lists of publications are queried from Engineering Village[1].

For each of these journals we consider all publications in the period of interest and construct networks where the node set Z is formed by all authors of the publications. To make networks with different numbers of publications comparable, we normalize all dissimilarities by the number of publications of the most prolific author in that network. By assuming that networks from the same community or constructed from the same journal have similar collaboration patterns, we show that the network clustering tools proposed here succeed in clustering research communities with similar research interests.

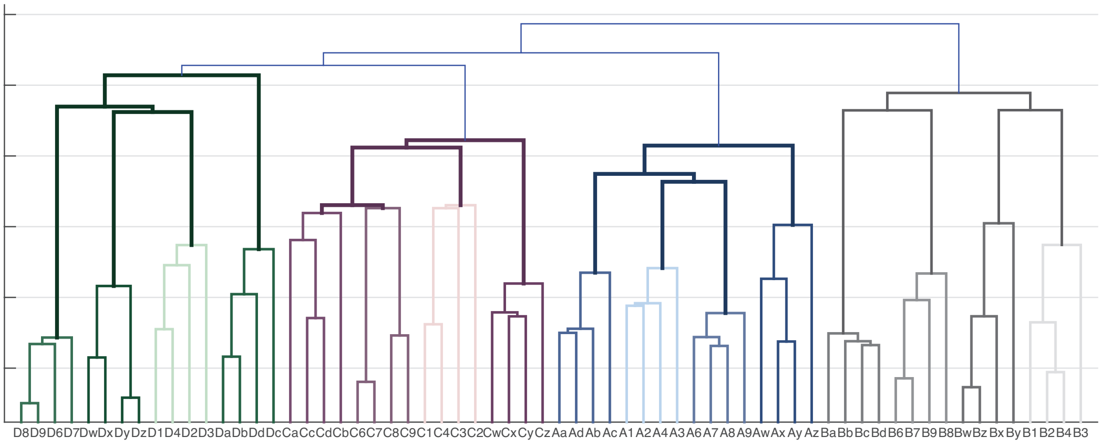
---

[1] http://www.engineeringvillage.com/search/quick.url

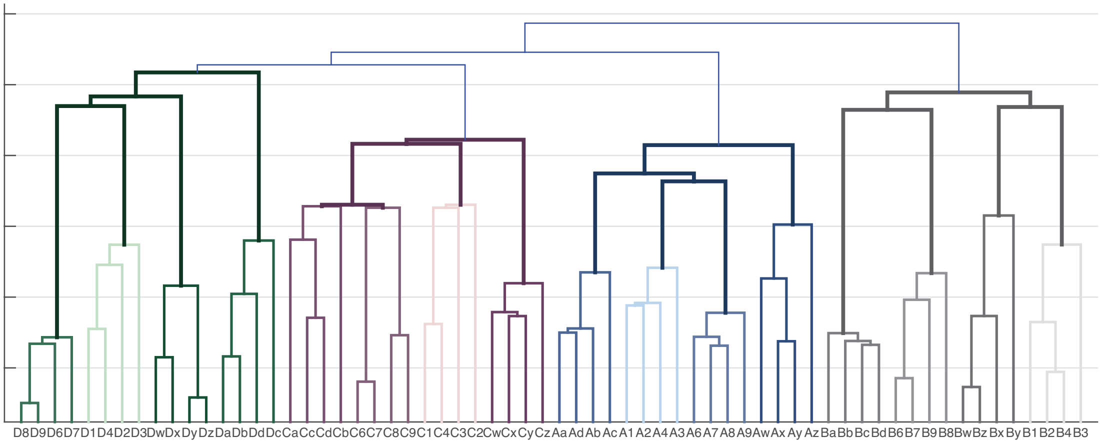
In order to cluster the networks, for each pair of coauthorship networks N_Z and N_W, we need to construct the distance lower bound $\underline{d}_{\mathcal{N}}(N_Z, N_W)$ and the distance upper bound $\bar{d}_{\mathcal{N}}(N_Z, N_W)$. The lower bounds are constructed using bottleneck distances between persistence diagrams established in our previous contribution [121, 76]. For the upper bound, denote the set of authors in the networks N_Z and N_W as Z and W respectively. In theory, any correspondence $C \subseteq Z \times W$ would witness an upper bound. However, to achieve good performance, the correspondence cannot be chosen arbitrarily. In specific, for each author $z \in Z$ in N_Z, we record her normalized number of publications as $s_Z(z)$; similarly, for each author $w \in W$ in N_W, we record her normalized number of publications as $s_W(w)$. The correspondence C is constructed in two steps. Firstly, starting with $C = \emptyset$, for each $z \in Z$, we add $(z, w^\star)$ to C where $w^\star = \operatorname{argmin}_{w \in W} |s_Z(z) - s_W(w)|$. If multiple authors in W yield the same discrepancy, a single author w is chosen randomly from the candidate set. Secondly, for each $w \in W$, we add $(z^\star, w)$ to C where $z^\star = \operatorname{argmin}_{z \in Z} |s_Z(z) - s_W(w)|$. The correspondence C constructed in this way is guaranteed to be a valid correspondence. The difference between the network $\Gamma_{Z,W}(C)$ with respect to this correspondence C defined in (2.8) is then used as the distance upper bound $\bar{d}_{\mathcal{N}}(N_Z, N_W)$.

We apply the proposed methods upon the metric distance bounds $(X, \underline{d}_X, \bar{d}_X)$ between these coauthorship networks with confidence level $\alpha = 0.5$. Figure 38 shows the two dendrograms corresponding to the two methods, where networks are labeled based on their journal names and the quinquennia they are constructed from (1 means 2004 - 2008, and 2 means 2009 - 2013). The two dendrograms are not highly different from each other, indicating again that the space of all admissible methods satisfying the axioms is not rich. Many interesting observations emerge from the figures: Firstly, there is a difference in its collaboration pattern between networks constructed from engineering journals (blue and cyan leafs in the dendrograms) and networks constructed from math journals (red and magenta leafs in the dendrograms). SPM1 is the only network that fall into the wrong cluster. Part of the reason is that SPM is a Magazine, and the collaboration pattern is slightly different from Transactions in engineering community. Both the dendrograms appear to be left-branching; the main reason is because networks from engineering journals are similar to each other in both the distance lower and upper bounds, whereas the networks from the mathematical journals are more different from each other in the distance bounds. Secondly, TSP, TIT, and TWC have similar collaboration patterns (blue leafs), which is different from the collaboration pattern in TAC, TPA (cyan leafs). This demonstrates the value of hierarchical clustering onto networks: cooperation pattern in engineering community is different from that in math community, and within engineering community, TSP, TIT, and TWC have different collaboration traits from TAC and TPA. Thirdly, compared to TSP, TIT, and TWC, there is more discrepancy between the networks of TAC and TPA constructed from different quinquennia. Fourthly, out of all 5 Transactions from engineering community, TPA is more different in collaboration patterns. Finally, despite the fact that networks from math community are more different to each other compared to networks from engineering community, the collaboration pattern in math journals in quinquennium 2004 - 2008 (mostly red leafs) are similar to each other, and are different from the pattern in math journals in quinquennium 2009 - 2013 (mostly magenta leafs). Clustering results using both distance upper and lower bounds are slightly better than the results using only the lower bound we investigated in previous contribution [121].

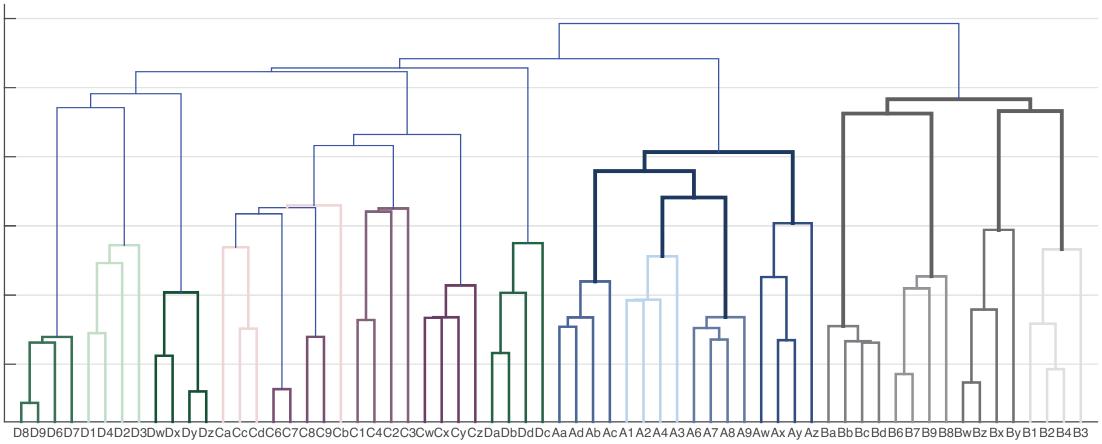
The proposed methods yield clustering structures only based on the upper and lower bound of actual network distance. No direct comparison can be made with other methods. Nonetheless, we compare the result with clustering using uncertainty in the feature space. To do so, we consider each network representing an uncertain point in a two-dimensional plane with the respective axis denoting the normalized number of publications, and the number of coauthors. Each actual author in this network denotes one sample in this space, with the coordinate in the two-dimensional plane denoting the number of publication and the number of coauthors of this author. We can then apply uncertainty k-means and uncertainty k-medians to cluster networks using this features. Setting $k = 2$, these exclusive clustering methods based on uncertainty would yield two cluster structures, one representing networks from engineering community, and the other denoting networks from math community; the cluster structures only misclassify SPM1. Methods to compare networks via uncertain features give us similar observations as those based on the proposed hierarchical clustering using distance bounds, despite the facts that these approaches use different information in the networks and originate from very different perspectives.



(a) Cluster-and-combine dendrogram



(b) Combine-and-cluster dendrogram



(c) Benchmark 1: dendrogram via mean distance

Figure 39: Resulting dendrograms of (a) cluster-and-combine method and (b) combine-and-cluster method applied upon the triplet $(X, \underline{d}_X, \bar{d}_X)$ in the synthetic experiment. (c) Benchmark dendrogram: single linkage applied upon the mean distance between any pair of nodes $\tilde{d}_X(x,x') = \frac{1}{T}\sum_{t=1}^{T} d_X^t(x,x')$.

Chapter 6

Network Comparison via Embeddings and Interiors

The idea of an embedding distance $d(A, B)$ is to analyze how much we have to modify network A to make it a subset of network B. This is an asymmetric relationship. In particular, having $d(A, B) = 0$ means that network A can be embedded in network B but the opposite need not be true. In this chapter, we first define network *embeddings* and a corresponding notion of partial embedding distances. Partial embedding distances define an embedding metric d such that $d(A, B) = 0$ if and only if A can be embedded in B. We attempt to use the MDS techniques in [51] to approximate the computation of embedding distances but observe that the methodology yields sub-optimal results – see Figure 40 for an illustration of why this is not unexpected. To improve these results we observe that when edge dissimilarities satisfy triangle inequality, an Euclidean interior is implicitly defined. In the case of arbitrary networks this is not true and motivates the definition of the interior of a network that we undertake in Section 6.2. The chapter then provides a definition of the *interior* of a network. The interior of a set of nodes is the set of points that can be written as convex combinations of the nodes. When the network forms a metric space, the dissimilarity between a pair of points in the interior is the distance on the shortest path between the pair. When the dissimilarities in the network do not form a metric space, e.g. representing travel time between nodes, such construction would yield conflict. The problem can be solved by defining the dissimilarity between a pair of points as the travel time on the shortest path between the pair. Having the ability to extend networks into their interiors, we extend different networks and compute partial embedding distances between their extensions. In principle, distances between two networks and their respective extensions need not be related. In Section 6.2.1 we show that a restriction in the embedding of the extended networks renders them identical. Therefore, we define embeddings for extended networks such that points in one of the original networks – prior to extension – can only be embedded into original points of the other network. We show that the embedding distance that results from this restriction is the same embedding distance between the original networks.

The definition of interior is somewhat arbitrary, however because of the identical network distance under embedding, the practical implication of interior definition is justified. We point out that a network extension is a dense set that includes all the convex combinations of sets of points. To make interior extensions practical we consider samplings of the interior in Section 6.2.2. It is not difficult to show that the embedding distance between a pair of networks extended to samples of their interiors is also identical to the embedding distance between the original pair of networks – if the restriction in the mapping of original nodes is retained.

We exploit these definitions to approximate the computation of embedding distances using the MDS techniques in [51] but applied to networks extended to their interiors. The definition of an interior improves the quality of MDS distance approximations. We illustrate this fact in Section 6.3 with illustrative examples and also demonstrate the ability to discriminate networks with different generative models.

6.1. Embeddings

A related notion to metric space defined in Definition 3 is that of an isometric embedding. We say that a map $\pi : X \to Y$ is an isometric embedding from $N_X = (X, r_X)$ to $N_Y = (Y, r_Y)$ if (2.1) holds for all points $x, x' \in X$. Since $r_X(x, x') = r_Y(\pi(x), \pi(x'))$ for any $x, x' \in X$, $r_X(x, x') > 0$ for $x \neq x'$ and $r_Y(y, y) = 0$, the map π is injective. This implies that the condition can only be satisfied when $N_X = (X, r_X)$ is a sub-network of $N_Y = (Y, r_Y)$. Such a map is called an isometric embedding. When N_X can be isometrically embedded into N_Y, we write $N_X \sqsubseteq N_Y$. Related to the notion of isometric embedding is the notion of an embedding metric that we state next.

Definition 25 *Given a space $\mathcal{S}$ and an isometric embedding $\sqsubseteq$, a function $d : \mathcal{S} \times \mathcal{S} \to \mathbb{R}$ is an embedding metric in $\mathcal{S}$ if for any $a, b, c \in \mathcal{S}$ the function d satisfies:*

(i) **Nonnegativity.** $d(a, b) \geq 0$.

(ii) **Embedding identity.** $d(a, b) = 0$ *if and only if* $a \sqsubseteq b$.

(iii) **Triangle inequality.** $d(a, b) \leq d(a, c) + d(c, b)$.

As is the case with correspondences, mappings also allow definition of associations between networks with different numbers of elements. We use this to define the distance from one network to another network by selecting the mapping that makes them most similar as we formally define next. In this chapter, we focus the case with $K = 2$ and remove the superscript K for the ease of notation.

Definition 26 *Given two networks $N_X = (X, r_X)$, $N_Y = (Y, r_Y)$, and a map $\pi : X \to Y$ from node space*

X to the node space Y, define the network difference with respect to π as

$$\Delta_{X,Y}(\pi) := \max_{x,x' \in X} \Big| r_X(x,x') - r_Y(\pi(x),\pi(x')) \Big|. \tag{6.1}$$

The partial embedding distance from N_X to N_Y is defined as

$$d_{\mathrm{PE}}(N_X, N_Y) := \min_{\pi : X \to Y} \Big\{ \Delta_{X,Y}(\pi) \Big\}. \tag{6.2}$$

Both, Definition 2 and Definition 26 consider a mapping between the node space X and the node space Y, compare dissimilarities, and set the network distance to the comparison that yields the smallest value in terms of maximum differences. The distinction between them is that in (2.2) we consider correspondence, which requires each point in any node spaces (X or Y) to have a correspondent in the other node space, whereas in (6.1) we examine mappings, which only require all points in node space X to have one correspondent in the node set Y. Moreover, in (2.2), a node $x \in X$ may have multiple correspondents, however, in (6.1), a node $x \in X$ can only have exactly one correspondent. Except for this distinction, Definition 2 and Definition 26 are analogous since $\Delta_{X,Y}(\pi)$ selects the difference $|r_X(x_1,x_2) - r_Y(y_1,y_2)|$ among all pairs. The distance $d_{\mathrm{PE}}(N_X, N_Y)$ is defined by selecting the mapping that minimizes these maximal differences. We show in the following proposition that the function $d_{\mathrm{PE}} : \mathcal{N} \times \mathcal{N} \to \mathbb{R}_+$ is, indeed, an embedding metric in the space of networks.

Proposition 9 *The function $d_{\mathrm{PE}} : \mathcal{N} \times \mathcal{N} \to \mathbb{R}_+$ defined in* (6.2) *is an embedding metric in the space $\mathcal{N}$.*

Proof: To prove that d_{PE} is an embedding metric in the space of networks, we prove the (i) nonnegativity, (ii) embedding identity, and (iii) triangle inequality properties in Definition 25.

Proof of nonnegativity property: Since $|r_X(x,x') - r_Y(\pi(x),\pi(x'))|$ is nonnegative, $\Delta_{X,Y}(\pi)$ defined in (6.1) also is. The partial embedding distance must then satisfy $d_{\mathrm{PE}}(N_X, N_Y) \geq 0$ because it is a minimum of nonnegative numbers. ■

Proof of embedding identity property: First, we need to show that if N_X can be isometrically embedded into N_Y, we must have $d_{\mathrm{PE}}(N_X, N_Y) = 0$. To see that this is true recall that for isometric embeddable networks, there exists a mapping $\pi : X \to Y$ that preserves distance functions (2.1). Then, under this mapping, we must have $\Delta_{X,Y}(\pi) = 0$. Since π is a particular mapping, taking a minimum over all mappings as in (6.2) yields

$$d_{\mathrm{PE}}(N_X, N_Y) \leq \Delta_{X,Y}(\pi) = 0. \tag{6.3}$$

Since $d_{\mathrm{PE}}(N_X, N_Y) \geq 0$, it must be that $d_{\mathrm{PE}}(N_X, N_Y) = 0$ when N_X can be isometrically embedded into N_Y.

Second, we need to prove $d_{\mathrm{PE}}(N_X, N_Y) = 0$ must imply that N_X can be isometrically embedded into N_Y. If $d_{\text{PE}}(N_X, N_Y) = 0$, there exists a mapping $\pi : X \to Y$ such that $r_X(x, x') = r_Y(\pi(x), \pi(x'))$ for any $x, x' \in X$. This implies that π is an isometric embedding and therefore N_X can be isometrically embedded into N_Y. ■

Proof of triangle inequality: To show that the triangle inequality, let the mapping π between X and Z and ω between Z and Y be the minimizing mappings in (6.2). We can then write

$$d_{\text{PE}}(N_X, N_Z) = \Delta_{X,Z}(\pi),\ d_{\text{PE}}(N_Z, N_Y) = \Delta_{Z,Y}(\omega). \tag{6.4}$$

Since both π and ω are mappings, $\omega \circ \pi$ would be a valid mapping from X to Y. The mapping $\omega \circ \pi$ may not be the minimizing mapping for the distance $d_{\text{PE}}(N_X, N_Y)$. However since it is a valid mapping with the definition in (6.2) we can write

$$d_{\text{PE}}(N_X, N_Y) \leq \Delta_{X,Y}(\omega \circ \pi). \tag{6.5}$$

Adding and subtracting $d_Z(\pi(x), \pi(x'))$ in the absolute value of $\Delta_{X,Y}(\omega \circ \pi) = \max_{x,x' \in X} \big| r_X(x, x') - r_Y(\omega(\pi(x)), \omega(\pi(x')))\big|$ and using the triangle inequality of the absolute value yields

$$\Delta_{X,Y}(\omega \circ \pi) \leq \max_{x,x' \in X} \Big\{ \big| r_X(x, x') - d_Z(\pi(x), \pi(x'))\big| + \Big| d_Z(\pi(x), \pi(x')) - r_Y\big(\omega\big(\pi(x)\big), \omega\big(\pi(x')\big)\big)\Big| \Big\}. \tag{6.6}$$

We can further bound (6.6) by taking maximum over each summand,

$$\begin{aligned} \Delta_{X,Y}(\omega \circ \pi) \leq & \max_{x,x' \in X} \big| r_X(x, x') - d_Z(\pi(x), \pi(x'))\big| \\ & + \max_{x,x' \in X} \Big| d_Z(\pi(x), \pi(x')) - r_Y\big(\omega\big(\pi(x)\big), \omega\big(\pi(x')\big)\big)\Big|. \end{aligned} \tag{6.7}$$

The first summand in (6.7) is nothing different from $\Delta_{X,Z}(\pi)$. Since $\pi(x), \pi(x') \in Z$, the second summand in (6.7) can be further bounded by

$$\begin{aligned} & \max_{x,x' \in X} \Big| d_Z(\pi(x), \pi(x')) - r_Y\big(\omega\big(\pi(x)\big), \omega\big(\pi(x')\big)\big)\Big| \\ & \leq \max_{z,z' \in Z} \big| d_Z(z, z') - r_Y\big(\omega(z), \omega(z')\big)\big| = \Delta_{Z,Y}(\omega). \end{aligned} \tag{6.8}$$

These two observations implies that

$$\Delta_{X,Y}(\omega \circ \pi) \leq \Delta_{X,Z}(\pi) + \Delta_{Z,Y}(\omega). \tag{6.9}$$

Substituting (6.4) and (6.5) into (6.9) yields triangle inequality. ■

Having proven all statements, the global proof completes. ■

The embedding distance $d_{\text{PE}}(N_X, N_Y)$ from one network N_X to another network N_Y is not a metric due to its asymmetry. We can construct a symmetric version from $d_{\text{PE}}(N_X, N_Y)$ by taking the maximum from the embedding distance $d_{\mathrm{PE}}(N_X, N_Y)$ and $d_{\mathrm{PE}}(N_Y, N_X)$. This would give us a valid metric distance in $\mathcal{N} \mod \cong$. A formal definition and theorem are shown next.

Definition 27 *Given two networks $N_X = (X, r_X)$, $N_Y = (Y, r_Y)$, define the embedding distance between the pair as*

$$d_{\mathrm{E}}(N_X, N_Y) := \max\left\{d_{\mathrm{PE}}(N_X, N_Y), d_{\mathrm{PE}}(N_Y, N_X)\right\}. \tag{6.10}$$

where partial embedding distances $d_{\mathrm{PE}}(N_X, N_Y)$ and $d_{\mathrm{PE}}(N_X, N_Y)$ are defined in Definition 26.

Theorem 13 *The function $d_{\mathrm{E}} : \mathcal{N} \times \mathcal{N} \to \mathbb{R}_+$ defined in* (6.10) *is a metric in the space $\mathcal{N} \mod \cong$.*

Proof: To prove that d_{E} is a metric in the space of networks modulo isomorphism, we prove the (i) nonnegativity, (ii) symmetry, (iii) identity, and (iv) triangle inequality properties in Definition 3.

Proof of nonnegativity property: Since $d_{\mathrm{PE}}(N_X, N_Y)$ as well as $d_{\mathrm{PE}}(N_Y, N_X)$ are both nonnegative, the embedding distance must then satisfy $d_{\mathrm{E}}(N_X, N_Y) \geq 0$. ■

Proof of symmetry property: Since $d_{\mathrm{E}}(N_X, N_Y) = d_{\mathrm{E}}(N_Y, N_X) = \max\{d_{\mathrm{PE}}(N_X, N_Y), d_{\mathrm{PE}}(N_Y, N_X)\}$, the symmetry property follows directly. ■

Proof of identity property: First, we need to show that if N_X and N_Y are isomorphic, we must have $d_{\mathrm{E}}(N_X, N_Y) = 0$. To see that this is true recall that for isomorphic networks there exists a bijective map $\pi : X \to Y$ that preserves distance functions (2.1). This implies π is also an injection, and we can find an injection $\omega : Y \to X$ that preserves distance functions (2.1). Then, under the injection π, we must have $\Delta_{X,Y}(\pi) = 0$. Since π is a particular mapping, taking a minimum over all mappings as in (6.2) yields

$$d_{\mathrm{PE}}(N_X, N_Y) \leq \Delta_{X,Y}(\pi) = 0. \tag{6.11}$$

Since $d_{\mathrm{PE}}(N_X, N_Y) \geq 0$, as already shown, it must be that $d_{\mathrm{PE}}(N_X, N_Y) = 0$ when N_X are isomorphic to N_Y. Similarly, we can show that $d_{\mathrm{PE}}(N_Y, N_X) = 0$, which combines with previous observation implies that $d_{\mathrm{E}}(N_X, N_Y) = 0$.

Second, we need to prove $d_{\mathrm{E}}(N_X, N_Y) = 0$ must imply that N_X and N_Y are isomorphic. By the definition of embedding distance, $d_{\mathrm{E}}(N_X, N_Y) = 0$ means $d_{\mathrm{PE}}(N_X, N_Y) = 0$ and $d_{\mathrm{PE}}(N_Y, N_X) = 0$. The observation $d_{\mathrm{PE}}(N_X, N_Y) = 0$ implies that there exists a mapping $\pi : X \to Y$ such that $r_X(x, x') = r_Y(\pi(x), \pi(x'))$ for any $x, x' \in X$. Moreover, this also implies the function π must be injective. If it were not, there would be a pair of nodes $x \neq x'$ with $\pi(x) = \pi(x') = y$ for some $y \in Y$. By the definition of networks, we have that

$$r_X(x, x') > 0, \quad r_Y(\pi(x), \pi(x')) = r_Y(y, y) = 0, \tag{6.12}$$

which contradicts the observation that $r_X(x, x') = r_Y(\pi(x), \pi(x'))$ for any $x, x' \in X$ and shows that π must be injective. Following similar derivation, the fact $d_{\text{PE}}(N_Y, N_X) = 0$ implies that there exists an injective mapping $\omega : Y \to X$ such that $r_Y(y, y') = r_X(\omega(y), \omega(y'))$ for any $y, y' \in Y$. By applying the Cantor-Bernstein-Schroeder Theorem (also known as Cantor-Bernstein or Schroeder-Bersteirn) [134, Section 2.6] to the reciprocal injections $\pi : X \to Y$ and $\omega : Y \to X$, the existence of a bijection between X and Y is guaranteed. This forces X and Y to have same cardinality and π and ω being bijections. Pick the bijection π and it follows $r_X(x, x') = r_Y(\pi(x), \pi(x'))$ for all nodes $x, x' \in X$. This shows that $N_X \cong N_Y$ and completes the proof of the identity statement. ■

Proof of triangle inequality: To show that the triangle inequality holds, from the definition of embedding distance, we have that

$$d_{\text{E}}(N_X, N_Y) = \max \{d_{\text{PE}}(N_X, N_Y), d_{\text{PE}}(N_Y, N_X)\}. \tag{6.13}$$

Since partial embedding distance is a valid embedding metric, it satisfies triangle inequality in Definition 25, therefore, we can bound (6.13) by

$$d_{\text{E}}(N_X, N_Y) \leq \max \big\{d_{\text{PE}}(N_X, N_Z) + d_{\text{PE}}(N_Z, N_Y), d_{\text{PE}}(N_Y, N_Z) + d_{\text{PE}}(N_Z, N_X)\big\}. \tag{6.14}$$

To further bound (6.14) we utilize the relationship as next.

Fact 5 *Given real numbers a, b, c, d, it holds that*

$$\max\{a, c\} + \max\{b, d\} \geq \max\{a + b, c + d\}. \tag{6.15}$$

Proof: If $a \geq c$ and $b \geq d$, the inequality holds since the left hand side is $a + b$ and the right hand side is also $a + b$. Similarly, if $c \geq a$ and $d \geq b$, the inequality also holds. What remains to consider are scenarios of $a \geq c, d \geq b$ as well as $c \geq a, b \geq d$. Since the order is irrelevant, it suffices to consider the first scenario with $a \geq c, d \geq b$. Under this scenario, the statement becomes

$$a + d \geq \max\{a + b, c + d\}. \tag{6.16}$$

It follows that the state is correct following the assumption. Since we have considered all scenarios, the proof concludes. ■

Back to the proof of triangle inequality, applying Fact 5 onto (6.14) yields

$$d_{\text{E}}(N_X, N_Y) \leq \max \big\{d_{\text{PE}}(N_X, N_Z), d_{\text{PE}}(N_Z, N_X)\big\} + \max \big\{d_{\text{PE}}(N_Y, N_Z), d_{\text{PE}}(N_Z, N_Y)\big\}. \tag{6.17}$$

Substituting the definition of $d_{\text{E}}(N_X, N_Z)$ and $d_{\text{E}}(N_Z, N_Y)$ into (6.17) yields

$$d_{\text{E}}(N_X, N_Y) \leq d_{\text{E}}(N_X, N_Z) + d_{\text{E}}(N_Z, N_Y), \tag{6.18}$$

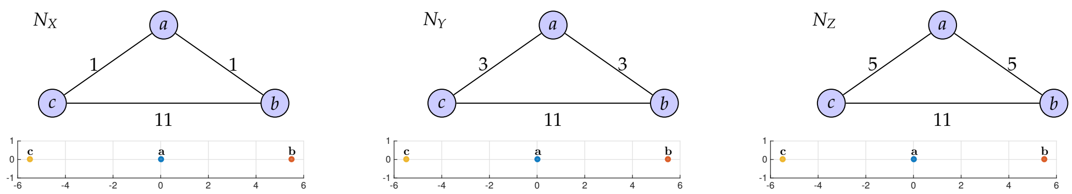


Figure 40: An example where different networks result in identical multi-dimensional scaling results. We emphasize that the number of dimension used in multi-dimensional scaling would not distinguish networks since the triangle inequality property for relationships between nodes in the networks is violated. Such a caveat would be solved by inducing semimetrics in the space defined by the given networks, as we develop throughout Section 6.2.

which is the triangle inequality and completes the proof. ■

Having proven all statements, the global proof completes. ■

Since embedding distances between two networks generate a well-defined metric, they provide a means to compare networks of arbitrary sizes. In comparing the embedding distance in (6.10) with the network distance in (2.3) we see that both find the bottleneck that prevents the networks to be matched to each other. It is not there surprising to learn that they satisfy the relationship that we state in the following proposition.

Proposition 10 *The network distance $d_{\mathcal{N}}(N_X, N_Y)$ defined in* (2.3) *can also be written as*

$$d_{\mathcal{N}}(N_X, N_Y) = \min_{\substack{\pi:X\to Y \\ \omega:Y\to X}} \max \left\{\Delta_{X,Y}(\pi), \Delta_{Y,X}(\omega), \delta_{X,Y}(\pi,\omega)\right\}, \tag{6.19}$$

where the network differences $\Delta_{X,Y}(\pi)$ and $\Delta_{Y,X}(\omega)$ with respect to mappings π and ω are defined in (6.1) *and $\delta_{X,Y}(\pi,\omega)$ measures how far the mappings π and ω are from being the inverse of each other, and is defined as*

$$\delta_{X,Y}(\pi,\omega) = \max_{x\in X, y\in Y} \left|r_X(x,\omega(y)) - r_Y(\pi(x),y)\right|. \tag{6.20}$$

A direct consequence of Proposition 10 is that the embedding distance (6.10) is a lower bound of the network distance (2.3).

Proof: Denote $d'_{\mathcal{N}}(N_X, N_Y)$ to represent $\min_{\pi:X\to Y, \omega:Y\to X} \max\{\Delta_{X,Y}(\pi), \Delta_{Y,X}(\omega), \delta_{X,Y}(\pi,\omega)\}$. In order to prove the statement, we show that given any networks N_X and N_Y, we have that (i) $d'_{\mathcal{N}}(N_X, N_Y) \leq d_{\mathcal{N}}(N_X, N_Y)$ and that (ii) $d_{\mathcal{N}}(N_X, N_Y) \leq d'_{\mathcal{N}}(N_X, N_Y)$.

Proof of $d'_{\mathcal{N}}(N_X, N_Y) \leq d_{\mathcal{N}}(N_X, N_Y)$: From the definition of $d_{\mathcal{N}}(N_X, N_Y)$, there exists a correspondent C such that $|r_X(x,x') - r_Y(y,y')| \leq d_{\mathcal{N}}(N_X, N_Y)$ for any $(x,y),(x',y') \in C$. Define a function $\pi : X \to Y$ that associates x with an arbitrary y chosen from the set that form a pair with

x in C,

$$\pi : x \mapsto y_0 \in \{y \mid (x, y) \in C\}. \tag{6.21}$$

Since C is a correspondence the set $\{y \mid (x, y) \in C\}$ is nonempty for any x implying that π is well-defined for any $x \in X$. Hence,

$$\left| r_X(x, x') - r_Y(\pi(x), \pi(x')) \right| \leq d_{\mathcal{N}}(N_X, N_Y), \tag{6.22}$$

for any $x, x' \in X$. Since (6.22) is true for any $x, x' \in X$, it also true for the maximum pair, and therefore

$$\Delta_{X,Y}(\pi) = \max_{x,x' \in X} \left| r_X(x, x') - r_Y(\pi(x), \pi(x')) \right| \leq d_{\mathcal{N}}(N_X, N_Y). \tag{6.23}$$

Define a function $\omega : Y \to X$ that associates y with an arbitrary x chosen from the set that form a pair with y in C,

$$\omega : y \mapsto x_0 \in \{x \mid (x, y) \in C\}. \tag{6.24}$$

Following the similar argument as above would yield us

$$\Delta_{Y,X}(\omega) \leq d_{\mathcal{N}}(N_X, N_Y). \tag{6.25}$$

Finally, recall that $\delta_{X,Y}(\pi, \omega)$ is defined as $\max_{x \in X, y \in Y} |r_X(x, \omega(y)) - r_Y(\pi(x), y)|$. In the same time, we have $(x, \omega(y)) \in C$ as well as $(\pi(x), y) \in C$, and therefore

$$\max_{x \in X, y \in Y} |r_X(x, \omega(y)) - r_Y(\pi(x), y)| \leq d_{\mathcal{N}}(N_X, N_Y). \tag{6.26}$$

Taking a maximum on both sides of inequlities (6.23), (6.25), and (6.26) yields

$$\max\{\Delta_{X,Y}(\pi), \Delta_{Y,X}(\omega), \delta_{X,Y}(\pi, \omega)\} \leq d_{\mathcal{N}}(N_X, N_Y). \tag{6.27}$$

The specific π and ω may not be the minimizing mappings for the left hand side of (6.27). Nonetheless, they are valid mappings and therefore taking a minimum over all mappings yields the desired inequality $d'_{\mathcal{N}}(N_X, N_Y) \leq d_{\mathcal{N}}(N_X, N_Y)$. ■

Proof of $d_{\mathcal{N}}(N_X, N_Y) \leq d'_{\mathcal{N}}(N_X, N_Y)$: From the definition of $d'_{\mathcal{N}}(N_X, N_Y)$, there exists a pair of mappings $\pi : X \to Y$ and $\omega : Y \to X$ such that

$$|r_X(x, x') - r_Y(\pi(x), \pi(x'))| \leq d'_{\mathcal{N}}(N_X, N_Y), \tag{6.28}$$

$$|r_X(\omega(y), \omega(y')) - r_Y(y, y')| \leq d'_{\mathcal{N}}(N_X, N_Y), \tag{6.29}$$

$$|r_X(x, \omega(y)) - r_Y(\pi(x), y)| \leq d'_{\mathcal{N}}(N_X, N_Y), \tag{6.30}$$

for any $x, x' \in X$ and $y, y' \in Y$. Define a correspondence by taking the union of the pairs associated by π and ω such that

$$C = \{(x, \pi(x)) \mid x \in X\} \cup \{(\omega(y), y) \mid y \in Y\}. \tag{6.31}$$

Since $\pi(x)$ is defined for any x and $\omega(y)$ is defined for any y, C is a well-defined correspondence. Notice that any pair $(x, y) \in C$ in the correspondence would be one of the following two forms: $(x, \pi(x))$ or $(\omega(y), y)$. Therefore, for any pairs $(x, y), (x', y') \in C$, they must be from one of the following three forms (i) $(x, \pi(x)), (x', \pi(x'))$, (ii) $(\omega(y), y), (\omega(y'), y')$, or (iii) $(x, \pi(x)), (\omega(y), y)$. If they are in the form (i), from (6.28), we can bound the difference between the respective relationship as

$$\left| r_X(x, x') - r_Y(y, y') \right| \leq d'_{\mathcal{N}}(N_X, N_Y). \tag{6.32}$$

If the pairs are in the form (ii), (6.29) also implies the correctness of (6.32). Finally, if the pairs are in the form (iii), (6.32) would be established from (6.30). Consequently, (6.32) holds for any $(x, y), (x', y') \in C$. Therefore, they must also hold true for the bottleneck pairs achieving the maximum $\Gamma_{X,Y}(C)$ in (2.2) which implies that $\Gamma_{X,Y}(C) \leq d'_{\mathcal{N}}(N_X, N_Y)$. The specific correspondence C may not be the minimizing one in defining $d_{\mathcal{N}}(N_X, N_Y)$. Nonetheless, they are valid mappings and therefore taking a minimum over all mappings yields the desired inequality $d_{\mathcal{N}}(N_X, N_Y) \leq d'_{\mathcal{N}}(N_X, N_Y)$. ■

Since we have proven the two inequalities, it follows that $d_{\mathcal{N}} \equiv d'_{\mathcal{N}}$ and this completes the proof of the statement. ■

Corollary 4 *Function d_{E} is a lower bound with $d_{\mathcal{N}}$ in* (2.3), *i.e.*

$$d_{\text{E}}(N_X, N_Y) \leq d_{\mathcal{N}}(N_X, N_Y), \tag{6.33}$$

for any networks N_X and N_Y.

Proof: The network distance $d_{\mathcal{N}}(N_X, N_Y)$ would be no smaller than the right hand side of (6.19), if we remove the term $\delta_{X,Y}(\pi, \omega)$ in the maximum, i.e.

$$d_{\mathcal{N}}(N_X, N_Y) \geq \min_{\pi: X \to Y, \omega: Y \to X} \max\left\{\Delta_{X,Y}(\pi), \Delta_{Y,X}(\omega)\right\}. \tag{6.34}$$

The right hand side of (6.34) would become smaller if we take the respective minimum for mappings π and ω before taking the maximum, yielding us

$$d_{\mathcal{N}}(N_X, N_Y) \geq \max\left\{\min_{\pi: X \to Y} \Delta_{X,Y}(\pi), \min_{\pi: Y \to X} \Delta_{Y,X}(\omega)\right\}. \tag{6.35}$$

From (6.2) and (6.10) in Definitions 27 and 26, it is not hard to observe that the right hand side of

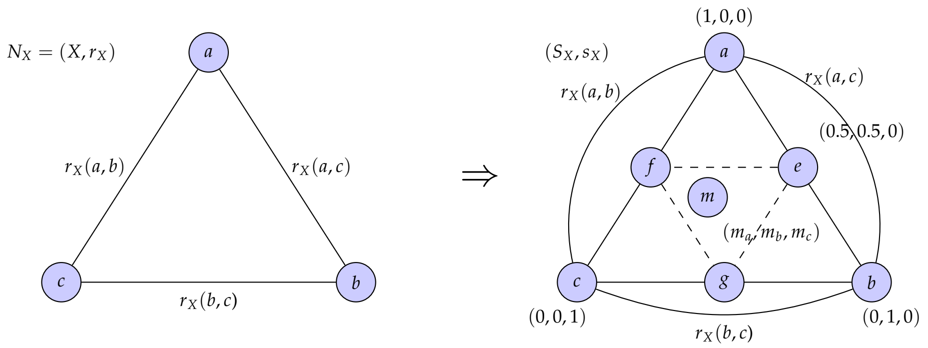


Figure 41: An example of induced space with points defined in the original network. We would like to induce a space (S_X, s_X) such that S_X includes infinite number of points formed by the convex of the original points a, b, and c, and s_X is a semimetric for any pair of points in S_X. We want to induce the semimetric from the original network such that $s_X(a,b) = r_X(a,b)$, $s_X(a,c) = r_X(a,c)$, and $s_X(b,c) = r_X(b,c)$. Middle points, e.g. e, can be considered as the entity represents 50% of a and 50% of b. An important observation is that any points in the space, e.g. m, can be written as a convex combination representing the proportion of their contents from original nodes – (m_a, m_b, m_c) with $m_a + m_b + m_c = 1$.

(6.35) is $\max\{d_{\text{PE}}(N_X, N_Y), d_{\text{PE}}(N_Y, N_X)\} =: d_{\text{E}}(N_X, N_Y)$, yielding the desired result $d_{\text{E}}(N_X, N_Y) \leq d_{\mathcal{N}}(N_X, N_Y)$. ■

The relationships in Proposition 10 and Corollary 4 are extensions of similar analyses that hold for the Gromov-Hausdorff distance between metrics spaces, [154, 155]. As in the case of metric spaces, these results imply that the embedding distance $d_{\text{E}}(N_X, N_Y)$ can be used to lower bound the network distance $d_{\mathcal{N}}(N_X, N_Y)$ [cf. (6.33)]. This value is in addition to the ability of the partial embedding distance $d_{\text{PE}}(N_X, N_Y)$ of Definition 26 to measure how far the network X is to being a subnetwork of network Y.

In the comparison of surfaces and shapes, the partial embedding distance $d_{\text{PE}}(N_X, N_Y)$ has the attractive property of being approximable using multidimensional scaling techniques [50, 52]. Our empirical analysis shows that the use of analogous techniques to estimate $d_{\text{PE}}(N_X, N_Y)$ for arbitrary networks yields sub-optimal results and that this is related to how far the dissimilarities in N_X and N_Y are from satisfying the triangle inequality – see the example in Figure 40 and the numerical analysis in Section 6.3. To improve the accuracy of multidimensional scaling estimates we propose to define the interior of a network by defining a space where dissimilarities between any pair of points represented by a convex combination of nodes in the given networks are defined (Section 6.2). We will further demonstrate that the proposed definition of the interior of a network is such that the partial embedding distances between networks with interiors are the same as the partial embedding distances between the corresponding original networks (Theorems 14 and 15). Empirical demonstrations will show that the comparison of networks with interiors using MDS techniques yields better results that are comparable to those obtained when comparing shapes and surfaces (Section 6.3). From an intuitive perspective, MDS fails mostly when networks do not satisfy triangle inequality. Because points in MDS approximation satisfy metric requirements, all information in a given network that deviates from triangle inequality is completely ignored. This will impact performance as the ignored information may be *the* unique feature distinguishing networks. Adding more points in a given network will add more points whose relationships (i) satisfy triangle inequality, and (ii) partly represent the relationships that do not satisfy triangle inequality in the original network. This helps mitigate the incompatibility of non-metric relationships in networks and MDS approximation, and therefore is expected to result in more accurate approximation that better represents true relationships in the networks.

6.2. Interiors

We provide a different perspective to think of networks as semimetric spaces where: (i) There are interior points defined by convex combinations of given nodes. (ii) Dissimilarities between these interior points are determined by the dissimilarities between the original points. To substantiate the formal definition below (Definition 28) we discuss the problem of defining the interior of a network with three points. Such network is illustrated in Figure 41 where nodes are denoted as a, b, and c and dissimilarities are denoted as r_X. Our aim is to induce a space (V_X, v_X) where the dissimilarities in the induced space are $v_X : V_X \times V_X \to \mathbb{R}_+$. We require that V_X preserve the distance of original points in N_X such that $v_X(a,b) = r_X(a,b)$, $v_X(a,c) = r_X(a,c)$, and $v_X(b,c) = r_X(b,c)$.

Points inside the network are represented in terms of convex combinations of the original points a, b, and c. Specifically, an arbitrary point m in the interior of the network is represented by the tuple (m_a, m_b, m_c) which we interpret as indicating that m contains an m_a proportion of a, an m_b proportion of b, and an m_c proportion of c. Points e, f, and g on Figure 41 contain null proportions of some nodes and are interpreted as lying on the edges. Do notice that although we are thinking of m as a point inside the triangle, a geometric representation does not hold.

First we consider the case that the triangle inequality is satisfied by r_X. For two arbitrary points p represented by the tuple (p_a, p_b, p_c) and m represented by the tuple (m_a, m_b, m_c) in the original network, to evaluate the dissimilarity between p and m using known dissimilarities in the original network, we need to find a path consisting of vectors parallel to the edges in the network that go to m from p. Specifically, denote w_{ab} as the proportion transversed in the direction from a to b in the path; see Figure 42 for illustration. For a positive value w_{ab}, compared to p, m becomes more similar to b by w_{ab} units and less similar to a by $-w_{ab}$ units; for a negative w_{ab}, compared to p, m becomes more similar to a and less similar to b. Proportion transversed in other directions, e.g. from a to c and from b to c, are denoted as w_{ac} and w_{bc}, respectively. For the path transversing w_{ab} from a to b, w_{ac} from a to c, and w_{bc} from b to c, the dissimilarity can be denoted as $|w_{ab}|r_X(a,b) + |w_{ac}|r_X(a,c) + |w_{bc}|r_X(b,c)$. There may be many different paths from p to m, as illustrated in Figure 42. Out of all paths, only the one yielding the smallest distance should be considered. This means the dissimilarity $v_X(p,m)$ between p and m can be defined by solving the

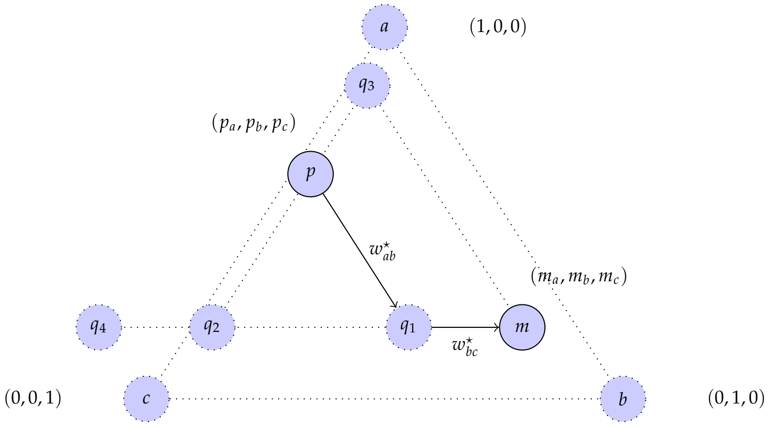


Figure 42: Comparing arbitrary points inside the space induced from networks of three nodes. Given a pair of nodes p and m in the induced space, we need to find paths from p to m that are consisted of vectors parallel to the direction of original nodes in the networks, e.g. a to b, a to c, and/or b to c. We assume the direction of original nodes in the networks have unit amount of transformation. Potential choices of paths from p to m include: p to m via q_1, via q_2, or via q_3. Of them, the path p to m via q_1 has the smallest amount of transformation traversed along the path. There are paths in the form which involves vectors p to q_2, q_2 to q_4, and q_4 to m; such paths would not give the optimal solution to (6.39).

following problem,

$$\begin{aligned} \min \quad & |w_{ab}|\, r_X(a,b) + |w_{ac}|\, r_X(a,c) + |w_{bc}|\, r_X(b,c) \\ \text{s. t.} \quad & m_a = p_a - w_{ab} - w_{ac}, \\ & m_b = p_b + w_{ab} - w_{bc}, \\ & m_c = p_c + w_{ac} + w_{bc}. \end{aligned} \tag{6.36}$$

The constraints make sure that the path starts with tuple (m_a, m_b, m_c) and ends with tuple (p_a, p_b, p_c). This is like the definition of Manhattan distance. In fact, if Manhattan was a triangle with three endpoints and the roads in Manhattan were in a triangle grid, then the distance between any pair of points in Manhattan would be evaluated as in (6.36).

When relationships in r_X do not satisfy triangle inequality, e.g. $r_X(a,b) + r_X(b,c) < r_X(a,c)$, however, the construction in (6.36) is problematic since the optimal solution in (6.36) would yield $v_X(a,c) = r_X(a,b) + r_X(b,c)$, which violates our requirement that $v_X(a,c)$ should be the same as $r_X(a,c)$. The problem arises because each segment in a given path contains two pieces of information – the proportion of transformation, and the dissimilarity created of such transformation. E.g. for the path segment pq_1 in Figure 42, it represents $w^\star_{ab}$ units of transformation from a to b, and also denotes a dissimilarity between p and q_1 as $|w^\star_{ab}| r_X(a,b)$. The two pieces of information unite when r_X is a metric, however, create conflicts for dissimilarities in a general network. To resolve such an issue, we could separate the amount of transformation from the dissimilarity incurred due to transformation. Firstly, we find the path with the smallest amount of transformation

$$\begin{aligned} \min \quad & |w_{ab}| + |w_{ac}| + |w_{bc}| \\ \text{s. t.} \quad & m_a = p_a - w_{ab} - w_{ac}, \\ & m_b = p_b + w_{ab} - w_{bc}, \\ & m_c = p_c + w_{ac} + w_{bc}. \end{aligned} \tag{6.37}$$

Then, for the optimal path $w^\star_{ab}$, $w^\star_{ac}$, and $w^\star_{bc}$ in (6.37), define the dissimilarity as the distance transversed on the path, i.e.

$$v_X(p, m) = |w^\star_{ab}|\, r_X(a, b) + |w^\star_{bc}|\, r_X(b, c) + |w^\star_{ac}|\, r_X(a, c). \tag{6.38}$$

The problem in (6.37) can always be solved since it is underdetermined due to the facts that $m_a + m_b + m_c = p_a + p_b + p_c = 1$. It traces back to (6.36) when relationships in network are metrics. Moreover, it satisfy our requirement $v_X(a, b) = r_X(a, b)$, $v_X(a, c) = r_X(a, c)$, and $v_X(b, c) = r_X(b, c)$ for any networks. Regarding our previous example of a triangle-shaped Manhattan with three endpoints, suppose relationships in the network denote the amount of travel time between the endpoints. These relationship may not necessarily satisfy triangle inequalities. Suppose roads in Manhattan form a triangle grid, the problem in (6.37) is finding the shortest path between a pair of locations in Manhattan. The dissimilarity in (6.38) describes the travel time between this pair of locations using the shortest path.

Given any network with arbitrary number of nodes, we define the induced space as a generalization to the case for nodes with three nodes we developed previously.

Definition 28 *Given a network $N_X = (X, r_X)$ with $X = \{1, 2, \ldots, n\}$, the induced space (V_X, v_X) is defined such that the space V_X is the convex hull of X with $V_X = \{m = (m_1, m_2, \ldots, m_n) \mid m_i \geq 0, \sum_{i \in X} m_i = 1\}$. Given a pair of nodes $m, p \in V_X$, the path yielding the smallest amount of transformation from p to m is obtained through the problem*

$$\begin{aligned} \left\{w^\star_{ij}\right\} = \operatorname{argmin} \quad & \sum_{i,j \in X, i<j} |w_{ij}| \\ \text{s. t.} \quad & m_i = p_i - \sum_{j \in X, j>i} w_{ij} + \sum_{j \in X, j<i} w_{ji}, \ \forall i \end{aligned} \tag{6.39}$$

The distance between p and m is then the distance traversed proportional to the original relationships weighted by the path,

$$v_X(p, m) = \sum_{i,j \in X, i<j} \left|w^\star_{ij}\right| r_X(i, j). \tag{6.40}$$

In scenarios where the optimal solution $\{w^\star_{ij}\}$ is not unique, the induced relationship $v_X(p, m)$ is defined using the one yielding the smallest $v_X(p, m)$ out of minimizing paths solving (6.39)..

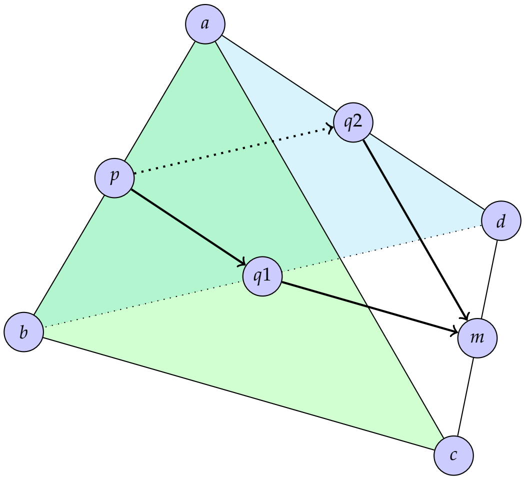


Figure 43: An example where the optimal solution in (6.39) is not unique. There are two paths from p to m, via q_1 or via q_2, which have the same objective cost in (6.39). The induced relationship $v_X(p, m)$ is defined using the path that has the smallest weighted sum.

The induced space V_X is the convex hull constructed by all nodes $i \in X$. Each node in the induced space $m \in V_X$ can be represented as a tuple $(m_1, m_2, \ldots, m_n)$ with $\sum_{i \in X} m_i = 1$ where m_i represents the percentage of m inheriting the property of node $i \in X$. To come up with distance between pairs of points $p, m \in V_X$ with the respective tuple representation $(p_1, p_2, \ldots, p_n)$ and $(m_1, m_2, \ldots, m_n)$, we consider each edge in the original space X, e.g. from i to j, represents one unit of cost to transform i into j. All edges are considered similarly with one unit of cost to transform the starting node into the ending node. We want to find the smallest amount of cost to transform p into m. This is solved via (6.39), which is always solvable since the problem is underdetermined due to the facts that $\sum_{i \in X} m_i = \sum_{i \in X} p_i = 1$. This gives us the optimal path with weights $\{w^{\star}_{ij}\}$ meaning that the most cost-saving transformation from p into m is to undertaking $w^{\star}_{ij}$ unit of transformation along the direction of transforming i into j. The distance in the induced space $v_X(p, m)$ is then the distance traversed proportional to the original relationships weighted by the path defined in (6.40). It is possible that the solution in (6.39) is not unique (see Figure. 43 for an example where two paths have the same objective cost). In these scenarios, the induced relationship $v_X(p, m)$ is defined using the path that has the smallest weighted sum $\sum_{i<j} |w^{\star}_{ij}| r_X(i, j)$. We need the path to firstly minimize the problem in (6.39) and then use (6.40) as a tie-breaker, because if we use (6.40) as the only criteria, the scenario discussed in the paragraph before (6.37) violating our requirement will happen. Back to our illustration of travel distance and travel time in Manhattan, if there exist multiple paths yielding the shortest travel distance between a pair of locations in Manhattan, the dissimilarity between this pair of locations is then defined as the shortest travel time among these paths.

Proposition 11 *The space (V_X, v_X) induced from $N_X = (X, r_X)$ defined in Definition 28 is a semimetric space in V_X. Moreover, the induced space preserves relationships: when $p, m \in X$, $v_X(p, m) = r_X(p, m)$.*

Proof: To prove that space (V_X, v_X) induced from $N_X = (X, r_X)$ is a semimetric space, we prove the (i) nonnegativity, (ii) symmetry, (iii) identity properties in Definition 3 and (iv) $v_X(p,m) = r_X(p,m)$ when $p, m \in X$.

Proof of nonnegativity property: Since $r_X(i,j) > 0$ for any different nodes in the original networks $i, j \in X, i \neq j$, $|w^\star_{ij}| r_X(i,j) \geq 0$ in (6.40). Therefore, the induced distance $v_X(p,m) = \sum_{i \neq j} |w^\star_{ij}| r_X(i,j) \geq 0$. ■

Proof of symmetry property: Given a pair of nodes $p, m \in V_X$, we would like to demonstrate that $v_X(p,m) = v_X(m,p)$. Denote $\{w^\star_{ij}\}$ as the collection of units of transformation along the direction from i to j in the original network. These vectors together make up the path from p to m with smallest amount of transformation. By definition, $\{w^\star_{ij}\}$ is the optimal solution to (6.39). Denote $\{v^\star_{ij}\}$ as the collection of units of transformation along the direction from i to j which makes up the path from m to p with the smallest amount of transformation. By definition, $\{v^\star_{ij}\}$ is the optimal solution to the following problem

$$\begin{aligned} \left\{v^\star_{ij}\right\} = \operatorname{argmin} & \sum_{i,j \in X, i<j} |v_{ij}| \\ \text{s.t. } & p_i = m_i - \sum_{j \in X, j>i} v_{ij} + \sum_{j \in X, j<i} v_{ji}, \ \forall i \end{aligned} \tag{6.41}$$

Comparing (6.39) with (6.41), it is easy to observe that if we take $v_{ij} = -w_{ij}$ for any $i < j$, the two problems becomes identical. Therefore, for the optimal solutions, we have the relationship $v^\star_{ij} = -w^\star_{ij}$ for any $i < j$. By definition in (6.40), this implies the two relationships are the same

$$v_X(p,m) = \sum_{\substack{i,j \in X, \\ i<j}} \left|w^\star_{ij}\right| r_X(i,j) = \sum_{\substack{i,j \in X, \\ i<j}} \left|v^\star_{ij}\right| r_X(i,j) = v_X(m,p), \tag{6.42}$$

and completes the proof. ■

Proof of identity property: First we want to show that if m and p are identical points, their induced relationship $v_X(p,m) = 0$. In such scenario, m and p must have same tuple representation $(m_1, \ldots, m_n)$ and $(p_1, \ldots, p_n)$ with $m_i = p_i$ for any $i \in X$. In this case, it is apparent that the optimal solution $\{w^\star_{ij}\}$ in (6.39) is $w^\star_{ij} = 0$ for any $i \neq j$. Therefore, $v_X(p,m) = 0$ shows the first part of the proof for identity property.

Second, we need to prove $v_X(p,m) = 0$ must imply that p and m are the same. By definition in (6.40), the induced relationship can be written $v_X(p,m) = \sum_{i \neq j} |w^\star_{ij}| r_X(i,j)$, where the original relationship is always positive with $r_X(i,j) > 0$ for any $i \neq j$. Therefore, $v_X(p,m) = 0$ must imply that $|w^\star_{ij}| = 0$ given any $i \neq j$. Combining this observation with the constraints in (6.39) imply that $p_i = m_i$ for any $i \in X$. Therefore, p and m are identical point in the induced space, and this completes the proof. ■

Proof of the property that $v_X(p, m) = r_X(p, m)$ when $p, m \in X$: When both $p, m \in X$, the respective tuple representation in the space is $p = (p_1, \dots, p_n)$ with $p_i = 1$ if $i = p$ and $p_i = 0$ otherwise, and $m = (m_1, \dots, m_n)$ with $m_i = 1$ if $i = m$ and $m_i = 0$. It is apparent that the path with the smallest amount of transformation from p into m is the exact vector from p to m. Here we give a geometric proof using Figure 11. The induced space is the $(n-1)$-simplex with interior defined. Nodes p and m correspond to the vertices in the simplex with their coordinates given by the tuple representations $p = (p_1, \dots, p_n)$ and $m = (m_1, \dots, m_n)$. The problem in (6.39) searches for the shortest path in the simplex joining p to m. It is then apparent that the shortest path should be the edge joining then; consequently $w^\star_{pm} = 1$ for the edge and $w^\star_{ij} = 0$ for any other edges ij. Taking this observation into (6.40) implies that $v_X(p, m) = r_X(p, m)$ and concludes the proof. ■

Having proven all statements, the global proof completes. ■

The semimetric established in Proposition 11 guarantees that the points in the induced space with their dissimilarity $v_X(p, m)$ are well-behaved. We note that semimetric is the best property we can expect, since the triangle inequality may not be satisfied even for the dissimilarities in the original networks. Next we show that the embedding distance is preserved when interiors are considered.

6.2.1. Distances between Networks Extended to Their Interiors

Since semimetrics are induced purely from the relationships in the original network, a pair of networks N_X and N_Y can be compared by considering their induced space, as we state next.

Definition 29 *Given two networks $N_X = (X, r_X)$ and $N_Y = (Y, r_Y)$ with their respective induced space (V_X, v_X) and (V_Y, v_Y), for a map $\pi : V_X \to V_Y$ from the induced space V_X to the induced space V_Y such that $\pi(x) \in Y$ for any $x \in X$, define the network difference with respect to π as*

$$\Delta_{V_X, V_Y}(\pi) := \max_{x, x' \in V_X} \Big| v_X(x, x') - v_Y(\pi(x), \pi(x')) \Big|. \tag{6.43}$$

The partial embedding distance from N_X to N_Y measured with respect to the induced spaces is then defined as

$$d_{\text{PE, V}}(N_X, N_Y) := \min_{\pi : V_X \to V_Y | \pi(x) \in Y, \forall x \in X} \Big\{ \Delta_{V_X, V_Y}(\pi) \Big\}. \tag{6.44}$$

The partial embedding distance $d_{\text{PE, V}}(N_X, N_Y)$ with respect to the induced space in (6.44) is defined similarly as the partial embedding distance $d_{\text{PE}}(N_X, N_Y)$ in (6.2) however considers the mapping between all elements in the induced spaces. Observe that we further require that the embedding satisfy $\pi(x) \in Y$ for any $x \in X$. This ensures the original nodes of network X are mapped to original nodes of network Y. The restriction is incorporated because it makes the embedding distance $d_{\text{PE, V}}(N_X, N_Y)$ with respect to the induced spaces identical to the original embedding distance $d_{\text{PE}}(N_X, N_Y)$ as we state next.

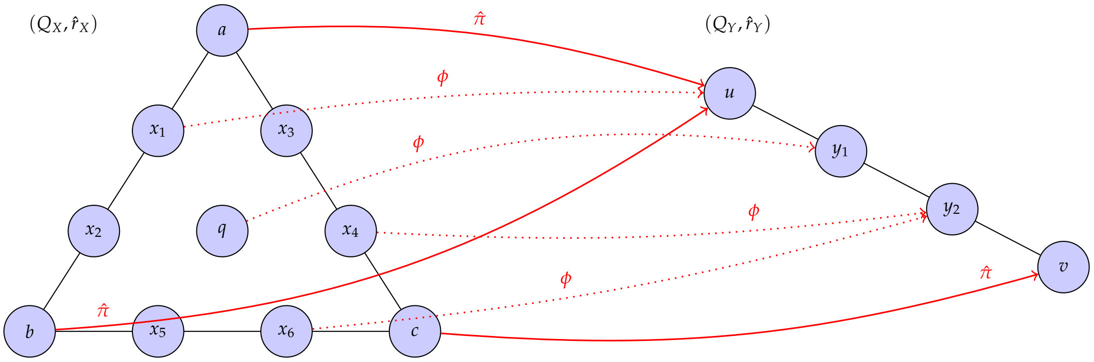


Figure 44: An example of regular sample pair $(Q_X, \hat{r}_X)$ and $(Q_Y, \hat{r}_Y)$, where $Q_X = \{a, b, c, x_1, x_2, x_3, x_4, x_5, x_6, q\}$ collects original points and induced points that are combination of one-third of original nodes, and $Q_Y = \{u, v, y_1, y_2\}$. Here we illustrate a specific mapping $\hat{\pi}$ with $\hat{\pi}(a) = \hat{\pi}(b) = u$ and $\hat{\pi}(c) = v$; it is apparent that $\pi(x) \in Q_Y$ for any $x \in Q_X$. Note that $\tilde{Q}_X = Q_X \setminus \{q\}$ also form a regular sample pair with Q_Y.

Theorem 14 *The function* $d_{\text{PE, V}} : \mathcal{N} \times \mathcal{N} \to \mathbb{R}_+$ *defined in* (6.44) *is an embedding metric in the space* $\mathcal{N}$ *and yields the same distance as the function* d_{PE} *defined in* (6.2),

$$d_{\text{PE, V}}(N_X, N_Y) = d_{\text{PE}}(N_X, N_Y), \quad \textit{for all } N_X, N_Y. \tag{6.45}$$

The statement in Theorem 14 justifies comparing networks via their respective induced space. Similar as in Definition 27, defining $\max\{d_{\text{PE, V}}(N_X, N_Y), d_{\text{PE, V}}(N_Y, N_X)\}$ would yield a metric in the space $\mathcal{N} \mod \cong$ and this maximum is the same as d_{E} defined in (6.10). Since the induced spaces incorporate more information of the original networks while at the same time $d_{\text{PE, V}}(N_X, N_Y) = d_{\text{PE}}(N_X, N_Y)$, an approximation to $d_{\text{PE, V}}(N_X, N_Y)$ via the induced space would be a better approximation to $d_{\text{PE}}(N_X, N_Y)$. It may appear that the evaluation of the induced space is costly. However, we demonstrate in the next subsection that the partial embedding distances have a nice property that if we sample a number of points in the induced spaces respectively according to the same rule, the distance between the sampled induced space is the same as the original distance. Despite that the definition of interiors of networks is somewhat arbitrary, its practical usefulness can be justified from Theorem 14.

6.2.2. *Sampling of Interiors*

In this section, we consider a practical scenario where we only take several samples in the induced space. We show that comparing the combination of nodes in the respective original networks and sampled nodes in the induced space would yield the same result as comparing the original networks. Given a network $N_X = (X, r_X)$, our aim is to define a sampled induced space $(Q_X, \hat{r}_X)$ where $Q_X \supset X$ includes more nodes compared to X. An example is in Figure 45, where the original node space is given by $\{a, b, c\}$, and one version of sampled induced node space is $Q_X = \{a, b, c, e, f, g\}$, the union of the original nodes and the nodes in the midpoints of the edges in

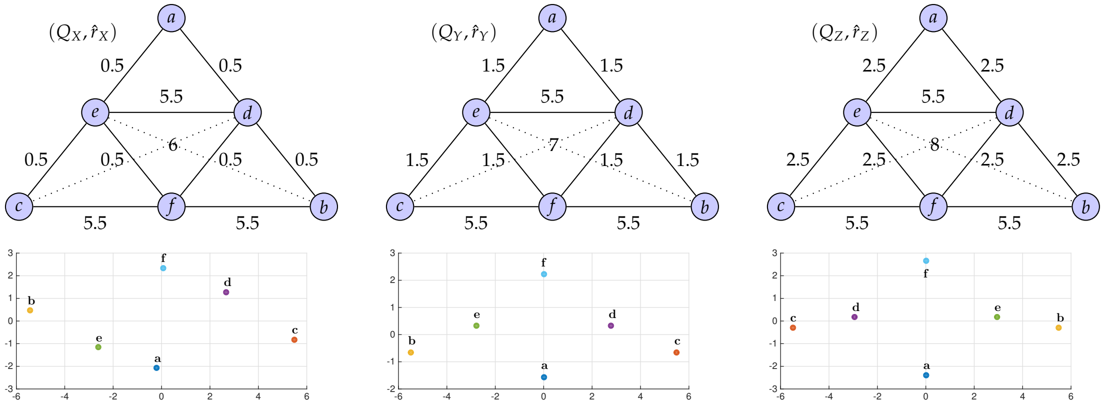


Figure 45: The caveat illustrated in Figure 40 where different networks results in identical multi-dimensional scaling results could be solved by considering the induced sample space where we utilize the same predetermined sampling strategy – taking midpoints for all edges – in the network. Multi-dimensional scaling by adding interiors would distinguish different networks. The dissimilarities between nodes a and f are not illustrated in the respective induced network due to space limit.

the original networks. The distance in the sampled induced space $\hat{r}_X : Q_X \times Q_X \to \mathbb{R}_+$ should preserve the distance of original points in X. A natural choice for $\hat{r}_X$ is the restriction of the distance v_X defined for the induced space V_X: i.e. given any pair of points $x, x' \in Q_X$, let $\hat{r}_X(x, x') := v_X(x, x')$. Our key observation for such construction is that if the nodes in the induced spaces of a pair of networks are sampled according to the same strategy, then the distance between the sampled induced space is identical to the original distance. We start by formally describing what do we mean by a pair of networks sampled according to the same rule as next.

Definition 30 *Given a pair of networks $N_X = (X, r_X)$ and $N_Y = (Y, r_Y)$, their respective sampled space $(Q_X, \hat{r}_X)$ and $(Q_Y, \hat{r}_Y)$ form a regular sample pair, if for any mapping $\hat{\pi} : X \to Y$ in the original node set, we have $\pi(x) \in Q_Y$ for any $x \in Q_X$, where $\pi : V_X \to V_Y$ is the map induced from $\hat{\pi}$ such that $\pi : x \mapsto \pi(x)$ whose the i-th element in the tuple representation $[\pi(x)]_i$ is*

$$[\pi(x)]_i = \sum_{j \in X} \mathbf{1}\left\{\hat{\pi}(j) = i\right\} x_j, \tag{6.46}$$

and for any mapping $\hat{\omega} : Y \to X$ in the original node set, we have $\omega(y) \in Q_X$ for any $y \in Q_Y$ where $\omega : V_Y \to V_X$ is the map induced from $\hat{\omega}$ such that the j-th element in the tuple representation of $\omega(y)$ is

$$[\omega(y)]_j = \sum_{i \in Y} \mathbf{1}\left\{\hat{\omega}(i) = j\right\} y_i. \tag{6.47}$$

In the definition, $\mathbf{1}\{\hat{\pi}(j) = i\}$ is the indicator function such that it equals one if $\hat{\pi}$ maps $j \in X$ to $i \in Y$ and $\mathbf{1}\{\hat{\pi}(j) = i\} = 0$ otherwise. The notation $[\pi(x)]_i$ denotes the proportion of $\pi(x)$ coming from i-th node in Y. It is easy to see that π in (6.46) is well-defined. Firstly, $[\pi(x)]_i \geq 0$ for any

$i \in Y$, and

$$\sum_{i \in Y} [\pi(x)]_i = \sum_{i \in Y} \sum_{j \in X} \mathbf{1} \{\hat{\pi}(j) = i\} x_j = \sum_{j \in X} x_j = 1, \tag{6.48}$$

ensuring $\pi(x)$ is in the induced convex hull space V_Y. Secondly, for any $j \in X$ in the original nodespace, its mapping $\pi(j)$ would have the tuple representation with $[\pi(j)]_i = \mathbf{1}\{\hat{\pi}(j) = i\}$, a node in the original node space of Y. Consequently, for any $j \in X$, we have that $\pi(j) \in Y$. Combining these two observations imply that $\pi : V_X \to V_Y$ is well-defined. Similarly, ω induced from $\hat{\omega}$ is also well-defined from V_Y to V_X. Definition 30 states that for any point x in Q_X, no matter how we relate points in Q_X to points in Q_Y, the mapped node $\pi(x)$ should be in the induced sample space Q_Y. An example of regular sample pair is illustrated in Figure 44, where $Q_X = \{a, b, c, x_1, x_2, x_3, x_4, x_5, x_6, q\}$ is the collection of original node space and points that are combination of one-third of original nodes and $Q_Y = \{u, v, y_1, y_2\}$. Figure 44 exemplifies the scenario for a specific mapping $\hat{\pi}$ with $\hat{\pi}(a) = \hat{\pi}(b) = u$ and $\hat{\pi}(c) = v$; it is apparent that $\pi(x) \in Q_Y$ for any $x \in Q_X$. We note that $\tilde{Q}_X = Q_X \setminus \{q\}$ also form a regular sample pair with Q_Y. A pair of networks N_X and N_Y can be compared by evaluating their difference in their respective sampled induced space as next.

Definition 31 *Given two networks $N_X = (X, r_X)$ and $N_Y = (Y, r_Y)$ with their respective sampled induced space $(Q_X, \hat{r}_X)$ and $(Q_Y, \hat{r}_Y)$, for a map $\pi : Q_X \to Q_Y$ such that $\pi(x) \in Y$ for any $x \in X$, define the difference with respect to π as*

$$\Delta_{Q_X, Q_Y}(\pi) := \max_{x, x' \in Q_X} \Big| \hat{r}_X(x, x') - \hat{r}_Y(\pi(x), \pi(x')) \Big|. \tag{6.49}$$

The partial embedding distance from N_X to N_Y measured with respect to the sampled induced spaces is then defined as

$$d_{\text{PE, Q}}(N_X, N_Y) := \min_{\pi : Q_X \to Q_Y | \pi(x) \in Y, \forall x \in X} \Big\{ \Delta_{Q_X, Q_Y}(\pi) \Big\}. \tag{6.50}$$

Our key result is that $d_{\text{PE, Q}}(N_X, N_Y)$ is the same as the partial embedding distance $d_{\text{PE}}(N_X, N_Y)$ defined in (6.2) when the sampled space form a regular sample pair.

Theorem 15 *When the sampled spaces Q_X and Q_Y form a regular sample pair, the function $d_{\text{PE, Q}} : \mathcal{N} \times \mathcal{N} \to \mathbb{R}_+$ defined in (6.49) is an embedding metric in the space $\mathcal{N}$. Moreover, it yields the same distance as the function d_{PE} defined in (6.2), i.e.*

$$d_{\text{PE, Q}}(N_X, N_Y) = d_{\text{PE}}(N_X, N_Y), \tag{6.51}$$

for any networks N_X and N_Y.

The statement in Theorem 15 gives proper reasoning for differentiating networks via their sampled induced space. Similar as previous treatments, we could define $\max\{d_{\text{PE, Q}}(N_X, N_Y), d_{\text{PE, Q}}(N_Y, N_X)\}$

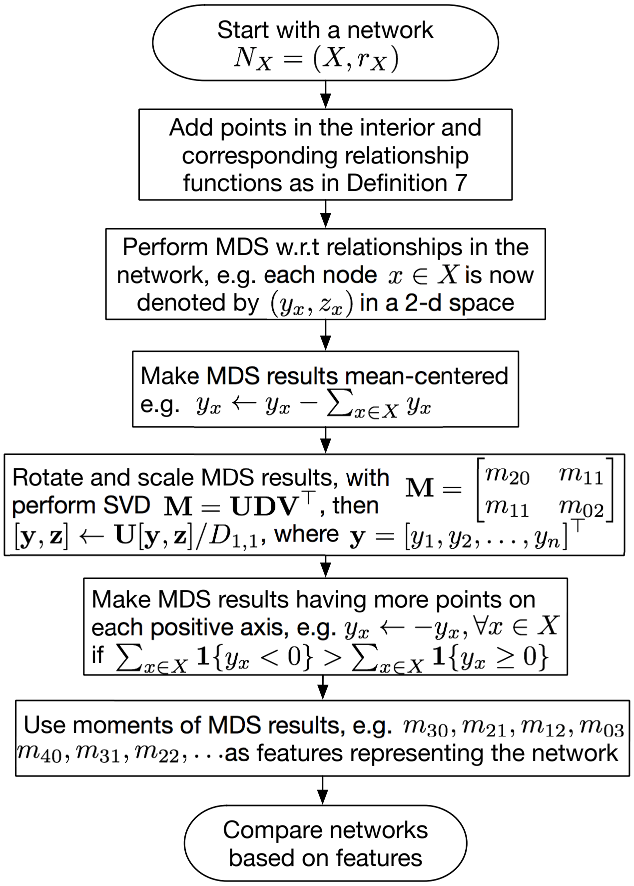
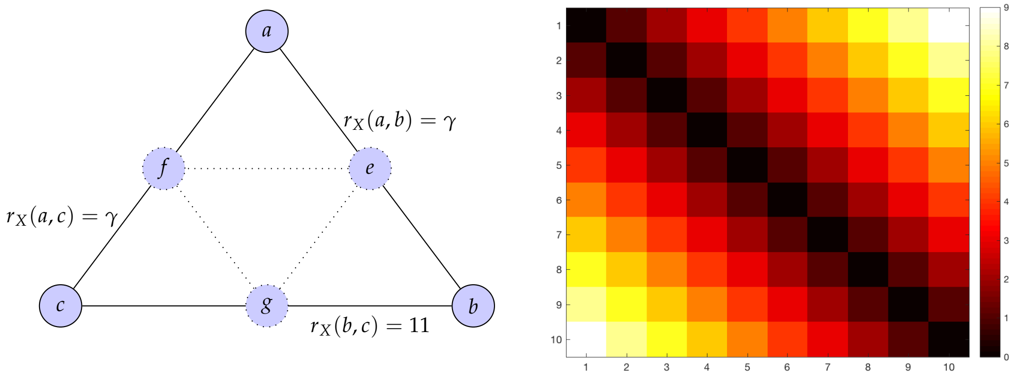


Figure 46: Flow chart of the embedding algorithm, where $m_{\alpha\beta} = \sum_{x \in X} y_x^\alpha z_x^\beta$ denotes the moment of the two-dimensional points. Starting with a network $N_X = (X, r_X)$, MDS is performed with respect to the relationship functions in the network. Post-processing is conducted to make MDS results mean-centered, along the same direction, having unit scale, and having more points on each positive axis. Network comparison is then conducted by using moments of the MDS results as features.

as a metric in the space $\mathcal{N} \mod \cong$. Since the sampled induced spaces incorporate more information of the original networks, an approximation to $d_{\text{PE, Q}}(N_X, N_Y)$ via the sampled induced space would be a better approximation to $d_{\text{PE}}(N_X, N_Y)$. Moreover, since we can construct the sampled induced space following some predetermined strategy – taking midpoints for all edges in the networks, comparing networks via their sampled induced space is plausible in terms of complexity. Figure 45 illustrate the same network considered in Figure 40 where the multi-dimensional scaling based on the sampled induced points would succeed in distinguishing networks that are different. We illustrate the practical usefulness of such methods in the next section.

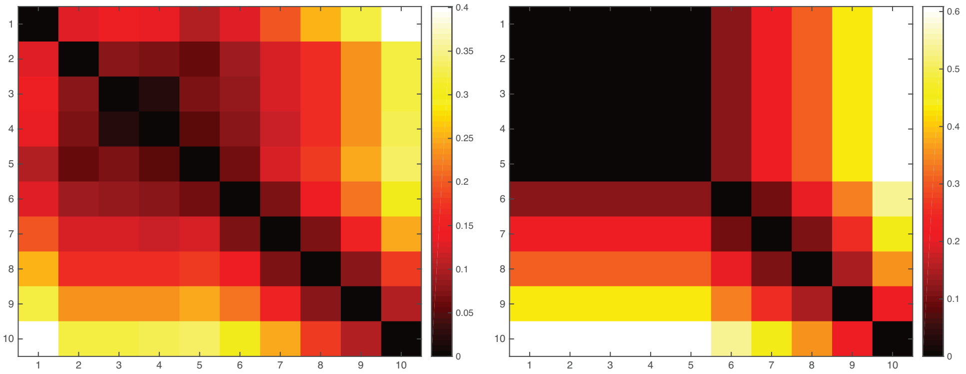
6.3. Application

We illustrate the value of adding interiors in performing network comparisons. A flowchart for the proposed method is available in Figure 46. For each network $N_X = (X, r_X)$, we add interiors



(a) 10 networks with $\gamma = 1, 2, \ldots, 10$

(b) true embedding distance



(c) with interiors

(d) without interiors

Figure 47: Example of the usefulness of considering the interior of networks. We consider 10 networks in the form (a) where $\gamma = 1, 2, \ldots, 10$. Approximations of network embedding distances are evaluated. Sub-figure (b) illustrates heat-map of the actual embedding distance d_{E} defined in Definition 27 where the indices in both horizontal and vertical directions denote γ in the networks. Sub-figures (c) and (d) denote the heat-maps of distance approximation with and without interiors being added to networks before approximation, respectively. When interiors are considered by adding midpoints of edges, e.g. nodes e, f, and g in (a), network distance approximations (c) yield more desired results, especially for $\gamma \leq 5$ where the relationships in the original networks fail to satisfy triangle inequality.

to the network and the corresponding relationship functions. Then we perform MDS with respect to the relationship in the network, such that each node $x \in X$ is represented by a point in the multi-dimensional space. The post-processing steps described in [50, 51] are detailed in Figure 46 for the completeness of the algorithm, where it makes the MDS results mean-centered, along the same direction, have unit scale, and have more points on each positive axis.

6.3.1. *Effect of Adding Interiors in Network Comparison*

We consider a small scale experiment to illustrate the effect of adding interiors in facilitating network comparisons. Specifically, we consider networks with three nodes in which the dissimilarity between b and c is fixed to $r_X(b,c) = 11$ and the dissimilarities between a and b and between a and c are set to $r_X(a,b) = r_X(a,c) = \gamma$, which we choose to vary within the set $\{1, \ldots, 10\}$; see

Figure 47-(a). According to this selection, the networks with $\gamma \in \{6, \dots, 10\}$ satisfy the triangle inequality and the networks with $\gamma \in \{1, \dots, 5\}$ do not.

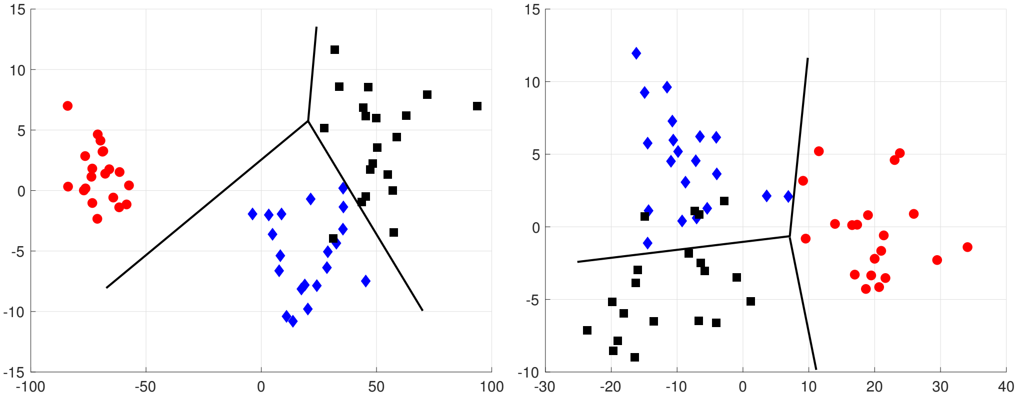
For this small network the actual embedding distances d_{E} in Definition 27 can be evaluated and are visualized in Figure 47-(b). Figures 47-(c) and 47-(d) illustrate distance estimates using moments of embedded points in the projected space [50] – the procedure is summarized in the flowchart in Figure 46 – with interiors included and not included, respectively. It is apparent that the addition of interiors yields better estimates of network distances. This improvement is very marked for networks that violate the triangle inequality. Using MDS techniques without the addition of interiors fails to distinguish networks with $\gamma \in \{1, \dots, 5\}$. The addition of interiors permits identification of the differences between these networks except for $\gamma = 3$ and $\gamma = 4$. Interestingly, the addition of interiors is also helpful when the triangle inequality holds, although the improvements is not as marked.

We also evaluate the approximation results when we add more points to the network. For example, add network interiors in the form of (m_a, m_b, m_c) with $m_a, m_b, m_c \in \{0, 1/3, 2/3, 1\}$, or in the form of (m_a, m_b, m_c) with $m_a, m_b, m_c \in \{0, 1/4, 1/2, 3/4, 1\}$. We found that the adding thirds or quarters points do not provide superior performance compared to just adding middle points as in Figure 47 (c). This is potentially due to the reason that the benefit of adding more points start to saturate. Therefore, for the better tradeoff between efficiency and quality of approximation, it may be sufficient to just add middle points.

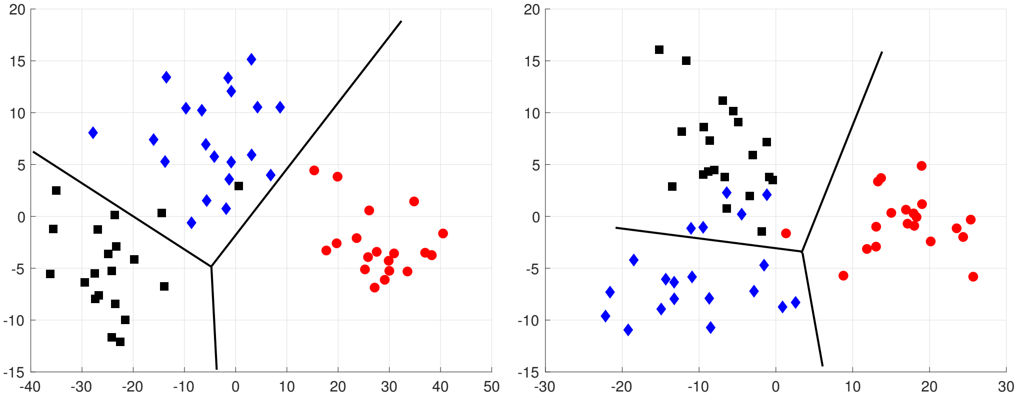
6.3.2. Identification of Generative Models

As an example application we consider the problem of classifying networks with different numbers of nodes according to their generative models. To that end we consider: (i) Weighted Erdős-Rényi random networks [144] with connection probability $p = 0.5$ and edge weights random and uniformly chosen from the unit interval $[0, 1]$. (ii) Random geometric networks where nodes are placed at random in the unit circle and edge weights are of the form $\exp(-d(i, j)^2/2\sigma^2)$, where $d(i, j)$ is the distance between vertices i and j in the unit circle and σ^2 is a kernel width parameter. We set $\sigma^2 = 0.5$. (iii) Random feature networks where edge weights are determined by the Pearson correlation coefficient ρ_{ij} between a pair of corresponding features $\mathbf{u}_i, \mathbf{u}_j \in \mathbb{R}^d$. Features are randomly chosen standard white Gaussian vectors in a space of dimension $d = 5$ and edge weights are chosen as $r_X(i, j) = (1 + \rho_{ij})/2$. Observe that in all three cases edge weights measure the relationship between pairs of vertices and take values in the unit interval.

We start with networks of equal size $|X| = 25$ and construct 20 random networks for each aforementioned type. We then use the MDS method in Figure 46 [50, 51] to approximate the embedding network distance d_{E} defined in Definition 27. To evaluate the effectiveness of considering interiors of networks described in Section 6.2, we add midpoints for all edges in a given network; it is apparent that any pair of networks with interiors defined in this way would form a regular sample pair. Approximations of the embedding network distance d_{E} between these networks with midpoints added are then evaluated. Figure 48 (a) and (b) plot the two dimensional Euclidean em-



(a) with interiors, 25 nodes (b) without interiors, 25 nodes



(c) with interiors, 20 to 25 nodes (d) without interiors, 20 to 25 nodes

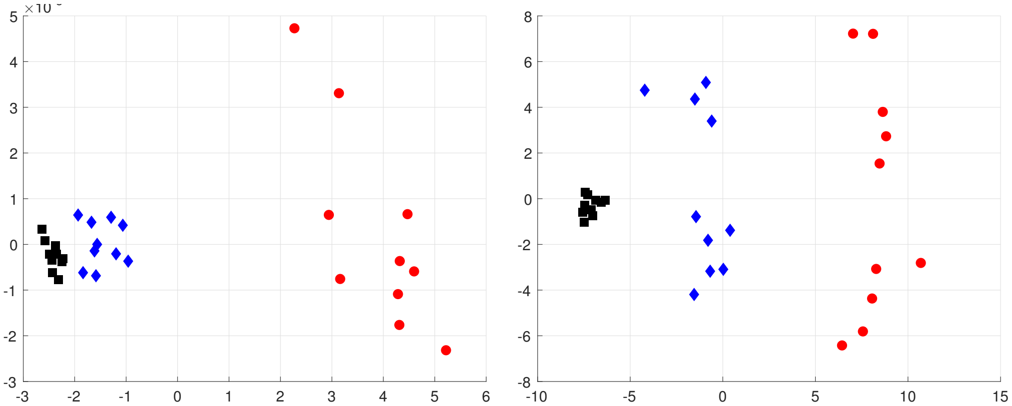
Figure 48: Two dimensional Euclidean embeddings of the networks constructed from three different models with different number of nodes with respect to the approximation to the network embedding distance. In the embeddings, red circles denote networks constructed from the Erdős-Rényi model, blue diamonds represent networks constructed from the unit circle model, and black squares the networks from the correlation model.

beddings [146] of the network metric approximations with and without interiors, respectively[1]. In both figures we see the emergence of clusters corresponding to each generative model, but the clusters are more clear when interiors are added – random networks are denoted by red circles, geometric networks with blue diamonds, and feature networks with black squares. For a formal performance evaluation we conduct unsupervised hierarchical clustering with Ward's linkage [156] method upon the approximated embedding network distance d_{E}; results are drawn using linear boundaries on the corresponding figures. There are 4 misclassifications when no interiors are added but only one misclassification after addition of interiors. This is an error rate of $1/60 \approx 1.67\%$. Similar results are obtained with the use of other unsupervised learning methods.

We further consider mixes in which the number of nodes ranges in the integer set $\{20\ldots,25\}$. Four networks are randomly generated for each type and each number of nodes, resulting in 60

---

[1] The visualization embeddings minimize the sum of squares of the inter-point distances. Visualizations look similar for other common choices of embedding distortion measures.



(a) graphs with 2 clusters (b) graphs with 3 clusters

Figure 49: Two dimensional Euclidean embeddings of unweighted graphs generated with stochastic block models, with interiors added to the weighted networks. (a) Graphs with 150 nodes, $S = 2$ clusters, in-diagonal connecting probability $p = 0.2$, and three different off-diagonal connecting probability $q = 0.05$ (red circles), $q = 0.1$ (blue diamonds), $q = 0.2$ (black squares). (b) Graphs with 150 nodes, $S = 3$ clusters, in-diagonal connecting probability $p = 0.5$, and three different off-diagonal connecting probability $q = 0.1$ (red circles), $q = 0.2$ (blue diamonds), $q = 0.4$ (black squares). In each figure, ten graphs are generated for each type of models.

networks in total. Interiors are examined similarly as before by adding midpoints for all edges in a given network. Figure 48 (c) and (d) illustrate the two dimensional Euclidean embeddings of the network metric approximations with and without interiors respectively. Despite the fact that networks with the same model have different number of nodes, dissimilarities between network distance approximations are smaller when their underlying networks are from the same process. As in comparison in Figure 47, considering interiors result in a more distinctive clustering pattern. An unsupervised classification would yield 2 out of 60 errors (3.33%) for networks with interiors added and 6 errors (10%) without interiors.

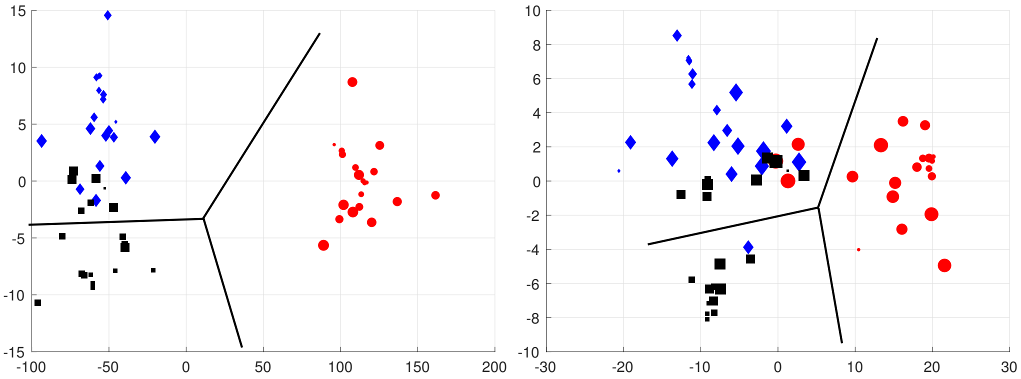
These results illustrate that: (i) Comparing networks by using embedding distances succeeds in identifying networks with different generative models. (ii) Adding interiors to networks to form regular sample pairs as in Section 6.2 would yield better approximations to the actual network distances. We must observe that other methods would exhibit similar success in this classification task. Weighted motifs [16] yields 1 misclassification for networks with constant size and 4 misclassifications for networks of varying sizes. Comparison with persistent homologies [76] yields no errors for networks with constant sizes and 3 errors for networks with varying sizes. This means alternative methods are comparable although they seem to be worse when networks of different sizes are considered. We will see in Section 6.3.4 that this is indeed the case. This is as expected because the strength of embedding distances is precisely on the possibility of embedding a network into another network of a larger size.

6.3.3. *Unweighted Networks*

The framework proposed in the manuscript is designed for weighted networks where relationships between nodes in a network are nonnegative values. In this section, we illustrate how the framework can also be adapted to evaluate distances between unweighted graphs. An unweighted graph is defined as a pair $U_X = (X, u_x)$, where X is a finite set of nodes and $u_X : X \times X \to \{0, 1\}$ denotes the existence of an edge when taking the value $u_X(x, x') = 1$. In this section we consider unweighted graphs generated with stochastic block models, where vertices in X form S different clusters O_τ, that are mutually exclusive with $O_\tau \cap O_{\tau'} = \emptyset$ for any $1 \leq \tau, \tau' \leq T$, and collectively exhaustive with $\cup_{\tau=1}^{T} O_\tau = X$. For any pair of nodes $x, x' \in X$ in the same cluster with $x, x' \in O_\tau$, the block stochastic model sets the probability of existence of a link to $\mathrm{P}\left[u_X(x, x') = 1\right] = p$. For nodes x and x' from different clusters with $x \in O_\tau, x' \in O_{\tau'}$ and $\tau \neq \tau'$, the probability of existence of link is $\mathrm{P}\left[u_X(x, x') = 1\right] = q$. In general, $p \gg q$ so that nodes are more likely to be connected if they belong to the same cluster.

Given an unweighted graph U_X, we construct a weighted network using shortest paths: For any $x, x' \in X$, the weighted relationship $r_X(x, x')$ is computed as the number of links traversed from x to x' in the unweighted graph. The relationship r_X constructed in this way satisfies the definition of networks in Section 2.1. We can then apply the proposed framework onto these constructed networks to evaluate the distance between the corresponding unweighted graphs. Despite the fact that networks constructed using shortest paths would satisfy metric requirements, we could still add interior points of networks to improve the quality of metric approximation. In the following experiment, we do find that adding interiors points to constructed networks yield slightly better clustering structure of graphs generated from different block stochastic models.

We consider block stochastic model unweighted graphs with $|X| = 150$ nodes, $S = 2$ clusters, and in-cluster connection probability $p = 0.2$. The inter-cluster connection probability varies in the set $q \in \{0.05, 0.1, 0.2\}$. For each different inter cluster probability we generate 10 unweighted graphs. Each unweighted graph is transformed into weighted network using shortest paths and the MDS approximation in Figure 49 is applied to evaluate distances between networks. Figure 49 (a) illustrates the two dimensional Euclidean embedding of the network distance approximations for the unweighted graphs generated using different q, where graphs with $q = 0.05$ are denoted by red circles, graphs with $q = 0.1$ are described by blue diamonds, and graphs with $q = 0.2$ represented as black squares. Graphs generated with different off-diagonal connecting probabilities form clear clusters in the distance approximations. In a separate experiment we consider unweighted graphs with $|X| = 150$ nodes, $S = 3$ clusters and in-cluster connection probability $p = 0.5$. We vary the inter-cluster connection probability in the set $q \in \{0.1, 0.2, 0.4\}$. We see in Figure 49 that the distance approximation succeeds again in distinguishing unweighted networks generated from different stochastic block models – networks with $q = 0.1$ are red circles, $q = 0.2$ are blue diamonds, and $q = 0.4$ are black squares. Again, the distance approximation succeeds in distinguishing unweighted networks generated from different stochastic block models.



(a) 20 to 100 nodes

(b) 20 to 1000 nodes

Figure 50: Two dimensional Euclidean embeddings of networks constructed from three different models with different number of nodes with respect to the approximation of the network embedding distance. (a) Midpoints added in interiors. (b) Randomly selected midpoints added to interiors using the framework presented in Section 6.3.4. In the embeddings, red circles denote networks constructed from the Erdős-Rényi model, blue diamonds represent networks constructed from the unit circle model, and black squares the networks from the correlation model; the sizes of the symbols in the embedding are proportional to number of nodes in corresponding networks.

6.3.4. *Large Scale Networks*

We consider the same setup of Section 6.3.2 but range the number of nodes in the network from 20 to 100. Figure 50 (a) illustrates the result of comparing networks with interiors added. The clustering of nodes is not as marked as in Figure 48 but still noticeable. For the most part, distances between networks coming from the same generative model are smaller than distances between networks generated by different processes. An unsupervised classification would yield 7 out of 60 errors (11% error rate) for networks with interiors added. The error rate raises to 15% if we use network distance approximations when interiors are not added. Of particular note, the weighted motif method [16] and the persistent homology method [76] do not yield clear clustering results. Misclassifications are in the order of 30%.

The proposed method would have difficulty when the number of nodes in a given network is in the range of thousands, for the reason that the number of added nodes in the interior of a given network is proportional to the number of edges in the network. An important observation here is that adding *some* point in the interior is helpful even if filling *all* of the interiors is intractable. Thus, a reasonable heuristic for large-scale networks is to add interior points in some randomly selected edges. Figure 50 (b) illustrates the result of classifying generative models when the number of interior points is limited to be the same as the number of nodes in the network. Network sizes vary between 20 and 1,000 nodes. The wide range of number of nodes makes the clustering structure less clear but we still see three clusters representing the three generating models. Unsupervised classification yields 13 out of 60 errors (21%) for networks with randomly chosen interiors added. The error rate raises to 26% if we use network distance approximations without interiors added.

The methods in [16] and [76] do not yield meaningful clustering structures, with misclassification rate higher than 40%; essentially these alternative methods do not recover different generative models used in constructing the networks.

Part II

Applied Graph Signal Processing

Chapter 7

Graph Signal Processing

Graph signal processing considers a directed graph G with a set of N nodes or vertices $\mathcal{N}$ and a set of links $\mathcal{E}$, such that if node n is connected to m, then $(i, j) \in \mathcal{E}$. The (incoming) neighborhood of n is defined as the set of nodes $\mathcal{N}_i = \{j \,|\, (j, i) \in \mathcal{E}\}$ connected to n. For any given graph we define the adjacency matrix $\mathbf{A}$ as a sparse $N \times N$ matrix with non-zero elements $A_{j,i}$ if and only if $(i, j) \in \mathcal{E}$. The value of $A_{j,i}$ captures the strength of the connection from i to j.

7.1. Graph Signals and Shift Operator

The focus of GSP is not on analyzing G, but graph signals defined on the set of nodes $\mathcal{N}$. Formally, each of these signals can be represented as a vector $\mathbf{z} = [z_1, ..., z_N]^\top \in \mathbb{R}^N$ where the n-th element represents the value of the signal at node n. Since this vectorial representation does not convey explicitly the topology of G, the graph is endowed with a sparse *graph-shift operator* that captures the local structure of G. Typical choices for this shift operator are the adjacency matrix [57, 58], and the graph Laplacian [59],

$$L_{ij} = -A_{ij}, \quad L_{ii} = \sum_{j=1}^{n} A_{ij}. \tag{7.1}$$

To facilitate the connections with NNM, in this work we chose as shift the adjacency matrix $\mathbf{A}$; however, our results can be easily generalized for other choices too. We assume henceforth that $\mathbf{S}$ is diagonalizable, so that $\mathbf{S} = \mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^{-1}$ with $\boldsymbol{\Lambda} = \operatorname{diag}(\boldsymbol{\lambda}) \in \mathbb{C}^{N \times N}$ being diagonal and $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_N]$. When $\mathbf{S}$ is symmetric we have that $\mathbf{V}$ is real and unitary, which implies $\mathbf{V}^{-1} = \mathbf{V}^\top$. The intuition behind looking at $\mathbf{S}$ as an operator is to represent a transformation that can be computed locally at the nodes of the graph. More rigorously, if $\mathbf{z}'$ is defined as $\mathbf{z}' = \mathbf{S}\mathbf{z}$, then node n can compute y'_n provided that it has access to the value of y_m at $m \in \mathcal{N}_n$.

Note that for classical time signals the adjacency network can be set to $\mathbf{S} = \mathbf{S}_{dc}$, where $\mathbf{S}_{dc}$ is the adjacency matrix of the directed chain graph, whose entries all zero except for $[\mathbf{S}_{dc}]_{1,N}$ and the

$N-1$ entries $[\mathbf{S}_{dc}]_{i+1,i}$, which are all one.

7.2. Graph Filters

Graph filters are a particular class of *linear* graph-signal operators that can be represented as matrix polynomials of the shift operator $\mathbf{S}$ [57]

$$\mathbf{H} := \sum_{l=0}^{L-1} h_l \mathbf{S}^l. \tag{7.2}$$

For a given input $\mathbf{z}$, the output of the filter is simply $\mathbf{z}' = \mathbf{H}\mathbf{z}$. The filter coefficients are collected into $\mathbf{h} := [h_0, \ldots, h_{L-1}]^\top$, with $L-1$ denoting the filter degree. Graph filters are of particular interest because they represent linear transformations that can be implemented distributedly with $L-1$ exchanges of information among neighbors [157].

7.3. Frequency Representations

The eigendecomposition of $\mathbf{S}$ is used to define the frequency representation of graph signals and filters as stated next.

Definition 32 *Consider a signal $\mathbf{z} \in \mathbb{R}^N$ and a graph shift operator $\mathbf{S} = \mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^{-1} \in \mathbb{R}$. Then, the vectors*

$$\tilde{\mathbf{z}} = \mathbf{V}^{-1}\mathbf{z} \quad \text{and} \quad \mathbf{z} = \mathbf{V}\tilde{\mathbf{z}} \tag{7.3}$$

form a Graph Fourier Transform (GFT) pair [59, 57].

There are several reasons that justify the association of the GFT with the Fourier transform. Mathematically, it is just a matter of definition that if the vectors $\mathbf{v}_k$ are of the form $\mathbf{v}_k = [1, e^{j2\pi k/n}, \ldots, e^{j2\pi k(n-1)/n}]^\top$, the GFT and iGFT in Definition 32 reduce to the conventional time domain Fourier and inverse Fourier transforms. More deeply, it is not difficult to see that if the graph $\mathcal{G}$ is a cycle, the vectors $\mathbf{v}_k$ in are of the form $\mathbf{v}_k = [1, e^{j2\pi k/n}, \ldots, e^{j2\pi k(n-1)/n}]^\top$. Since cycle graphs are representations of discrete periodic signals, it follows that the GFT of a time signal is equivalent to the conventional discrete Fourier transform.

The GFT encodes a notion of variability for graph signals akin to one that the Fourier transform encodes for temporal signals [57, 58]. Specifically, the smaller the distance between λ_p and $|\lambda_{\max}|$ in the complex spectrum, the lower the frequency it represents. This idea is based on defining the total variation of a graph signal $\mathbf{z}$ as $\mathrm{TV}(\mathbf{z}) = \|\mathbf{z} - \mathbf{S}\mathbf{z}/\lambda_{\max}(\mathbf{S})\|_1$, with smoothness being associated to small values of TV. Then, given a $(\lambda_p, \mathbf{v}_p)$ pair, one has that $\mathrm{TV}(\mathbf{v}_p) = \|(1 - \lambda_p/\lambda_{\max}(\mathbf{S}))\mathbf{v}_p\|_1$, which provides an intuitive way to order the different frequencies. This statement holds true for real and complex and eigenvalues and eigenvectors.

Definition 33 *Consider a graph shift operator $\mathbf{S} = \mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^{-1} \in \mathbb{R}^{N\times N}$, a graph filter $\mathbf{H} = \sum_{l=0}^{L-1} h_l \mathbf{A}^l$, and form the Vandermonde matrix $\boldsymbol{\Psi} \in \mathbb{C}^{N\times L}$ with $\Psi_{p,l} = \lambda_p^{l-1}$. Then, the frequency response $\tilde{\mathbf{h}} \in \mathbb{C}^N$ of* *the filter is*

$$\mathbf{H} := \mathbf{V}\mathrm{diag}(\tilde{\mathbf{h}})\mathbf{V}^{-1} \quad \textit{where} \quad \tilde{\mathbf{h}} = \mathbf{\Psi}\mathbf{h} \tag{7.4}$$

and $\mathbf{\Psi}$ *is the GFT for filters [57] with*

$$\mathbf{\Psi} := \begin{bmatrix} 1 & \lambda_0 & \cdots & \lambda_0^{n-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \lambda_{n-1} & \cdots & \lambda_{n-1}^{n-1} \end{bmatrix}. \tag{7.5}$$

From the previous definitions is follows readily that if $\mathbf{z}' = \mathbf{H}\mathbf{z}$, then $\tilde{\mathbf{z}}' = \mathrm{diag}(\tilde{\mathbf{h}})\tilde{\mathbf{z}}$. In words, graph filters are orthogonal operators in the frequency domain.

Chapter 8

Diffusion Filtering and Application to Cancer Subtype Classification

We begin this chapter with a brief introduction of basic concepts in graph theory and metric geometry followed by a formal description of diffusion dynamics in networks (Section 8.1). This preliminary discussion provides the necessary elements for a formal definition of the superposition and diffusion distances. In Section 8.2 we define the superposition distance between two signals with respect to a given graph and a given input norm. To determine this distance the signals are diffused in the graph, the input norm of their difference is computed for all times, and the result is discounted by an exponential factor and integrated over time. We show that the superposition distance is a valid metric between vectors supported in the node set of a graph.

The diffusion distance with respect to a given graph and a given input norm is introduced in Section 8.3 as an alternative way of measuring the distance between two signals in a graph. In this case the diffused signals are also exponentially discounted and integrated over time but the input norm is taken after time integration. The diffusion distance is shown to also be a valid metric in the space of signals supported on a given graph and is further shown to provide a lower bound for the superposition distance. Different from the superposition distance, the diffusion distance can be reduced to a closed form expression with computational cost dominated by one matrix inversion. The superposition distance requires numerical integration of the time integral of the norm of a matrix exponential.

We further address stability with respect to uncertainty in the specification of the network (Section 8.4). Specifically, we prove that when the input norm is either the 1-norm, the 2-norm, or the infinity-norm a small perturbation in the underlying network transports linearly to a small perturbation in the values of the superposition and diffusion distances. In Section 8.5 we demonstrate that the diffusion and superposition distances can be applied to classify signals in graphs with better accuracy than comparisons that utilize traditional vector distances. We illustrate the differences using synthetic data (Section 8.5.1) and establish the practical advantages through the classification of ovarian cancer histologies from gene mutation profiles of different patients (Section 8.5.2). In Section 8.5.3, we utilize diffusion in label propagation process in semi-supervised learning and present its benefit through the classification of handwritten digits.

8.1. Norms and Diffusion Dynamics

In some of our proofs we encounter norms induced in the vector space of matrices $\mathbb{R}^{n\times n}$ by norms defined in the vector space $\mathbb{R}^n$. For a given vector norm $\|\cdot\| : \mathbb{R}^n \to \mathbb{R}_+$ the induced matrix norm $\|\cdot\| : \mathbb{R}^{n\times n} \to \mathbb{R}_+$ is defined as

$$\|\mathbf{A}\| := \sup_{\|\mathbf{x}\|=1} \|\mathbf{A}\mathbf{z}\|. \tag{8.1}$$

I.e., the induced norm of matrix $\mathbf{A}$ is equal to the maximum achievable vector norm when multiplying $\mathbf{A}$ by a vector with unit norm. Apart from satisfying the three requirements in the definition of norms, induced matrix norms are compatible and submultiplicative [158, Section 2.3]. That they are submultiplicative means that for any given pair of matrices $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{n\times n}$ the norm of the product does not exceed the product of the norms,

$$\|\mathbf{A}\mathbf{B}\| \leq \|\mathbf{A}\|\,\|\mathbf{B}\|. \tag{8.2}$$

That they are compatible means that for any vector $\mathbf{z} \in \mathbb{R}^n$ and matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ it holds,

$$\|\mathbf{A}\mathbf{z}\| \leq \|\mathbf{A}\|\,\|\mathbf{z}\|. \tag{8.3}$$

I.e., the vector norm of the product $\mathbf{Az}$ does not exceed the product of the norm of the vector $\mathbf{z}$ and the induced norm of the matrix $\mathbf{A}$.

Compared to metric spaces defined in Definition 3, norms are more stringent than metrics because they require the existence of a null element with null norm. However, whenever a norm is defined on a vector space V it induces a distance in the same space as we formally state next [131, Section 1].

Lemma 2 *Given any norm $\|\cdot\|$ on some vector space V, the function $d : V \times V \to \mathbb{R}_+$ defined as $d(\mathbf{x}, \mathbf{y}) := \|\mathbf{y} - \mathbf{x}\|$ for all pairs $\mathbf{x}, \mathbf{y} \in V$ is a metric.*

Consider the graph G with Laplacian matrix $\mathbf{L}$ and a vector $\mathbf{x}$. For a given constant $\alpha > 0$, define the time-varying vector $\mathbf{x}(t) \in \mathbb{R}^n$ as the solution of the linear differential equation

$$\frac{\mathrm{d}\,\mathbf{x}(t)}{\mathrm{d}\,t} = -\alpha\,\mathbf{L}\,\mathbf{x}(t), \qquad \mathbf{x}(0) = \mathbf{x}. \tag{8.4}$$

The differential equation in (8.4) represents heat diffusion on the graph G because $-\mathbf{L}$ can be shown to be the discrete approximation of the continuous Laplacian operator used to describe the diffusion of heat in physical space [159]. The given vector $\mathbf{x} = \mathbf{x}(0)$ specifies the initial temperature distribution and $\mathbf{x}(t)$ represents the temperature distribution at time t. The constant α is the thermal conductivity – which depends on the units used to measure the weights on the graph – and controls the heat diffusion rate. Larger α results in faster changing $r(t)$. The solution of (8.4) is given by

$$\mathbf{r}(t) = e^{-\alpha \mathbf{L} t}\, \mathbf{r}, \tag{8.5}$$

where, for an arbitrary matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$, the matrix exponential $e^{\mathbf{A}}$ is defined as [160]

$$e^{\mathbf{A}} = \sum_{k=0}^{\infty} \frac{1}{k!} \mathbf{A}^k. \tag{8.6}$$

Direct substitution is enough to confirm that indeed $\mathbf{x}(t) = e^{-\alpha \mathbf{L} t}\mathbf{x}$ is a solution of (8.4). The expression in (8.5) allows us to compute the temperature distribution at any point in time given the initial heat configuration $\mathbf{x}$ and the structure of the underlying network through its Laplacian $\mathbf{L}$. Notice that as time grows, $\mathbf{x}(t)$ settles to an isothermal equilibrium – all nodes have the same temperature – if the graph is connected.

It is instructive to rewrite (8.4) componentwise. If we focus on the variation of the i-th component of $\mathbf{x}(t)$, it follows that (8.4) implies

$$\frac{\mathrm{d}\, \mathbf{x}_i(t)}{\mathrm{d}\, t} = \sum_{j=1}^{n} \alpha\, \mathbf{A}_{ij} \left(x_j(t) - x_i(t) \right). \tag{8.7}$$

(8.7) is describing the flow of heat through edges of the graph. The flow of heat on an edge grows proportionally with the temperature differential $x_j(t) - x_i(t)$ as well as with the proximity A_{ij}. Nodes with larger proximity tend to equalize their temperatures faster, other things being equal. In particular, two initial vectors $\mathbf{x}(0) = \mathbf{x}$ and $\mathbf{y}(0) = \mathbf{y}$ result in similar temperature distributions across time if they are themselves similar – all x_i and y_i components are close –, or if they have similar initial levels at nodes with larger proximity – each component x_i need not be similar to s_i itself but might be similar to the component y_j of a neighboring node for which the edge weight A_{ij} is large. This latter fact suggests that the diffused vectors $\mathbf{x}(t)$ and $\mathbf{y}(t)$ define a notion of proximity between $\mathbf{x}$ and $\mathbf{y}$ associated with the underlying graph structure. We exploit this observation to define distances between signals supported on graphs in the following two sections.

8.2. Superposition Distance

Given a graph G with Laplacian matrix $\mathbf{L}$, an input vector norm $\|\cdot\|$, and two signals $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ defined in the node space $\mathcal{V}$, we define the superposition distance $d^{\mathbf{L}}_{\text{sps}}(\mathbf{x}, \mathbf{y})$ between $\mathbf{x}$ and $\mathbf{y}$ as

$$d^{\mathbf{L}}_{\text{sps}}(\mathbf{x}, \mathbf{y}) := \int_0^{+\infty} e^{-t} \left\| e^{-\alpha \mathbf{L} t} (\mathbf{x} - \mathbf{y}) \right\| \mathrm{d}t, \tag{8.8}$$

where $\alpha > 0$ corresponds to the diffusion constant in (8.4). As we mentioned in the discussion following (8.7), the distance $d^{\mathbf{L}}_{\text{sps}}(\mathbf{x}, \mathbf{y})$ defines a similarity between $\mathbf{x}$ and $\mathbf{y}$ that incorporates the underlying network structure. Indeed, notice that the term inside the input norm corresponds to the difference $\mathbf{x}(t) - \mathbf{y}(t)$ between the vectors that solve (8.4) for initial conditions $\mathbf{x}$ and $\mathbf{y}$ [cf. (8.5)]. This means that we are looking at the difference between the temperatures $\mathbf{x}(t)$ and $\mathbf{y}(t)$ at time t, which we then multiply by the dampening factor e^{-t} and integrate over all times. These temperatures are similar if $\mathbf{x}$ and $\mathbf{y}$ are similar, or, if $\mathbf{x}$ and $\mathbf{y}$ have similar values at similar nodes. The dampening factor gives more relative importance to the differences between $\mathbf{x}(t)$ and $\mathbf{y}(t)$ for early times. This is necessary because after prolonged diffusion times the network settles into an isothermal equilibrium and the structural differences between $\mathbf{x}$ and $\mathbf{y}$ are lost.

Exploiting the same interpretation, we can define the superposition norm of a vector $\mathbf{z} \in \mathbb{R}^n$ for a given graph with Laplacian matrix $\mathbf{L}$ and a given input norm $\|\cdot\|$ as

$$\|\mathbf{z}\|^{\mathbf{L}}_{\text{sps}} := \int_0^{+\infty} e^{-t} \left\| e^{-\alpha\, \mathbf{L}\, t} \mathbf{z} \right\| \, \mathrm{d}t. \tag{8.9}$$

Although we are referring to $d^{\mathbf{L}}_{\text{sps}}(\mathbf{x}, \mathbf{y})$ as the superposition distance between $\mathbf{x}$ and $\mathbf{y}$ and $\|\mathbf{z}\|^{\mathbf{L}}_{\text{sps}}$ as the superposition norm of $\mathbf{z}$ we have not proven that they indeed are valid definitions of distance and norm functions. As it turns out, they are. We begin by showing that $\|\cdot\|^{\mathbf{L}}_{\text{sps}}$ is a valid norm as we claim in the following proposition.

Proposition 12 *The function $\|\cdot\|^{\mathbf{L}}_{sps}$ in* (8.9) *is a valid norm on $\mathbb{R}^n$ for every Laplacian $\mathbf{L}$ and every input norm $\|\cdot\|$.*

Proof: As stated in Section 8.1, we need to show positiveness, positive homogeneity and subadditivity of $\|\cdot\|^{\mathbf{L}}_{\text{sps}}$. To show positive homogeneity, utilize the positive homogeneity of the input norm and the linearity of integrals to see that for every vector $\mathbf{z} \in \mathbb{R}^n$ and scalar β, it holds

$$\begin{aligned} \|\beta\mathbf{z}\|^{\mathbf{L}}_{\text{sps}} &= \int_0^{+\infty} e^{-t} \, \|e^{-\alpha\, \mathbf{L}\, t} \beta \mathbf{z}\| \, \mathrm{d}t \\ &= |\beta| \int_0^{+\infty} e^{-t} \, \|e^{-\alpha\, \mathbf{L}\, t} \mathbf{z}\| \, \mathrm{d}t = |\beta| \|\mathbf{z}\|^{\mathbf{L}}_{\text{sps}}. \end{aligned} \tag{8.10}$$

In order to show subadditivity, pick arbitrary vectors $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ and use the subadditivity of the input norm $\|\cdot\|$ and the linearity of integrals to see that

$$\begin{aligned} \|\mathbf{x} + \mathbf{y}\|^{\mathbf{L}}_{\text{sps}} &= \int_0^{+\infty} e^{-t} \, \|e^{-\alpha\, \mathbf{L}\, t} (\mathbf{x} + \mathbf{y})\| \, \mathrm{d}t \\ &\leq \int_0^{+\infty} e^{-t} \left(\|e^{-\alpha\, \mathbf{L}\, t} \mathbf{x}\| + \|e^{-\alpha\, \mathbf{L}\, t} \mathbf{y}\| \right) \mathrm{d}t = \|\mathbf{x}\|^{\mathbf{L}}_{\text{sps}} + \|\mathbf{y}\|^{\mathbf{L}}_{\text{sps}}, \end{aligned} \tag{8.11}$$

To show positiveness, first observe that for every $\mathbf{z} \in \mathbb{R}^n$ we have that $\|\mathbf{z}\|^{\mathbf{L}}_{\text{sps}} \geq 0$ since for every time t the argument of the integral in the definition (8.9) is the product of two nonnegative terms, an exponential and a norm which itself satisfies the positiveness property. The fact that

$\|\vec{0}\|_{\text{sps}}^{\mathbf{L}} = 0$ is an immediate consequence of the definition (8.9). Hence, we are only left to show that $\|\mathbf{z}\|_{\text{sps}}^{\mathbf{L}} \neq 0$ for $\mathbf{z} \neq 0$. To show this, it suffices to prove that the argument of the integral in (8.9) is strictly positive for every time t which is implied by the fact that the matrix $e^{-\alpha\,\mathbf{L}\,t}$ is strictly positive definite for every t. To see why this is true, notice that $-\alpha\,\mathbf{L}\,t$ is a real symmetric matrix, thus, it is diagonalizable and has real eigenvalues. Consequently, the eigenvalues of $e^{-\alpha\,\mathbf{L}\,t}$ are the exponentials of the eigenvalues of $-\alpha\,\mathbf{L}\,t$ which are strictly positive. ■

If the superposition norm is a valid norm as shown by Proposition 12 it induces a valid metric as per the construction in Lemma 2. This induced metric is the superposition distance defined in (8.8) as we show in the following corollary.

Corollary 5 *The function* $d_{sps}^{\mathbf{L}}$ *in* (8.8) *is a valid metric on* $\mathbb{R}^n$ *for every Laplacian* $\mathbf{L}$ *and every input norm* $\|\cdot\|$.

Proof: Since $d_{\text{sps}}^{\mathbf{L}}(\mathbf{x},\mathbf{y}) = \|\mathbf{x}-\mathbf{y}\|_{\text{sps}}^{\mathbf{L}}$ for all vectors $\mathbf{x},\mathbf{y} \in \mathbb{R}^n$ and $\|\cdot\|_{\text{sps}}^{\mathbf{L}}$ is a well-defined norm [cf. Proposition 12], Lemma 2 implies that $d_{\text{sps}}^{\mathbf{L}}$ is a metric on $\mathbb{R}^n$. ■

The distance $d_{\text{sps}}^{\mathbf{L}}$ incorporates the network structure to compare two signals r and s supported in a graph with Laplacian $\mathbf{L}$. As a particular case the edge set E of the underlying graph G may be empty. In this case, the Laplacian $\mathbf{L} = \mathbf{0}$ is identically null and we obtain from (8.8) that $d_{\text{sps}}^{\mathbf{0}}(\mathbf{x},\mathbf{y}) = \|\mathbf{y}-\mathbf{x}\|$. This is consistent with the fact that when no edges are present, the network structure adds no information to aid in the comparison of r and s and the superposition distance reduces to the standard distance induced by the input norm. The same effect is obtained when the thermal conductivity α is set to zero.

The computational cost of evaluating the superposition distance is significant in general. To evaluate $d_{\text{sps}}^{\mathbf{L}}(\mathbf{x},\mathbf{y})$ we approximate the improper integral in (8.8) with a finite sum and evaluate the norm of the matrix exponential $\left\|e^{-\alpha\,\mathbf{L}\,t}(\mathbf{y}-\mathbf{x})\right\|$ at the points required by the appropriate discretization. Notice that the decaying exponential modulation in (8.8) renders the first time points more relevant for the approximation, thus, a finer discrete time grid should be used for smaller times. An alternative notion of distance for graph-supported signals that is computationally more tractable comes in the form of the diffusion distance that we introduce in the next section.

8.3. Diffusion Distance

Given an arbitrary graph G with Laplacian $\mathbf{L}$, an input vector norm $\|\cdot\|$ and two signals $\mathbf{x},\mathbf{y} \in \mathbb{R}^n$ defined in the node space $\mathcal{V}$, the diffusion distance $d_{\text{diff}}^{\mathbf{L}}(\mathbf{x},\mathbf{y})$ between $\mathbf{x}$ and $\mathbf{y}$ is given by

$$d_{\text{diff}}^{\mathbf{L}}(\mathbf{x},\mathbf{y}) := \left\| \int_0^{+\infty} e^{-t}\, e^{-\alpha\,\mathbf{L}\,t}(\mathbf{y}-\mathbf{x})\,\mathrm{d}t \right\|, \tag{8.12}$$

with $\alpha > 0$ corresponding to the diffusion constant in (8.4). As in the case of the superposition distance in (8.8), the diffusion distance incorporates the graph structure in determining the proximity between $\mathbf{x}$ and $\mathbf{y}$ through the solutions $\mathbf{x}(t)$ and $\mathbf{y}(t)$ of (8.4) for initial conditions $\mathbf{x}$ and $\mathbf{y}$ [cf. (8.5)]. The difference is that in the diffusion distance the input norm of the difference between $\mathbf{x}(t)$ and $\mathbf{y}(t)$ is taken *after* discounting and integration, whereas in the superposition distance the input norm is applied *before* discounting and integration. An interpretation in terms of heat diffusion is that the diffusion distance compares the total (discounted) energy that passes trough each node. The superposition distance compares the energy difference at each point in time and integrates that difference over time. Both are reasonable choices. Computational aspects aside, whether the superposition or diffusion distance is preferable depends on the specific application.

A definite advantage of the diffusion distance is that the matrix integral in (8.12) can be resolved to obtain a closed solution that is more amenable to computation. To do so, notice that the primitive of the matrix exponential $e^{-t}e^{-\alpha\mathbf{L}t} = e^{-(\mathbf{I}+\alpha\mathbf{L})t}$ is given by $-(\mathbf{I}+\alpha\mathbf{L})^{-1}e^{-(\mathbf{I}+\alpha\mathbf{L})t}$ to conclude that (8.12) is equivalent to

$$d^{\mathbf{L}}_{\mathrm{diff}}(r,s) = \left\|(\mathbf{I}+\alpha\mathbf{L})^{-1}(\mathbf{x}-\mathbf{y})\right\|. \tag{8.13}$$

As in the case of the superposition distance of Section 8.2 a vector norm can be defined based on the same heat diffusion interpretation used to define the distance in (8.12). Therefore, consider a given a graph with Laplacian $\mathbf{L}$ and a given input norm $\|\cdot\|$ and define the diffusion norm of the vector $\mathbf{z} \in \mathbb{R}^n$ as

$$\|\mathbf{z}\|^{\mathbf{L}}_{\mathrm{diff}} := \left\|\int_0^{+\infty} e^{-t}\, e^{-\alpha\,\mathbf{L}\,t}\mathbf{z}\,\mathrm{d}t\right\| = \left\|(\mathbf{I}+\alpha\mathbf{L})^{-1}\mathbf{z}\right\|, \tag{8.14}$$

where the second equality follows from the same primitive expression used in (8.13).

The superposition distance is a proper metric and the superposition norm is a proper norm. We show first that $\|\cdot\|^{\mathbf{L}}_{\mathrm{diff}}$ is a valid norm as we formally state next.

Proposition 13 *The function* $\|\cdot\|^{\mathbf{L}}_{\mathit{diff}}$ *in* (8.14) *is a valid norm on* $\mathbb{R}^n$ *for every Laplacian* $\mathbf{L}$ *and every input norm* $\|\cdot\|$.

Proof: To prove the validity of $\|\cdot\|^{\mathbf{L}}_{\mathrm{diff}}$ we need to show positiveness, positive homogeneity and subadditivity; see Section 8.1. Positive homogeneity follows directly from the positive homogeneity of the input norm, i.e. for any vector $v \in \mathbb{R}^n$ and scalar β we have that

$$\|\beta v\|^{\mathbf{L}}_{\mathrm{diff}} = \|(\mathbf{I}+\alpha\mathbf{L})^{-1}\beta\mathbf{z}\| = |\beta|\|(\mathbf{I}+\alpha\mathbf{L})^{-1}\mathbf{z}\| = |\beta|\|\mathbf{z}\|^{\mathbf{L}}_{\mathrm{diff}}. \tag{8.15}$$

In order to show subadditivity, pick arbitrary vectors $\mathbf{z}, \mathbf{z}' \in \mathbb{R}^n$ and use the subadditivity of the input norm $\|\cdot\|$ to see that

$$\begin{aligned}\|\mathbf{z}+\mathbf{z}'\|^{\mathbf{L}}_{\mathrm{diff}} &= \|(\mathbf{I}+\alpha\mathbf{L})^{-1}(\mathbf{z}+\mathbf{z}')\| \\ &\leq \|(\mathbf{I}+\alpha\mathbf{L})^{-1}\mathbf{z}\| + \|(\mathbf{I}+\alpha\mathbf{L})^{-1}\mathbf{z}'\| = \|\mathbf{z}\|^{\mathbf{L}}_{\mathrm{diff}} + \|\mathbf{z}'\|^{\mathbf{L}}_{\mathrm{diff}}.\end{aligned}$$

Given the positiveness property of the input norm $\|\cdot\|$, to show positiveness of the diffusion norm $\|\cdot\|^{\mathbf{L}}_{\mathrm{diff}}$ it is enough to show that $(\mathbf{I}+\alpha\mathbf{L})^{-1}\mathbf{z} \neq \vec{0}$ for all vectors $\mathbf{z}t \in \mathbb{R}^n$ different from the null vector. This is implied by the fact that $(\mathbf{I}+\alpha\mathbf{L})^{-1}$ is a positive definite matrix. To see why $(\mathbf{I}+\alpha\mathbf{L})^{-1}$ is positive definite, first notice that $\mathbf{L}$ is positive semidefinite as stated in Section 8.1. Consequently, $\alpha\mathbf{L}$ is also positive semidefinite since $\alpha > 0$ and $\mathbf{I}+\alpha\mathbf{L}$ is positive definite since every eigenvalue of $\mathbf{I}+\alpha\mathbf{L}$ is a unit greater than the corresponding eigenvalues of $\alpha\mathbf{L}$, thus, strictly greater than 0. Finally, since inversion preserves positive definiteness, the proof is completed. ■

From Proposition 12 and Lemma 2 it follows directly that that the diffusion distance defined in (8.12) is a valid metric as we prove next.

Corollary 6 *The function $d_{\mathit{diff}}^{\mathbf{L}}$ in* (8.12) *is a valid metric on $\mathbb{R}^n$ for every Laplacian $\mathbf{L}$ and every input norm $\|\cdot\|$.*

Proof: Since $d_{\text{diff}}^{\mathbf{L}}(\mathbf{x},\mathbf{y}) = \|\mathbf{y}-\mathbf{x}\|_{\text{diff}}^{\mathbf{L}}$ for all vectors $\mathbf{x},\mathbf{y} \in \mathbb{R}^n$ and $\|\cdot\|_{\text{diff}}^{\mathbf{L}}$ is a well-defined norm [cf. Proposition 13], Lemma 2 implies that $d_{\text{diff}}^{\mathbf{L}}$ is a metric on $\mathbb{R}^n$. ■

As in the case of the superposition norm and distance, the diffusion norm and distance reduce to the input norm and its induced distance when the set edge is empty. In that case we have $\mathbf{L} = \mathbf{0}$ and it follows from the definitions in (8.14) and (8.12) that $\|\mathbf{z}\|_{\text{diff}}^{\mathbf{L}} = \|\mathbf{z}\|_{\text{diff}}^{\mathbf{0}} = \|\mathbf{z}\|$ and that $d_{\text{diff}}^{\mathbf{L}}(r,s) = d_{\text{diff}}^{\mathbf{0}}(\mathbf{x},\mathbf{y}) = \|\mathbf{y}-\mathbf{x}\|$.

The superposition and diffusion distance differ in the order in which the input norm and time integral are applied. It is therefore reasonable to expect some relationship to hold between their values. In the following proposition we show that the diffusion distance is a lower bound for the value of the superposition distance.

Proposition 14 *Given any graph G with Laplacian $\mathbf{L}$, any two signals $r,s \in \mathbb{R}^n$ defined in V and any input vector norm $\|\cdot\|$, the diffusion distance $d_{\mathit{diff}}^{\mathbf{L}}(\mathbf{x},\mathbf{y})$ defined in* (8.12) *is a lower bound on the superposition distance $d_{\mathit{sps}}^{\mathbf{L}}(\mathbf{x},\mathbf{y})$ defined in* (8.8)

$$d_{\mathit{sps}}^{\mathbf{L}}(\mathbf{x},\mathbf{y}) \geq d_{\mathit{diff}}^{\mathbf{L}}(\mathbf{x},\mathbf{y}). \tag{8.16}$$

Proof: Since the exponential e^{-t} in (8.8) is nonnegative, we may replace it with its absolute value to obtain

$$\begin{aligned} d_{\text{sps}}(\mathbf{x},\mathbf{y}) &= \int_0^{+\infty} |e^{-t}|\, \|e^{-\alpha\,\mathbf{L}\,t}(\mathbf{y}-\mathbf{x})\|\, \mathrm{d}t \\ &= \int_0^{+\infty} \|e^{-t}e^{-\alpha\,\mathbf{L}\,t}(\mathbf{y}-\mathbf{x})\|\, \mathrm{d}t, \end{aligned} \tag{8.17}$$

where we used the positive homogeneity property of the input norm to write the second equality. Further using the subadditivity property of the input norm we may write

$$d_{\text{sps}}(\mathbf{x},\mathbf{y}) \geq \left\| \int_0^{+\infty} e^{-t}e^{-\alpha\,\mathbf{L}\,t}(\mathbf{y}-\mathbf{x})\, \mathrm{d}t \right\|. \tag{8.18}$$

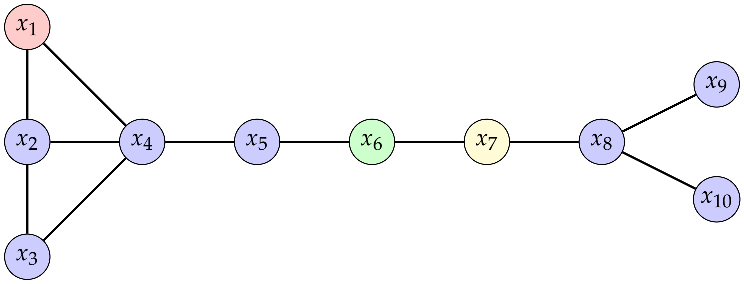
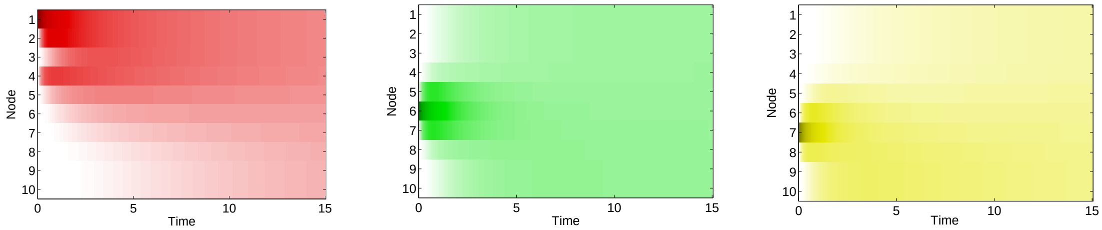


Figure 51: Example of an underlying graph used to compute the superposition and diffusion distances. Three signals **r**, **g** and **y** are compared taking a value of 1 in the red, green, and yellow nodes respectively, and zero everywhere else.



Diffusion of **r** Diffusion of **g** Diffusion of **y**

Figure 52: Heat maps of the diffused signals for **r**, **g**, and **y** as diffusion evolves for every node in the network in Figure 51. Darker colors represent stronger signals. The heat maps of **g** and **y** are more similar, entailing smaller diffusion and superposition distances.

The right hand side of (8.18) is the definition of the diffusion distance $d_{\text{diff}}(\mathbf{x}, \mathbf{y})$ in (8.12). Making this substitution in (8.18) yields (8.16). ■

For applications in which the superposition distance is more appropriate, the diffusion distance is still valuable because, as it follows from Proposition 14, it can be used as a lower bound on the superposition distance. This lower bound is useful because computing the diffusion distance is less expensive than computing the superposition distance.

8.3.1. Discussion

In order to illustrate the superposition and diffusion distances and their difference with the standard vector distances, consider the undirected graph in Figure 51 where the weight of each undirected edge is equal to 1. Define three different vectors supported in the node space and having exactly one component equal to 1 and the rest equal to 0. The vector **r** has its positive component for node x_1, colored in red, the vector **g** has its positive for node x_6, colored in green, and the vector **y** has its positive component for node x_7, colored in yellow.

For the traditional vector metrics, the distances between each of the vectors **r**, **g** and **y** are the same. In the case when, e.g., the ℓ_2 distance is used as input metric, we have that $\|\mathbf{r}-\mathbf{g}\|_2 = \|\mathbf{g}-\mathbf{y}\|_2 = \|\mathbf{y}-\mathbf{r}\|_2 = \sqrt{2}$. In the case of the ℓ_1 and ℓ_∞ distances we have that $\|\mathbf{r}-\mathbf{g}\|_1 =$

$\|\mathbf{g}-\mathbf{y}\|_1 = \|\mathbf{y}-\mathbf{r}\|_1 = 2$ and $\|\mathbf{r}-\mathbf{g}\|_\infty = \|\mathbf{g}-\mathbf{y}\|_\infty = \|\mathbf{y}-\mathbf{r}\|_\infty = 1$. However, by observing the network in Figure 51, it is intuitive that signals **g** and **y** should be more alike than they are to r since they affect nodes that are closely related. E.g., if we think of the vectors **r**, **g** and **y** as signaling faulty nodes in a communication network, it is evident that the impact of nodes x_6 and x_7 failing would disrupt the communication between the right and left components of the graph, whereas the failure of x_1 would entail a different effect. This intuition is captured by the diffusion and superposition distances. Indeed, if we fix $\alpha = 1$ and we use the ℓ_2 norm as input norm to the diffusion distance, we have that the distance between the vectors that signal faults at x_6 and x_7 are [cf. (8.13)]

$$d^{\mathbf{L}}_{\text{diff}}(\mathbf{g},\mathbf{y}) = \|(\mathbf{I}+\mathbf{L})^{-1}(\mathbf{g}-\mathbf{y})\|_2 = 0.418, \tag{8.19}$$

where **L** is the Laplacian of the graph in Figure 51. However, the diffusion distances from these green and yellow vectors to the red vector that signals a fault at node x_1 are

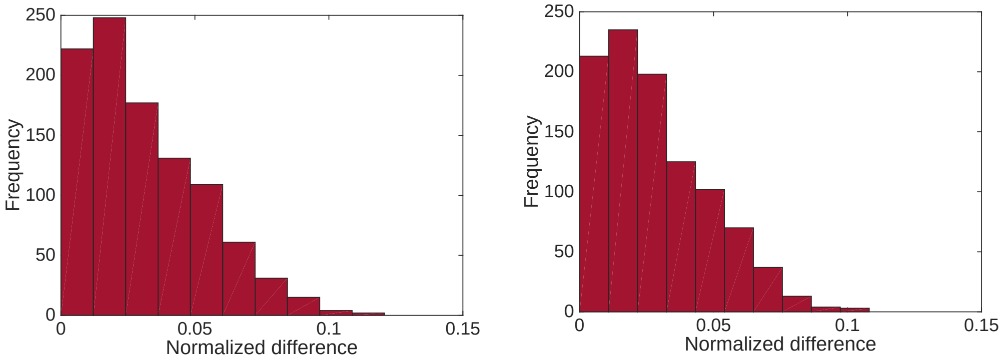
$$\begin{aligned} d^{\mathbf{L}}_{\text{diff}}(\mathbf{r},\mathbf{g}) &= \|(\mathbf{I}+\mathbf{L})^{-1}(\mathbf{r}-\mathbf{g})\|_2 = 0.664, \\ d^{\mathbf{L}}_{\text{diff}}(\mathbf{r},\mathbf{y}) &= \|(\mathbf{I}+\mathbf{L})^{-1}(\mathbf{r}-\mathbf{y})\|_2 = 0.698. \end{aligned} \tag{8.20}$$

The distances in (8.20) are larger than the distance in (8.19) signaling the relative similarity of the **g** and **y** vectors with respect to the **r** vector. The differences are substantial – almost 60% increase –, thus allowing identification of **g** and **y** as somehow separate from **r**. Further observe that the distance between **r** and **g** is slightly smaller than the distance between **r** and **y**. This is as it should be, because node x_1 is closer to node x_6 than to node x_7 in the underlying graph.

Repeating the exercise, but using the superposition distance instead [cf. (8.8)], we obtain that $d^{\mathbf{L}}_{\text{sps}}(\mathbf{r},\mathbf{g}) = 0.701$, $d^{\mathbf{L}}_{\text{sps}}(\mathbf{r},\mathbf{y}) = 0.742$, and $d^{\mathbf{L}}_{\text{sps}}(\mathbf{g},\mathbf{y}) = 0.456$. Although the numbers are slightly different, the qualitative conclusions are the same as those obtained for the diffusion distance. We can tell that g and y are more like each other than they are to r, and we can tell that g is slightly closer to r than y is. Also note that the diffusion distances are smaller than the superposition distances between the corresponding pairs, i.e., $d^{\mathbf{L}}_{\text{sps}}(\mathbf{r},\mathbf{g}) \geq d^{\mathbf{L}}_{\text{diff}}(\mathbf{r},\mathbf{g})$, $d^{\mathbf{L}}_{\text{sps}}(\mathbf{r},\mathbf{y}) \geq d^{\mathbf{L}}_{\text{diff}}(\mathbf{r},\mathbf{y})$, and $d^{\mathbf{L}}_{\text{sps}}(\mathbf{g},\mathbf{y}) \geq d^{\mathbf{L}}_{\text{diff}}(\mathbf{g},\mathbf{y})$. This is consistent with the result in Proposition 14.

To further illustrate the intuitive idea behind the diffusion and superposition distances, Figure 52 plots the evolution of the diffused signals $\mathbf{r}(t)$, $\mathbf{g}(t)$ and $\mathbf{y}(t)$ for each of the respective initial conditions **r**, **g**, and **y**. At time $t = 0$ each of the signals is concentrated at one specific node. The signals are, as a consequence, equally different to each other. At very long times, the signals are completely diffused and therefore indistinguishable. For intermediate times, the signal distributions across nodes for the green and yellow signals are more similar than between the green and red or yellow and red signals. This difference between the evolution of the diffused signals results in different values for the superposition and diffusion distances.

Remark 12 Computation of the diffusion distance using the closed form expression in (8.13) requires the inversion of the $n \times n$ identity plus Laplacian matrix followed by multiplication with the difference vector $\mathbf{y}-\mathbf{x}$. The cost of this computation is of order n^3, but is much smaller when



Diffusion distance

Superposition distance

Figure 53: Histogram of the absolute value of the normalized difference, i.e. $|d^{\mathbf{L}'}(\mathbf{g},\mathbf{r}) - d^{\mathbf{L}}(\mathbf{g},\mathbf{r})|/\|\boldsymbol{\Delta}\|_2$, for the diffusion and superposition distances. For this particular network and perturbations, the difference is considerably lower than the theoretical upper bound of 2.

the matrix $\mathbf{L}$ is sparse, as is typically the case. Further observe that most computations can be reused when computing multiple distances, because the vectors change, but the matrix inverse $(\mathbf{I} + \alpha\mathbf{L})^{-1}$ stays unchanged.

8.4. Stability

The superposition and diffusion distances depend on the underlying graphs through their Laplacian $\mathbf{L}$. It is therefore important to analyze how a perturbation of the underlying network impacts both distances. We prove in this section that these distances are well behaved with respect to perturbations of the underlying graph. I.e., we show that if the network perturbation is small, the change in the diffusion and superposition distances is also small. We think of a perturbation of a given network as noise added to its edge weights, thus, we quantify the network perturbation as the matrix p-norm of the difference between the Laplacians of the original and perturbed networks. We focus our analysis on the most frequently used norms where $p \in \{1, 2, \infty\}$. We begin with a formal statement for the case of the superposition distance defined by (8.8).

Proposition 15 *Given any graph with Laplacian* $\mathbf{L}$, *an input* ℓ_p *norm* $\|\cdot\|_p$ *with* $p \in \{1, 2, \infty\}$, *and bounded signals* $\mathbf{x}$ *and* $\mathbf{y}$ *on the network with* $\|\mathbf{x}\|_p \le \gamma$ *and* $\|\mathbf{y}\|_p \le \gamma$, *if we perturb the network such that the resulting Laplacian* $\mathbf{L}' = \mathbf{L} + \boldsymbol{\Delta}$ *where the perturbation* $\boldsymbol{\Delta}$ *is such that* $\|\boldsymbol{\Delta}\|_p \le \delta\|\mathbf{L}\|_p < 1$, *then*

$$\left| d_{sps}^{\mathbf{L}'}(\mathbf{x},\mathbf{y}) - d_{sps}^{\mathbf{L}}(\mathbf{x},\mathbf{y}) \right| \le 2\gamma\|\mathbf{L}\|_p\delta. \tag{8.21}$$

Proof: The following lemma is central to the proof of Proposition 15.

Lemma 3 *Given the Laplacian* $\mathbf{L}$ *for some undirected network, the matrix exponential of nonpositive multiples of the Laplacian* $e^{-\tau\mathbf{L}}$ *with* $\tau \ge 0$ *is a doubly stochastic matrix.*

Proof: Since $\mathbf{L} = \mathbf{D} - \mathbf{A}$, all off-diagonal components of $-\tau\mathbf{L}$ are nonnegative, making $-\tau\mathbf{L}$ a Metzler matrix [161]. Since the exponentials of Metzler matrices are nonnegative [161, Theorem 8.2], we are guaranteed that all elements of $e^{-\tau\mathbf{L}}$ are nonnegative. From the power series of matrix exponentials, we have

$$e^{-\tau\mathbf{L}} = \sum_{k=0}^{\infty} \frac{1}{k!}(-\tau\mathbf{L})^k = \mathbf{I} - \tau\mathbf{L} + \frac{\tau^2\mathbf{L}^2}{2} - \frac{\tau^3\mathbf{L}^3}{3!} + \cdots . \tag{8.22}$$

If we are able to show that all rows and columns of $\mathbf{L}^k$ add up to 0 for any integer $k \geq 1$, then we know that all rows and columns of $\sum_{k=1}^{\infty}(-\tau\mathbf{L})^k/k!$ also add up to 0. Therefore, when we add the identity matrix to this summation to obtain the exponential $e^{-\tau\mathbf{L}}$ as in (8.22) we are guaranteed that the rows and columns sum up to 1. Combining this with the non negativity of $e^{-\tau\mathbf{L}}$ implies doubly stochasticity, as wanted. To see that the rows and columns of $\mathbf{L}^k$ indeed add up to 0 for any integer $k \geq 1$, denote by $\mathbf{1}$ and $\mathbf{0}$ the vectors of all-ones and all-zeros, respectively. Then, by the definition of the graph Laplacian, it follows that $\mathbf{1}^T\mathbf{L} = \mathbf{L}\mathbf{1} = \mathbf{0}$ which immediately implies that $\mathbf{1}^T\mathbf{L}^k = \mathbf{L}^k\mathbf{1} = \mathbf{0}$ for all $k \geq 1$. ■

We now use Lemma 3 to show Proposition 15. Given the definition of $\mathbf{L}'$, from (8.8) we have that

$$d_{\text{sps}}^{\mathbf{L}'}(\mathbf{x},\mathbf{y}) = \int_0^{\infty} e^{-t} \left\| e^{-(\mathbf{L}+\mathbf{\Delta})t}(\mathbf{y}-\mathbf{x}) \right\|_p dt, \tag{8.23}$$

where without loss of generality we assume $\alpha = 1$. If $\alpha \neq 1$, then $\alpha\mathbf{L}'$ defines a Laplacian and we can think of the distance $d_{\text{sps}}^{\alpha\mathbf{L}'}(\mathbf{x},\mathbf{y})$ where the new α parameter is equal to 1. If we focus on the input norm $\|\cdot\|_p$ inside the integral in (8.23), we may add and subtract $e^{-\mathbf{L}t}(\mathbf{x}-\mathbf{y})$ to obtain

$$\begin{aligned}\left\| e^{-(\mathbf{L}+\mathbf{\Delta})t}(\mathbf{x}-\mathbf{y}) \right\|_p &= \left\| \left(e^{-(\mathbf{L}+\mathbf{\Delta})t} - e^{-\mathbf{L}t}\right)(\mathbf{x}-\mathbf{y}) + e^{-\mathbf{L}t}(\mathbf{x}-\mathbf{y}) \right\|_p \\ &\leq \left\| \left(e^{-(\mathbf{L}+\mathbf{\Delta})t} - e^{-\mathbf{L}t}\right)(\mathbf{x}-\mathbf{y}) \right\|_p + \left\| e^{-\mathbf{L}t}(\mathbf{x}-\mathbf{y}) \right\|_p ,\end{aligned} \tag{8.24}$$

where we used the subadditivity property of the input norm. To further bound the first term on the right hand side of (8.24) we apply the compatibility property of p-norms (8.3) followed by the subadditivity property to obtain that

$$\begin{aligned}&\left\| \left(e^{-(\mathbf{L}+\mathbf{\Delta})t} - e^{-\mathbf{L}t}\right)(\mathbf{x}-\mathbf{y}) \right\|_p \leq \left\| e^{-(\mathbf{L}+\mathbf{\Delta})t} - e^{-\mathbf{L}t} \right\|_p \|(\mathbf{x}-\mathbf{y})\|_p \\ &\leq \left\| e^{-(\mathbf{L}+\mathbf{\Delta})t} - e^{-\mathbf{L}t} \right\|_p \left(\|\mathbf{x}\|_p + \|\mathbf{y}\|_p\right).\end{aligned} \tag{8.25}$$

In order to bound the first term on the right hand side of (8.25), we use a well-known result in matrix exponential analysis [160, 162] that allows us to write the difference of matrix exponentials in terms of an integral,

$$\begin{aligned}\left\|e^{-(\mathbf{L}+\boldsymbol{\Delta})t}-e^{-\mathbf{L}t}\right\|_p &= \left\|\int_0^t e^{-\mathbf{L}(t-\tau)}\boldsymbol{\Delta}e^{-(\mathbf{L}+\boldsymbol{\Delta})\tau}d\tau\right\|_p \\ &\leq \int_0^t \left\|e^{-\mathbf{L}(t-\tau)}\boldsymbol{\Delta}e^{-(\mathbf{L}+\boldsymbol{\Delta})\tau}\right\|_p d\tau \\ &\leq \|\boldsymbol{\Delta}\|_p \int_0^t \left\|e^{-\mathbf{L}(t-\tau)}\right\|_p \left\|e^{-(\mathbf{L}+\boldsymbol{\Delta})\tau}\right\|_p d\tau, \end{aligned} \tag{8.26}$$

where the first inequality follows from subadditivity of the input p-norm and the second one from submultiplicativity (8.2).

We now bound each of the three terms on the right hand side of (8.26). For the first term, $\|\boldsymbol{\Delta}\|_p \leq \delta\|\mathbf{L}\|_p$ by assumption. From Lemma 3, the doubly stochasticity of $e^{-\mathbf{L}(t-\tau)}$ implies that $\|e^{-\mathbf{L}(t-\tau)}\|_1 = \|e^{-\mathbf{L}(t-\tau)}\|_\infty = 1$. For $p = 2$, notice that $-\mathbf{L}(t-\tau)$ is a negative semi-definite matrix with an eigenvalue at 0. Since the eigenvalues of $e^{-\mathbf{L}(t-\tau)}$ are equal to the exponentials of the eigenvalues of $-\mathbf{L}(t-\tau)$, it follows that the largest eigenvalue of $e^{-\mathbf{L}(t-\tau)}$ is 1 and hence $\|e^{-\mathbf{L}(t-\tau)}\|_2 = 1$. For the term $\left\|e^{-(\mathbf{L}+\boldsymbol{\Delta})\tau}\right\|_p$, notice that $\mathbf{L}+\boldsymbol{\Delta} = \mathbf{L}'$ is in itself a Laplacian, meaning that we can follow the aforementioned argument and upper bound this term by 1. Substituting these bounds in (8.26) and solving the integral yields

$$\left\|e^{-(\mathbf{L}+\boldsymbol{\Delta})t}-e^{-\mathbf{L}t}\right\|_p \leq \delta\|\mathbf{L}\|_p\, t. \tag{8.27}$$

Further substitution in (8.25) combined with the fact that $\|\mathbf{x}\|_p \leq \gamma$ and $\|\mathbf{y}\|_p \leq \gamma$, results in

$$\left\|\left(e^{-(\mathbf{L}+\boldsymbol{\Delta})t}-e^{-\mathbf{L}t}\right)(\mathbf{x}-\mathbf{y})\right\|_p \leq 2\gamma\delta\|\mathbf{L}\|_p\, t. \tag{8.28}$$

By substituting this result in (8.24) and inputing the resultant inequality in the integral in (8.23) we conclude that

$$d_{\text{sps}}^{\mathbf{L}'}(\mathbf{x},\mathbf{y}) \leq \int_0^\infty te^{-t}2\gamma\delta\|\mathbf{L}\|_p dt + \int_0^\infty e^{-t}\left\|e^{-\mathbf{L}t}(\mathbf{x}-\mathbf{y})\right\|_p dt. \tag{8.29}$$

Notice that the rightmost summand in (8.29) is exactly equal to $d_{\text{sps}}^{\mathbf{L}}(\mathbf{x},\mathbf{y})$ [cf. (8.8)]. Thus, solving the integral in the first summand we get that

$$d_{\text{sps}}^{\mathbf{L}'}(\mathbf{x},\mathbf{y}) - d_{\text{sps}}^{\mathbf{L}}(\mathbf{x},\mathbf{y}) \leq 2\gamma\delta\|\mathbf{L}\|_p. \tag{8.30}$$

Following the same methodology but starting from the definition of $d_{\text{sps}}^{\mathbf{L}'}(\mathbf{x},\mathbf{y})$, it can be shown that

$$d_{\text{sps}}^{\mathbf{L}}(\mathbf{x},\mathbf{y}) - d_{\text{sps}}^{\mathbf{L}'}(\mathbf{x},\mathbf{y}) \leq 2\gamma\delta\|\mathbf{L}\|_p. \tag{8.31}$$

Finally, by combining (8.30) and (8.31), we obtain (8.21), concluding the proof. ■

Proposition 15 guarantees that for any two vectors, the difference between their superposition distances computed based on different underlying graphs is bounded by a term which is bilinear in a bound on the magnitude of the input vectors γ and a bound on the difference between the Laplacians of both underlying graphs $\|\mathbf{\Delta}\|_p \leq \delta \|\mathbf{L}\|_p$. This implies that vanishing perturbations on the underlying network have vanishing effects on the distance between two signals defined on the network.

Similarly to the case of the superposition distance, perturbations have limited effect on the diffusion metric defined in (8.12) as shown next.

Proposition 16 *For the same setting described in Proposition 15, we have that*

$$\left| d_{\mathit{diff}}^{\mathbf{L}'}(\mathbf{x}, \mathbf{y}) - d_{\mathit{diff}}^{\mathbf{L}}(\mathbf{x}, \mathbf{y}) \right| \leq 2\gamma \|\mathbf{L}\|_p \delta + o(\delta). \tag{8.32}$$

Proof: In the proof of Proposition 16 we use two lemmas. The first one is similar to Lemma 3 and shows that $(\mathbf{I} + \mathbf{L})^{-1}$ is doubly stochastic.

Lemma 4 *Given the Laplacian* $\mathbf{L}$ *for some undirected network, the inverse of the Laplacian plus identity matrix* $(\mathbf{I} + \mathbf{L})^{-1}$ *is a doubly stochastic matrix.*

Proof: Since all the off-diagonal entries of $\mathbf{I} + \mathbf{L}$ are less than or equal to zero, $\mathbf{I} + \mathbf{L}$ is a Z-matrix [163]. Moreover, due to the fact that all eigenvalues of $\mathbf{I} + \mathbf{L}$ have positive real parts, $\mathbf{I} + \mathbf{L}$ is an M-matrix. Since the inverse of an M-matrix is elementwise nonnegative [164], $(\mathbf{I} + \mathbf{L})^{-1}$ is a nonnegative matrix. Thus, to show doubly stochasticity, we only need to prove that all rows and columns of $(\mathbf{I} + \mathbf{L})^{-1}$ add up to 1. Recall that $\mathbf{1}$ and $\mathbf{0}$ stand for the vectors of all-ones and all-zeros, respectively, and that $\mathbf{L}\mathbf{1} = \mathbf{0}$ [cf. (7.1)] Thus, we may write $(\mathbf{I} + \mathbf{L})\mathbf{1} = \mathbf{1}$ from which we have that

$$\mathbf{1} = (\mathbf{I} + \mathbf{L})^{-1} (\mathbf{I} + \mathbf{L})\mathbf{1} = (\mathbf{I} + \mathbf{L})^{-1} \mathbf{1}, \tag{8.33}$$

showing that all the rows of $(\mathbf{I} + \mathbf{L})^{-1}$ sum up to 1. Similarly, it can be shown that all the columns of $(\mathbf{I} + \mathbf{L})^{-1}$ sum up to 1, concluding the proof. ■

The second lemma is a statement about the stability of inverse matrices.

Lemma 5 *If* $\mathbf{A}$ *is nonsingular and* $\|\mathbf{A}^{-1}\mathbf{\Delta}\|_p < 1$, *then* $\mathbf{A} + \mathbf{\Delta}$ *is nonsingular and it is guaranteed that*

$$\left\| (\mathbf{A} + \mathbf{\Delta})^{-1} - \mathbf{A}^{-1} \right\|_p \leq \frac{\|\mathbf{\Delta}\|_p \|\mathbf{A}^{-1}\|_p^2}{1 - \|\mathbf{A}^{-1}\mathbf{\Delta}\|_p}. \tag{8.34}$$

Proof: See [158, Theorem 2.3.4]. ■

We now use Lemmas 4 and 5 to show Proposition 16. Given the definition of $\mathbf{L}'$, from (8.13) we have that

$$d_{\text{diff}}^{\mathbf{L}'}(\mathbf{x}, \mathbf{y}) = \left\| (\mathbf{I} + \mathbf{L} + \boldsymbol{\Delta})^{-1} (\mathbf{x} - \mathbf{y}) \right\|_p . \tag{8.35}$$

As in the proof of Proposition 15, we can assume that $\alpha = 1$ without loss of generality. Subtracting and adding $(\mathbf{I} + \mathbf{L})^{-1}(\mathbf{x} - \mathbf{y})$ from (8.35) and applying the subadditivity property of the p-norm implies

$$d_{\text{diff}}^{\mathbf{L}'}(\mathbf{x}, \mathbf{y}) \leq \left\| \left((\mathbf{I} + \mathbf{L} + \boldsymbol{\Delta})^{-1} - (\mathbf{I} + \mathbf{L})^{-1} \right) (\mathbf{x} - \mathbf{y}) \right\|_p + \left\| (\mathbf{I} + \mathbf{L})^{-1} (\mathbf{x} - \mathbf{y}) \right\|_p , \tag{8.36}$$

where the second term in the sum is exactly $d_{\text{diff}}^{\mathbf{L}}(\mathbf{x}, \mathbf{y})$ [cf. (8.13)]. Therefore we may write

$$d_{\text{diff}}^{\mathbf{L}'}(\mathbf{x}, \mathbf{y}) - d_{\text{diff}}^{\mathbf{L}}(\mathbf{x}, \mathbf{y}) \leq \| ((\mathbf{I} + \mathbf{L} + \boldsymbol{\Delta})^{-1} - (\mathbf{I} + \mathbf{L})^{-1}) (\mathbf{x} - \mathbf{y}) \|_p . \tag{8.37}$$

By applying compatibility of p-norms (8.3) followed by the subadditivity property we obtain that

$$\begin{aligned} d_{\text{diff}}^{\mathbf{L}'}(\mathbf{x}, \mathbf{y}) - d_{\text{diff}}^{\mathbf{L}}(\mathbf{x}, \mathbf{y}) &\leq \| ((\mathbf{I} + \mathbf{L} + \boldsymbol{\Delta})^{-1} - (\mathbf{I} + \mathbf{L})^{-1}) \|_p \, \| (\mathbf{x} - \mathbf{y}) \|_p \\ &\leq \| ((\mathbf{I} + \mathbf{L} + \boldsymbol{\Delta})^{-1} - (\mathbf{I} + \mathbf{L})^{-1}) \|_p \, (\|\mathbf{x}\|_p + \|\mathbf{y}\|_p) \end{aligned} \tag{8.38}$$

Given that $\mathbf{I} + \mathbf{L}$ is nonsingular we have to show that $\|(\mathbf{I} + \mathbf{L})^{-1}\boldsymbol{\Delta}\|_p < 1$ in order to be able to apply Lemma 5 with $\mathbf{A} = (\mathbf{I} + \mathbf{L})$ and further bound (8.38).

Due to doubly stochasticity [cf. Lemma 4], we have that $\|(\mathbf{I} + \mathbf{L})^{-1}\|_1 = \|(\mathbf{I} + \mathbf{L})^{-1}\|_\infty = 1$. Moreover, $\|(\mathbf{I} + \mathbf{L})^{-1}\|_2 = 1$ comes from the fact that the smallest eigenvalue of $(\mathbf{I} + \mathbf{L})$ and hence the largest eigenvalue of $(\mathbf{I} + \mathbf{L})^{-1}$ is equal to 1. Consequently, we may write

$$\|(\mathbf{I} + \mathbf{L})^{-1}\boldsymbol{\Delta}\|_p \leq \|(\mathbf{I} + \mathbf{L})^{-1}\|_p \|\boldsymbol{\Delta}\|_p < 1, \tag{8.39}$$

for $p \in \{1, 2, \infty\}$, as wanted, where the first inequality follows from submultiplicativity (8.2). Hence, applying Lemma 5 with $\mathbf{A} = (\mathbf{I} + \mathbf{L})$ yields

$$\left\| (\mathbf{I} + \mathbf{L} + \boldsymbol{\Delta})^{-1} - (\mathbf{I} + \mathbf{L})^{-1} \right\|_p \leq \frac{\|\boldsymbol{\Delta}\|_p \|(\mathbf{I} + \mathbf{L})^{-1}\|_p^2}{1 - \|(\mathbf{I} + \mathbf{L})^{-1}\boldsymbol{\Delta}\|_p} . \tag{8.40}$$

Recalling that $\|(\mathbf{I} + \mathbf{L})^{-1}\|_p = 1$ for any $p \in \{1, 2, \infty\}$ allows us to further bound (8.40) to obtain

$$\left\| (\mathbf{I} + \mathbf{L} + \boldsymbol{\Delta})^{-1} - (\mathbf{I} + \mathbf{L})^{-1} \right\|_p \leq \frac{\|\boldsymbol{\Delta}\|_p}{1 - \|\boldsymbol{\Delta}\|_p} \leq \frac{\delta \|\mathbf{L}\|_p}{1 - \delta \|\mathbf{L}\|_p} , \tag{8.41}$$

where we used that $\|\boldsymbol{\Delta}\|_p \leq \delta \|\mathbf{L}\|_p < 1$ for the last inequality.

Utilizing the Taylor series of $1/(1 - \delta\|\mathbf{L}\|_p)$ and substituting (8.41) into (8.38) combined with the

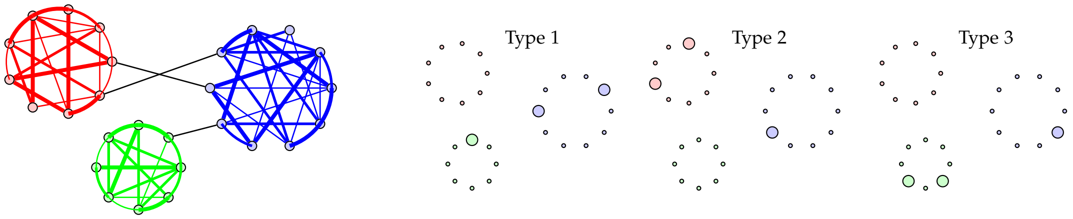


Figure 54: (a) The three-cluster network on which signals to be classified are defined. The width of the links is proportional to the weights of the corresponding edges. (b) Sample signals for the three types considered. Type 1 signals have stronger presence in the blue cluster, type 2 in the red, and type 3 in the green cluster.

fact that $\|s\|_p \leq \gamma$ and $\|r\|_p \leq \gamma$ we have that

$$d_{\text{diff}}^{\mathbf{L}'}(\mathbf{x},\mathbf{y}) - d_{\text{diff}}^{\mathbf{L}}(\mathbf{x},\mathbf{y}) \leq \sum_{n=1}^{\infty} 2\gamma(\delta\|\mathbf{L}\|_p)^n = 2\gamma\|\mathbf{L}\|_p\delta + o(\delta). \tag{8.42}$$

In a similar manner but starting from the definition of $d_{\text{diff}}^{\mathbf{L}}(\mathbf{x},\mathbf{y})$, it can be shown that

$$d_{\text{diff}}^{\mathbf{L}}(\mathbf{x},\mathbf{y}) - d_{\text{diff}}^{\mathbf{L}'}(\mathbf{x},\mathbf{y}) \leq 2\gamma\|\mathbf{L}\|_p\delta + o(\delta). \tag{8.43}$$

Finally, by combining (8.42) and (8.43), we obtain (8.32) and the proof concludes. ■

In contrast to Proposition 15, the bound in (8.32) contains higher order terms that depend on the magnitude of the perturbation. Hence, since the other terms of the bound in (8.32) tend to zero super linearly, we may divide (8.32) by $\delta\|\mathbf{L}\|_p$ and compute the limit as the perturbation vanishes

$$\lim_{\delta\to 0} \frac{\left| d_{\text{diff}}^{\mathbf{L}'}(\mathbf{x},\mathbf{y}) - d_{\text{diff}}^{\mathbf{L}}(\mathbf{x},\mathbf{y}) \right|}{\delta\|\mathbf{L}\|_p} \leq 2\gamma, \tag{8.44}$$

which implies that for small perturbations the difference in diffusion distances grows linearly.

When constructing the underlying graph to compare signals in a real-world application, noisy information can be introduced. This means that the similarity weight between two nodes in the underlying graph contains inherent error. Propositions 15 and 16 show that the superposition and diffusion distances are impervious to these minor perturbations.

In order to illustrate the stability results presented, consider again the underlying network in Figure 51. We perturb this network by multiplying every edge weight – originally equal to 1 – by a random number uniformly picked from $[0.95, 1.05]$ and then compute the diffusion and superposition distances between vectors r and g with the perturbed graph as underlying network. For these illustrations we pick the input norm to be ℓ_2 and observe that $\gamma = 1$ given the definitions of r and g. In Figure 53 we plot histograms of the absolute value of the difference in the distances when using the original and the perturbed graphs as underlying networks normalized by the norm of the perturbation for 1000 repetitions of the experiment. From (8.21) we know that this value should be less than 2 for the superposition distance and from (8.44) we know this should also be the case for the diffusion distance for vanishing perturbations. Indeed, as can be seen from Figure 53, all perturbations are below the threshold of 2 by a considerable margin. This stability property is essential for the practical utility of the diffusion and superposition distances as seen in the next section.

Remark 13 In Propositions 15 and 16 we focus our analysis on the input norms $\| \cdot \|_p$ for $p \in \{1, 2, \infty\}$ because these norms lead to the simple bounds in (8.21) and (8.32). The simplicity of these bounds is derived from the fact that $\|e^{-\mathbf{L}t}\|_p \leq 1$ and $\|(\mathbf{I} + \mathbf{L})^{-1}\|_p \leq 1$ for the values of p previously mentioned. For other matrix norms satisfying (8.2) and (8.3), including all induced matrix norms, the equivalence of norms guarantees that bounds analogous to those in (8.21) and (8.32) must exist with more involved constant terms.

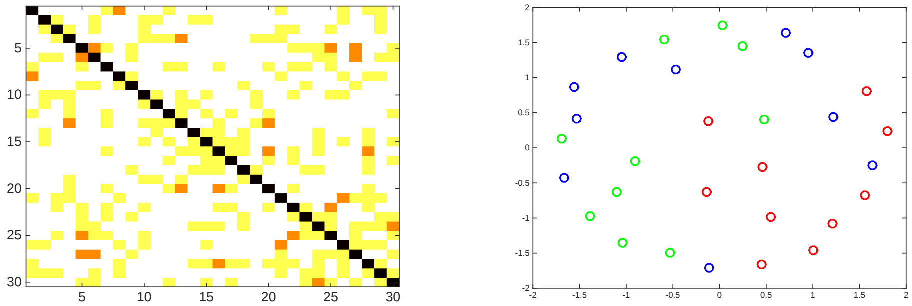
8.5. Applications

We illustrate the advantages of the superposition and diffusion distances developed in Sections 8.2 and 8.3 respectively through numerical experiments in both synthetic (Section 8.5.1) and real-world data (Sections 8.5.2 and 8.5.3).

8.5.1. Classification of Synthetic Signals on Networks

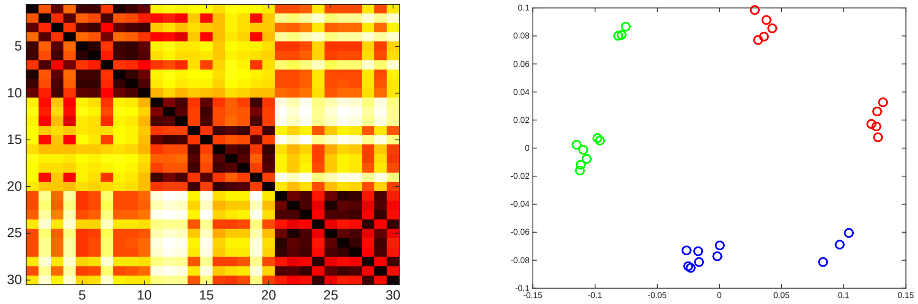
The diffusion and superposition distances lead to better classification of signals on networks compared to traditional vector distances such as the Euclidean ℓ_2 metric. Consider the network presented in Figure 54 (left) containing three clusters – blue, red, and green – where nodes within each cluster are highly connected and there exist few connections between nodes in different clusters. This network was generated randomly, where an undirected edge between a pair of nodes in the same cluster is formed with probability 0.4 and its weight is picked uniformly between 1 and 3. In addition, three edges were added with weight 1 between random pairs of nodes in different clusters. We consider three types of signals on this network. The strength of all signals is equal to 1 on three nodes in the network and 0 on the remaining ones. Among the three nodes with value 1 for the first type of signals, two of them are randomly selected from the blue cluster and the remaining one is randomly chosen from the other clusters. Similarly, for the second type of signals, exactly two out of the three nodes with positive value belong to the red cluster and the remaining one is chosen randomly between the blue and green clusters. Finally, the third type of signal has two positive values on the green cluster and the third value randomly chosen from the rest of the network. Sample signals for each type are illustrated in Figure 54 (right) where positive signal values are denoted by larger nodes.

We generate ten signals of each type and measure the distance between them with the superposition, diffusion, and ℓ_2 metrics. For the superposition and diffusion metrics we use ℓ_2 as input norm and $\alpha = 1$. The use of each metric generates a different metric space with the thirty signals as the common underlying set of points. In order to illustrate these higher dimensional spaces, in Figure 55 (left) we present heat maps of the distance functions, where darker colors represent



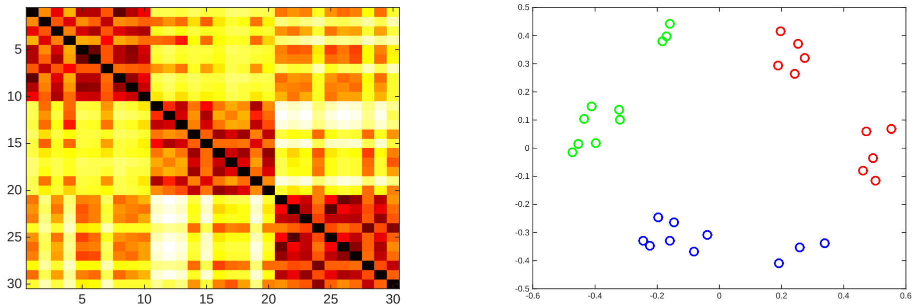
ℓ_2 heat map

MDS for ℓ_2



Diffusion heat map

MDS for diffusion



Superposition heat map

MDS for superposition

Figure 55: Heat maps (left) and 2D multi dimensional scaling (MDS) [146] representations (right) for the metric spaces generated by the ℓ_2 (top), diffusion (middle) and superposition (bottom) distances. The diffusion and superposition metrics perfectly classify the signals into the three types while ℓ_2 does not reveal any clear classification.

closer signals. It is clear that for the diffusion and superposition distances, three blocks containing ten points each appear along the diagonal in exact correspondence with the three types of signals. In contrast, the heat map corresponding to the ℓ_2 metric does not present any clear structure. To further illustrate these implications, in Figure 55 (right) we present 2D multi dimensional scaling (MDS) [146] representations of the three metric spaces. The points corresponding to type 1 signals are represented as blue circles, type 2 as red circles, and type 3 as green circles. The MDS representations for diffusion and superposition are fundamentally different from the one obtained for

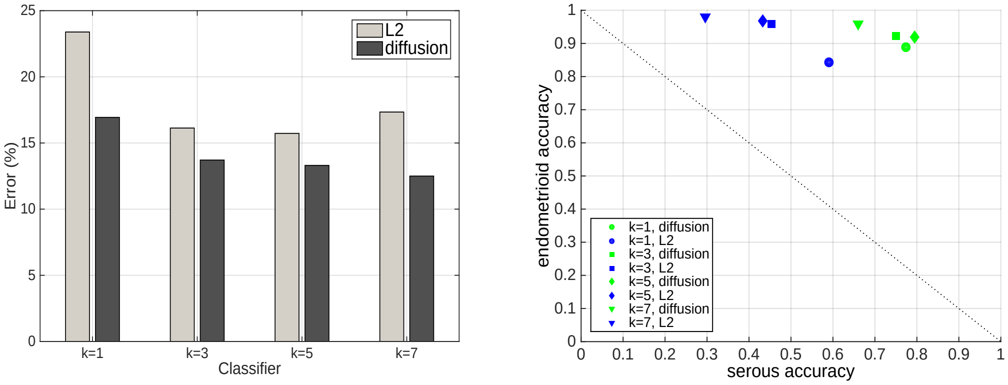


Figure 56: Histology classification of ovarian cancer patients based on k nearest neighbors with respect to the ℓ_2 and diffusion distances of their genetic profile. (a) Light bars denote the error when patients are classified using the ℓ_2 distance while the dark bars denote the error when diffusion distance is used for different k-NN classifiers. The diffusion distance reduces the classification error consistently across classifiers. (b) Accuracy of serous subtype vs. endometrioid subtype. Classifiers using diffusion (green) are closer to the top right corner, i.e. perfect classification, than those using the ℓ_2 distance (blue).

ℓ_2. For the latter, the circles of different colors are spread almost randomly on the plane, with no clear clustering structure. For diffusion and superposition, in contrast, signals of different colors are clearly separated so that any clustering method is able to recover the original signal type.

8.5.2. Ovarian Cancer Histology Classification

We demonstrate that the diffusion distance can provide a better classification of histology subtypes for ovarian cancer patients than the traditional ℓ_2 metric. To do this, we consider 240 patients diagnosed with ovarian cancer corresponding to two different histology subtypes [165]: serous and endometrioid. Our objective is to recover the histology subtypes from patients' genetic profiles.

For each patient i, her genetic profile consists of a binary vector $\mathbf{z}_i \in \{0,1\}^{2458}$ where, for each of the 2458 genes studied, $\mathbf{z}_i$ contains an 1 in position k if patient i presents a mutation in gene k and 0 otherwise. One way of building a metric in the space of 240 patients is by quantifying the distance between patients i and j as the ℓ_2 distance between their genetic profiles,

$$d_{\ell_2}(i,j) = \|\mathbf{z}_i - \mathbf{z}_j\|_2. \tag{8.45}$$

In this approach, every gene is considered orthogonal to each other and compared separately across patients. An alternative approach is to take into account the relational information across genes when comparing patients. In order to do so, we apply the diffusion distance on an underlying gene-to-gene network built based on publicly available data [166]. In order to build this network, we first extract the pairwise gene-gene interactions from [166] using the *NCI_Nature* database. After normalization, every edge weight is contained between 0 and 1, which we interpret as a probability of interaction between genes. We assign to each path the probability obtained by multiplying the probabilities in the edges that form the path. For every pair of genes in the network, we compute a similarity value between them corresponding to the maximum probability achievable by a path that links both genes. Finally, we apply normalization and thresholding operations to obtain the gene-to-gene network that we use in our experiments. Observe that the gene-to-gene network contains accepted relations between genes in humans in general and is not patient dependent, hence, it defines a common underlying network for all subjects being compared. Thus, denoting as $\mathbf{L}$ the Laplacian of the gene-to-gene network and using the ℓ_2 as input norm we compute the diffusion distances between patients i and j as [cf. (8.13)]

$$d_{\text{diff}}^{\mathbf{L}}(i,j) = \|(\mathbf{I} + \alpha\mathbf{L})^{-1}(\mathbf{z}_i - \mathbf{z}_j)\|_2, \tag{8.46}$$

where α was set to 15, however, results are robust to this particular choice. Given that in Section 8.5.1 we obtained similar performance between the diffusion and superposition distances, combined with the fact that the latter is computationally expensive, we do not implement the superposition distance in this data set.

In order to evaluate the classification power of both approaches – ℓ_2 and diffusion distance – we perform 240-fold cross validation for a k nearest neighbors (k-NN) classifier. More precisely, for a particular patient, we look at the k nearest patients as given by the metric being evaluated and assign to this patient the most common cancer histology among the k nearest patients. We then compare the assigned histology with her real cancer histology and evaluate the accuracy of the classifier. Finally, we repeat this process for the 240 women considered and obtain a global classification accuracy of both approaches.

In Figure 56 (left) we show the reduction in histology classification error when using the diffusion distance (8.46) compared to using the ℓ_2 distance (8.45) when comparing genetic profiles. The four groups of bars correspond to classifiers built using different numbers of neighbors $k \in \{1, 3, 5, 7\}$. Notice that the reduction in error is consistent across all classifiers analyzed with an average improvement of over 21% in the error rates, unveiling the value of incorporating the network information in the classification process.

To further analyze the obtained results, in Figure 56 (right) we present the accuracy obtained for the serous subtype versus the accuracy obtained for the endometrioid subtype for different classifiers based on the diffusion (green) and ℓ_2 (blue) distances. Points on the top right corner of the plot are ideal, obtaining perfect classification for both subtypes. When using diffusion, accuracies shift towards the ideal position since the accuracies for the serous subtypes increase by 20% to 40% whereas the accuracies for endometrioid subtypes decrease by less than 5%. Furthermore, among the 240 patients analyzed, there are 196 of them with endometrioid subtype and only 44 with serous subtype. Hence, a nearest neighbor classifier based on an uninformative distance would tend to have a high classification accuracy for the former but a low one for the latter. This is the case for the ℓ_2 metric. The diffusion distance, in contrast, by exploiting the gene-to-gene interaction can overcome this limitation.

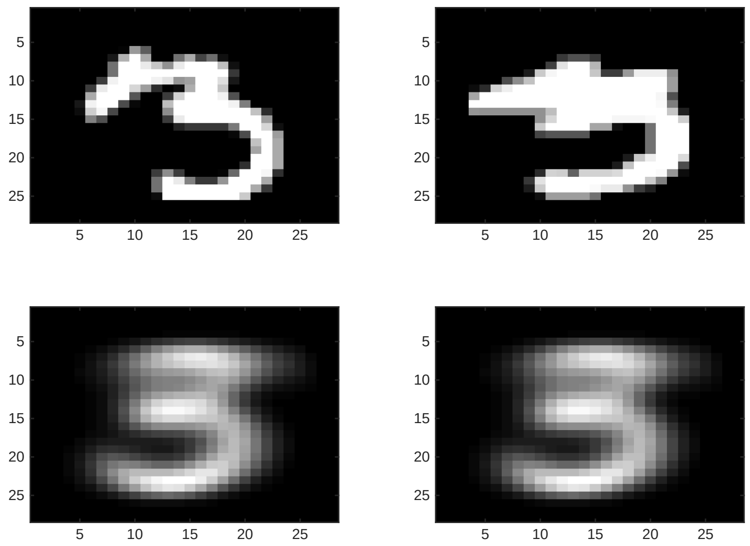


Figure 57: Two instances of handwritten threes (top) which are interpreted as fives by the classical *k*-NN approach and their corresponding diffused image (bottom). Diffusion averages out irregularities, achieving higher classification accuracy.

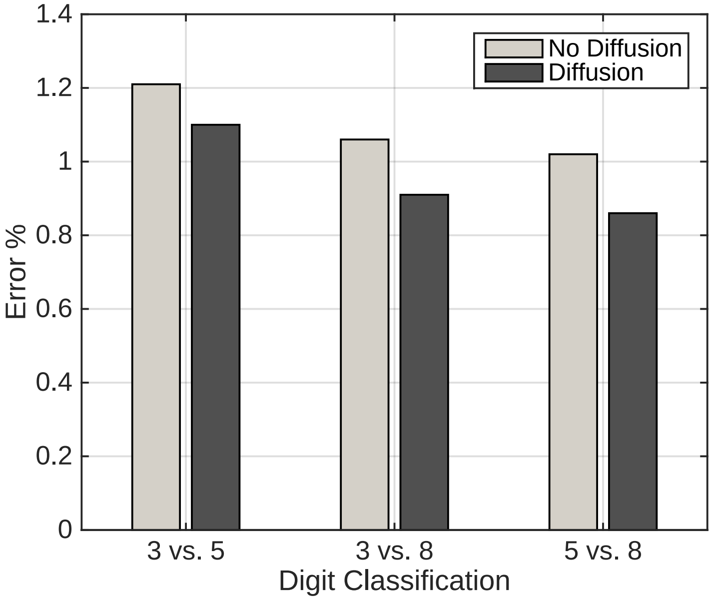


Figure 58: Error rates for three binary classification problems of written digits given by the traditional and diffused *k*-NN approaches. The error is reduced by diffusion in the three cases.

Diffusion distance can be instrumental in the classification of digits via semi-supervised learning. To illustrate this, consider the well-known MNIST handwritten digit database [167]. Each observation consists of a square gray-scaled image of a handwritten digit with 28×28 pixels. Consequently, we can think of each observation as a vector $\mathbf{x} \in \mathbb{R}^{784}$ where the value of each component corresponds to the intensity of the associated pixel. A subset of these images – the training set – are labeled, i.e. we know the digit that the image represents. The rest of the images – the testing set – are unlabeled and our objective is to correctly identify the digits they represent.

K nearest neighbors is a simple conventional approach used to solve this problem. In order to implement it, we first compute the ℓ_2 pairwise distance between all the vectors x representing the images. Then, to obtain the estimated label of an image in the testing set, we look at the labels of the k closest images among those in the training set and assign the most popular label to the image in the testing set. We repeat this procedure for each image in the testing set until all of them have an assigned label.

An alternative k-NN approach can be designed using diffusion by defining a network whose nodes are the handwritten digits. To do this, we draw an edge – with weight 1 – between two digits if the ℓ_2 pairwise distance between the corresponding vectors is less than a threshold τ. We can interpret a digit in this network as a signal on top of the network with value 1 at the node that corresponds to that digit and 0 everywhere else. Aggregating the data as in $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n]$ where n is the total number of observations (either labeled or unlabeled) as well as the number of vertices in this network such that each column in $\mathbf{X}$ corresponds to the pixels of one digit, the i-th original handwritten digit corresponding to node i in the network can be thought as the weighted average $\mathbf{X}\mathbf{e}_i$ of all the digits in the network where the weights are given by the i-th canonical vectors $\mathbf{e}_i$, i.e., all entries of $\mathbf{e}_i$ are zero except the ith one, which is 1. The diffused representation of the i-th handwritten digit is computed as a weighted average of all the digits in the network where the weights are given by $(\mathbf{I} + \mathbf{L})^{-1}\mathbf{e}_i$ [cf. (8.14)]. Using such a network, the *diffusion* distance between two images of handwritten digits, say the image i and j can be computed as

$$d_{\text{diff}}^{\mathbf{L}}(i, j) = \|\mathbf{X}(\mathbf{I} + \alpha\mathbf{L})^{-1}(\mathbf{e}_i - \mathbf{e}_j)\|_2, \tag{8.47}$$

We can then train a k-NN classifier based on the diffusion distance and compare the results with the conventional k-NN based on the ℓ_2 distance without diffusion.

In Figure 58 we present the attribution error comparison between both approaches when performing a binary attribution task between hard-to-distinguish digits: 3 vs. 5, 3 vs. 8, and 5 vs. 8. For each of these cases, we use the entire MNIST training set and testing set with $k \in \{3, 5, 7\}$. It is immediate to see that the diffusion approach outperforms the traditional k-NN in the three tasks. To see why this is the case, in Figure 57 (top) we present two handwritten images that correspond to 3 but are misclassified as 5 by the traditional k-NN method. In Figure 57 (bottom) we present their diffused representations. It is clear that diffusion averages out irregularities found in particular handwritten digits and drives them towards a canonical representation of the number 3.

If we replicate the comparison for a ten class classification problem, i.e. for all digits between 0 and 9, diffusion still improves the accuracy by reducing the error rates from 4.43% to 4.21% (training set of 8600 digits, testing set of 1400 digits and $k = 3$). The results using $k \in \{5, 7\}$ are very similar where we see the outperformance of diffusion distances. Moreover, further accuracy improvements can be obtained by combining the traditional and the diffused k-NN methods by choosing the most popular label among the k nearest neighbors in the traditional approach and the $k + 1$ nearest neighbors in the diffused approach. The error rate is further reduced to 3.93%. We pick k neighbors from one approach and $k + 1$ from the other to obtain an odd total number of neighbors, reducing the possibility of ties.

Notice that this application of the diffusion distance is fundamentally different from the one presented in Section 8.5.2. In the ovarian cancer case, the nodes in the network represented genes and each signal on the network represented a patient. On the contrary, in the current case, both the nodes in the network and the signals compared represent handwritten digits. This approach can be used in general for label propagation problems in graphs.

Chapter 9

Graph Filter and Motor Learning Task

We begin the chapter with the introduction of basic notions of graphs and graph signals. We start by describing two different experiments involving the learning of different visual-motor tasks by different sets of participants (Section 9.1). We visualize the decomposed graph frequencies relating to the functional brain network (Section 9.2). We find that high graph frequencies of functional networks concentrate on visual and sensorimotor modules of the brain – the two brain areas well-known to be associated with motor learning [168, 169]. This motivates us to consider graph frequencies other than low frequency components, whereas the PCA-oriented approach has been focusing on low frequencies. We also describe the construction of a simple model to establish artificial networks with a few network descriptive parameters (Section 9.2.1). We observe that the model is able to mimic the properties of actual functional brain networks and we use them to analyze spectral properties of the brain networks (Section 9.2.2). The paper then utilizes graph frequency decomposition to visualize and investigate brain activities with different levels of spatial variation (Section 9.3). It is noticed that the decomposed signals associated to different graph frequencies exhibit different levels of temporal variation throughout learning (Section 9.3.1). Finally, we also define learning capabilities of subjects, and examine the importance of brain frequencies at different task familiarity by evaluating their respective correlation with learning performance at different task familiarities (Section 9.4). We find as learning progresses, we favor different levels of graph frequency components.

9.1. Brain Signals during Learning

We considered two experiments in which subjects learned a simple motor task [170, 171, 172]. In the experiments, fourty-seven right-handed participants (29 female, 18 male; mean age 24.13 years) volunteered with informed consent in accordance with the University of California, Santa Barbara

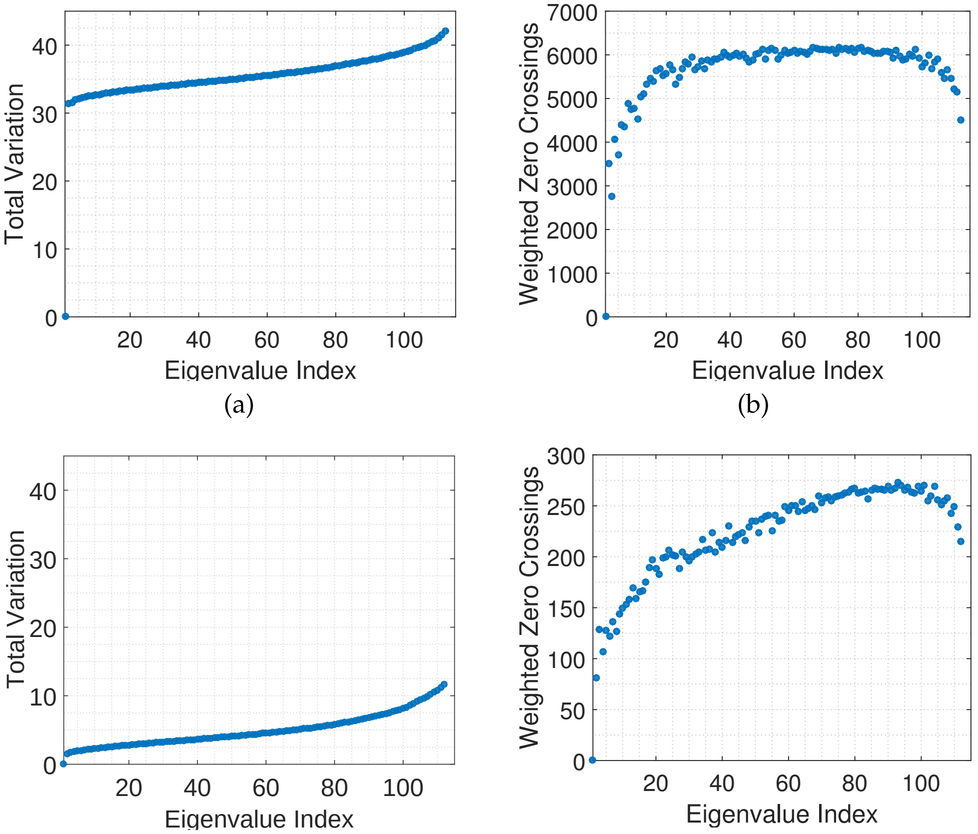
	Session 1	Session 2	Session 3	Session 4
MIN Sequences	50	110	170	230
MOD Sequences	50	200	350	500
EXT Sequences	50	740	1430	2120

Figure 59: Relationship between training duration, intensity, and depth for the first experimental framework. The values in the table denote the number of trials (i.e., "depth") of each sequence type (i.e., "intensity") completed after each scanning session (i.e., "duration") averaged over the 20 participants.

Internal Review Board. After exclusions for task accuracy, incomplete scans, and abnormal MRI, 38 participants were retained for subsequent analysis.

Twenty individuals participated in the first experimental framework. The experiment lasted 6 weeks, in which there were 4 scanning sessions, roughly at the start of the experiment, at the end of the 2nd week, at the end of the 4th week, and at the end of the experiment, respectively. During each scanning session, individuals performed a discrete sequence-production task in which they responded to sequentially presented stimuli with their dominant hand on a custom response box. Sequences were presented using a horizontal array of 5 square stimuli with the responses mapped from left to right such that the thumb corresponded to the leftmost stimulus. The next square in the sequence was highlighted immediately following each correct key press; the sequence was paused awaiting the depression of the appropriate key if an incorrect key was pressed. Each participant completed 6 different 10-element sequences. Each sequence consists of two squares per key. Participants performed the same sequences at home between each two adjacent scanning sessions, however, with different levels of exposure for different sequence types. Therefore, the number of trials completed by the participants after the end of each scanning session depends on the sequence type. There are 3 different sequence types (MIN, MOD, EXT) with 2 sequences per type. The number of trials of each sequence type completed after each scanning session averaged over the 20 participants is summarized in Figure 59. During scanning sessions, each scan epoch involved 60 trials, 20 trials for each sequence type. Each scanning session contained a total of 300 trials (5 scan epochs) and a variable number of brain scans depending on how quickly the task was performed by the specific individual.

Eighteen subjects participated in the second experimental framework. The experiment had 3 scanning sessions spanning the three days. Each scanning session lasted roughly 2 hours and no training was performed at home between adjacent scanning sessions. Subjects responded to a visually cued sequence by generating responses using the four fingers of their nondominant hand on a custom response box. Visual cues were presented as a series of musical notes on a pseudo-musical staff with four lines such that the top line of the staff mapped to the leftmost key pressed with the pinkie finger. Each 12-note sequence randomly ordered contained three notes per line. Each training epoch involved 40 trials and lasted a total of 245 repetition times (TRs), with a TR of 2,000 ms. Each training session contained 6 scan epochs (240 trials) and lasted a total of 2,070 scan TRs.



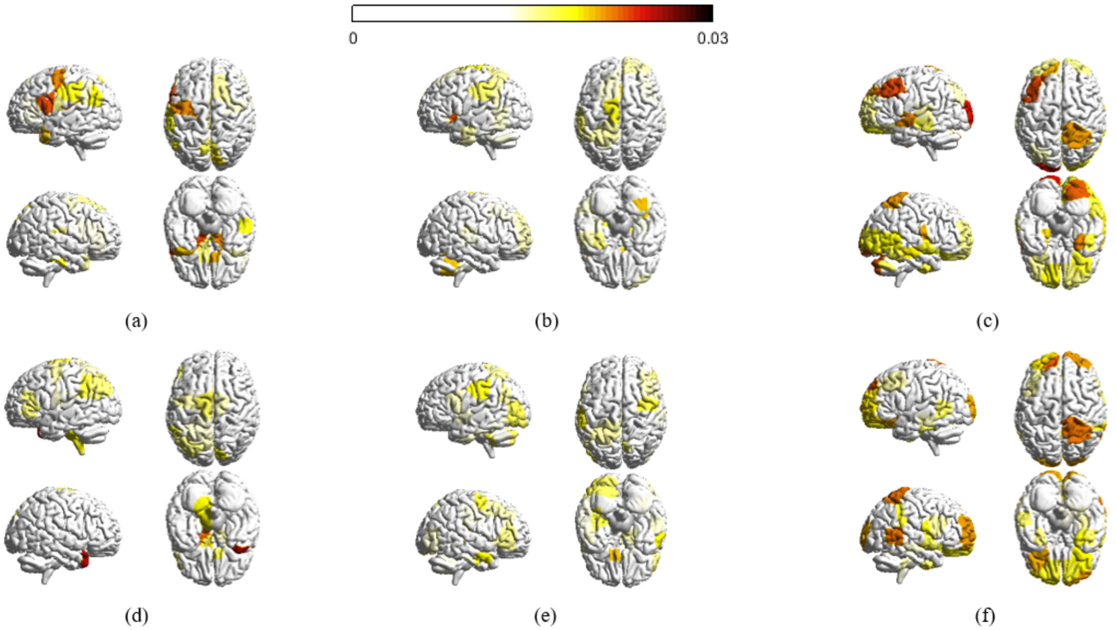
(a) (b)

(c) (d)

Figure 60: (a) Total variation $\mathrm{TV}(\mathbf{v}_k)$ and (b) weighted zero crossings $\mathrm{ZC}(\mathbf{v}_k)$ of the graph Laplacian eigenvectors for the brain networks averaged across participants in the 6 week training experiment. (c) and (d) present the values for the 3 day experiment. In both cases, the Laplacian eigenvectors associated with larger indexes vary more on the network and cross zero relatively more often, confirming the interpretation of the Laplacian eigenvalues as notions of frequencies. Besides, note that total variation increases relatively linearly with indexes.

In both experiments participants were instructed to respond promptly and accurately. Repetitions (e.g., “11”) and regularities such as trills (e.g., “121”) and runs (e.g., “123”) were excluded in all sequences. The order and number of sequence trials were identical for all participants. Participants completed the tasks inside the MRI scanner for scanning sessions.

Reordering with fMRI was conducted using a 3.0 T Siemens Trio with a 12-channel phased-array head coil. For each functional run, a single-shot echo planar imaging sequence that is sensitive to blood oxygen level dependent (BOLD) contrast was utilized to obtain 37 (the first experiment) or 33 (the second experiment) slices (3mm thickness) per repetition time (TR), an echo time of 30 ms, a flip angle of 90°, a field of view of 192 mm, and a 64 × 64 acquisition matrix. Image preprocessing was performed using the Oxford Center for Functional Magnetic Resonance Imaging of the Brain (FMRIB) Software Library (FSL), and motion correction was performed using FMRIB’s linear



(a) (b) (c)

(d) (e) (f)

Figure 61: Absolute magnitude at each of the n cortical structures averaged across participants in the 6 week experiment and averaged across all frequency components in (a) the set of low graph frequencies $\{\mathbf{v}_k\}_{k=0}^{K_\mathrm{L}-1}$, (b) the set of middle graph frequencies $\{\mathbf{v}_k\}_{k=K_\mathrm{L}}^{K_\mathrm{L}+K_\mathrm{M}-1}$, and (c) the set of high graph frequencies $\{\mathbf{v}_k\}_{k=K_\mathrm{L}+K_\mathrm{M}}^{n-1}$. (d)-(f) presents the average absolute magnitudes for the 3 day experiment. Only brain regions with absolute magnitudes higher than a fixed threshold (0.015) are colored. The magnitudes at different brain regions across the datasets are significantly similar in the low and high graph frequencies (correlation coefficients 0.5818 and 0.6616, respectively). The brain regions with high magnitude values significantly overlap with the visual and sensorimotor modules, in which more than 60% of values greater than the threshold belong to the visual and sensorimotor modules.

image registration tool. The whole brain is parcellated into a set of $n = 112$ regions of interest that correspond to the 112 cortical and subcortical structures anatomically identified in FSL's Harvard-Oxford atlas. The choice of parcellation scheme is the topic of several studies in resting-state [173], and task-based [174] network architecture. The question of the most appropriate delineation of the brain into nodes of a network is open and is guided by the particular question one wants to ask. We use Harvard-Oxford atlas here because it is consistent with previous studies of task-based functional connectivity during learning [171, 170]. The threshold in probability cutoff settings of Harvard Oxford atlas parcellation is 0 so that no voxels were excluded.

For each individual fMRI dataset, we estimate regional mean BOLD time series by averaging voxel time series in each of the n regions. We evaluate the magnitude squared spectral coherence [175] between the activity of all possible pairs of regions to construct $n \times n$ functional connectivity matrices $\mathbf{A}$. Besides, for each pair of brain regions i and j, we use t-statistical testing to evaluate the probability $p_{i,j}$ of observing the measurements by random chance, when the actual data are uncorrelated [176]. In the 3 day dataset, the value of all elements with no statistical significance ($p_{i,j} > 0.05$) [177] are set to zero; the values remain unchanged otherwise. In the 3 day experiment, a single brain network is constructed for each participant. Thresholding is applied because the networks are for the entire span of the experiment and many entries in $\mathbf{A}$ would be close to zero without threshold correction. In the 6 week experiment, due to the long duration of the experiment, we build a different brain network per scanning session, per sequence type for each subject. Because each network describes the functional connectivity for one training session given a subject, not many entries will be removed even in the presence of threshold correction; consequently, no thresholding is applied for the 6 week dataset. We normalize the regional mean BOLD observations $\hat{\mathbf{x}}(t)$ at any sample time t and consider $\mathbf{x}(t) = \hat{\mathbf{x}}(t)/\|\hat{\mathbf{x}}(t)\|_2$ such that the total energy of activities at all structures is consistent at different t to avoid extreme spikes due to head motion or drift artifacts in fMRI.

9.2. Brain Network Frequencies

In this section, we analyze the graph spectrum brain networks of the dataset considered. For the brain network $\mathbf{A}$ of each subject, we construct its Laplacian $\mathbf{L} = \mathbf{D} - \mathbf{A}$, and evaluate the total variation $\mathrm{TV}(\mathbf{v}_k)$ defined as

$$\mathrm{TV}(\mathbf{v}_k) = \mathbf{v}_k^H \mathbf{L} \mathbf{v}_k = \lambda_k. \tag{9.1}$$

for each eigenvector $\mathbf{v}_k$. Figure 60 (a) and (c) plot the total variation of all graph eigenvectors averaged across participants of the 6 week training experiment and 3 day experiment, respectively. In both experiments, the Laplacian eigenvectors associated with larger indexes fluctuate more on the network. Another observation is that with respect to graph frequency indices $0 < k < 100$, the total variation increases almost linearly.

Besides total variation, the number of zero crossings is used as a measure of the smoothness of signals with respect to an underlying network [59]. Since brain networks are weighted, we adapt a slightly modified version – weighted zero crossings – to investigate the given graph eigenvector $\mathbf{v}_k$

$$\mathrm{ZC}(\mathbf{v}_k) = \frac{1}{2} \sum_{i \neq j} A_{ij} \mathbb{I} \left\{ v_k(i) v_k(j) < 0 \right\}. \tag{9.2}$$

In words, weighted zero crossings evaluate the weighted sum of the set of edges connecting a vertex with a positive signal to a vertex with a negative signal. Figure 60 (b) and (d) demonstrate the weighted zero crossings of all graph eigenvectors averaged across subjects of the 6 week and 3 day experiments, respectively. The weighted zero crossings increase almost proportionally with graph frequency index k until they eventually level off for $0 \leq k \leq 100$. For k greater than 100, though, eigenvectors associated with higher graph frequencies exhibit lower weighted zero crossings.

It would be interesting to examine where the associated eigenvectors lie anatomically, and the relative strength of their values. To facilitate the presentation, we consider three sets of eigenvectors, $\{\mathbf{v}_k\}_{k=0}^{K_\mathrm{L}-1}, \{\mathbf{v}_k\}_{k=K_\mathrm{L}}^{K_\mathrm{L}+K_\mathrm{M}-1}$ and $\{\mathbf{v}_k\}_{k=K_\mathrm{L}+K_\mathrm{M}}^{n-1}$, and compute the absolute magnitude at each of the n cortical and subcortical regions averaged across participants and across all graph frequencies belonging to each of the three sets. Figure 61 presents the average magnitudes for the two experimental frameworks considered in the paper using BrainNet [178], where brain regions with absolute magnitudes lower than a fixed threshold are not colored. Throughout the paper, the parameter K_L is set as 40 and K_M is set as 32. This combination yields three roughly equally-sized components with one piece corresponding to the 40 lowest graph frequencies and another piece corresponding to the 40 highest frequencies. The results presented in the paper are robust with the choice of parameters: we examined the results for K_L and K_M in the range of 32 to 42, inclusive, and found similar observations as the ones presented. To demonstrate that, Figure 62 presents the range of correlation coefficients calculated between the frequency ranges selected for this paper and all frequency ranges for K_L and K_M between 32 and 42, inclusive, giving 120 correlation values for each box plot. The correlation coefficients reported are a quantification of similarity measure when we examine the similarity between two vectors, given as the absolute magnitudes passing the given threshold across all brain regions. An investigation of cosine similarity gives high similarities as well. We have also conducted robustness testing in our analysis of learning rate (Section 9.4) and have plotted our results obtained from using all 121 possible frequency ranges (Figure 70) and have quantified the robustness of parameters in Figure 69.

9.2.1. *Artificial Functional Brain Networks*

An approach to analyze the complex networks is to define a model to generate artificial networks [179, 180]. The main motivation of an artificial network model is to use them to analyze complex brain networks. Examples of such models include the Barabási-Albert model for scale free networks [181] and recent developments and insights on weighted network models [180]. Here we present a framework to construct artificial networks that can be used to mimic the functional brain networks with only a few parameter inputs. The model is related to weighted block stochastic model [182], but involves more aspects like individual variance and analyzes links independent of their connectivity strength to other brain regions. The output of the method would be a symmetric network with edge weights between 0 and 1 without self-loops.

To begin, suppose the desired network has two clusters of nodes $\mathcal{V}_1$ and $\mathcal{V}_o$. The algorithm requires the average edge weight μ_1 for connections between nodes of the first cluster $\mathcal{V}_1$, average edge weight μ_o for links between nodes of the other cluster $\mathcal{V}_o$, and average edge weight μ_{1o} for inter-cluster connections. To reflect the fact that the edge weights on some links are independent of their joining vertices, for each edge within $\mathcal{V}_1$, with probability $p_\epsilon < 1$, its weight is randomly generated with respect to uniform distribution $\mathcal{U}[0, 1]$ between 0 and 1, and with probability $1 - p_\epsilon$, its weight is randomly generated with respect to uniform distribution $\mathcal{U}[\mu_1 - \delta, \mu_1 + \delta]$. The parameter p_ϵ determines the percentage of edges whose weights are selected irrespective of their actual locations To further simulate the observation that different participants may possess distinctive brain networks, if the edge weight is randomly generated from a uniform distribution

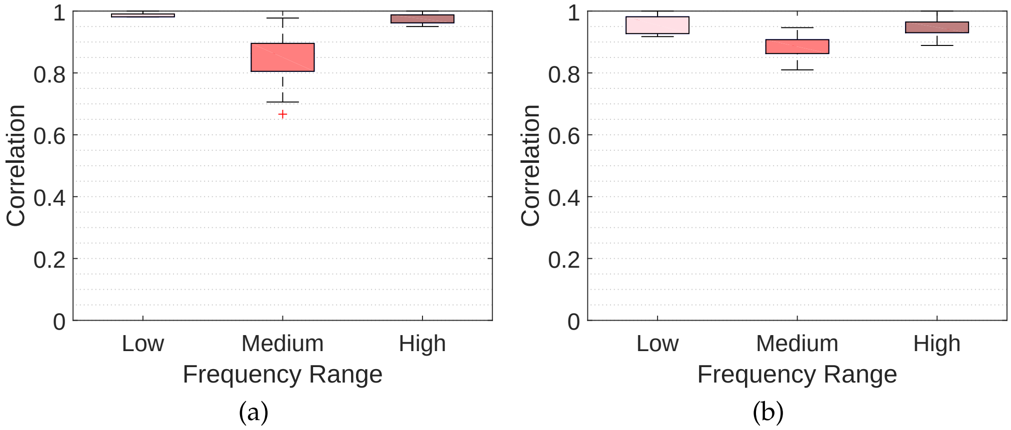


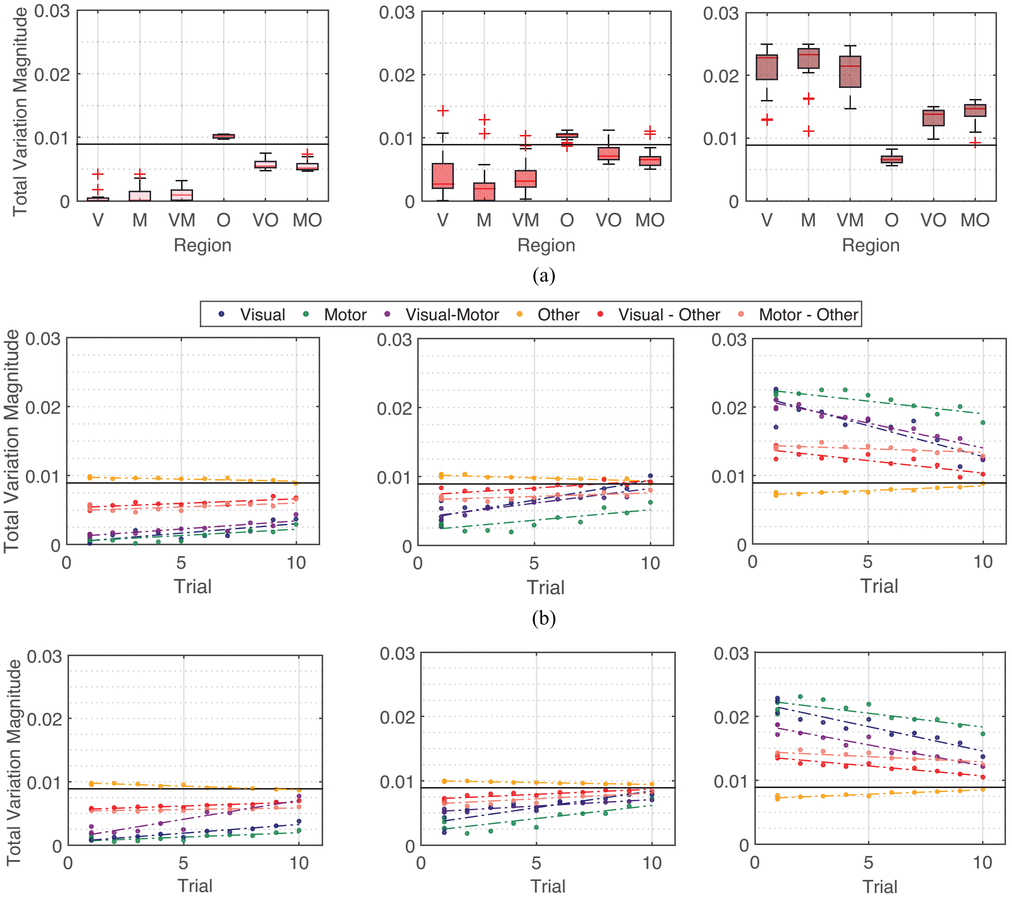
Figure 62: Box plots demonstrating the robustness of parameters chosen for different frequency ranges for the absolute magnitudes across brain regions for (a) 6 week and (b) 3 day experiments. Each box plot presents the correlation coefficients between the frequency range selected for this paper and all frequency ranges for K_L and K_M between 32 and 42, inclusive.

$\mathcal{U}[\mu_1 - \delta, \mu_1 + \delta]$, it is then perturbed by $A_u \sim \mathcal{U}[-u_\epsilon/2, u_\epsilon/2]$ where u_ϵ controls the level of perturbation. The edge weights for connections within cluster $\mathcal{U}_o$ are generated similarly: with probability $1 - p_\epsilon$, the edge weight is randomly chosen from the uniform distribution $\mathcal{U}[\mu_o - \delta, \mu_o + \delta]$ before being contaminated by $A_u \sim \mathcal{U}[-u_\epsilon/2, u_\epsilon/2]$. The edge weights for connections between clusters $\mathcal{U}_1$ and $\mathcal{U}_o$ are formed analogously using μ_{1o}. The method presented here can be easily generalized to analyze brain networks with more regions of interest, i.e. by specifying sets of regions of interest and by detailing the expected correlation values on each type of connection between different regions.

Remark 14 At one extreme we can make each node i belonging to a different set $\mathcal{V}_i = \{i\}$. Then the method requires the inputs of expected weights for all nodes, or alternatively speaking, the expected network. At the other extreme, there is only one set of nodes $\mathcal{V}$, and then the method is highly akin to a network with edge weights completely randomly generated. Any construction of interest would have some prior knowledge regarding the community structure. Therefore, the method proposed here can be used to see if the network constructed with the specific choice of community structure highly simulate the key properties of the actual network, and can be used to examine the evolution of community structure in the brain throughout the process to master a particular task.

9.2.2. Spectral Properties of Brain Networks

In this section, we analyze graph spectral properties of brain networks. Given the graph Laplacian, we examine the fluctuation of its eigenvectors on different types of connections in the brain network [169]. More specifically, given an eigenvector $\mathbf{v}_k$, its variation on the visual module is



(a)

(b)

(c)

Figure 63: Spectral property of brain networks in the 6 week experiment. (a) Left: Averaged total variation of eigenvectors $\mathbf{v}_k$ for 6 different types of connections of the brain averaged across all eigenvectors associated with low graph frequencies $\mathbf{v}_k \in \{\mathbf{v}_k\}_{k=0}^{K_\mathrm{L}-1}$, across all participants and scan sessions. Middle: Across all eigenvectors associated with mid-range graph frequencies $\mathbf{v}_k \in \{\mathbf{v}_k\}_{k=K_\mathrm{L}}^{K_\mathrm{L}+K_\mathrm{M}-1}$. Right: Across all eigenvectors with high graph frequencies $\mathbf{v}_k \in \{\mathbf{v}_k\}_{k=K_\mathrm{L}+K_\mathrm{M}}^{n-1}$. (b) Median total variations of brain networks across participants of different scanning sessions and different sequence types with respect to the level of exposure of participants to the sequence type at the scanning session. Relationship between training duration, intensity, and depth is summarized in Figure 59. Value of 1 on the x-axis in the figures refers to minimum exposure to sequences (all 3 sequence types of the first session), and value of 10 on the x-axis denotes the maximum exposure to sequences (EXT sequence types of the fourth session). An association between spectral property of brain networks and the level of exposure is clearly observed (average correlation coefficient 0.8164). (c) Median total variations evaluated upon artificial networks. Spectral properties of actual brain networks can be closely simulated using a few parameters. The main text gives all correlation values for similarity between variance among subjects and between correlations of training intensity.

defined as

$$\mathrm{TV}_{\text{visual}}(\mathbf{v}_k) = \frac{\sum_{i,j\in\mathcal{V}_\mathrm{v},i\neq j} A_{ij}\left(v_k(i)-v_k(j)\right)^2}{\sum_{i,j\in\mathcal{V}_\mathrm{v},i\neq j} A_{ij}}, \tag{9.3}$$

where $\mathcal{V}_\mathrm{v}$ denotes the set of nodes belonging to the visual module. The measure $\mathrm{TV}_\mathrm{visual}(\mathbf{v}_k)$ computes the difference for signals on the visual module for each unit of edge weight. To facilitate interpretation, we only consider three sets of eigenvectors $\{\mathbf{v}_k\}_{k=0}^{K_\mathrm{L}-1}, \{\mathbf{v}_k\}_{k=K_\mathrm{L}}^{K_\mathrm{L}+K_\mathrm{M}-1}$, and $\{\mathbf{v}_k\}_{k=K_\mathrm{L}+K_\mathrm{M}}^{n-1}$. We then compute the visual module total variation $\mathrm{TV}_\mathrm{visual}^\mathrm{L}$ averaged over eigenvectors $\{\mathbf{v}_k\}_{k=0}^{K_\mathrm{L}-1}$, and $\mathrm{TV}_\mathrm{visual}^\mathrm{M}$ as well as $\mathrm{TV}_\mathrm{visual}^\mathrm{H}$ similarly. Besides $\mathrm{TV}_\mathrm{visual}$, we also examine the level of fluctuation of eigenvectors on edges within the motor module, denoted as $\mathrm{TV}_\mathrm{motor}$, and on connections belonging to brain modules other than the visual and motor module $\mathrm{TV}_\mathrm{others}$. Further, there are links between two separate brain modules, and to assess the variation of eigenvectors on those links, we define total variations between the visual and motor modules

$$\mathrm{TV}_\text{visual-motor}(\mathbf{v}_k) = \frac{\sum_{i\in\mathcal{V}_\mathrm{v}, j\in\mathcal{V}_\mathrm{m}} A_{ij}(v_k(i) - v_k(j))^2}{\sum_{i\in\mathcal{V}_\mathrm{v}, j\in\mathcal{V}_\mathrm{m}} A_{ij}}, \tag{9.4}$$

where $\mathcal{V}_\mathrm{m}$ denotes the set of nodes belonging to the motor module. Total variations $\mathrm{TV}_\text{visual-others}$ between the visual and other modules, and total variations $\mathrm{TV}_\text{motor-others}$ between the motor and other modules are defined analogously. We chose to study visual and motor modules separately from other brain modules because of their well-known associations with motor learning [168, 169].

Figure 63 (a) presents boxplots of the variation for eigenvectors of different graph frequencies measured over different types of connections across participants, at the start of the six week training. Despite that total variation of eigenvectors should increase with their frequencies, the variation on the other module $\mathrm{TV}_\mathrm{others}^\mathrm{L}$ of eigenvectors associated with low frequencies are higher than $\mathrm{TV}_\mathrm{others}^\mathrm{H}$ (pass t-test with $p < 0.0001$). This observation is discussed in detail in Section 9.2.3.

Next we study how the graph spectral properties of brain networks evolve as participants become more familiar with the tasks. Figure 63 (b) illustrates the median of the variation for eigenvectors of different graph frequencies measured over different types of connections across subjects, at 10 different levels of exposure in the six week training. As participants become more acquainted with the assignment, their brain networks display lower variation in the visual and motor modules and higher variation in the other modules for low and middle graph frequencies, and the exact opposite is true for high graph frequencies. The association with training intensity is statistically significant (average correlation coefficient $r = 0.8164$).

9.2.3. Discussion

Firstly, we examine why we see a decrease in zero crossings of graph frequencies when k is greater than 100 in Figure 60. A detailed analysis shows this is because the functional brain networks are highly connected with nearly homogeneous degree distribution, and consequently each high graph frequency tends to have a value with high magnitude at one vertex of high degree and similar values at other nodes, resulting in a smaller global zero crossings for eigenvectors associated with very high frequencies.

Secondly, in terms of the visualization of graph frequencies in Figure 61, the most interesting find-

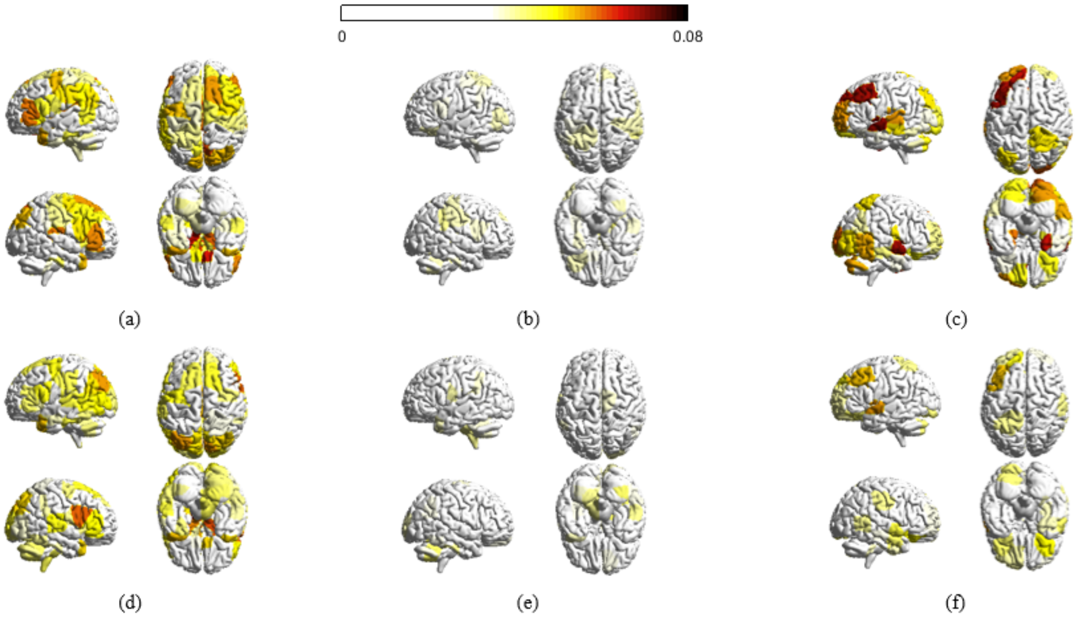


Figure 64: Distribution of decomposed signals for the 6 week experiment. (a) Absolute magnitudes for all brain regions with respect to $\mathbf{x}_L$ – brain signals varing smoothly across the network – averaged across all sample points for each individual and across all participants at the first scan session of the 6 week dataset. (b) With respect to $\mathbf{x}_M$ and (c) with respect to $\mathbf{x}_H$ – signals rapidly fluctuating across the brain. (d), (e), and (f) are averaged $\mathbf{x}_L, \mathbf{x}_M$ and $\mathbf{x}_H$ at the last scan session of the 6 week dataset, respectively. Only regions with absolute magnitudes higher than a fixed threshold are colored.

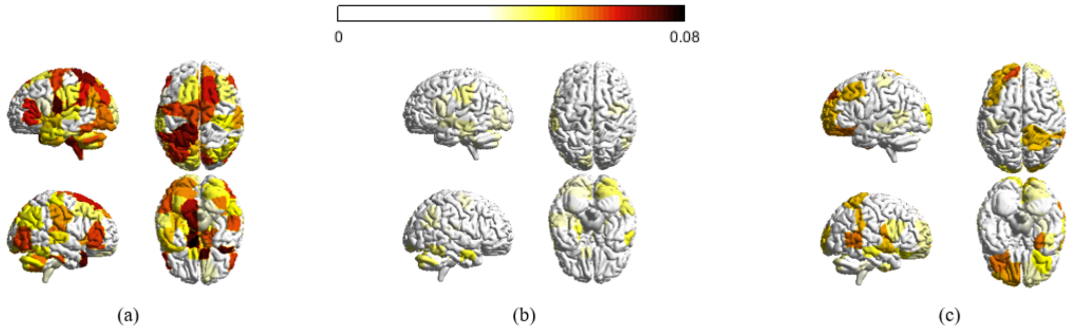


Figure 65: Distribution of decomposed signals for the 3 day experiment. (a), (b), and (c) are the absolute magnitudes for all brain regions with respect to $\mathbf{x}_L, \mathbf{x}_M$ and $\mathbf{x}_H$, averaged across all sample points for each subject and across participants in the 3 day experiment, respectively. Regions with absolute value less than a threshold are not colored.

ing relates to the eigenvectors associated with high graph frequencies. The magnitudes at different brain regions for high frequencies are significantly similar across the two datasets investigated (correlation coefficient 0.6616). There are very few noticeable brain regions in which the absolute magnitude highlighted in the first dataset is not likewise highlighted in the second. Given the different experiment setups, it would not be uncommon to observe large variations across datasets.

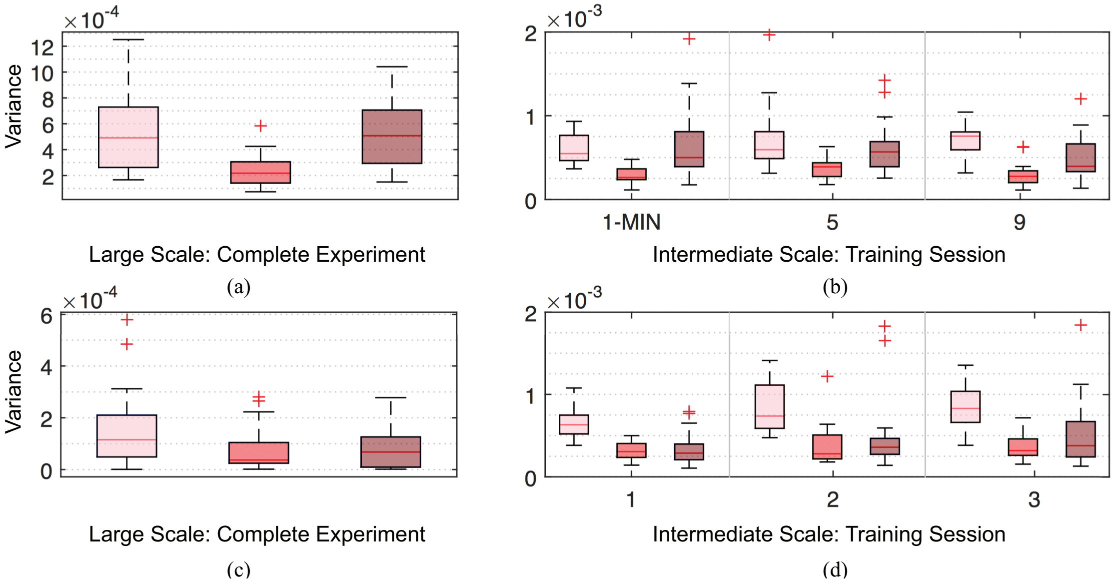


Figure 66: Temporal adaptations of spatial variations. Boxplots showing differences in temporal adaptabilities between brain activities with smooth (pink), moderate (red) and rapid (maroon) spatial variations, measured over the complete experiment for 6 week (a) and 3 day (c), and individual training sessions for 6 week (b) and 3 day (d) experiments. We measured the temporal adaptations using the variance of the averaged activities over the complete experiment or with individual training sessions. Compared to activities with moderate spatial variations, smooth (95% sessions pass t-test with $p < 0.01$) and rapid (65% sessions pass t-test with $p < 0.005$) spatial variations have significantly higher temporal adaptations.

However, the fact that we see the majority of brain regions similarly highlighted in the two experiments solidifies our understanding that eigenvector decomposition captures general signatures, as opposed to task-specific realizations. Additionally, brain regions with high magnitude values are highly alike (greater than 60% overlap) to the visual and sensorimotor cortices [183]. This is likely to be a consequence of the fact that visual and motor regions are more strongly connected with other structures, and hence an eigenvector with a high magnitude on visual or motor structures would result in high global spatial variation. The eigenvectors of low graph frequencies are more spread across the networks, resulting in low global variations. The middle graph frequencies are less interesting – the magnitudes at most regions (greater than 90%) do not pass the threshold, and little associations (correlation coefficient 0.3529) can be found between the eigenvectors of the 6 week and 3 day experiments.

Thirdly, to better interpret the meaning of variations for specific types of connections, we construct artificial networks as described in Section 9.2.1 with visual and motor modules as regions of interest, and consider other modules to be brain regions other than visual and motor modules. We observe that there are three contributing factors that cause the variation within a specific module to become higher for higher eigenvectors and to become lower for lower eigenvectors: (i) Increases in the average edge weight for connections within the module, (ii) Increments in the average edge weight for links between this module to other module, and (iii) Escalation in the average edge weight for associations within the other module. This can also be observed by analyzing closely the definition of total variation. If a module is highly connected, in order for the eigenvector associated with a low graph frequency to be smooth on the entire network, it has to be smooth on the specific module, resulting in a low value in the variation of an eigenvector associated with a low graph frequency with respect to the module of interest. Similarly, the increase in the variation of connections between two modules, e.g. between visual and other modules are resulted from: (i) The growth in the average edge weight for connections between visual and other modules, or (ii) The augmentation of average weight for links within the other module. The graph spectral properties as in Figure 63 (a) are observed because (i) visual and motor modules are themselves highly connected, and (ii) visual module is also strongly linked with motor module.

Finally, in analyzing the evolution of graph spectral properties as participants become more familiar with the tasks, following the interpretations based on artificial network analysis, this evolution in graph spectral properties of brain networks is mainly caused by the decrease in values of connections within visual and motor modules and between the visual and motor modules. An interesting observation is that the values in the variation of eigenvectors associated with high frequencies decline with respect to the visual module much faster than that of motor module, even though the visual module is more strongly connected throughout training compared to the motor module. A deep analysis using artificial networks shows that this results from the following three factors: (i) Though more strongly connected compared to the motor module, connections within the visual module weaken very quickly, (ii) The motor module is more closely connected with the other module than the link between the visual module to the other module, and (iii) Association levels within the other module stay relatively constant. Therefore, as participants become more exposed to the tasks, compared to the visual module, the motor module becomes more strongly connected. The graph spectral properties of actual brain networks and their evolution can be closely imitated using artificial networks as plotted in Figure 63 (c). The artificial network created for our analysis best imitated the real brain networks with parameters p_ϵ of 0.10, u_ϵ of 0.10, and δ of 0.01. The average edge weights μ for visual (v), motor (m), other (o), and inter-connecting regions are $\mu_v = 0.6028$, $\mu_m = 0.4902$, $\mu_o = 0.3098$, $\mu_{vm} = 0.3985$, $\mu_{vo} = 0.3181$, and $\mu_{mo} = 0.3271$. The correlation coefficients of association with training intensity between real and artificial networks for low, medium, and high graph frequencies are 0.6436, 0.7187 and 0.8457, respectively. Additionally, the variation among participants in real dataset can be closely mimicked using artificial network model we proposed, with correlation coefficients 0.9338, 0.9660, and 0.9486 for low, medium, and high graph frequencies, respectively. The analysis for the three day training dataset is highly similar (correlation coefficients 0.9834, 0.9186, and 0.9674 for low, medium, and high graph frequencies, respectively) and for this reason we do not present and analyze it separately here.

9.3. Frequency Decomposition of Brain Signals

The previous sections focus on the study of brain networks and their graph spectral properties. In this section, we investigate brain signals from a GSP perspective, and analyze the brain signals by examining the decomposed graph signals $\mathbf{x}_{\mathrm{L}}, \mathbf{x}_{\mathrm{M}}$, and $\mathbf{x}_{\mathrm{H}}$ with respect to the underlying brain networks. We compute the absolute magnitude of the decomposed signal $\mathbf{x}_{\mathrm{L}}$ for each brain region averaged across all sample signals for each individual during a scan session and then averaged across all participants. Similar aggregation is applied for $\mathbf{x}_{\mathrm{M}}$ and $\mathbf{x}_{\mathrm{H}}$.

Figure 64 presents the distribution of the decomposed signals corresponding to different levels of spatial variations for the first scan session (top row) and the last scan session (bottom row) in the 6 week experiment. Figure 65 exhibits how the decomposed signals are distributed across brain regions in the 3 day experiment. Brain regions with absolute magnitudes lower than a fixed threshold are not colored.

9.3.1. Temporal Variation of Graph Frequency Components

We analyze temporal variation of decomposed signals with respect to different levels of spatial variations. To this end, we evaluate the variance of the decomposed signals over multiple temporal scales – over days and minutes – for the two experiments. We describe the method specifically for $\mathbf{x}_{\mathrm{L}}$ for simplicity and similar computations were conducted for $\mathbf{x}_{\mathrm{M}}$ and $\mathbf{x}_{\mathrm{H}}$. At the macro timescale, we average the decomposed signals $\mathbf{x}_{\mathrm{L}}$ for all sample points within each scanning session with different sequence type, and evaluate the variance of the magnitudes of the signals [60] across all the scanning sessions and sequence types. For the 6 week experiment, there are 4 scanning sessions and 3 different sequence types, so the variance is with respect to 12 points. For the 3 day experiment, there are 3 scanning sessions and only 1 sequence type, so the variance is for 3 points. As for the micro or minute-scale, we average the decomposed signals $\mathbf{x}_{\mathrm{L}}$ for all sample points within each minute, and evaluate the variance of the magnitudes of the averaged signals across all minute windows for each scanning session with different sequence types. The evaluated variance is then averaged across all participants of the experiment of interest.

Figure 66 displays the variance of the decomposed signals $\mathbf{x}_{\mathrm{L}}, \mathbf{x}_{\mathrm{M}}$ and $\mathbf{x}_{\mathrm{H}}$ at two different temporal scales of the two experiments. For the 6 week dataset, 3 session-sequence combinations, with the number proportional to the level of exposure of participants to the sequence (1-MIN refers to MIN sequence at session 1, 5 denotes MIN sequence at session 4, 9 entails EXT sequence at session 3) are selected out of the 12 combinations in total for a cleaner illustration, but all the other session-sequence combinations exhibit similar properties.

9.3.2. Discussion

A deep analysis of Figs. 64 and 65 yields many interesting aspects of graph frequency decomposition. First, for $\mathbf{x}_{\mathrm{L}}$, the magnitudes on adjacent brain regions tend to possess highly similar values, resulting in a more evenly spread brain signal distribution, where as for $\mathbf{x}_{\mathrm{H}}$, neighboring signals can exhibit highly dissimilar values; this corroborates the motivation to use graph frequency decomposition to segment brain signals into pieces corresponding to different levels of spatial fluctuations. Second, decomposed signals for a specific level of variation, notedly $\mathbf{x}_{\mathrm{H}}$, are highly similar with respect to different scan sessions in an experiment as well as with respect to the two experiments with different sets of participants. The correlation coefficient between datasets for high graph frequencies is 0.6469. Third, recall that we normalize the brain signals at every sample point for all subjects, and for this reason signals $\mathbf{x}_{\mathrm{L}}, \mathbf{x}_{\mathrm{M}}$ and $\mathbf{x}_{\mathrm{H}}$ would be similarly distributed across the brain if nothing interesting happens at the decomposition. However, in both Figs. 64 and 65, it is observed that many brain regions possess magnitudes higher than a threshold in $\mathbf{x}_{\mathrm{L}}$ ($\sim 60\%$ pass) and $\mathbf{x}_{\mathrm{H}}$ ($\sim 20\%$ pass) while not many brain regions pass the thresholding with respect to $\mathbf{x}_{\mathrm{M}}$ ($\sim 3\%$ pass). It has long been understood that the brain combines some degree of disorganized behavior with some degree of regularity and that the complexity of a system is high when order and disorder coexist [1]. $\mathbf{x}_{\mathrm{L}}$ varies smoothly across the brain network and therefore can be regarded as regularity (order), whereas $\mathbf{x}_{\mathrm{H}}$ fluctuates rapidly and consequently can be considered as randomness (disorder). This evokes the intuition that graph frequency decomposition segments a brain signal $\mathbf{x}$ into pieces $\mathbf{x}_{\mathrm{L}}$ and $\mathbf{x}_{\mathrm{H}}$, which reflect order and disorder (and are therefore more interesting), as well as the remaining $\mathbf{x}_{\mathrm{M}}$.

For the variance analysis, it is expected for the low graph frequency components (smooth spatial variation) to exhibit the smallest temporal variations, exceeded by medium and then high counterparts. Nonetheless, it is observed that brain activities with smooth spatial variations exhibit the most rapid temporal variation. Because it has been shown that temporal variation of observed brain activities is associated with better performance in tasks [60], this indicates a stronger contribution of low graph frequency components during the learning process. Furthermore, since the measurements were normalized such that the total energy of overall brain activities stayed constant at different sampling points, the rapid temporal changes of low graph frequency components should be accompanied by fast temporal variation of some other components, which are found to be high frequency components in all cases. Because these results were consistent for all of the temporal scales and datasets that we examined, and the association between temporal variability and positive performance has been established [61], we concluded that brain activities with smooth or rapid spatial variations offer greater contributions during learning. The graph frequency signatures at different stages of learning is analyzed in the next section.

9.4. Frequency Signatures of Task Familiarity

Given that the decomposed signals exhibit interesting perspectives, it is natural to probe whether the signals corresponding to different levels of spatial variations associate with learning. To this end, we first describe how learning rate is evaluated. Given a participant, for each sequence completed, we defined the movement time M as the difference between the time of the first button press and the time of the last button press during a single sequence. We then estimate the participant's learning rate by fitting an exponential function (plus a constant) using the robust

	$\|\mathbf{x}_{\mathrm{L}}\|_2$	$\|\mathbf{x}_{\mathrm{M}}\|_2$	$\|\mathbf{x}_{\mathrm{H}}\|_2$
6 week experiment (linear scale)	-0.3155	0.0897	0.4125
6 week experiment (logarithm scale)	-0.5409	0.3992	0.3565
3 day experiment	-0.9873	0.8443	0.9605

Figure 67: Pearson correlation coefficients between the number of trials (level of task familiarity) and R values, defined as correlations between learning rate parameters and the norm of the decomposed signal of interest. More obvious adaptability between decomposed signals and learning across training is observed for $\mathbf{x}_{\mathrm{L}}$ and $\mathbf{x}_{\mathrm{H}}$, with decreasing association with exposure to tasks for the former and increasing importance for the latter.

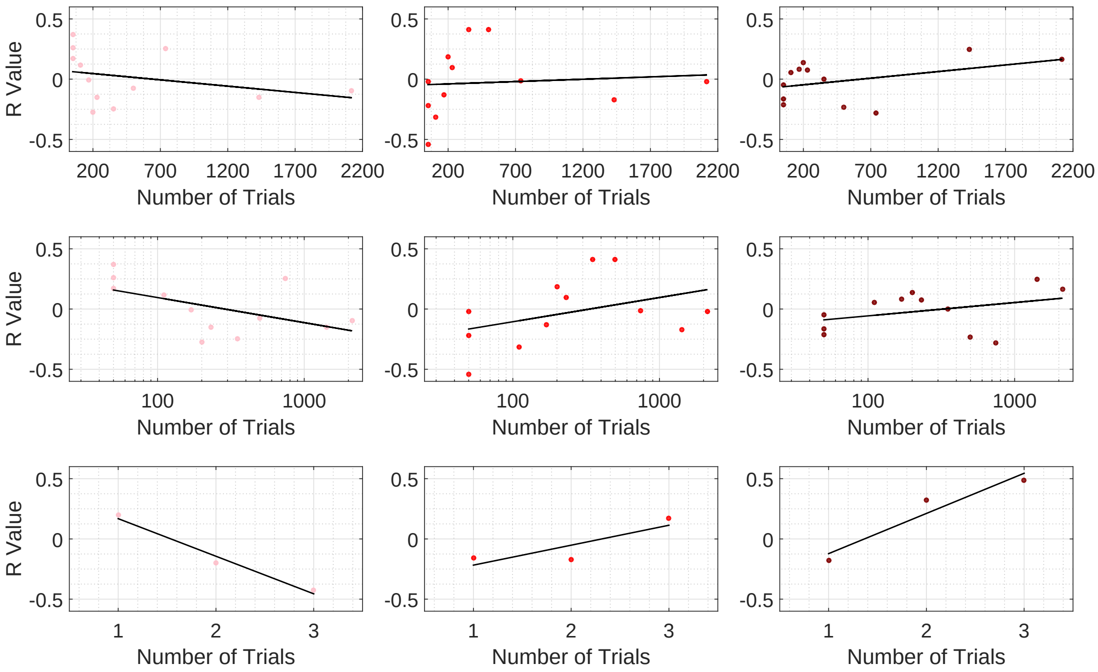


Figure 68: Scatter plots in which each point is for a specific training session (level of task familiarity), depicting the R value defined here as correlations between learning rate parameters and the norm of the decomposed signal of interest (Pink points in the Left: $\mathbf{x}_{\mathrm{L}}$, Red points in the Middle: $\mathbf{x}_{\mathrm{M}}$, and Maroon points in the Right: $\mathbf{x}_{\mathrm{H}}$). Top row: 6 week experiment with number of trials described in linear scale. Middle row: 6 week experiment withe number of trials evaluated in logarithm scale. We examine 6 week experiment by ordering the number of trials in both linear and logarithm scales to alleviate the fact that number of trials are densely distributed towards small values. Bottom row: 3 day experiment in which the number of trials is represented by the 3 scanning sessions in the experiment.

outlier correction [184] to the sequence of movement times $\mathbf{M}$

$$\mathbf{M} = c_1 e^{\mathbf{t}/\kappa} + c_2. \tag{9.5}$$

where $\mathbf{t}$ is a sequence representing the time index, κ is the exponential drop-off parameter (which we call the "learning rate parameter") used to describe the early and fast rate of improvement, and c_1 and c_2 are nonnegative constants. Their sum $c_1 + c_2$ is an estimation of the starting speed of the participant of interest prior to training, while the parameter c_2 entails the fastest speed to complete the sequence attained by that participant after extended training. A negative value of κ indicates a decrease in movement time $M(t)$, which is thought to indicate that learning is occurring [185]. We chose exponential because it is viewed as the most statistically robust choice [186]. Further, the approach that we used has the advantage of estimating the rate of learning independent of initial performance or performance ceiling.

We evaluate the learning rate for all participants at each scanning session, and then compute the correlation between the norm $\|\mathbf{x}_\mathrm{L}\|_2$ of the decomposed signal corresponding to low spatial variation and the learning rates across subjects. The correlation (R value) between the norms $\|\mathbf{x}_\mathrm{M}\|_2$ as well as $\|\mathbf{x}_\mathrm{H}\|_2$ and learning rates are also calculated. Figure 68 plots the Pearson correlation coefficients at all scanning sessions of the two experiments considered. The horizontal axis denotes the level of exposure of participants to the sequence – which day in the 3 day experiment and how many number of trials participants have completed at the end of the scanning session in the 6 week experiment. Points are densely distributed for small number of trials in the 6 week experiment, so to mitigate this effect, we also plot the points by taking the logarithm of numbers of trials completed. We emphasize that due to normalization at each sampling point, the correlation values would all be 0 if graph frequency decomposition segments brain signals into three equivalent pieces. There are scan sessions where the correlation is of particular interest, however the most noteworthy observation is the change of correlation values with the level of exposure for participants.

In general, for $\mathbf{x}_\mathrm{L}$ corresponding to smooth spatial variation, we see a gradually decreasing trend in correlation with learning as training progresses. Although not all training sessions can be fit to this pattern (i.e. trials 500 and 740), it is still visible that the correlation with learning is above zero (≈ 0.25) at the start of the training when participants perform the task for the first time and gradually shifts to below zero (≈ -0.25) at the end of the experiment when individuals are highly familiar with the sequence. For $\mathbf{x}_\mathrm{H}$ corresponding to vibrant spatial variation, its correlation with learning is below zero (≈ -0.2) at the start of the training, and gradually increases throughout training until it is above zero (≈ 0.25) at the end of the experiment, with the exception of trials 500 and 740. This is the exact opposite of $\mathbf{x}_\mathrm{L}$. For $\mathbf{x}_\mathrm{M}$, correlation between its norm $\|\mathbf{x}_\mathrm{M}\|_2$ with learning rate generally increases with the intensity of training. However, this trend is not as obvious compared to other decomposition counterparts, and there are a greater number of sessions that cannot be fit to this pattern. The correlation between the number of trials and R values is summarized in Figure 67. For robustness testing, we conduct similar analysis using the 120 other sets of parameters described in Section 9.2. The plots (similar to Figure 68) for the R values resulting from all parameter choices are presented in Figure 70 and the correlation between the number of trials and the average R value from considering all parameter choices is summarized in Figure 69. Again, similar observations are found in different experiments involving different learning tasks and different sets of participants.

	$\|\mathbf{x}_\mathrm{L}\|_2$	$\|\mathbf{x}_\mathrm{M}\|_2$	$\|\mathbf{x}_\mathrm{H}\|_2$
6 week experiment (linear scale)	-0.2944	0.0486	0.3808
6 week experiment (logarithm scale)	-0.5334	0.3872	0.3189
3 day experiment	-0.9880	-0.9238	0.9976

Figure 69: Pearson correlation coefficients for robustness testing, as comparable to Figure 67. Each correlation coefficient is between the number of trials (level of task familiarity) and the average R value obtained at each trial. As such, each trial contains 121 R values for the different frequency ranges considered for K_L and K_M between 32 and 42, inclusive. Each R value is defined as the correlation between learning rate parameters and the norm of the decomposed signal of interest for a given frequency range.

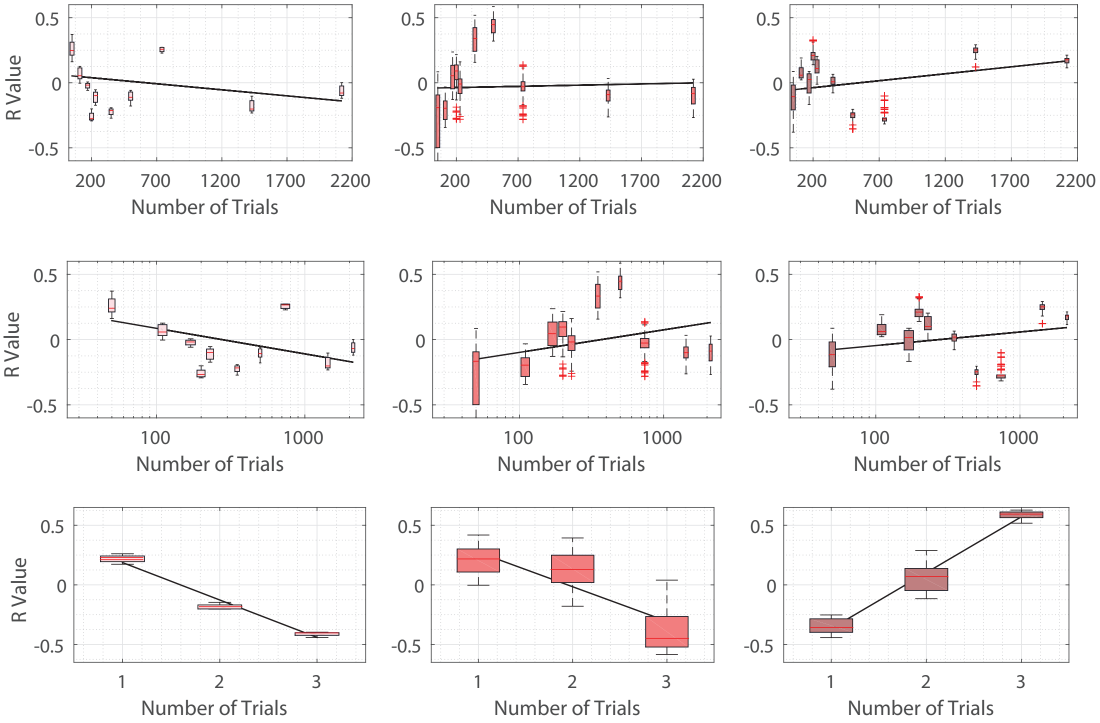


Figure 70: Robustness testing to show similar trends as observed in Figure 68. Each box is for a specific training session (level of task familiarity), depicting the R values obtained from changing the frequency ranges of K_L and K_M between 32 and 42, inclusive. As such, the R value is defined here as correlations between learning rate parameters and the norm of the decomposed signal of interest for a specific frequency range. Each box contains R values for 121 different combinations of frequency ranges.

9.4.1. Discussion

This result further implies that the most association between learning or adaptability during the training process comes from the brain signals that either vary smoothly ($\mathbf{x}_\mathrm{L}$, regularity) or rapidly ($\mathbf{x}_\mathrm{H}$, randomness) with respect to the brain network. Therefore, the graph frequency decomposition could be used to capture more informative brain signals by filtering out non-informative counterparts, most likely associated with middle graph frequencies. Besides, the positive association between $\|\mathbf{x}_\mathrm{L}\|_2$ and learning rates as well as the negative association between $\|\mathbf{x}_\mathrm{H}\|_2$ and learning rates at the start of training indicates that it favors learning to have more *smooth*, *spread*, and *cooperative* brain signals when we face an unfamiliar task. As we gradually become familiar with the task, the smooth and cooperative signal distribution becomes less and less important, and there is a level of exposure when such signal distribution becomes destructive instead of constructive. We note that the task in the 3 day experiment is more difficult compared to that of the 6 week experiment, and therefore the time when the cooperative signal distribution starts to become detrimental (the point where the regression line intercepts the horizontal line of R value equaling 0) is also comparable in the two experiments, describing a certain level of familiarity to the task. When we become highly familiar with the task, it is better and favors further learning to have *varied*, *spiking*, and *competitive* brain signals.

In the dataset evaluated here, we utilize the average coherence between time series at pairs of brain cortical and subcortical regions during the training as the network. Hence, a concentration of brain activities towards low graph frequencies would imply that activities on brain regions that are generally cooperative are indeed similar. Simultaneously, the interpretation of concentration of brain activities towards high graph frequencies is that brain activities on brain regions that are generally cooperative are in fact dissimilar. In terms of learning, one possible explanation is that there are two different stages in learning: we start by grasping the big picture of the task to perform relatively well, and then we refine the details to perform better and to approach our limits.

Because the graph frequency analysis method presented in this paper applies to any setting where signals are defined on top of a network structure representing proximities between nodes, it would be interesting in future to use this method to investigate other types of signals and networks in neuroscience problems. As an example, in situations given fMRI measurements on structural networks, concentration of signals in low graph frequency components would imply functional activities do behave according to the structural networks.

Besides, it has been understood that learning is different when one is unfamiliar or familiar with a particular task – it is easy to improve performance at first exposure due to the fact that one is far from their performance ceiling. It would therefore be interesting to utilize graph frequency decomposition to further analyze the difference between learning scenarios at different stages of familiarity, e.g. adaptability at first exposure and creativity when one fully understands the components of the specific tasks.

Chapter 10

Graph Filter and Attention Switching

In this chapter, we broadly cover how GSP can be applied for an elegant and principled analysis of brain activity. In Section 10.1, we start by constructing a graph from structural connectivity—the backbone of the brain—and considering brain activity as graph signals. Then, in Section 10.2, we follow the definitions of graph spectrum analysis introduced in Chapter 7 to introduce a number of graph signal operations that are particularly useful for processing the activity time courses measured at the nodes of the graph; i.e., filtering in terms of anatomically-aligned or -liberal modes, randomization preserving anatomical smoothness, and localized decompositions that can incorporate additional domain knowledge. In the following sections, we review a recent study in [62] demonstrating the relevance of these GSP tools as an integrated framework to consider structure and function: in the context of an attention task, we discuss the potential of GSP operations to capture cognitively relevant brain properties (Section 10.3). We also provide avenues for utilizing GSP tools in the structure-informed study of functional brain dynamics (Section 10.4), through the extraction of significant excursions in a particular structure/function regime (Section 10.4.1), and by more elaborate uses of GSP building blocks that can broaden the analysis to the temporal frequency domain, or narrow it down to a localized subset of selected regions (Section 10.4.2).

10.1. Brain Graphs and Brain Signals

Following the notations defined in Chapter 7, the brain regions encoded in the nodes of $\mathcal{V}$ are macro-scale parcels of the brain that our current understanding of neuroscience deems anatomically or functionally differentiated. There are various parcellations in use in the literature that differ mostly in their level of resolution [188, 1]. As an example, the networks that we study here consist of $N = 82$ regions from the Desikan-Killiany anatomical atlas [189] combined with the Harvard-Oxford subcortical parcels [190]. A schematic representation of a few labeled brain regions is shown in Figure 71 (left).

The entries A_{ij} of the adjacency matrix $\mathbf{A}$ measure the strength of the axonal connection between

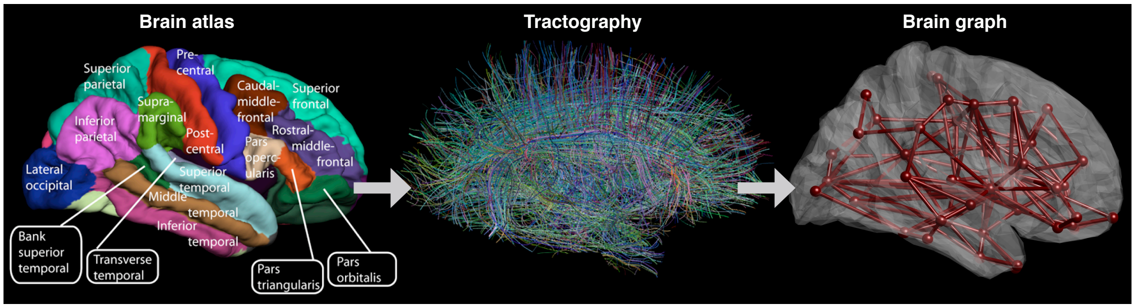


Figure 71: **Estimating brain graphs**. Knowledge from an anatomical atlas based on anatomical features such as gyri and sulci (left) is combined with MRI structural connectivity extracted from diffusion-weighted MRI (middle), which can then be used to estimate the brain graph (right). [Adapted from [187]].

region i and region j. This strength is a simple count of the number of streamlines that connect the regions, and can be estimated with diffusion spectrum imaging (DSI) [191]. In specific, MRI allows the acquisition of detailed structural information about the brain. The brain graph investigated in the present article was acquired on a Siemens 3.0T Tim Trio with a T1-weighted anatomical scan. Twenty-eight healthy individuals volunteered for the experiment. We followed a parallel strategy for data acquisition and construction of streamline adjacency matrices as in [191]. First, DSI scans sampled 257 directions using a Q5 half-shell acquisition scheme with a maximum b-value of 5,000 and an isotropic voxel size of 2.4 mm. We utilized an axial acquisition with repetition time (TR) = 5 s, echo time (TE)= 138 ms, 52 slices, field of view (FoV) (231, 231, 125 mm). We acquired a three-dimensional SPGR T1 volume (TE = minimal full; flip angle = 15 degrees; FOV = 24 cm) for anatomical reconstruction. Second, diffusion spectrum imaging (DSI) was performed to establish structural connectivity. DSI data were reconstructed in DSI Studio using q-space diffeomorphic reconstruction (QSDR)[192]. QSDR computes the quantitative anisotropy in each voxel, which is used to warp the brain to a template QA volume in Montreal Neurological Institute (MNI) space. Then, spin density functions were again reconstructed with a mean diffusion distance of 1.25 mm using three fiber orientations per voxel. Fiber tracking was performed in DSI studio with an angular cutoff of 35°, step size of 1.0 mm, minimum length of 10 mm, spin density function smoothing of 0.0, maximum length of 400 mm, and a QA threshold determined by diffusion-weighted imaging (DWI) signal in the colony-stimulating factor. Deterministic fiber tracking using a modified FACT algorithm was performed until 1,000,000 streamlines were reconstructed for each individual. Third, each anatomical scan was segmented using FreeSurfer[193], and parcellated using the connectome mapping toolkit [194]. A parcellation scheme including $N = 87$ regions was registered to the B0 volume from each subject's DSI data. The B0 to MNI voxel mapping produced via QSDR was used to map region labels from native space to MNI coordinates. To extend region labels through the grey-white matter interface, the atlas was dilated by 4mm [195]. We used FSL to nonlinearly register the individual T1 scans to MNI space. By combining parcellation and streamline information, we constructed subject-specific structural connectivity matrices, whose elements represent the number of streamlines connecting two different regions [196], divided by the sum

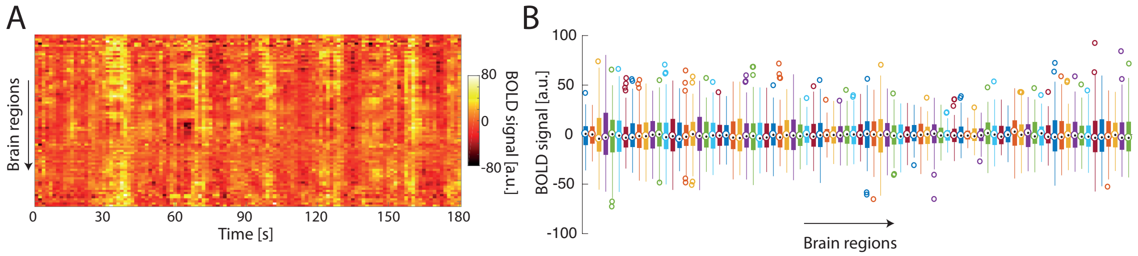


Figure 72: **Example brain activity signals**. (**A**) For an example subject, the heat map of BOLD magnitude activity across brain regions (vertically) and time points (horizontally). Brain activity signals can be considered as a two-dimensional matrix, indexed in both the temporal and spatial domains. From the temporal perspective, there are certain time instances (e.g., in this case, between 30 and 40 seconds and between 70 and 80 seconds) when BOLD magnitudes are in general stronger than for others. From the spatial perspective, signals on most brain regions change in the same direction, but there are certain brain regions where their changes do not follow the main trend. As we will see, low and high graph frequency components, respectively, can be used to extract these two different pieces of information. (**B**) For the same subject, distribution of fMRI BOLD values for each brain region (horizontally) across all time points. Different brain regions exhibit different levels of variability, but in general, the wide variance of BOLD signals complicates data analysis. For each brain region, edges of the box denote 25th and 75th percentiles respectively; whiskers extend to the extreme points not considered to be outliers; circles denote outliers, which are values beyond 1.5 times the interquartile range away from the edges of the box.

of their volumes [187]. This process yields the weighted adjacency matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$ for each individual considered here. Inter-subject variability of structural connectivity has demonstrated clinical value as it has been reliably associated with neurological [116, 197] and psychological [198] disorders.

Besides structural connectivity, it is also possible to acquire brain activity signals $\mathbf{x} \in \mathbb{R}^N$ such that the value of the i^{th} component x_i quantifies neuronal activity in brain region i. In specific, To derive the studied brain activity signals, functional MRI (fMRI) runs were acquired during the same scanning sessions as the DSI data on a 3.0T Siemens Tim Trio whole-body scanner with a whole-head elliptical coil by means of a single-shot gradient-echo T2* (TR = 1500 ms; TE = 30 ms; flip angle = 60°; FOV = 19.2 cm, resolution 3mm x 3mm x 3mm). Preprocessing was performed using FEAT [199], and included skull-stripping with BET [200] to remove non-brain material, motion correction with MCFLIRT [199], slice timing correction (interleaved), spatial smoothing with a 6mm 3D Gaussian kernel, and high-pass temporal filtering to reduce low-frequency artifacts. We also performed EPI unwrapping with fieldmaps in order to improve subject registration to standard space. Native image transformation to a standard template was completed using FSL's affine registration tool, FLIRT [199]. Subject-specific functional images were co-registered to their corresponding high-resolution anatomical images via a Boundary Based Registration technique [201] and were then registered to the standard MNI-152 structural template via a 12-parameter linear transformation. Finally, we extracted region-averaged BOLD signals using the same atlas as for the structural analysis. At the end of this pipeline, we are thus left with a signal matrix $\mathbf{X} \in \mathbb{R}^{N \times T}$ for each subject, reflecting the activity levels of all brain regions over time.

BOLD signals for all the N studied brain regions are acquired over T successive time points, and therefore, we define the matrix $\mathbf{X} \in \mathbb{R}^{N \times T}$ such that its j^{th} column codifies brain activity at time j. An example of such a brain signal matrix is provided in Figure 72A, with the corresponding distribution of values for each brain region illustrated in Figure 72B.

Brain activity signals carry dynamic information that is not only useful for the study of pathology [197, 202, 203], but also enables us to gain insight into human cognitive abilities [204, 205, 206]. Whereas physical connectivity can be seen as a long-term property of individuals that changes slowly over the course of years, brain activity signals display meaningful fluctuations at second or sub-second time scales that reflect how different parts of the brain exchange and process information in the absence of any external stimulus, and how they are recruited to meet emerging cognitive challenges. There is increasing evidence that differences in activation patterns across individuals tightly relate to behavioral variability [207, 208, 209, 76].

To the extent that brain activity signals are generated on top of the physical connectivity substrate, brain graphs and brain signals carry complementary information and should be studied in conjunction. This has been a challenge in neuroscience due to the unavailability of appropriate methods for performing this joint analysis. Here, we advocate for the use of GSP tools, as detailed in the following section.

10.2. Graph Surrogate Signals and Graph Wavelets

In this section, we generalize surrogate signals used to evaluate the significance of obtained results as well as wavelet transforms to graph domains.

10.2.1. Generation of Graph Surrogate Signals

A pivotal aspect in any research field is to assess the significance of obtained results through statistical testing. More precisely, one aims to invalidate the *null hypothesis*, which expresses the absence of the effect of interest. Standard parametric tests such as the well-known t-test assume independent and identically distributed Gaussian noise, which makes a weak null hypothesis for most applications. Non-parametric tests such as the permutation test provide a powerful alternative by mimicking the distribution of the empirical data. For correlated data, the Fourier phase-randomization procedure [210] has been widely applied as it preserves temporal autocorrelation structure under stationarity assumptions. This standard method can be applied to the temporal dimension of our graph signals:

$$\mathbf{Y} = \mathbf{X}\mathbf{V}_{\text{F}}^{H}\mathbf{\Theta}_{\text{time}}\mathbf{V}_{\text{F}}, \tag{10.1}$$

where and $\mathbf{V}_{\text{F}}$ is the Fourier matrix, the diagonal of $\mathbf{\Theta}_{\text{time}}$ contains random phase factors according to the windowing function $\Theta(\lambda_l) = \exp(j2\pi\theta_l)$, with θ_l realizations[1] of a random variable uniformly distributed in the interval $[0, 1]$. From the surrogate signals, one can then compute a

---

[1] In practice, some additional constraints are added such as preservation of Hermitian symmetry.

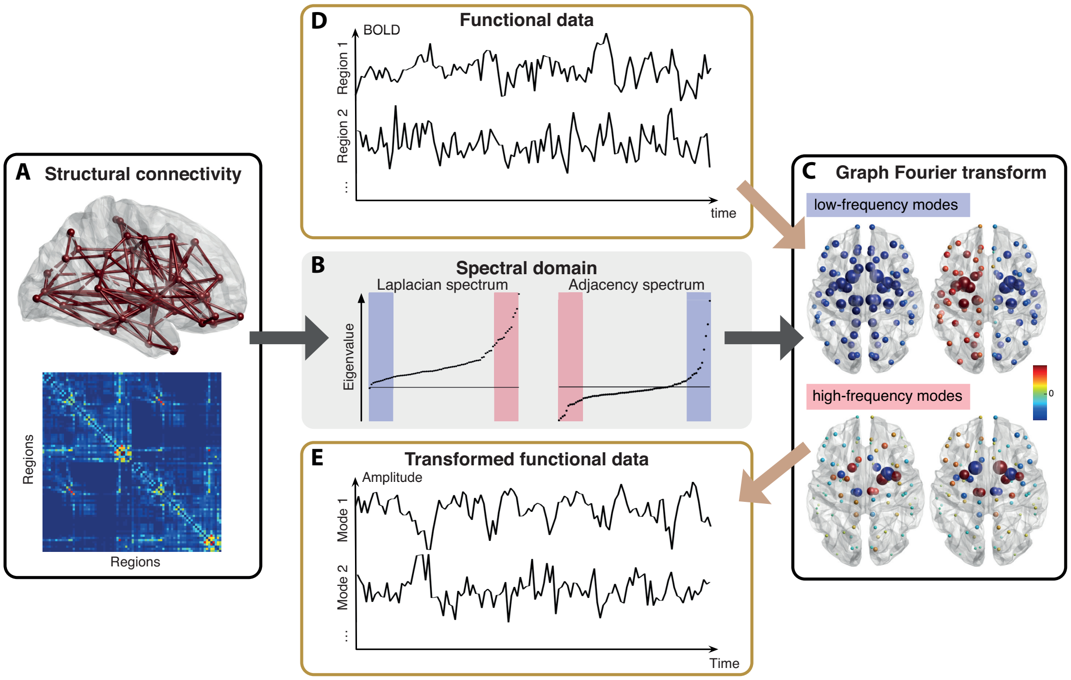


Figure 73: **Graph signal processing for brain imaging**. (**A**) Structural connectivity from diffusion-weighted MRI, as seen in the form of a sagittal brain view (top) or of an adjacency matrix where the weights represent the strength of the structural connections (bottom), is used to build a graph representing the brain's wiring scaffold. (**B**) Through the eigendecomposition of the Laplacian (left plot) or adjacency (right plot) matrix, this structural graph can be analyzed in the spectral domain. The smallest Laplacian eigenvalues (or most positive adjacency eigenvalues) (labeled in blue) are associated with low-frequency modes on the graph (**C**, top brain views), while the largest Laplacian eigenvalues (or most negative adjacency eigenvalues) (labeled in red) are associated with high-frequency modes (**C**, bottom brain views). Together, these modes define the graph Fourier transform. Functional MRI data measured at the nodes of the graph (**D**) can be decomposed using these modes, and transformed by means of graph signal processing tools (**E**).

test statistic and establish its distribution under the null hypothesis by repeating the randomization procedure; i.e., the power spectrum density of the surrogate data is dictated by the empirical data. Note that in this setting, the spatial features of null realizations are identical to the ones of the actual data, while temporal non-stationary effects are destroyed.

The phase randomization procedure can be generalized to the graph setting [211] by considering the GFT. In particular, the graph signal can be decomposed on the GFT basis and then, the graph spectral coefficients can be randomized by flipping their signs. Assuming that the random sign flips are stored on the diagonal of $\mathbf{\Theta}_{\text{graph}}$, we can formally write the procedure as:

$$\mathbf{Y} = \mathbf{V}\mathbf{\Theta}_{\text{graph}}\mathbf{V}^{\top}\mathbf{X}. \tag{10.2}$$

This procedure generates surrogate graph signals in which the smoothness as measured on the graph is maintained, but in which the non-stationary spatial effects is destroyed. The temporal properties of null realizations are identical to those observed in the actual data.

10.2.2. Wavelets and Slepians on the Graph

The wavelet transform is another fundamental tool of signal processing [212] providing localized, multiscale decompositions. Several designs have been proposed to generalize this concept to graphs, such as approaches in the vertex domain [213, 214, 215], based on diffusion processes [216, 217], or using the spectral domain [218, 219, 220]. The latter design builds upon the GFT and has been applied for multiscale community mining [221] or to investigate uncertainty principles [67, 68, 69, 70].

Here, we detail a more recent design of a localized decomposition for graph signals that is based on a generalization of Slepian functions [222] and that can deal with additional domain knowledge. Let us consider the problem of retrieving a signal $\mathbf{x} \in \mathbb{R}^N$ that is maximally concentrated within a subset of nodes from the graph at hand, while at the same time setting a maximal bandwidth on the solution. As the global concentration of a signal is given by $\mathbf{x}^\top \mathbf{x}$, we end up maximizing

$$\mu = \frac{\tilde{\mathbf{x}}^\top \bar{\mathbf{V}}^\top \mathbf{M} \bar{\mathbf{V}} \tilde{\mathbf{x}}}{\tilde{\mathbf{x}}^\top \tilde{\mathbf{x}}}, \tag{10.3}$$

where $\mathbf{M}$ is the diagonal *selectivity matrix* with elements $M_{i,i} = 0$ or 1 to respectively exclude, or include, a node into the sub-graph of interest, and $\bar{\mathbf{V}} \in \mathbb{R}^{N \times M}$ is a trimmed GFT matrix where only low-frequency basis vectors are kept. The interpretation here is that we aim at finding the linear combination of band-limited graph spectral coefficients enabling the best localization of the signal within the sub-graph. Note that the sub-graph is selected using prior information, and not optimized over.

If we define the *concentration matrix* as $\mathbf{C} = \bar{\mathbf{V}}^\top \mathbf{M} \bar{\mathbf{V}}$, then the problem amounts to solving its eigendecomposition, and $\{\tilde{\mathbf{p}}_k\}, k = 0, 1, \ldots, M-1$, are the weighting coefficients obtained as solutions. We assume that they are ordered in decreasing eigenvalue amplitude ($\mu_0 > \mu_1 > \ldots > \mu_{M-1}$), so that $\tilde{\mathbf{p}}_0$ is the optimal (maximally concentrated) solution. From the set of coefficients, the *Slepian matrix* can then be retrieved as:

$$\mathbf{P} = \bar{\mathbf{V}} \tilde{\mathbf{P}}, \tag{10.4}$$

where $\mathbf{P} \in \mathbb{R}^{N \times M}$ and each column contains one of the Slepian vectors $\mathbf{p}_k$. Slepian vectors are not only orthonormal within the whole set of nodes ($\mathbf{p}_k^\top \mathbf{p}_l = \delta_{k-l}$), but also orthogonal over the chosen subset ($\mathbf{p}_k^\top \mathbf{M} \mathbf{p}_l = \mu_k \delta_{k-l}$).

Now, in order to make Slepian vectors more amenable to the application of GSP tools, let us consider an alternative optimization criterion in which the modified concentration matrix is given as $\mathbf{C_2} = \bar{\mathbf{\Lambda}}^{1/2} \mathbf{C} \bar{\mathbf{\Lambda}}^{1/2}$, with $\bar{\mathbf{\Lambda}} \in \mathbb{R}^{M \times M}$ the trimmed diagonal matrix of eigenvalues. The new quantity to optimize then reads:

$$\xi = \frac{\tilde{\mathbf{x}}^\top \bar{\mathbf{\Lambda}}^{1/2} \bar{\mathbf{V}}^\top \mathbf{M} \bar{\mathbf{V}} \bar{\mathbf{\Lambda}}^{1/2} \tilde{\mathbf{x}}}{\tilde{\mathbf{x}}^\top \tilde{\mathbf{x}}}. \tag{10.5}$$

The set of solution Slepian vectors are still orthonormal, but this time, they satisfy $\mathbf{p}_k^\top \mathbf{M} \mathbf{p}_l = \xi_k \delta_{k-l}$. Observe that, when using the Laplacian matrix as our graph shift operator, if all nodes are selected as the subset of interest ($\mathbf{M} = \mathbf{I}$) while enabling a full bandwidth ($\bar{\mathbf{\Lambda}} = \mathbf{\Lambda}$, $\bar{\mathbf{V}} = \mathbf{V}$), then we fall back on the classical Laplacian embedding case discussed in Chapter 7, and as such, this modified criterion can be seen as a generalization of Laplacian embedding (i.e., a *modified embedded distance* criterion) under user-defined bandwidth and selectivity constraints.

Analogously to the GFT setting, solution Slepian vectors of increasing eigenvalue ξ_k can then be regarded as building blocks of increasing graph frequency, but within the chosen sub-graph, i.e., of increasing *localized* frequency. The conceptual difference between both optimization schemes is illustrated in an example dataset of leopard mesh in Figure 74, where the sub-graph is the head of the leopard as shown in Figure 74A. Four of the Slepian vectors derived from (10.5) are shown with their localized frequency ξ, their energy concentration μ computed from (10.3), and their embedded distance $\lambda = \mathbf{p}^\top \mathbf{L} \mathbf{p}$. The leftmost example denotes a low frequency on the whole graph, with very weak signal within the selected sub-graph, and thus both low localized frequency and energy concentration. The second Slepian vector shows fairly uniform negative signal within the sub-graph, resulting in a quite large energy concentration, but a very low localized frequency. The last two examples reflect Slepian vectors that are both strongly concentrated (high μ) and of high localized frequency (high ξ).

If Laplacian embedding is performed on the full graph (Figure 74C, left plot), the resulting eigenvectors linearly span the graph frequency spectrum (black line). If the energy concentration criterion is used for generating Slepian vectors (middle plot), there is a well-defined transition point past which Slepian vectors become strongly concentrated within the selected subset of nodes. If the modified embedded distance criterion is used (right plot), then, past a point where Slepian vectors become concentrated within the subset (around 600 in this example), they also linearly span the localized graph frequency space.

As a result, it becomes possible to apply similar GSP tools as for the GFT, but for a decomposition that can be tailored in terms of localization by utilizing different subgraphs, and the choice of the bandwidth. In fact, the Slepian matrix can be seen as an alternative set of basis vectors, themselves obtained as a linear combination of Laplacian eigenvectors under the localized concentration constraint. For example, the temporal signal matrix $\mathbf{X}$ at hand can be projected on the Slepian building blocks as $\mathbf{P}^\top \mathbf{X}$, and if we define the diagonal matrix $\mathbf{\Gamma}_L$ as a localized low-pass filter by setting $[\mathbf{\Gamma}_L]_{i,i} = 1$ if $\xi_i < \hat{\xi_L}$ (low localized frequency) and $\mu_i > \epsilon$ (concentrated solution), or 0 otherwise, the locally filtered output signal would be given by:

$$\mathbf{Y}_{\mathbf{\Gamma}_L} = \mathbf{P} \mathbf{\Gamma}_L \mathbf{P}^\top \mathbf{X}. \tag{10.6}$$

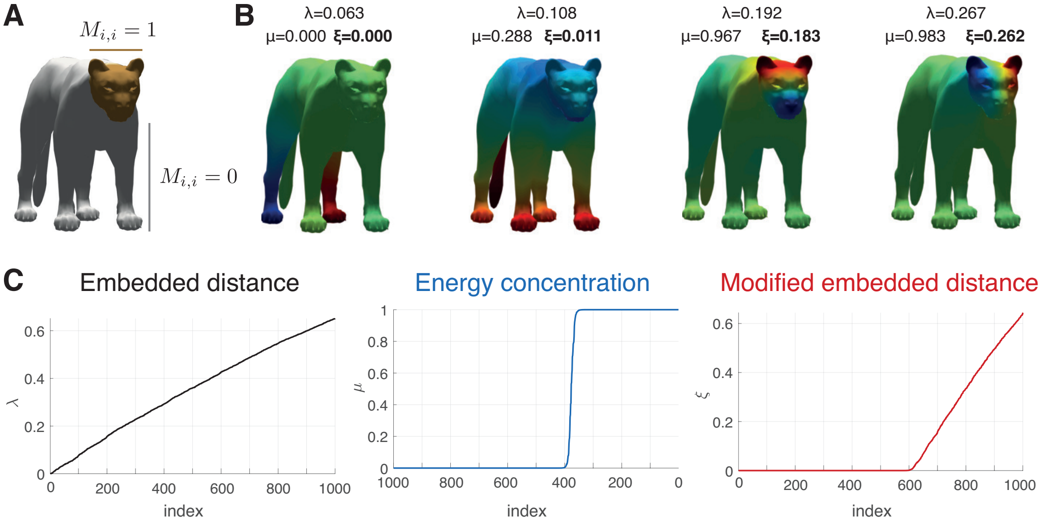


Figure 74: **Illustration of Slepian vectors and their properties**. (**A**) Within the considered graph (a leopard mesh), the head is selected as the subset of nodes of interest. (**B**) Example Slepian vectors obtained from the modified embedded distance optimization criterion (10.5). In each case, alongside localized frequency (ξ), embedded distance (λ), and energy concentration (μ) are also shown. (**C**) For a bandwidth $M = 1000$ and Laplacian embedding (left), energy concentration (middle) or modified embedded distance (right) optimizations, sorting of the obtained eigenvalues (respectively λ, μ or ξ).

10.3. A Brain GSP Case Study: Deciphering the Signatures of Attention Switching

We now discuss how the aforementioned GSP methods can be applied in the context of functional brain imaging. Figure 75 is reproduced from [62]; Figures 76A and B are adapted from [62]. To do so, we focus on the data whose acquisition was described in Section 10.1, Callouts. For each volunteer, fMRI recordings were obtained when performing a Navon switching task, where local-global perception is assessed using classical Navon figures [223]. Local-global stimuli were comprised of four shapes – a circle, cross, triangle, or square – that were used to build the global and local aspects of the cues (see Figure 75A for examples).

A response (button press) to the local shape was expected from the participants in the case of white stimuli, and to the global shape for green ones. Two different block types were considered in the experiment: in the first one (Figure 75B), the color of the presented stimuli was always the same, and the subjects thus responded consistently to the global or to the local shapes. In the second block type (Figure 75C), random color switches were included, so that slower responses were expected. The difference in response time between the two block types, which we refer to as *switch cost*, quantifies the behavioral ability of the subjects.

To study the association between brain signal and attention switching, we decomposed the functional brain response into two separate components: one representing *alignment* with structural connectivity (i.e., the regions that activate together are also physically wired), and one describing

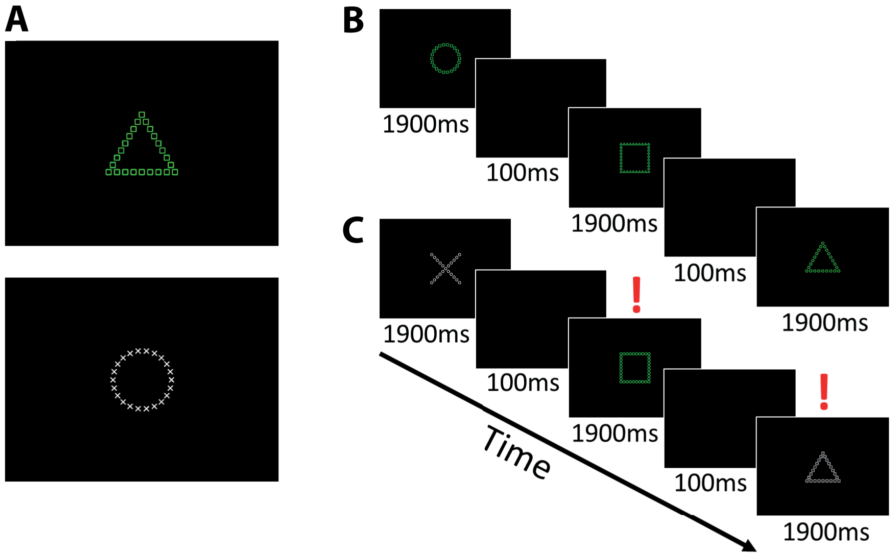


Figure 75: **Cognitive task requiring perceptual switching**. (**A**) Example stimuli based on Navon local-global features. Subjects were trained to respond to the larger (or "global") shapes if the stimulus was green and to the smaller (or "local") shapes if it was white. (**B**) An example of the non-switching condition for responses. Subjects viewed a sequence of images and were instructed to respond as quickly and as accurately as possible. (**C**) An example of the switching condition between stimuli requiring global and local responses. Here, trials with a red exclamation mark are switches from the previous stimulus.

liberality (i.e., the areas that exhibit high signal variability with respect to the underlying graph structure). To do so, we performed graph signal filtering (Section 7.2) with two different filtering matrices: (1) $\boldsymbol{\Psi}_{\mathrm{Al}}$, so that $\mathbf{Y}_{\boldsymbol{\Psi}_{\mathrm{Al}}} = \mathbf{V}\boldsymbol{\Psi}_{\mathrm{Al}}\mathbf{V}^{\top}\mathbf{X}$ is the transformed (low-pass filtered) functional data in which only the 10 lowest frequency modes are expressed at each time point; and (2) $\boldsymbol{\Psi}_{\mathrm{Lib}}$, for which $\mathbf{Y}_{\boldsymbol{\Psi}_{\mathrm{Lib}}}$ only represents the temporal expression of the 10 largest frequency modes (high-pass filtering). At a given time point, the filtered functional signal varies in sign across brain regions. Thus, to derive a subject-specific scalar quantifying alignment or liberality, we considered the norms of those signals as measures of concentration, which were eventually averaged across all temporal samples of a given subject. We used the ℓ_2 norm because it provides an interpretation of energy for each graph frequency component; other reasonable choices of norm, including the ℓ_1 norm, yield similar results. Also, presented results are obtained using the adjacency matrix as the graph shift operator, but similar findings were recovered using the Laplacian matrix instead (see Callout 3).

To relate signal alignment and liberality to cognitive performance of the participants, we computed partial Pearson's correlation between our concentration measures and switch cost (median additional response time during switching task blocks compared to non-switching task blocks). Age and motion were included as covariates to remove their impact from the results. Regarding alignment, there was no significant association ($p > 0.35$; Figure 76A). In other words, the extent with which functional brain activity was in line with the underlying brain structural connectivity

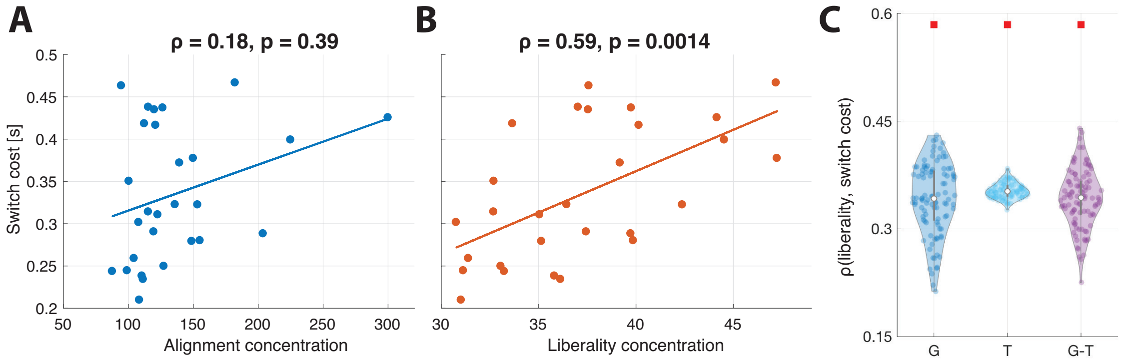


Figure 76: **Switch cost correlates with the concentration in liberal signal**. (**A**) Switch cost does not significantly relate to the concentration of the low-frequency functional signal component (alignment). (**B**) A lower concentration of graph high-frequency components is associated with a lower switch cost, that is, with faster attention switching. (**C**) The correlation between switch cost and liberal signal concentration is much stronger in the actual data than in null realizations, irrespective of whether the statistical randomization is performed in the graph domain (denoted as 'G' in the figure), in the temporal domain (denoted as 'T' in the figure), or jointly performed in both (denoted as 'G-T' in the figure). Blue, cyan and purple data points denote the correlation coefficients obtained from surrogate signals under the three null models, while the red rectangle indicates the real correlation coefficient ($\rho = 0.59$). ρ, partial Pearson's correlation coefficient; p, p-value.

did not relate to cognitive abilities in the assessed task. However, we observed a significant positive correlation between liberal signal concentration and switch cost ($\rho = 0.59$, $p < 0.0015$; see Figure 76B). Thus, the subjects exhibiting most liberality in their functional signals were also the ones for whom the attention switching task was the hardest. We verified that the high-frequency modes involved in those computations were not solely localized to a restricted set of nodes by evaluating the distribution of the average decomposed signal across all brain regions. When averaged across all time points and subjects, 27 brain regions had their decomposed signals higher than 1.5 times the mean of the distribution (approximately 3), confirming that a wide area of the brain was spanned by high-frequency modes. From these results, one can see that a GSP framework may provide a way to disentangle brain signals that exhibit different levels of association with attention switching.

To more thoroughly examine the significance of the association between liberal signals and switch cost, we performed a null permutation test by generating graph surrogate signals as described in Section 10.2.1. Specifically, we generated 100 graph surrogate signals by randomly flipping the signs stored on the diagonal of $\mathbf{\Theta}_{\text{graph}}$, as in (10.2). Then, we evaluated the association between the null surrogate signals and switch cost. As seen in Figure 76C (case 'G'), the actual correlation coefficient between liberal signal concentration and switch cost (denoted by the red rectangle) is significantly larger than when computed on any of the null graph surrogate signals. We also performed the same process using phase randomization in the time domain to generate surrogate signals (see Figure 76C, case 'T'), which preserves the temporal stationarity assumption, and com-

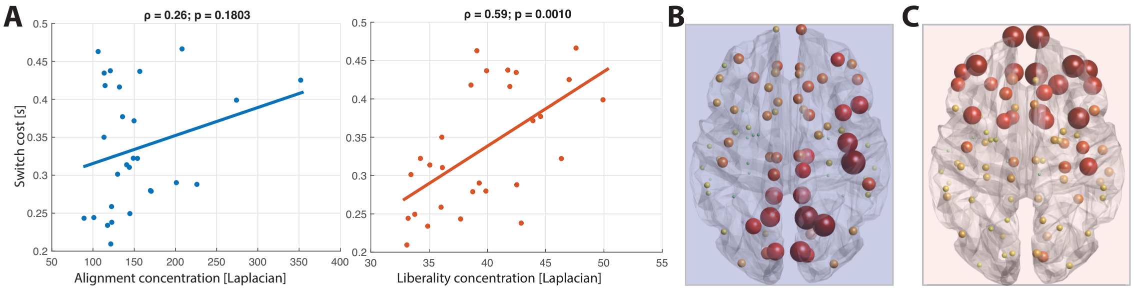


Figure 77: **Results carry over to alternative graph shift operator – graph Laplacian** (**A**) Switch cost correlates with the concentration in liberal signal and not aligned signal using Laplacian as a shift operator; results are similar as in Figure 76A and B. Horizontal brain views depicting excursion occurrence across brain nodes with Laplacian for alignment (**B**) and liberality (**C**); results are similar as in Figure 79B.

bining phase randomization in the time domain and randomly flipping the signs of graph spectral coefficients (Figure 76C, case'G-T'). Again, the actual correlation coefficient between liberal signal concentration and switch cost was significantly larger than for any of the null realizations.

To confirm that the graph frequency decomposition framework is insensitive to the level of resolution used in the considered parcellation, we examined the data recorded during the same experiment, on the same subjects, but at a higher resolution ($N = 262$ different brain regions). In other words, we considered the same experiment, but defined the network differently by having each node of $\mathcal{V}$ consisting of a smaller volume of the brain. We followed the same graph frequency decomposition, using the adjacency matrix as graph shift operator, on this finer graph. We observed that the results still held, as switch cost did not significantly relate to the concentration of the low-frequency signal component ($\rho = 0.3408, p = 0.0759$), whereas a lower concentration of the high-frequency component was associated with faster attention switching ($\rho = 0.4232, p = 0.0249$). Here and above, the results were also robust to the number of largest/smallest frequency components used in the decomposition.

In sum, in this section we reviewed a recent study [62] demonstrating that individuals whose most liberal fMRI signals were more aligned with white matter architecture could switch attention faster. In other words, relative alignment with anatomy is associated with greater cognitive flexibility. This observation complements prior studies of executive function that have focused on node-level, edge-level, and module-level features of brain networks [224, 225]. The importance of this finding illustrates the usefulness of GSP tools in extracting relevant cognitive features.

Up to this point, we have been dealing with a graph frequency decomposition considered at the level of the whole brain. However, GSP tools also allow us to independently evaluate separate nodes, or sets of nodes, from the graph at hand. In the present case, this flexibility permits a more in-depth study of which brain regions are specifically responsible for the observed association between liberality and switch cost. For this purpose, we considered 9 different, previously defined functional brain systems [191], each of which included a distinct set of regions. We assessed,

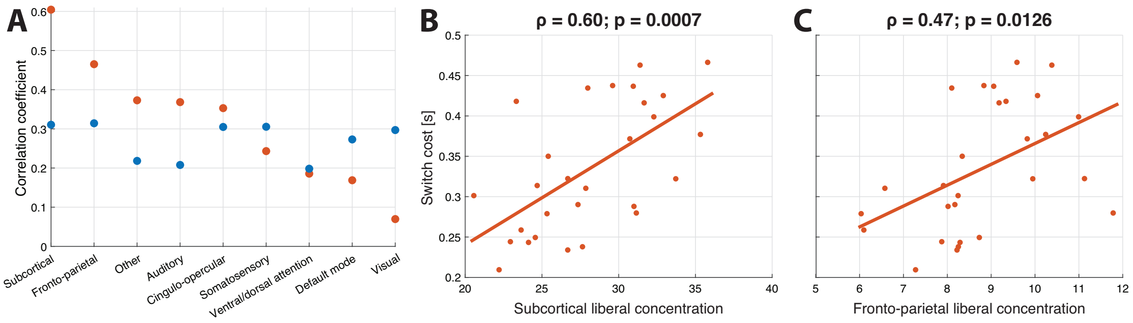


Figure 78: **Pinpointing the brain systems involved in attention switching**. (**A**) Separate partial correlation assessments between switch cost and alignment (blue) or liberality (red) signal concentration on the brain areas belonging to different functional systems, using age and motion as covariates. Systems are ordered in decreasing liberality correlation coefficient order. Liberality concentrations of subcortical and fronto-parietal systems exhibit the highest and most significant contributions to the association with switch cost. Liberality concentrations of other systems and alignment concentrations of any system exhibited no significant association ($p > 0.05$). (**B**) A lower concentration of graph high-frequency components in the subcortical system is associated with faster attention switching. (**C**) A lower concentration of graph high-frequency components in the fronto-parietal system is associated with faster attention switching.

separately for each system, the correlation between switch cost and alignment or liberality. In the former case (alignment), there was no significant association, whereas in the latter (liberality), the relationship seen in Figure 76B could be narrowed down to two significant contributors: the subcortical and the fronto-parietal systems (Figure 78). Those results highlight the ability of GSP tools to not only decompose signals in the graph *frequency domain*, but also in the graph *spatial domain* (examining different nodes in the graph). Combining those two analytical axes enables us to gather deeper insights into functional brain activity and its relation to cognition.

Multiple graph shift operators could be used to decompose graph signals. Most of the material presented in this work uses the adjacency matrix as graph shift operator, but results remain very similar if the Laplacian matrix is used instead. More specifically, we reevaluated the association with switch cost illustrated in Figure 76, and the set of brain regions most frequently undergoing alignment or liberality excursions as displayed in Figure 79B, using the Laplacian matrix as graph shift operator. Figure 77, presented below, illustrates the similarity in the obtained results. There exist other types of graph shift operators, e.g. the normalized Laplacian, for which results can also be expected to remain relatively similar.

10.4. Perspectives for Brain GSP: Studying Functional Dynamics

10.4.1. *Resolving Excursions in Alignment or Liberality Regimes*

We now illustrate, on the same data as above, how GSP tools can be applied to provide insights into the dynamics of functional brain activity. For every subject, we generated 1000 null signal matrices using the strategy outlined in (10.2) (graph domain randomization). We combined this

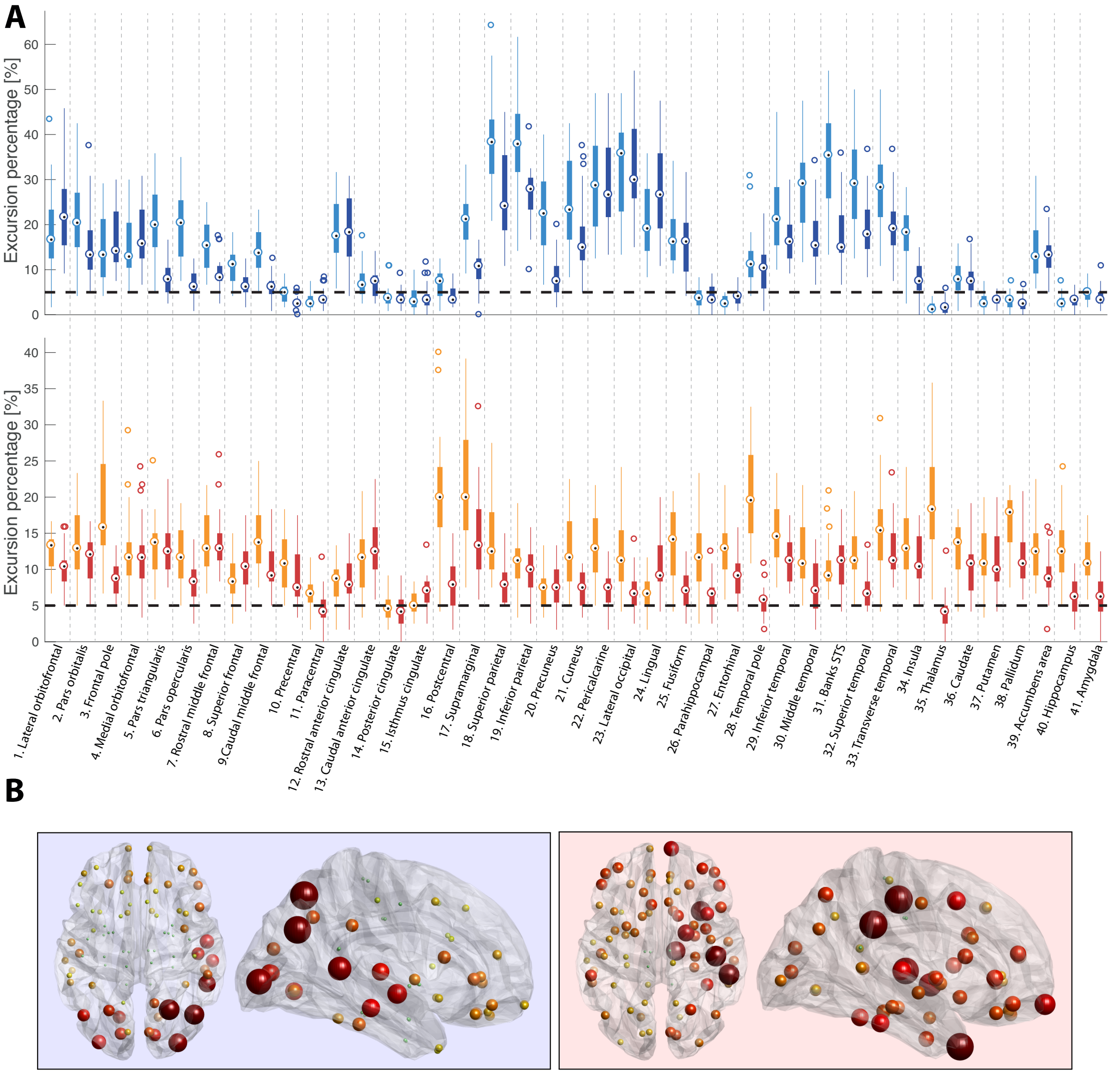


Figure 79: **Significant excursions of aligned and liberal functional signals across regions**. (**A**) For all 82 nodes, percentage of significant excursions for alignment (top panel, light and dark blue box plots) or liberality (bottom panel, red and orange box plots) across subjects. The horizontal dashed line denotes chance level ($\alpha = 5\%$), and light gray vertical dashed lines separate the box plots from different regions. Light colors denote regions from the left side of the brain, and dark colors from the right side. (**B**) For alignment (left box) and liberality (right box), horizontal and sagittal brain views depicting excursion occurrence across brain nodes. A larger amount of significant excursions is denoted by a bigger and redder sphere. Left on the brain slices stands for the right side of the brain.

operator ($\mathbf{\Theta}_{\text{graph}}$) with the alignment/liberality filtering operations, to generate null data for the aligned and liberal signal components. Formally, we thus computed a null realization as $\mathbf{Y} = \mathbf{V}\mathbf{\Theta}_{\text{graph}}\mathbf{\Psi}_{\text{Al}}\mathbf{V}^{\top}\mathbf{X}$ or $\mathbf{Y} = \mathbf{V}\mathbf{\Theta}_{\text{graph}}\mathbf{\Psi}_{\text{Lib}}\mathbf{V}^{\top}\mathbf{X}$, respectively. At an α-level of 5%, we then used the generated null data to threshold the filtered signals, in order to locate significant signal *excursions* – particular moments in time when entering a regime of strong alignment, or liberality, with the underlying brain structure. In doing so, we considered absolute graph signals. Presented results are obtained using the adjacency matrix as graph shift operator, but similar findings were recovered using the Laplacian matrix (see Callout 3).

Figure 79A highlights the percentage of time points showing significant excursions for the aligned (light blue and dark blue box plots) and liberal (red and orange box plots) signal components across brain regions. An excursion percentage value of 5% (horizontal dashed line) denotes chance level. Such a case was, for instance, observed for the paracentral and posterior cingulate areas (nodes 11 and 14), both in terms of aligned and liberal signal contributions. As null data realizations were generated in the graph domain, this observation means that those nodes did not show signal fluctuations going beyond what could be accounted for by the underlying spatial smoothness of the brain's structural graph.

Most brain regions did display very significant excursion percentages: considering alignment, occipital (nodes 21-25), parietal (nodes 18 and 19) and temporal (nodes 29-33) regions were the strongest contributors, while for liberality, key areas were located in temporal (nodes 29-33), subcortical (nodes 34, 36-39) or frontal (nodes 1-9) regions. Figure 79B displays the anatomical location of the main contributing regions. Qualitatively similar findings were also obtained when resorting to a finer parcellation of the brain ($N = 262$ regions; see Supplementary Figure 1). The observation that the majority of brain nodes show frequent moments of strong alignment or liberality with respect to brain structure is consistent with current knowledge on spontaneous brain dynamics, since an alternation between time points with and without global similarity to the structural scaffold has previously been documented from second-order connectivity analyses [226, 227]. A GSP approach can also reveal these subtle relationships, with the added advantage of conserving a frame-wise temporal resolution.

To better grasp the signal features at the root of alignment or liberality excursions, we compared the outcomes obtained using the graph surrogate method to the ones generated with the more classical Fourier phase-randomization procedure to generate null data, or to the outcomes resulting from the combination of those two surrogate approaches (see Supplementary Figure 2). Excursions in terms of liberality with respect to brain structure were not resolved anymore under those two other null models, for which null realizations conserve similar stationary temporal properties. This implies that the liberal signal component can be explained by stationary temporal features. On the other hand, alignment excursions remained, in particular when including graph domain randomization. Thus, the aligned signal component relates to spatial features that cannot be explained by stationary smoothness alone.

10.4.2. Combining Graph Excursions with Fourier Analysis

Other ingredients from the GSP pallet can be appended to the pipeline we have introduced, in order to further expand our understanding of brain activity. For example, to examine whether alignment and liberality would change along frequency, referring this time to the *temporal frequency* of the signal, we simply combined our null and alignment/liberality operators with the classical

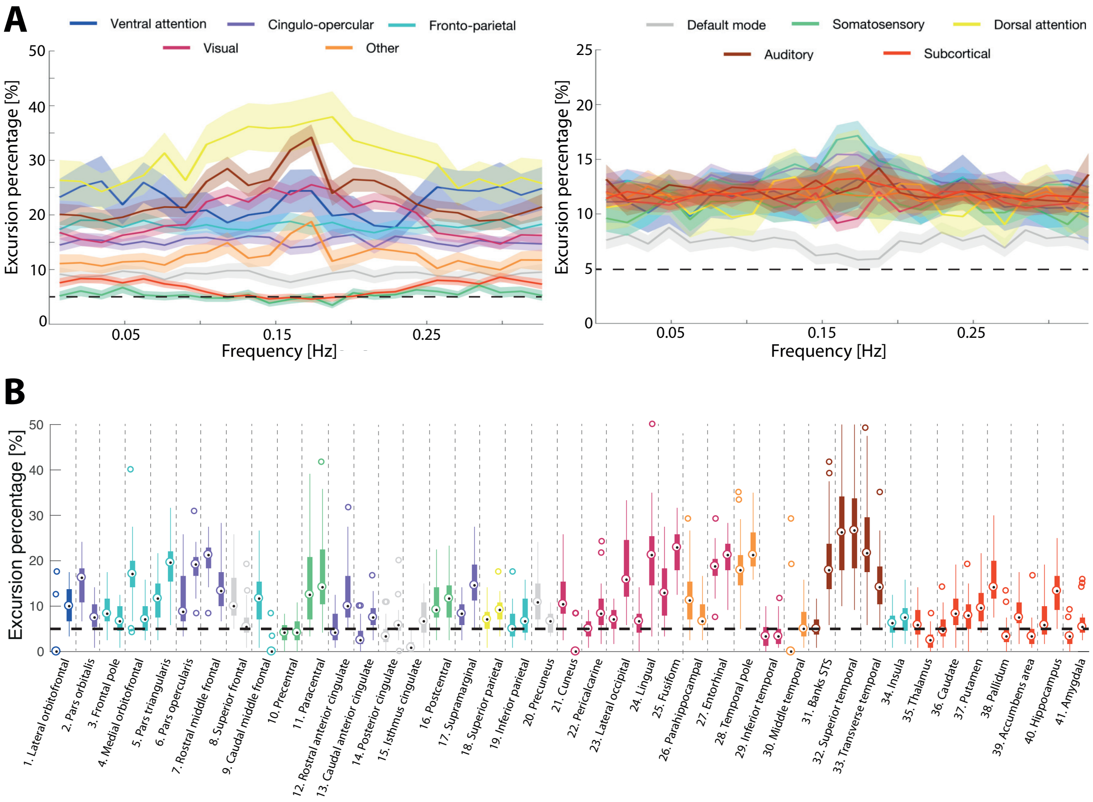


Figure 80: **Further disentangling functional brain signals by more elaborate GSP building blocks**. (**A**) Percentage of significant excursions for key functional brain systems across temporal frequency sub-bands in the case of the aligned (left graph) or liberal (right graph) signal contributions. Two-tailed 95% confidence intervals are displayed for each curve, and the horizontal dashed line represents the excursion chance level ($\alpha = 5\%$). (**B**) As a quantification of local alignment, percentage of significant excursions for all brain nodes when applying the graph Slepian design with bandwidth $M = 80$. Color coding reflects the functional system to which a region belongs, and for a given region, the left box plot stands for the left side of the brain.

Fourier decomposition highlighted in Chapter 7, and computed the percentage of significant excursions for all the functional brain systems introduced in [191] (Figure 80A). For alignment (left graph), different systems were observed to vary in terms of excursion occurrence, with dorsal attention and auditory areas as primary contributors while subcortical and somatosensory regions stood at around chance level. Interestingly, in a few cases, alignment with the structural brain scaffold appeared to be maximized at particular frequencies: for instance, the dorsal attention, ventral attention and auditory systems showed more frequent excursions in the $0.15 - 0.2$Hz range.

Regarding liberality (right graph), almost all systems showed similar excursion percentages, with the exception of the default mode network (gray line), whose regions appeared to more rarely diverge from the activation patterns expected from structural connectivity. In addition, excursions further decreased close to chance level in the $0.15 - 0.2$Hz range, while at the same time, positive peaks could be seen, amongst others, for the fronto-parietal and cingulo-opercular systems.

This antagonistic relationship between those functional brain systems could be the reflection of a hallmark feature of brain activity: the anti-correlation between the default mode (also known as *task-negative*) and so called *task-positive* networks [228]. The GSP approach enables, a more accurate characterization of these networks in terms of both temporal and graph frequencies.

10.4.3. Probing Excursions within a Sub-graph with Slepians

Finally, another way to dig deeper into the functional signals is to consider them at a *local scale*, rather than at the whole-brain level. For this purpose, we computed a basis of Slepian vectors through the process detailed in Section 10.2.2 (using the modified embedded distance optimization criterion). We started from the eigendecomposition of the Laplacian matrix, and iteratively focused the analysis on a subset of nodes being part of only one given functional brain system. Every time, we derived $M = 80$ Slepian vectors, and used the 10 lowest localized frequency (i.e., with lowest ξ_i), concentrated (i.e., satisfying $\mu_i > \epsilon$) elements of this new basis to extract the part of the functional signals aligned with local structural brain features, generate null data, and quantify significant excursions.

As can be seen in Figure 80B, some nodes stand out as undergoing particularly frequent excursions in terms of local alignment to brain structure. This is for example seen for regions from the visual (nodes 23-25) and auditory (nodes 31-33) systems, reflecting the presence of moments when there is strong alignment of the functional signals with the underlying structure *at the local scale of the considered system*, which is encoded in the Slepian basis. We note that the same nodes already showed high excursion percentages in Figure 79A, where alignment was assessed at the global (not local) level, and thus, what was captured by this less focused analysis may have largely involved local alignment with structure. Conversely, there are also many cases in which regions exhibited frequent global alignment with the structural scaffold, without displaying it at the local scale (for example, nodes 18-19). In such cases, global alignment to structure instead reflects cross-network interactions. Overall, surrogate analyses are conducted from three aspects in the preceding subsections (vanilla as in Section 10.4.1, combined with Fourier analysis as in Section 10.4.2, and combined with Slepians sub-graph as in Section 10.4.3). Some consistent observations inherited from the surrogate analysis itself are found across the subsections, while some different results reflect the different perspectives and features of the respective approach.

Chapter 11

Recommendation System

The organization of the chapter is as follows. Section 11.1 introduces the problem formulations for collative filtering, and summarizes the GSP tools that are relevant for a recommendation system setup. Section 11.2 reinterprets NN approaches under a GSP perspective, showing that NN predictors can be viewed as a graph *band-stop* filter, and proposes novel methods for recommendation systems including higher order band-stop graph filters in either the user-based network or the item-based network, and higher order band-stop graph filters on a joint user-item network. Section 11.3 reinterprets LF from a GSP perspective, showing that LF schemes can be viewed as reconstruction algorithms that give rise to *bandlimited* rating prediction signals. It also presents novel GSP-inspired methods to interpolate the sampled ratings under the assumption of graph bandlimitedness. Numerical experiments comparing the performance of the proposed methods and illustrating their practical relevance are presented in Section 11.4 using the MovieLens 100k dataset.

11.1. Collaborative Filtering

Consider a recommender system setup with U users indexed by u, and I items indexed by i. The rating that user u would give to item i is represented by the unobserved variable $\bar{Y}_{ui}$, but only a few of these ratings are available; see Figure 81. Available ratings are denoted as Y_{ui} and we further denote as $\mathcal{S}_u \subseteq \{1, \ldots, I\}$ the set of items that have been rated by user u, as $\mathcal{T}_i \subseteq \{1, \ldots, U\}$ the set of users that have rated item i, and as $\mathcal{R} \subseteq \{1, \ldots, U\} \times \{1, \ldots, I\}$ the set of user-item pairs with available ratings. Although ratings are typically recorded as nonnegative scores, they are usually converted to mean centered scores. To center them around the user mean we define the average score of user u as $\mu_u := (1/|\mathcal{S}_u|) \sum_{i \in \mathcal{S}_u} Y_{ui}$ and define the user mean centered ratings

$$X^{\mathrm{U}}_{ui} \ := \ Y_{ui} - \mu_u \ = \ Y_{ui} - \big(1/|\mathcal{S}_u|\big) \sum_{i \in \mathcal{S}_u} Y_{ui}. \tag{11.1}$$

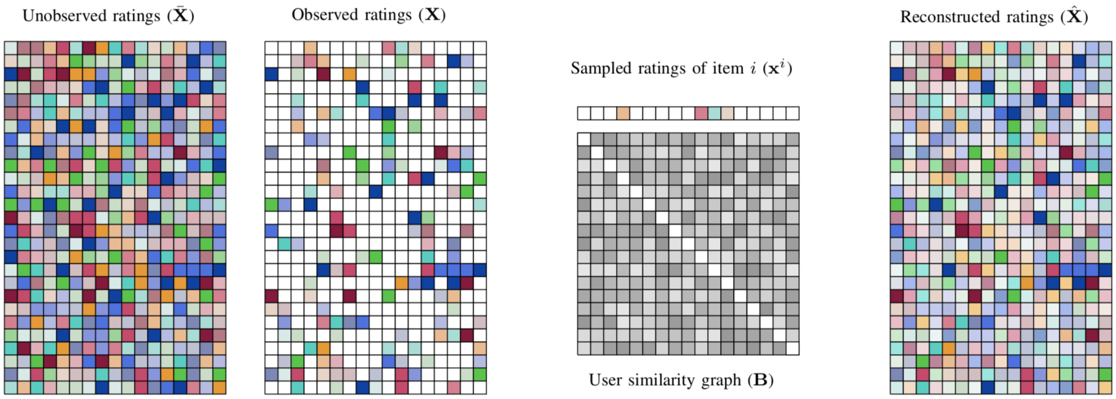


Figure 81: The ratings that users would give to items are represented by the unobserved ratings $\bar{\mathbf{X}}$ (left). Available ratings represented by $\mathbf{X}$ (middle) form a small proportion of the unobserved ratings $\bar{\mathbf{X}}$. Collaborative filtering aims to exploit the observed ratings $\mathbf{X}$ to determine estimates $\hat{\mathbf{X}}$ (right) of the unobserved ratings. This is a sampling and reconstruction problem, where, unlike conventional formulations, signals of interest are supported in an irregular domain. The ratings $\mathbf{x}^i$ given by all users for a certain item i is a signal on top of the graph $\mathbf{B}$ representing similarities between users. The goal of this paper is to exploit these similarity graphs in reconstructing the estimates $\hat{\mathbf{X}}$.

To center around the item mean we define the average score of item i as $\nu_i := (1/|\mathcal{T}_i|) \sum_{u \in \mathcal{T}_i} Y_{ui}$ and define the item mean centered ratings

$$X^{\mathrm{I}}_{ui} \; := \; Y_{ui} - \nu_i \; = \; Y_{ui} - \big(1/|\mathcal{T}_i|\big) \sum_{u \in \mathcal{T}_i} Y_{ui}. \tag{11.2}$$

We can also center around both the item and user mean by defining the average score of all entries as $\rho := (1/|\mathcal{R}|) \sum_{(u,i) \in \mathcal{R}} Y_{ui}$ to construct the user-item mean centered ratings

$$X^{\mathrm{UI}}_{ui} \; := \; Y_{ui} - \frac{\mu_u \nu_i}{\rho} \; = \; Y_{ui} - \frac{\mu_u \nu_i}{(1/|\mathcal{R}|) \sum_{(u,i) \in \mathcal{R}} Y_{ui}}. \tag{11.3}$$

Distinguishing user, item, and user-item mean centered scores is important in practice but is not conceptually relevant. Throughout Sections 11.2 and 11.3 we use X_{ui} and $\bar{X}_{ui}$ to denote observed and unobserved mean centered scores without making specific assumptions on whether they are centered around user means, item means or user-item means. The distinction is important in the practical implementations in Section 11.4.

Observed mean centered ratings can be collected either into the rating matrix $\mathbf{X} \in \mathbb{R}^{U \times I}$, or into the rating vector $\mathbf{x} = \mathrm{vec}(\mathbf{X}) \in \mathbb{R}^{UI}$. Additionally, vectors $\mathbf{x}_u = [X_{u,1}, ..., X_{u,I}]^\top \in \mathbb{R}^I$ and $\mathbf{x}^i = [X_{1,i}, ..., X_{U,i}]^\top \in \mathbb{R}^U$ represent the ratings by the u-th user and i-th item, respectively. We adopt the convention that $X_{ui} = 0$ for all the ratings that have not been observed, since zero entries in the mean-centered X_{ui} denote lack of information. Analogously, we collect the unobserved mean centered ratings into the matrix $\bar{\mathbf{X}} \in \mathbb{R}^{U \times I}$. Although $\bar{\mathbf{X}}$ is represented as a full matrix, many of its entries are not available. The goal of a recommender system is to compute estimates

$\hat{X}_{ui}$ of the unobserved ratings $\bar{X}_{ui}$. We state this problem for future reference:

Problem 1 (Collaborative Filtering) *Given the observed ratings in the matrix* $\mathbf{X}$*, determine estimates* $\hat{\mathbf{X}}$ *of the full rating matrix* $\bar{\mathbf{X}}$*.*

Problem 1 is a sampling and reconstruction problem which differs from conventional formulations in that the signals of interest are supported in an irregular domain; see Figure 81. If one represents such an irregular domain by a graph, Problem 1 can then be addressed as the sampling and *reconstruction of a graph signal* [63, 64]. Interestingly, ratings $\mathbf{X}$ can be considered as signals either on a graph representing similarity between users, on a graph denoting similarity between items, or, on a graph representing similarity between user-item pairs.

Formally let $\mathbf{B} \in \mathbb{R}^{U \times U}$ be a matrix whose entry B_{uv} measures the similarity between the pair of users (u, v). This similarity matrix can be determined in different ways but one that is commonly used in practice is to compute Pearson correlations restricted to items that have been rated by pairs of users [229]. Specifically, define the set $\mathcal{S}_{uv} := \mathcal{S}_u \cap \mathcal{S}_v$ containing items rated by both users u and v. Further define the mean intersection score $\mu_{uv} = (1/|\mathcal{S}_{uv}|) \sum_{i \in \mathcal{S}_{uv}} Y_{ui}$ and use it to compute the correlation coefficient

$$\Sigma^{\mathrm{U}}_{uv} := (1/|\mathcal{S}_{uv}|) \sum_{i \in \mathcal{S}_{uv}} (Y_{ui} - \mu_{uv})(Y_{vi} - \mu_{uv}). \tag{11.4}$$

The graph $\mathbf{B}$ that measures similarities between user preferences is defined by normalizing via individual user variances,

$$\mathbf{B} := \left[\mathrm{diag}^{-1/2}\left(\boldsymbol{\Sigma}^{\mathrm{U}}\right)\right] \boldsymbol{\Sigma}^{\mathrm{U}} \left[\mathrm{diag}^{-1/2}\left(\boldsymbol{\Sigma}^{\mathrm{U}}\right)\right] - \mathbf{I}, \tag{11.5}$$

where we have subtracted the identity matrix to zero the diagonal elements of $\mathbf{B}$.

Similarly, we can define a matrix $\mathbf{C} \in \mathbb{R}^{I \times I}$ whose C_{ij} entry measures the similarity between the pair of items (i, j). We define the set $\mathcal{T}_{ij} := \mathcal{T}_i \cap \mathcal{T}_j$ containing users that rated both items i and j, the mean intersection score $\nu_{ij} = (1/|\mathcal{T}_{ij}|) \sum_{u \in \mathcal{T}_{ij}} Y_{ui}$ and use it to compute the correlation

$$\Sigma^{\mathrm{I}}_{ij} := (1/|\mathcal{T}_{ij}|) \sum_{u \in \mathcal{T}_{ij}} (Y_{ui} - \nu_{ij})(Y_{ui} - \nu_{ij}). \tag{11.6}$$

The graph $\mathbf{C}$ that measures similarities between items is then defined by [cf. (11.5)]

$$\mathbf{C} := \left[\mathrm{diag}^{-1/2}\left(\boldsymbol{\Sigma}^{\mathrm{I}}\right)\right] \boldsymbol{\Sigma}^{\mathrm{I}} \left[\mathrm{diag}^{-1/2}\left(\boldsymbol{\Sigma}^{\mathrm{I}}\right)\right] - \mathbf{I}, \tag{11.7}$$

The definitions of $\mathbf{B}$ and $\mathbf{C}$ in (11.4) and (11.6) yield symmetric matrices with entries $B_{uv} \in [-1, 1]$ and $C_{ij} \in [-1, 1]$. Alternative choices for B_{uv} and C_{ij} include cosine similarities, adjusted cosine similarities [230], or the percentage of items rated by both u and v [85]. As in the case of mean centered ratings, the specific form of the matrices $\mathbf{B}$ and $\mathbf{C}$ is important in Section 11.4 but is not germane to Sections 11.2 and 11.3.

11.2. NN from a Graph SP Perspective

The main idea behind NN predictors is that when estimating the rating X_{ui}, only the ratings X_{vi} from users v that are highly *similar* to u should be used. To do so, denote $\mathcal{K}_{ui}$ as the set of k users who are the most similar to u (largest value of B_{uv}) and have rated the item i. Leveraging these definitions, the unknown ratings are predicted as

$$\hat{X}_{ui} = \frac{\sum_{v \in \mathcal{K}_{ui}} B_{uv} X_{vi}}{\sum_{v \in \mathcal{K}_{ui}} B_{uv}}. \tag{11.8}$$

As in user-based collaborative filtering, with $\mathcal{K}^{iu}$ denoting the set of k items which are the most similar to i and have been rated by user u, the unknown ratings can be predicted using the item similarity scores as

$$\hat{X}_{ui} = \frac{\sum_{j \in \mathcal{K}^{iu}} C_{ij} X_{uj}}{\sum_{j \in \mathcal{K}^{iu}} C_{ij}}. \tag{11.9}$$

11.2.1. Graph-SP Interpretation

The goal in this section is to show that the user-based (item-based) NN methods presented can be viewed as the application of a *band-stop graph filter* to an input rating signal. To be more precise, let us focus on the generation of $\hat{\mathbf{x}}^i$, i.e., the predicted ratings for item i, using the ratings from other *users* and the similarities among them.

The major step is to construct the user-similarity network, which will serve as the adjacency matrix used in the graph filter. To this end, we start with the matrix $\mathbf{B}$ given in (11.5). Then, in order to account for the fact that ratings from users who do not rate i should not be considered when predicting i, we remove any edges starting from v if X_{vi} is unknown. This implies that the similarity network, which will be denoted as $\mathbf{B}_i$, will depend on the particular item i. The final steps are to keep only the edges corresponding to the k most similar users and normalize each row so that the resultant matrix is right stochastic [cf. the denominator in (11.8)]. Mathematically, this implies that the matrix $\mathbf{B}_i \in \mathbb{R}^{U \times U}$ is defined as

$$[\mathbf{B}_i]_{uv} = \begin{cases} B_{uv} / \sum_{v' \in \mathcal{K}_{ui}} B_{uv'} & \text{if} \quad v \in \mathcal{K}_{ui} \\ 0 & \text{if} \quad v \notin \mathcal{K}_{ui} \end{cases}, \tag{11.10}$$

where we recall that $\mathcal{K}_{ui}$ contains the k users that are the most similar to u and have rated item i. An example of this procedure using the MovieLens 100k dataset is illustrated in Figure 82, where the top network represents the original $\mathbf{B}$ and the subsequent plots represent $\mathbf{B}_i$ for several items, where edge weights are proportional to the width of the edges. For the case of the user-similarity network for item $i = 1$, we observe that the edges starting from u_3 and u_4 that were present in $\mathbf{B}$

are absent in $\mathbf{B}_1$, because users u_3 and u_4 have not rated item i. In the case of $\mathbf{B}_2$ we observe that the edges (u_3, u_2) and (u_4, u_3) present in $\mathbf{B}$ have been removed because $k = 2$ and their respective edge weight was small (their edge widths are small in the visualization).

Once the item-specific adjacency matrix $\mathbf{B}_i$ is defined, the predicted ratings are simply given by

$$\hat{\mathbf{x}}^i = \mathbf{B}_i \mathbf{x}^i, \tag{11.11}$$

cf. (11.8). In words, the estimated ratings are obtained after applying the graph filter $\mathbf{H} = \mathbf{B}_i$ of order one to the input signal $\mathbf{x}^i$.

Prediction in the frequency domain

We now analyze the behavior of (11.11) in the frequency domain, to conclude that $\mathbf{H} = \mathbf{B}_i$ acts a band-stop graph filter. Given a certain item i, consider the eigen-decomposition for the user-similarity network $\mathbf{S} = \mathbf{B}_i = \mathbf{V}_i \mathbf{\Lambda} \mathbf{V}_i^{-1}$. Since the set of diagonalizable matrices is dense and $\mathbf{B}_i$ is constructed from real (noisy) data, matrix $\mathbf{B}_i$ is typically diagonalizable. Even if it were not, one could always add an arbitrarily small perturbation to $\mathbf{B}_i$ and render it diagonalizable. Denote the GFT of the known input signal as $\tilde{\mathbf{x}}^i = \mathbf{V}_i^{-1} \mathbf{x}^i$, and the GFT of the predicted rating as $\tilde{\hat{\mathbf{x}}}^i = \mathbf{V}_i^{-1} \hat{\mathbf{x}}^i$. The two GFTs are related via

$$\tilde{\hat{\mathbf{x}}}^i = \mathbf{V}_i^{-1} \hat{\mathbf{x}}^i = \mathbf{V}_i^{-1} \mathbf{B}_i \mathbf{x}^i = \mathbf{V}_i^{-1} \mathbf{V}_i \mathbf{\Lambda} \mathbf{V}_i^{-1} (\mathbf{V}_i \tilde{\mathbf{x}}^i) = \mathbf{\Lambda} \tilde{\mathbf{x}}^i. \tag{11.12}$$

Therefore, the frequency response of the filter implementing NN is $\text{diag}(\tilde{\mathbf{b}}_i) = \text{diag}(\tilde{\mathbf{h}}) = \mathbf{\Lambda}$ and the p-th frequency coefficient of the predicted output is $[\tilde{\hat{\mathbf{x}}}^i]_p = \lambda_p [\tilde{\mathbf{x}}^i]_p$. Since the selection of the k nearest neighbors introduces some non-symmetries, λ_p will also have an imaginary part. Recall that $\lambda_{\max}(\mathbf{B}_i)$ is always 1 because of right stochasticity and the Perron–Frobenius theorem, and that eigenvectors can be ordered according to $\text{TV}(\mathbf{v}_p) = |\lambda_p - 1|$; see [58] and the related discussion after Definition 1. As a result smooth (low-frequency) eigenvectors are signals where $\|\mathbf{v}_q - \mathbf{B}_i \mathbf{v}_q\|_2 \approx 0$; i.e., full rating signals where users that are similar tend to agree. In practice, the imaginary part of λ_p tends to be small. This is not surprising since $\mathbf{B}_i$ is constructed by removing some of the edges of the (mostly) symmetric matrix $\mathbf{B}$, followed by a renormalization to make it right stochastic. The normalization by a diagonal positive-valued matrix does not introduce an imaginary part. To understand the effect of the edge removal, one can see the eigenvalues of the resultant matrix as those of an originally symmetric matrix (which are always real valued) slightly perturbed with a complex part associated with the potential presence of non-symmetries.

To gain further intuition on the spectral behavior of (11.12), we examine the frequency response of $\mathbf{B}_i$ for the MovieLens 100k database. Specifically, for each $\mathbf{B}_i$, we order its eigenvalues according to $|\lambda_p - 1|$, and record the frequency response for low, middle, and high frequencies. The I frequency responses obtained using this procedure are then averaged across $\mathbf{B}_i$, giving rise to the single transfer function depicted in Figure 84 (a). The horizontal axis represents the index p

of the frequency (assuming that the order satisfies $|\lambda_p - \lambda_{\max}| \leq |\lambda_{p+1} - \lambda_{\max}|$) and the vertical axis denotes the corresponding frequency response of the examined filters. To help visualization, the scale in the horizontal axis is not homogeneous and only the real part of the eigenvalues is shown (the imaginary part is small). The main finding is that the frequency response is zero for more than 90% of the frequencies, implying that the predicted signal will be graph bandlimited. This results from the way the (graph) matrix $\mathbf{B}_i$ is constructed in (11.10): NN graphs typically have many zeros and a clusterizable structure, which leads itself to a low rank matrix representation. Another observation of interest is that the frequencies not rejected by the filter and that are present in the predicted output are the ones associated with the first eigenvectors (low values of p) and the last eigenvectors (high values of p). The first eigenvectors represent signals of small total variation, while the last ones are associated with signals of high variation. Since the diagonal elements of each matrix $\mathbf{B}_i$ are zero, the sum of the eigenvalues is zero, with the eigenvalues associated with low frequencies being positive, and those associated with signals of large total variation being negative. To corroborate this, Figure 85 shows the low-pass and high-pass components of a rating profile $\mathbf{x}$. The low-pass components represent signals where similar users tend to have similar ratings, providing *the big picture* for the predicted rating. Differently, the high-pass component focuses on the differences between users with similar taste for the particular item. With this interpretation one can see (11.12) as a filter that eliminates the irrelevant features (middle frequencies), smoothes out the similar components (low frequencies) and preserves the discriminative features (high frequencies). This band-stop behavior where both high and low graph frequencies are preserved is not uncommon in image processing (image de-noising and image sharpening, respectively) [231], and have been observed in brain signal analytics as well [114, 62].

Item-based NN

Following similar steps, an *item-based* NN predictor can also be casted as a band-stop graph filter. In this case, the input signal is $\mathbf{x}_u \in \mathbb{R}^I$ and the user-specific adjacency matrix $\mathbf{C}_u$ is found after setting to zero the entries in $[\mathbf{C}]_{ij}$ corresponding to items not rated by user u and normalizing each of the rows; see Figure 83. The prediction for user u generated by the item-based NN in (11.9) can be alternatively written as $\hat{\mathbf{x}}_u = \mathbf{C}_u \mathbf{x}_u$. That is, a graph filter of order one with $g_0 = 0$ and $g_1 = 1$. After ordering the eigenvalues of $\mathbf{C}_u$ according to $|\lambda_p - \lambda_{\max}(\mathbf{C}_u)|$, it also holds true that item-based NN has a band-stop frequency response, with eigenvectors associated with small $|\lambda_p|$ being rejected. To assess this behavior in real data, the frequency response of $\mathbf{G} = \mathbf{C}_u$ (averaged across all u) is visualized in Figure 84 (d). There exist certain $\mathbf{C}_u$ with some rows being entirely zero, making their largest eigenvalue being strictly less than 1 in the figure. We show in the next section that we can generalize the idea in collaborative filtering, and design other types of band stop filters with similar frequency response.

11.2.2. Higher Order Graph Filters

Since NN predictors can be understood as the implementation of a band-stop graph filter of order one, our proposal here is, using $\mathbf{S} = \mathbf{B}_i$ as the shift, to design other types of *band-stop* graph filters $\mathbf{H}(\mathbf{S})$ to perform rating predictions. Consider first $\mathbf{H} = \mathbf{B}_i^2$, whose frequency response is $\text{diag}(\tilde{\mathbf{h}}) = \mathbf{\Lambda}^2$. The fact of $\mathbf{B}_i$ being a band-stop filter implies that many of the entries of its frequency response $\tilde{\mathbf{h}}_i$ are zero. As a result, $\mathbf{B}_i^2$ has a band-stop behavior too and the same holds true for any positive power of $\mathbf{B}_i$. Figure 86 (a) shows this for the filter $\mathbf{H} = \mathbf{B}_i^2$ and $\mathbf{H} = \mathbf{B}_i^3$ corresponding to the MovieLens 100k database. Since all powers of $\mathbf{B}_i$ are band-stop operators, the unknown ratings predicted with graph filters of the form

$$\hat{\mathbf{x}}^i = \mathbf{H}\mathbf{x}^i \;\; \text{with} \;\; \mathbf{H} = \sum_{l=0}^{L} h_l \mathbf{B}_i^l, \tag{11.13}$$

will also give rise to bandlimited signals. Hence, predictions in (11.13) are generalizations of the traditional NN method in (11.11), which estimates $\hat{\mathbf{x}}_i$ using a filter $\mathbf{H} = \mathbf{B}_i$ of order one. This graph-frequency interpretation can be complemented by understanding the effect of $\mathbf{B}_i$ on the graph vertex domain. To do so, note that $\mathbf{B}_i^0\mathbf{x}^i = \mathbf{x}^i$ coincides with the original signal, $\mathbf{B}_i\mathbf{x}^i$ is an average of the ratings given by one-hop neighbors, $\mathbf{B}_i^2\mathbf{x}^i$ is an average of elements in nodes that interact via intermediate common neighbors, and, in general, $\mathbf{B}_i^l\mathbf{x}^i$ describes interactions between l-hop neighbors. Therefore, on top of relying on the ratings of highly similar users to make predictions, the powers of the matrix $\mathbf{B}_i^l$ in the graph filter in (11.13) also account for chains of users with similar taste, exploiting them to generate enhanced predictions.

Compared to classical NN, the filter coefficients $\mathbf{h}$ are not known a priori, and therefore need to be learned from a training set. Moreover, h_0 is irrelevant since $\mathbf{B}_i^0\mathbf{x}^i = \mathbf{x}^i$ and therefore would not be helpful in predictions. Then, the filter coefficients are found by solving

$$\min_{\mathbf{h}} \sum_{(u,i)\in\mathcal{R}} \left| \left[\left(\sum_{l=1}^{L} h_l \mathbf{B}_i^l \right) \mathbf{x}^i \right]_u - X_{ui} \right|^2 . \tag{11.14}$$

where we recall that $\mathcal{R}$ contains the indexes of the available ratings. To avoid overfitting, (11.14) can be augmented with a regularizer

$$\min_{\mathbf{h}} \sum_{(u,i)\in\mathcal{R}} \left| \left[\left(\sum_{l=1}^{L} h_l \mathbf{B}_i^l \right) \mathbf{x}^i \right]_u - X_{ui} \right|^2 + r\|\mathbf{h}\|_2^2, \tag{11.15}$$

where r is a regularization parameter that can be tuned by cross-validation on the training set. Note that formulations in (11.14) and (11.15) are least squares problems that, using the Moore–Penrose pseudo inverse, admit "closed form" solutions.

Mimicking the previous steps, we can use the following generalized *item-based* NN scheme to predict user ratings

$$\hat{\mathbf{x}}_u = \mathbf{G}\mathbf{x}_u, \text{ with } \mathbf{G} = \sum_{q=0}^{Q} g_q \mathbf{C}_u^q. \tag{11.16}$$

The filter above has more degrees of freedom than its order-one counterpart in Section 11.2.1, and accounts for *chains* of similar items to improve prediction accuracy. The vector of filter coefficients **g** is found by solving

$$\min_{\mathbf{g}} \sum_{(u,i)\in\mathcal{R}} \left| \left[\left(\sum_{q=1}^{Q} g_q \mathbf{C}_u^q \right) \mathbf{x}_u \right]_i - X_{ui} \right|^2 + r\|\mathbf{g}\|_2^2. \tag{11.17}$$

in the training set.

Remark 15 (Selecting the filter order) An important task when designing the graph filters in (11.15) and (11.17) is to decide the number of taps L and Q. Large values of L and Q can capture higher-order dependencies and reduce the error on the training set, but they increase the complexity of the training and prediction phases as well as the the risk of overfitting. From an optimization point of view, a common way to deal with this tradeoff is to set a large value for L and Q and augment the cost with a regularizer penalizing the ℓ_1 norm of the filter coefficients. Such a norm serves as a proxy of the ℓ_0 pseudo norm (which adds a fixed cost for each new coefficient that is activated), is convex, and comes with advanced solvers that allow for an efficient (low cost) optimization [232]. Better (more accurate) non-convex surrogates can be used, at the expense of increasing the complexity of the training phase. Alternatively, one can cast the optimization from a statistical perspective and modify the formulation to account for the complexity of the model using criteria such as the Akaike information criterion (AIC) or the Bayesian information criterion (BIC); see, e.g., [233].

11.2.3. Mirror Filtering

The graph filter in (11.13) relies on chains of up to L users with similar taste to make a prediction for a certain item. The same holds true for the estimator in (11.16), which in this case exploits chains of items. Since the original collaborative filtering (Problem 1) depends both on users and items, combining these two filters is a natural generalization. One can think of first passing the rating matrix **X** through a filter **H** on user-similarity and then passing the filtered rating matrix **HX** through another filter **G** on item-similarity to yield the final prediction $\mathbf{H}\mathbf{X}\mathbf{G}^\top$. We term such cascade of filters as Mirror Filtering (MiFi). Note that the particular order in which the filters are applied is inconsequential, since matrix multiplication is associative. More formally, we propose MiFi as a generalization of the NN method to operate in the joint graph. The predicted ratings are obtained as

$$\hat{X}_{ui} = \left[\mathbf{H}\mathbf{X}\mathbf{G}^\top\right]_{ui}, \text{ with}$$
$$\mathbf{H} = \sum_{l=0}^{L} h_l \mathbf{B}_i^l \text{ and } \mathbf{G} = \sum_{q=0}^{Q} g_q \mathbf{C}_u^q. \tag{11.18}$$

In contrast with the previous section, the filter coefficients h_0 and g_0 will play an important role in (11.18). The reason being that setting $h_0 = g_0 = 0$ would force all the terms of the form $\mathbf{B}_i^l\mathbf{X}\mathbf{C}_u^\top$ to depend on both $\mathbf{B}_i$ and $\mathbf{C}_u$, preventing the possibility of having terms of the form $\mathbf{B}_i^l\mathbf{X}$ or $\mathbf{X}(\mathbf{C}_u^q)^\top$. At an intuitive level, MiFi uses ratings X_{vj} to predict $\hat{X}_{ui}$ if either exists a path of similar users from v to u or a path or similar items from j to i. The cost is that more filter coefficients $L+Q$ need to be found.

To that end, the training problem that yields the filter coefficients $\mathbf{h}$ and $\mathbf{g}$ is

$$\min_{\mathbf{h},\mathbf{g}} \sum_{(u,i)\in\mathcal{R}} \left| \left[\left(\sum_{l=0}^{L} h_l \mathbf{B}_i^l \right) \mathbf{X} \left(\sum_{q=0}^{Q} g_q \mathbf{C}_u^q \right)^\top \right]_{ui} - X_{ui} \right|^2 + r\left(\|\mathbf{h}\|_2^2 + \|\mathbf{g}\|_2^2 \right). \tag{11.19}$$

Preprocessing steps can also be applied to compute efficiently $\mathbf{B}_i^l\mathbf{X}\mathbf{C}_u^q$ for the user-item pairs (u,i) of interest and different orders l and q. The main difference with respect to the formulations in (11.15) and (11.17) is that the problem in (11.19) is bilinear on $\mathbf{h}$ and $\mathbf{g}$. For a fixed $\mathbf{h}$, (11.19) is a least squares problem on $\mathbf{g}$, and the same holds true for a fixed $\mathbf{g}$. Hence, a natural approach is to use a computationally efficient alternating least squares scheme to (approximately) solve (11.19). Initialization plays an important role for alternating least squares problems. Let $\mathbf{h}^{(0)}$ be the optimal coefficients when $\mathbf{g}$ is set as $g_0 = 1$ and $g_q = 0$ for $q \neq 0$; similarly, let $\mathbf{g}^{(0)}$ be the optimal coefficients when $\mathbf{h}$ is set as $h_0 = 1$ and $h_l = 0$ for $l \neq 0$. Our simulations show that using $\mathbf{h}^{(0)}$ or $\mathbf{g}^{(0)}$ as initial points, leads to solutions with good prediction performance.

An alternative to bypass the bilinearity on the filter coefficients is to reformulate (11.19) using the vectorial representation of the ratings. To this end, note first that $(\sum_{l=0}^{L} h_l\mathbf{B}_i^l)\mathbf{X}(\sum_{q=0}^{Q} g_q\mathbf{C}_u^q)^\top$ can be rewritten as $(\sum_{q=0}^{Q} g_q\mathbf{C}_u^q) \otimes (\sum_{l=0}^{L} h_l\mathbf{B}_i^l)\mathbf{x}$. Defining now the Kronecker shift as $\mathbf{S}_{ui} := \mathbf{C}_u \otimes \mathbf{B}_i$, it can be shown that our MiFi can be written as a graph filter on $\mathbf{S}_{ui}$ and, hence, we can estimate the filter coefficients $\mathbf{h}^\otimes$ that minimize

$$\min_{\mathbf{h}^\otimes} \sum_{r\in\mathcal{R}} \left| \left[\left(\sum_{l=0}^{L^\otimes} h_l^\otimes \mathbf{S}_{ui}^l \right) \mathbf{x} \right]_r - x_r \right|^2 + r\|\mathbf{h}^\otimes\|_2^2, \tag{11.20}$$

where $r \in \{1,\ldots,U\} \times \{1,\ldots,I\}$ represents the index of a certain user-item pair. The problem above is not bilinear, and has a similar form as those in Section 11.2.2. Note, however, that the number of coefficients $L^\otimes$ can be significantly larger than $L+Q$. Since the filter coefficients must be learned during the training phase, the increase in dimensionality can lead to problems related to estimation accuracy and overfitting. Simulations will corroborate this point for the MovieLens 100k database, showing that the predictor based on (11.19) enjoys a estimation performance superior to that of (11.20) .

Remark 16 As done in classical NN methods, we considered graph filters that use shift operators $\mathbf{B}_i$ and $\mathbf{C}_u$. Such shifts have been adjusted according to the item and user of interest, and normalized to make them stochastic (normalization-then-filtering). An alternative generalization can be carried out by using the correlation matrices $\mathbf{B}$ and $\mathbf{C}$ as shifts, so that the prediction filters are $\sum_l h_l \mathbf{B}^l$ and $\sum_q g_q \mathbf{C}^q$. This would require performing adjustment and normalization after filtering (filtering-then-normalization). We focused on the first formulation because it is closer to the (original) setup considered in NN methods and gives rise to better prediction accuracy. In any case, under minor modifications our algorithms can also handle the second formulation.

11.3. LF from a Graph-SP Perspective

Latent factor models try to approximate the rating user-item matrix by identifying a few latent features (factors) and then characterize both users and items in terms of these factors. In the linear case, this amounts to project the original user ratings $\mathbf{x}_u$ and item ratings $\mathbf{x}^i$ into a feature vector space of dimension F. The ratings are then obtained as inner products in such a space. Since in collaborative filtering the features are not known a priori but learned from the data, the SVD decomposition plays a key role in identifying the latent factors as well as the underlying mapping. To be more specific, let us rely on the SVD factorization to write the rating matrix as $\mathbf{X} = \mathbf{W}\text{diag}(\sigma)\mathbf{Z}^\top$. Next, use the singular vectors to define $\boldsymbol{\phi}_u(\mathbf{W})$ (the u-th row of $\mathbf{W}$) as the *feature* profile of user u and $\boldsymbol{\phi}_i(\mathbf{Z})$ (the i-th row of $\mathbf{Z}$) as the feature profile of item i. With these notational conventions, any rating X_{ui} can be obtained as $X_{ui} = \sum_{f=1}^{F} \sigma_f [\boldsymbol{\phi}_u(\mathbf{W})]_f [\boldsymbol{\phi}_i(\mathbf{Z})]_f$, with the f-th singular value σ_f representing the weight of factor f in explaining the rating. While the value of F is related to the rank of the rating matrix, in real scenarios one expects F to be small, so that $\mathbf{X}$ is (or can be approximated as) a low-rank matrix.

Low-rankness can be achieved by computing the SVD and keeping only the largest singular values. A major hurdle to implement this approach is that the computation of the SVD requires full knowledge of $\mathbf{X}$, but in the context of collaborative filtering only the rating values in $\mathcal{R}$ are known. This implies that one must solve instead the problem of minimizing rank($\hat{\mathbf{X}}$) subject to $\hat{X}_{ui} = X_{ui}, \ \forall \ (u,i) \in \mathcal{R}$. Since the rank function renders the problem non-convex, a widely used approach is to relax the rank using the nuclear-norm [107] and set $\hat{\mathbf{X}}$ as the solution to

$$\begin{aligned} \min_{\hat{\mathbf{X}}} \quad & \|\hat{\mathbf{X}}\|_* \\ \text{s.t.} \quad & |\hat{X}_{ui} - X_{ui}|^2 \leq \epsilon, \quad \forall \ (u,i) \in \mathcal{R}, \end{aligned} \tag{11.21}$$

where a tolerance ϵ to account for potential observation noise in the known ratings has been included too. Note also that an additional advantage of (11.21) is that it can be easily augmented to incorporate additional sources of (a priori) information [83].

11.3.1. Searching Sparse Frequency Coefficients

The underlying idea behind LF is to model the ratings as a matrix $\mathbf{X} = \mathbf{W}\text{diag}(\sigma)\mathbf{Z}^\top$ with rank $F = \|\sigma\|_0$ small. The goal in this section is to interpret LF from a graph SP perspective, showing that LF predictors also give rise to bandlimited signals.

To that end, let us consider the graph shift operators given by the user-to-user and item-to-item covariance matrices

$$\mathbf{D} = \frac{1}{I}\mathbf{X}\mathbf{X}^\top \;\text{ and }\; \mathbf{E} = \frac{1}{U}\mathbf{X}^\top\mathbf{X}. \tag{11.22}$$

Since the shifts $\mathbf{S} = \mathbf{D}$ and $\mathbf{S} = \mathbf{E}$ are symmetric, they admit an orthonormal eigen-decomposition as

$$\mathbf{D} = \mathbf{V}\mathbf{\Lambda_D}\mathbf{V}^\top \;\text{ and }\; \mathbf{E} = \mathbf{U}\mathbf{\Lambda_E}\mathbf{U}^\top. \tag{11.23}$$

For the shift $\mathbf{S} = \mathbf{D}$, the frequency representation of the ratings for a given item $\mathbf{x}^i$ is therefore given by $\tilde{\mathbf{x}}^i = \mathbf{V}^\top\mathbf{x}^i$ and the matrix collecting the frequency representations for all item signals is $[\tilde{\mathbf{x}}^1, ..., \tilde{\mathbf{x}}^I] = \mathbf{V}^\top\mathbf{X}$. Similarly, for $\mathbf{S} = \mathbf{E}$ the frequency representation of $\mathbf{x}_u$ is given by $\tilde{\mathbf{x}}_u = \mathbf{U}^\top\mathbf{x}_u$, with matrix $[\tilde{\mathbf{x}}_1, ..., \tilde{\mathbf{x}}_U] = \mathbf{U}^\top\mathbf{X}^\top$ collecting the frequency signal for all users. Notice that the two GFTs can be applied jointly, giving rise to the joint (two-dimensional) frequency representation of the user-item rating matrix as

$$\tilde{\mathbf{X}} = \mathbf{V}^\top\mathbf{X}\mathbf{U}. \tag{11.24}$$

The previous matrix can be vectorized to yield

$$\tilde{\mathbf{x}} = \text{vec}\left(\mathbf{V}^\top\mathbf{X}\mathbf{U}\right) = \mathbf{U}^\top \otimes \mathbf{V}^\top\mathbf{x}. \tag{11.25}$$

The equation shows that the if the ratings are expressed in the vector form $\mathbf{x}$, the unitary matrix $\mathbf{U}^\top \otimes \mathbf{V}^\top$ represents the associated GFT. This also implies that one can view $\mathbf{x} \in \mathbb{R}^{UI}$ as a signal defined on a shift $\mathbf{S} \in \mathbb{R}^{UI\times UI}$ that has $\mathbf{U} \otimes \mathbf{V}$ as eigenvectors. Two natural choices for such a shift are $\mathbf{S} = \mathbf{E} \otimes \mathbf{D}$, which is the Kronecker graph product graph of the two shifts in (11.22), and $\mathbf{S} = \mathbf{E} \otimes \mathbf{I} + \mathbf{I} \otimes \mathbf{D}$, the Cartesian graph product of the shifts in (11.22) illustrated in Figure 87. For further details on graph SP and product graphs, we refer readers to [58].

Since LF approaches try to approximate the ratings matrix using the SVD factorization $\mathbf{X} = \mathbf{W}\text{diag}(\sigma)\mathbf{Z}^\top$ with $F = \|\sigma\|_0$ small, from the expressions in (11.22)-(11.24) it follows readily that:

(i) The eigenvectors of $\mathbf{D}$ and $\mathbf{E}$ are the singular vectors of $\mathbf{X}$.

(ii) The corresponding GFTs are $\mathbf{V}^\top = \mathbf{W}^\top$ and $\mathbf{U}^\top = \mathbf{Z}^\top$.

(iii) The matrix collecting the frequency coefficients in (11.24) is $\tilde{\mathbf{X}} = \text{diag}(\sigma)$.

Therefore, since $\tilde{\mathbf{X}}$ is diagonal with rank F, *vector* $\tilde{\mathbf{x}} \in \mathbb{R}^{UI}$ in (11.25) *will have at most* F non-zero entries, with $F \ll UI$. In words, LF approaches are essentially *modeling the ratings* $\mathbf{x}$ *as* a signal that is *bandlimited* in a frequency domain associated with the product graph of the *covariance shifts* given in (11.22).

From a practical point of view, it is important to notice that since only the subset of ratings in $\mathcal{R}$ is known, unless a priori information exists, the covariance matrices in (11.22) cannot be computed beforehand. This implies that one can view the LF formulation in (11.21) as a way to search jointly for a sparse $\tilde{\mathbf{x}}$ as well as for the eigenvectors of the user-to-user and item-to-item covariance matrices. Alternatively, one can try to infer first the covariance graphs, obtain its eigenvectors, and then look for the spectral coefficients. This is the approach followed in the ensuing section, but first a remark is in order.

Remark 17 The discussion in this section revealed that LF algorithms that predict the ratings by minimizing the rank of matrix $\hat{\mathbf{X}}$ are implicitly viewing the columns of $\hat{\mathbf{X}}$ as bandlimited signals defined on the user-to-user covariance graph $\mathbf{S} = \mathbf{D}$ and, likewise, the rows of $\hat{\mathbf{X}}$ as bandlimited signals defined on the item-to-item covariance graph $\mathbf{S} = \mathbf{E}$. The key to notice this fact is that the low-rank matrix completion algorithm aims at expressing the ratings as combinations of a few of the columns of $\mathbf{W}$ and $\mathbf{Z}$, which are the eigenvectors (graph frequencies) of $\mathbf{D}$ and $\mathbf{E}$. Interestingly, this interpretation also holds for any other graph $\mathbf{S}$ with the same eigenvectors than $\mathbf{D}$ and $\mathbf{E}$. Those include all the powers (polynomials) of $\mathbf{D}$ and $\mathbf{E}$. In the network topology inference literature, the two more widely used choices in statistics and machine learning are the covariance graph itself as well as its inverse, the precision graph.

11.3.2. Sampling Bandlimited Graph Signals

Section 11.3.1 revealed that, when interpreted from a graph SP perspective, LF methods build on two fundamental assumptions: (i) the rating signal $\mathbf{x}$ is bandlimited, and (ii) $\mathbf{x}$ is defined on top of a graph which can be obtained by combining the user-to-user and item-to-item covariance matrices.

As a result, one can reinterpret the prediction carried out by an LF scheme as a sampling and reconstruction problem (cf. Problem 1): given $\mathbf{x}$, the goal is to recover the full signal $\bar{\mathbf{x}}$; and the key assumption is that $\bar{\mathbf{x}}$ is bandlimited and can be written as a linear combination of a few columns of $\mathbf{U} \otimes \mathbf{V}$. To be rigorous, recall that (11.25) stated that $\tilde{\mathbf{x}} = \mathbf{U}^\top \otimes \mathbf{V}^\top \mathbf{x}$ and we have that $\tilde{\mathbf{x}} = \text{vec}(\tilde{\mathbf{X}}) = \text{vec}(\text{diag}(\boldsymbol{\sigma}))$. Since only F of the singular values in $\boldsymbol{\sigma}$ are non-zero, the support of $\tilde{\mathbf{x}}$ will have cardinality F and correspond to the F first diagonal elements of $\tilde{\mathbf{X}}$. This readily implies that only F frequencies will be active, those corresponding the Kronecker product of the f-th column of $\mathbf{U}$ with the f-th column of $\mathbf{V}$. Using $\odot$ to denote the Khatri-Rao product and $\mathbf{U}_\text{F}$ to represent the first F columns of $\mathbf{U}$, we then have

$$\bar{\mathbf{x}} = (\mathbf{U}_\text{F} \odot \mathbf{V}_\text{F})\tilde{\mathbf{x}}_\text{F}, \text{ with } \tilde{\mathbf{x}}_\text{F} \in \mathbb{R}^F. \tag{11.26}$$

If the eigenvectors are known, the procedure is clear [63, 64]: Given $|\mathcal{R}| \geq F$ samples of $\mathbf{x}$, invert (11.26) to estimate $\tilde{\mathbf{x}}_{\mathrm{F}}$, and then use the estimate $\hat{\tilde{\mathbf{x}}}_{\mathrm{F}}$ to recover the full rating signal as $\hat{\mathbf{x}} = (\mathbf{U}_{\mathrm{F}} \odot \mathbf{V}_{\mathrm{F}})\hat{\tilde{\mathbf{x}}}_{\mathrm{F}}$. This interpretation is useful not only to come up with new reconstruction schemes, but also to apply results from sampling of graph signals to recommender systems. For example, different papers have shown that when the number of observations is small, the set of sampled nodes plays a critical role on recovery performance [63, 64]. Consequently, schemes to select the nodes to be sampled have been developed [63]. This will amount to identifying user-item pairs that, if known, would contribute to increase the prediction performance. In this context, one can envision active sampling schemes where some users are exposed to particular items so that the overall prediction of the recommendation systems improves.

The rest of the section is devoted to leverage the interpretation to present different alternatives for the LF prediction. The main problem to implement the described approach is that the full covariances $\mathbf{D}$ and $\mathbf{E}$ in (11.22), which give rise to $\mathbf{V}$ and $\mathbf{U}$, are not known. If the available ratings are uniformly distributed across users and items, a simple strategy is to replace the covariances in (11.22) with the approximations $\mathbf{\Sigma}^{\mathrm{U}}$ and $\mathbf{\Sigma}^{\mathrm{I}}$ presented in (11.4) and (11.6), which rely only on the ratings that are known. Let $\hat{\mathbf{V}}$ and $\hat{\mathbf{U}}$ represent the approximated eigenvectors, the recovery problem to solve in this case is

$$\begin{aligned} &\text{find} \quad \tilde{\mathbf{x}}_{\mathrm{F}} \\ &\text{s.t.} \quad \left|\left[\hat{\mathbf{U}}_{\mathrm{F}} \odot \hat{\mathbf{V}}_{\mathrm{F}}\tilde{\mathbf{x}}_{\mathrm{F}}\right]_r - x_r\right|^2 \leq \epsilon, \quad \forall\, r \in \mathcal{R}, \end{aligned} \tag{11.27}$$

where the main difference is that the solution in the sampled set is not forced to coincide with the original observations.

Another interesting alternative is to enlarge the set of active frequencies both inside and outside the diagonal of $\tilde{\mathbf{X}}$. Suppose first that the frequency support $\mathcal{F}$, is small and known, the formulation is then

$$\begin{aligned} &\text{find} \quad \tilde{\mathbf{x}} \\ &\text{s.t.} \quad \left|\left[\hat{\mathbf{U}} \otimes \hat{\mathbf{V}}\tilde{\mathbf{x}}\right]_r - x_r\right|^2 \leq \epsilon, \quad \forall\, r \in \mathcal{R}, \\ &\qquad \tilde{x}_f = 0, \quad f \notin \mathcal{F}, \end{aligned} \tag{11.28}$$

and the predicted ratings are simply $\hat{\mathbf{x}} = (\hat{\mathbf{U}} \otimes \hat{\mathbf{V}})\tilde{\mathbf{x}}^*$, with $\tilde{\mathbf{x}}^*$ being the solution to (11.28). If $\mathcal{F}$ is not known, a regularizer $\|\tilde{\mathbf{x}}\|_0$ penalizing the number of nonzero coefficients can be added to the optimization. Since the ℓ_0 norm is non-convex, the convex surrogate $\|\tilde{\mathbf{x}}\|_1$ is used instead to yield

$$\begin{aligned} &\min_{\tilde{\mathbf{x}}} \quad \|\tilde{\mathbf{x}}\|_1 \\ &\text{s.t.} \quad \left|\left[\hat{\mathbf{U}} \otimes \hat{\mathbf{V}}\tilde{\mathbf{x}}\right]_r - x_r\right|^2 \leq \epsilon, \quad \forall\, r \in \mathcal{R}. \end{aligned} \tag{11.29}$$

The above optimization is a classical sparse recovery problem whose performance depends on the number of observed ratings $|\mathcal{R}|$, the tolerance ϵ, as well as on the properties (including coherence) of the so-called sensing matrix $\hat{\mathbf{U}} \otimes \hat{\mathbf{V}}$ [234].

The last algorithm in this section is inspired by NN. While LF schemes are implicitly considering the same user-to-user and item-to-item shifts for all the predictions, NN schemes use a different graph for each user and item. This can be incorporated to the current sampling setup. Suppose that the focus is on predicting $x_r = X_{ui}$. Then, let $\mathbf{V}_i$ and $\mathbf{U}_u$ be the eigenvectors of $\mathbf{B}_i$ and $\mathbf{C}_u$, respectively, and consider the problem [cf. (11.28)]

$$\begin{aligned} \text{find} \quad & \tilde{\mathbf{x}} \\ \text{s.t.} \quad & \left| \left[\mathbf{U}_u \otimes \mathbf{V}_i \tilde{\mathbf{x}} \right]_r - x_r \right|^2 \leq \epsilon, \quad \forall\, r = (u,i) \in \mathcal{R}, \\ & \tilde{x}_f = 0, \quad f \notin \mathcal{F}, \end{aligned} \tag{11.30}$$

The main difficulty with this approach is that (11.30) needs to be solved for every r in $\mathcal{R}$. On the positive side, since matrices $\mathbf{B}_i$ and $\mathbf{C}_u$ are very sparse and only a few of their eigenvectors are required (the ones associated with the largest eigenvalues), those eigenvectors can be found efficiently.

Summarizing, by reinterpreting LF predictors as the recovery of a bandlimited graph signal from a subset of samples, a number of novel prediction algorithms have been proposed in this section. All the considered algorithms proceed in two steps. Firstly, the user-to-user and item-to-item networks are built and their eigenvectors are found. Secondly, using those eigenvectors as input, the prediction is formulated as a sparse recovery problem. The different algorithms correspond to different ways to build the similarity shifts as well as different formulations of the sparse recovery. Clearly, alternative definitions for the shifts and modifications in the sparse optimization are also worth considering, but left as future work.

11.4. Numerical experiments

The main purpose of this section is to illustrate how the proposed methods (especially those in Sections 11.2.2 and 11.2.3) improve the rating accuracy in real data. For that purpose we use the MovieLens 100k dataset [108], which contains ratings from 943 users on 1,682 movies. The number of available ratings is 100,000, i.e., the 6.3% of the total number of user-item pairs. The values for the movie ratings belong to the *discrete* set $\{1,2,3,4,5\}$. The main focus of the analysis is on band-stop filters and MiFi, since they will give rise to the best performance. We randomly select 100 ratings as the testing set, and use the rest as training set. The sets containing the indexes of elements in testing sets and training sets are denoted as $\mathcal{R}_{\text{ts}}$ and $\mathcal{R}_{\text{tr}}$, respectively. The networks and filter coefficients are only trained on the training set. As a performance metric, we use the global root mean squared error (RMSE) between the actual ratings and predicted ratings in the testing set, i.e.,

$$\text{RMSE} = \sqrt{\textstyle\sum_{(u,i) \in \mathcal{R}_{\text{ts}}} \left| Y_{ui} - \hat{Y}_{ui} \right|^2 \Big/ |\mathcal{R}_{\text{ts}}|}, \tag{11.31}$$

where Y_{ui} is the available non-centered rating and $\hat{Y}_{ui}$ is the reconstructed rating scaled back by adding mean. We will specify the mean centering procedure undertaken for each of the algorithms. User-based NN, item-based NN, and conventional matrix completion are used as benchmark algorithms. To get an estimate for the regularization constant r used in (11.15), (11.17), and (11.19), we perform cross-validation by splitting the ratings in the training set into ten equally sized subsets.

11.4.1. Complexity Analysis

Before evaluating the performance, we start by analyzing the computational complexity associated with the proposed methods, and compare it with that of other traditional methods in rating prediction. Recall that U denotes the number of users, I the number of items, $|\mathcal{R}|$ the total observed ratings, and k the number of most similar users (items) used to construct the graphs [cf. (11.8) and (11.9)]. Moreover, we use β to denote the *average* number of items (users) rated by a pair of users (items) [cf. (11.10)]. Without loss of generality we suppose that $U > I$ and note that, in practice, we have that $|\mathcal{R}| \sim O(U)$ and, therefore, $|\mathcal{R}| \ll UI$. Though there are more items than users in the MovieLens 100k dataset examined in the paper, for massive dataset it is more common to have $U > I$, e.g. Netflix Prize.

Traditional user-based NN methods involve two steps: i) a pre-computation step to obtain the correlation matrix $\mathbf{B}$ in (11.5), and ii) a prediction step to find the unknown ratings [cf. (11.8)]. The computation of $\mathbf{B}$ requires U^2 evaluations of the expression in (11.4), each of them with an average complexity of $O(\beta)$; hence the overall complexity of this step is $O(\beta U^2)$. The prediction step in (11.8) has a complexity $O(k)$ and needs to be run U times (one per user), resulting in an overall complexity of $O(kU)$. Since $k \ll U$, the joint complexity of the two steps is $O(\beta U^2)$. Item-based NN methods have a similar complexity.

The user-based high-order NN graph filters proposed in Section 11.2.2 involve three steps: i) a pre-computation step to obtain $\mathbf{B}_i$ in (11.10), ii) a training step to find the filter coefficients [cf. (11.15)], and iii) a prediction step to find the unknown ratings using a graph filter. [cf. (11.11)].

- For the pre-computation step, after obtaining of $\mathbf{B}$ with a complexity of $O(U^2\beta)$, the normalization in (11.10) needs to be implemented for all I items. Since for each row of matrix for the i-th item, we need to adjust the value in k entries, the resultant complexity is $O(IUk)$. Therefore, the pre-computation step has overall complexity of $O(U^2\beta + UIk) \sim O((\beta + k)U^2)$, comparable with that of NN.
- For the prediction step $\sum_{l=1}^{L} h_l \mathbf{B}_i^l \mathbf{x}^i$, notice that the sparse matrix $\mathbf{B}_i$ has to be applied to the input data L times and then implement the linear combination dictated by $\{h_l\}_{l=1}^{L}$. As explained for the case of traditional NN methods, each application of $\mathbf{B}_i$ involves $O(kU)$ operations, so that the overall cost is $O(kLU)$, L times larger than that of NN and with L being typically small.
- For the training step, we first evaluate $[\mathbf{B}_i^l \mathbf{x}^i]_u$ for all l and $\mathbf{x}^i \in \mathcal{R}$, and then solve the minimization problem in (11.14). Since $\mathbf{B}_i$ is sparse and we are only interested in the u-th entry of $\mathbf{B}_i^l \mathbf{x}^i$, the evaluation across all ratings has complexity $O(Lk|\mathcal{R}|)$. The least squares

minimization has L variables and $|\mathcal{R}|$ observations, so that its complexity is $O(L^2|\mathcal{R}|)$. As a result, the overall complexity in the training phase is $O(L(L+k)|\mathcal{R}|)$, linear in the number of observed ratings.

Overall, the bottleneck is in the pre-computation step, so that the complexity is $O(U^2\beta + UIk) \sim O((\beta+k)U^2)$, which is in the same order than that of NN. To have a better sense of these values in practical applications, for simulations shown in the subsequent sections we have that $U = 943$, $I = 1682$), and the values of k are set to 40 or less. Following the same analysis, the pre-computation step for Mi-Fi predictors has complexity $O(\beta U^2 + \beta I^2 + kUI) \sim O((\beta+k)U^2)$, the prediction step has complexity $O(Lk(U+I)) \sim O(LkU)$, and the training step has complexity $O(L(L+k)|\mathcal{R}|)$, where the evaluation of $(\sum_l h_l \mathbf{B}_i^l)\mathbf{X}(\sum_q g_q \mathbf{C}_u^q)$ across ratings $(u,i) \in \mathcal{R}$ has complexity $O(Lk|\mathcal{R}|)$ and the least square minimization has $2L$ variables and hence $O(L^2|\mathcal{R}|)$ complexity. This means that the complexity is still linear in the number of observed ratings. Again, the aggregated complexity is $O(U^2\beta + I^2\beta + UIk) \sim O((\beta+k)U^2)$, the same as that of NN.

The traditional low-rank matrix completion via semidefinite programming has complexity $O(\max\{U,I\}^4) \sim O(U^4)$ [55] and [235, Section 11.8]. The bandlimited LF model proposed in Section 11.3.2 requires running three steps: i) computing the covariance matrix $\mathbf{D}$ or $\mathbf{E}$ [cf. (11.22)], ii) obtaining of the eigenvectors $\mathbf{V}$ and $\mathbf{U}$ [cf. (11.23)], and iii) finding the frequency coefficients [cf. (11.34)]. Obtaining the covariance matrices and their eigenvectors has complexity $O(\max\{U,I\}^3) \sim O(U^3)$; lower complexity is possible with more advanced methods or by considering sparsity, but we examine the most straightforward implementation here. The problem in (11.34) has dimension UI and therefore has empirical complexity of $O((UI)^{1.2})$ [232]. Therefore, the bandlimited sampling proposed in Section 11.3.2 has an overall complexity of $O(\max\{U,I\}^3) \sim O(U^3)$, which is an order of magnitude smaller than that of matrix completion.

11.4.2. High-Order NN Graph Filters

We now look at user-based band-stop graph filters. User mean centering as in (11.1) is used to remove biases of different users. Also, since the quality of the constructed graph in (11.4) and (11.5) depends on the cardinality of $|\mathcal{S}_{uv}|$, we set $B_{uv} = 0$ for any pair of users such that $|\mathcal{S}_{uv}|$ is smaller than a threshold $\underline{\mathcal{S}} = 2$. Before we start to compare different approaches, the first task is to assess the best performance that one can achieve in the setup at hand. To this end, we use the networks $\mathbf{B}_i$ learned on the training set and learn the filter coefficients by solving the least squares problem in (11.15) using not the training but the *testing* set

$$\min_{\mathbf{h}} \sum_{(u,i)\in\mathcal{R}_{\text{ts}}} \left| \left[\left(\sum_{l=1}^{L} h_l \mathbf{B}_i^l \right) x^i \right]_u - X_{ui} \right|^2 + r\|\mathbf{h}\|_2^2. \tag{11.32}$$

Since the coefficients above are biased towards the data in $\mathcal{R}_{\text{ts}}$ and all other schemes will be trained using $\mathcal{R}_{\text{tr}}$, the performance achieved by (11.32) on $\mathcal{R}_{\text{ts}}$ will serve as a benchmark for all other schemes. The RMSE across ratings in the testing set $\mathcal{R}_{\text{ts}}$ using the $\mathbf{h}$ trained in (11.32) for different values of L and r is presented in Table 1. There are several interesting observations.

Table 1: User-based filtering – limit behavior: RMSE for different number of taps with coefficients learned on the testing set

Number of taps	RMSE ($r = 0$)	RMSE ($r = 0.5$)
$L = 1$	0.9036	0.9036
$L = 2$	0.8921	0.8921
$L = 3$	0.8226	0.8735
$L = 4$	0.8218	0.8643
$L = 5$	0.8128	0.8593
$L = 6$	0.8073	0.8572
$L = 7$	0.8068	0.8560
$L = 8$	0.7922	0.8554
$L = 9$	0.8026	**0.8550**

Table 2: User-based filtering – proper training: RMSE for different number of taps with coefficients learned on the training set

Number of taps	RMSE ($r = 0$)	RMSE (r learned from cross-validation)
benchmark NN	0.9116	
matrix completion	0.8723	
$L = 1$	0.9175	0.9175
$L = 2$	0.8875	0.8875
$L = 3$	0.8647	0.8647
$L = 4$	0.8661	0.8661
$L = 5$	0.8557	0.8554
$L = 6$	0.8609	**0.8551 (6.20% improvement to NN)**

Firstly, the RMSE for both $r = 0$ and $r = 0.5$ decreases as the number of filter taps L increases from 1 to 6, remaining flat for $L > 6$. This suggests that considering chains of more than 6 users probably does not improve prediction performance. Secondly, the RMSE with large L and $r = 0$ is around 0.80, which will be the value considered as the benchmark for algorithms that learn $\mathbf{h}$ in the training set and test their performance in the testing set.

When we solve the actual problem in (11.15) with coefficients learned on the training set $\mathcal{R}_{\text{tr}}$, we rely on the results in Table 1 to limit the maximum number of taps to 6. The RMSE on the testing set $\mathcal{R}_{\text{ts}}$ for different values of L and r is presented in Table 2. The main observations are: i) higher order filters perform better than the traditional order-one NN filter (a user-based NN filter attains an RMSE of 0.9116, while for $L = 6$ and $r = 0.5$ our method attains an error of 0.8551, which is an improvement of 6.10%); and ii) the prediction performance, especially that for the case where $r = 0.5$, is not much worse than that in Table 1, with the trends being also similar to those in Table 1. Moreover, when proper regularization is applied, the optimal coefficients learned from the training set are also close to the coefficients learned from the testing set. User-based filtering

Table 3: Item-based filtering – limit behavior: RMSE for different number of taps with coefficients learned on the testing set

Number of taps	RMSE ($r = 0$)	RMSE ($r = 0.5$)
$Q = 1$	0.9052	0.9052
$Q = 2$	0.8787	0.8852
$Q = 3$	0.8785	0.8829
$Q = 4$	0.8785	**0.8824**
$Q = 5$	0.8660	0.8826
$Q = 6$	0.8646	0.8827
$Q = 7$	0.8503	0.8828
$Q = 8$	0.8485	0.8828
$Q = 9$	0.8485	0.8828

Table 4: Item-based filtering – proper training: RMSE for different number of taps with coefficients learned on the training set

Number of taps	RMSE ($r = 0$)	RMSE (r learned from cross-validation)
benchmark NN	0.9053	
matrix completion	0.8723	
$Q = 1$	0.9165	0.9165
$Q = 2$	0.8800	0.8799
$Q = 3$	0.8909	**0.8797 (2.83% improvement to NN)**
$Q = 4$	0.8921	0.8895
$Q = 5$	0.8958	0.8858
$Q = 6$	0.9235	0.8854

outperforms conventional matrix completion by 1.97%.

Another interesting observation is that the optimal coefficients learned from either the training set in (11.15) or the testing set in (11.32) tend to satisfy that $\sum_{l=1}^{L} h_l \simeq 1$, as illustrated in Figure 88 (a). Such a property does not seem to depend on the number of taps used in the filter. This is in accordance with the classical NN predictor that, being a a graph filter of order $L = 1$, sets $h_0 = 0$ and $h_1 = 1$.

To gain further insights, the frequency response for the user-based filter with $L = 6$ and $\mathbf{h}$ learned in $\mathcal{R}_{\text{tr}}$ is illustrated in Figure 84 (b). The frequency response resembles that of user-based NN in Figure 84 (a), with both of them being band-stop filters. The major difference is that the range of frequencies with non-zero frequency response is now larger, which is the reason potentially explaining the enhanced prediction performance (an RMSE reduction of 6.10%).

We move now to item-based filters, where item-mean centering as in (11.2) is used to remove differences between items. Similar as in user-based filtering, we set $C_{ij} = 0$ in (11.7) for any pair of items such that $|\mathcal{S}_{ij}|$ is smaller than the threshold $\underline{\mathcal{S}} = 2$. The first experiment tries to assess the best performance that we are able to have with the available data. To do so, we use the networks $\mathbf{C}_u$ learned on the training set $\mathcal{R}_{\text{tr}}$ and solve a similar least squares problem by finding the coefficients yielding the best fit for the *testing* set $\mathcal{R}_{\text{ts}}$, i.e.

$$\min_{\mathbf{g}} \sum_{(u,i)\in\mathcal{R}_{\text{ts}}} \left| \left[\left(\sum_{q=1}^{Q} g_q \mathbf{C}_u^q \right) \mathbf{x}_u \right]_i - X_{ui} \right|^2 + r\|\mathbf{g}\|_2^2. \tag{11.33}$$

The RMSE across ratings in $\mathcal{R}_{\text{ts}}$ using $\mathbf{g}$ trained in (11.33) for $r = 0$ and $r = 0.5$ is listed in Table 3. Compared to the results for user-based band-stop filters, there are two major differences. Firstly, the RMSE (for both $r = 0$ and $r = 0.5$) stops to improve once the number of filter taps reach 4. This implies that in predicting the ratings for an item i, only items j connected to i via at most three intermediate items are informative. Recall that in user-based filtering, paths with length up to 6 are still helpful. A potential explanation for this phenomenon is that it is easier to find a path $u_1u_2u_3u_4u_5u_6$ of 6 users in the network, such that each adjacent pair of users u_t and u_{t+1} possess highly similar taste; for item-based filtering, however, it is not that easy to find paths of more than 4 items such that each adjacent pair is conceived as highly similar by most users. Secondly, the limit RMSE for item-based filtering, 0.8485, is higher compared to its user-based counterpart, 0.7922, pointing out that for the setup at hand user networks could be more informative. Regardless of these differences, in both Tables 1 and 3, the RMSE differences between $r = 0$ and $r = 0.5$ are relatively small, indicating that adding a regularizer still yields results close to the limits.

When we solve the actual problem in (11.17) with coefficients learned on the training set $\mathcal{R}_{\text{tr}}$, we limit the maximum number of taps to 6. Similar to user-based filtering, the RMSE on the testing set $\mathcal{R}_{\text{ts}}$ shown in Table 4 does not deteriorate significantly compared to the limits reported in Table 3. In fact, results for $r = 0.5$ are highly similar to the ones listed in Table 3, with the best RMSE achieving 0.8797, a 2.83% improvement compared to the benchmark item-based NN. Moreover, when proper regularization is applied, the optimal coefficients learned from the training set are also close to the coefficients learned from the testing set. In this case we also observe that $\sum_{q=1}^{Q} g_q \simeq 1$, irrespective of the number of taps Q, as plotted in Figure 88 (b). The frequency response for the item-based filter with $Q = 3$ and the optimal coefficients learned from the training set are illustrated in Figure 84 (e). The frequency response is highly similar to the one of item-based NN in Figure 84 (d), since both of them are band-stop filters. Item-based filtering slightly underperforms relative to conventional matrix completion.

11.4.3. MiFi

To investigate the prediction performance of MiFi, user-mean centering as in (11.1) is applied. The reason for this is because user-based filtering improves performance better than item-based filtering, and user-based filtering utilizes user-mean centering. We start by assessing its limit behavior, where we use the networks $\mathbf{B}_i$ and $\mathbf{C}_u$ learned on the training set $\mathcal{R}_{\text{tr}}$ and solve a least

Table 5: MiFi – limit behavior: RMSE for different number of taps (horizontally for item-based $\mathbf{C}_u^q$ and vertically for user-based $\mathbf{B}_i^l$) with coefficients learned on the testing set

	$Q=1$	$Q=2$	$Q=3$	$Q=4$
$L=1$	0.8513	0.8269	0.8258	0.8259
$L=2$	0.8430	0.8200	0.8186	0.8187
$L=3$	0.8107	0.7916	0.7910	0.7914
$L=4$	0.8087	0.7886	0.7882	0.7885
$L=5$	0.8089	0.7884	**0.7879**	0.7882

Table 6: MiFi – proper training: RMSE for different number of taps (horizontally for item-based $\mathbf{C}_u^q$ and vertically for user-based $\mathbf{B}_i^l$) with coefficients learned on the training set

	$Q=1$	$Q=2$	$Q=3$	$Q=4$
$L=1$	0.8817	0.8653	0.8572	0.8518
$L=2$	0.8701	0.8565	0.8523	0.8534
$L=3$	0.8561	0.8426	0.8418	0.8430
$L=4$	0.8558	0.8426	0.8413	0.8423
$L=5$	0.8422	**0.8229 (9.10% improvement)**	0.8363	0.8381

squares problem similar to (11.19) but finding the coefficients yielding that minimize the error in the *testing* set $\mathcal{R}_{\text{ts}}$, i.e.

$$\min_{\mathbf{h},\mathbf{g}} \sum_{(u,i)\in\mathcal{R}_{\text{ts}}} \left| \left[\left(\sum_{l=0}^{L} h_l \mathbf{B}_i^l \right) \mathbf{X} \left(\sum_{q=0}^{Q} g_q \mathbf{C}_u^q \right)^{\top} \right]_{ui} - X_{ui} \right|^2 + r\left(\|\mathbf{h}\|_2^2 + \|\mathbf{g}\|_2^2 \right).$$

As described in Section 11.2.3, this is a bilinear minimization on $\mathbf{g}$ and $\mathbf{h}$. Since user-based filtering outperforms its item-based counterpart, we use as initialization the optimal filter coefficients $\mathbf{h}^{(0)}$ solving (11.15). We then solve a succession of least-squares problems that alternates between $\mathbf{g}$ and $\mathbf{h}$, and stops when the change in the solution is sufficiently small. In our experiments, 30 iterations seem to be enough.

The RMSE across ratings in $\mathcal{R}_{\text{ts}}$ using $\mathbf{g}$ and $\mathbf{h}$ trained in (11.34) for $r = 0.5$ is listed in Table 5. We observe that the gains associated with increasing L are larger than those when increasing Q, pointing out that user-based filtering seems to be more important. The results also show that higher order filters perform better, but there exists a saturation effect. These two observations corroborate the findings in Section 11.4.2. More importantly, the results also demonstrate that MiFi outperforms the item-based and user-based graph filter approaches. To elaborate on this, we move to the case where the coefficients are learned on $\mathcal{R}_{\text{tr}}$, r is learned via cross-validation, and the solution is tested on $\mathcal{R}_{\text{ts}}$. The results for this setup, listed in Table 6, show that the optimal RMSE is 0.8229, which is achieved for $L = 5$ and $Q = 2$. Since the benchmark RMSE was 0.9053 (the better among the benchmarks for user-based NN and item-based NN), this means that a gain of 9.10% was achieved. Compared to conventional matrix completion, MiFi is superior by 5.66%.

Equally important, this gain is larger than the combination of the gains observed by user-based filtering and item-based filtering, which is $1 - (1 - 6.20\%) \times (1 - 2.83\%) = 8.85\%$. This implies that MiFi is able to create a *synergy* between the two filters on the user and item graphs.

As an example, the available rating X_{vj} is informative in predicting the unknown rating X_{ui}, as long as their corresponding users u and v are similar and their respective items i and j are similar. Despite the fact of such a model being very intuitive, the combination of two graph domains has not been investigated in NN. The reason could be as follows. Traditional NNs (either user-based or item-based) require input signals $\mathbf{x}^i$ or $\mathbf{x}_u$, where many entries are zeros and non-zero values correspond to observed ratings; these methods produce output signals $\mathbf{B}_i\mathbf{x}^i$ or $\mathbf{C}_u\mathbf{x}_u$, where all the values denote predictions. Therefore, if we combine user-based NN with item-based NN directly, the output of the first NN will be the input of the second NN. However, the output of the first NN are all predictions whereas the second NN needs input to be observed ratings. It might be this incompatibility that results in suboptimal performance. Using graph filters, this issue can be properly addressed by adding more coefficients (including those associated with the identity matrix $h_0\mathbf{B}_i^0$ and $g_0\mathbf{C}_u^0$) on both the user-based filter and item-based filter. The frequency response for the optimal MiFi filter with $L = 5$ and $Q = 2$ is visualized in Figure 84 (c) and (f). While the original frequency response is not band-stop, after removing the effect of h_0 and g_0 it becomes again a band-stop filter, reinforcing the idea that, when using the two filters, the role of h_l and g_q for l and q different than one is critical. We emphasize that the specific frequency response as in Figure 84 (b), (c), (e), and (f) is robust to small changes on L and Q.

We also evaluate the performance using the Kronecker shift $\mathbf{S}_{ui}^l$ by solving the problem in (11.20) and training the coefficients on the training set. The RMSE on testing set achieved is 0.8444, which is 6.73% better than NN benchmark, 3.20% better than conventional matrix completion, and 2.61% worse than MiFi. The RMSE decreases as the number of filter taps increases. A saturation effect on this decrement is observed for $L \geq 10$. This suggests that, for the database at hand, the consideration of more complicated shifts does not help explaining the data; most of the interactions between chains of users and items can be explained well under the MiFi model.

11.4.4. Sampling Bandlimited Graph Signals

Finally, we examine the performance of sampling bandlimited graph signals by solving (11.29). We consider (11.29) because the support of frequency $\mathcal{F}$ is unknown. The RMSE on the testing set achieved is 0.8674, which is slightly better than conventional matrix completion by 0.56%. This shows that the approximated eigenvectors $\hat{\mathbf{V}}$ and $\hat{\mathbf{U}}$ can be used to represent the actual eigenvectors well, and therefore the single-step matrix completion can be done separately by first evaluating the sample eigenvectors $\hat{\mathbf{V}}$ and $\hat{\mathbf{U}}$, and then estimating the bandlimited frequency components $\tilde{\mathbf{x}}$. We emphasize that this two-step strategy lowers the complexity significantly, since the (dominant) sample eigenvectors $\hat{\mathbf{V}}$ and $\hat{\mathbf{U}}$ can be efficiently found, and (11.29) can be written

Table 7: RMSE for the proposed methods and some common algorithms in rating predictions as benchmarks

Round	User-based NN	Item-based NN	Matrix completion	User-based proposed	Item-based proposed	**MiFi**	Bandlimited sampling	
1st	0.9116	0.9053	0.8723	0.8551	0.8797	**0.8229**	0.8674	0
2nd	0.8919	0.8696	0.8772	0.8596	0.8314	**0.8287**	0.8703	0
3rd	0.9498	0.9404	0.9345	0.9058	0.9282	**0.8733**	0.9249	0
4th	1.0228	1.0390	0.9938	0.9733	1.0097	**0.9391**	0.9874	1
5th	0.8788	0.8615	0.8494	0.8478	0.8395	**0.8173**	0.8517	0
Average	0.9310	0.9232	0.9054	0.8883	0.8977	**0.8563**	0.9003	0

Table 8: Percentage of RMSE improvements of the proposed methods compared to the benchmark algorithms

Round	User-based proposed	Item-based proposed	Bandlimited sampling	MiFi	MiFi	MiFi	MiFi
			compared to				
	User-based NN	Item-based NN	Matrix completion	User-based NN	Best NN	Matrix completion	[87]
1st	6.20%	2.83%	0.56%	9.73%	9.10%	5.66%	0.95%
2nd	3.62%	4.39%	0.79%	7.09%	4.70%	5.53%	1.67%
3rd	4.63%	1.30%	1.03%	8.05%	7.14%	6.55%	-3.41%
4th	4.84%	2.82%	0.64%	8.18%	8.18%	5.50%	6.57%
5th	3.53%	2.55%	-0.27%	7.00%	5.13%	3.78%	-3.22%
Average	4.56%	2.78%	0.55%	8.01%	6.85%	5.40%	0.51%

equivalently as

$$\min_{\tilde{\mathbf{x}}} \quad \sum_{r \in \mathcal{R}} \left| \left[\hat{\mathbf{U}} \otimes \hat{\mathbf{V}} \tilde{\mathbf{x}} \right]_r - x_r \right|^2 + \alpha \|\tilde{\mathbf{x}}\|_1, \tag{11.34}$$

which can be solved with computationally efficient algorithms, see, e.g., [232]. The next experiment relaxes the requirement that the frequency is a diagonal matrix $\tilde{\mathbf{X}} = \operatorname{diag}(\boldsymbol{\sigma})$ to allow some off-diagonal elements to be non-zero. In specific, we may allow entries $\tilde{X}_{t\tau}$ to be non-zero if the location is not far from diagonal with $|t-\tau| \leq \Delta$. This yields the following problem

$$\begin{aligned} \min_{\tilde{\mathbf{X}}} \quad & \left\| \operatorname{vec}(\tilde{\mathbf{X}}) \right\|_1 \\ \text{s. t.} \quad & \left| \left[\hat{\mathbf{U}} \tilde{\mathbf{X}} \hat{\mathbf{V}} \right]_{ui} - X_{ui} \right|^2 \leq \epsilon, \quad \forall\, (u,i) \in \mathcal{R}, \\ & \tilde{X}_{t\tau} = 0, \quad \text{if } |t-\tau| > \Delta. \end{aligned} \tag{11.35}$$

Setting $\Delta = 0$ would reduce the problem (11.35) back to (11.29). We evaluated the performance of (11.35) for $\Delta = 1$ and $\Delta = 2$, and found that the off diagonal elements $\tilde{X}_{t\tau}$ with $t \neq \tau$ generated by the solver were zero. In other words, the solution and performance was the same compared to that of (11.29). This implies that the model in (11.29) captures the key features of our data, with the estimated sample eigenvectors being close to the actual eigenvectors.

11.4.5. More Rounds of Testing

The results and discussion in the previous sections were based on networks constructed using a training set with 99,900 ratings and tested using a set with 100 ratings. The reason for splitting the ratings in such a way was to ensure that the graphs, which are the building blocks of all the algorithms in this paper, were very well estimated. However, since we only kept 100 ratings, which is a small number, for testing, a natural question is whether the reported performance is very sensitive to that particular testing set. To shed light on that question, in this section we repeat the experiment five times, each time with a different randomly selected (100 ratings) testing set and leaving the remaining 99,900 ratings in the training set. As before, the computation of the graphs and the filter coefficients is carried out using only the ratings in the corresponding training set.

The RMSE for the proposed methods as well as the benchmark algorithms across the five rounds of randomization is presented in Table 7, where the first round was the one analyzed in the previous sections. The RMSE gains relative to the benchmark algorithms are listed in Table 8. The boxplots illustrating the overall RMSE performance across the five rounds are presented in Figure 89.

As a general conclusion, the results obtained for this setup validate the analysis carried out in the previous sections. In particular, the proposed graph-SP-inspired methods yield better performance compared to the benchmarks: in average user-based higher order filtering is 4.56% better than user-based NN, item-based higher order filtering is 2.78% better than item-based NN, and bandlimited sampling is slightly better (0.55%) than low-rank matrix completion. As discussed in the previous sections, the improvement achieved by higher order graph filters in the user domain is larger than that achieved by higher order filters in the item domain. Moreover, MiFi is 6.85% better than both user-based NN and item-based NN, and 5.40% better than matrix completion. Recently, graph SP has been used to extract features to train a recurrent neural network to improve matrix completion [87]. Figure 89 reveals that MiFi yields an overall performance similar to that of the proposed graph-based recurrent neural network. More specifically, Table 8 shows that for the considered setup the RMSE of MiFi predictors is slightly (0.51%) better than that incurred by the graph-based recurrent neural network in [87].

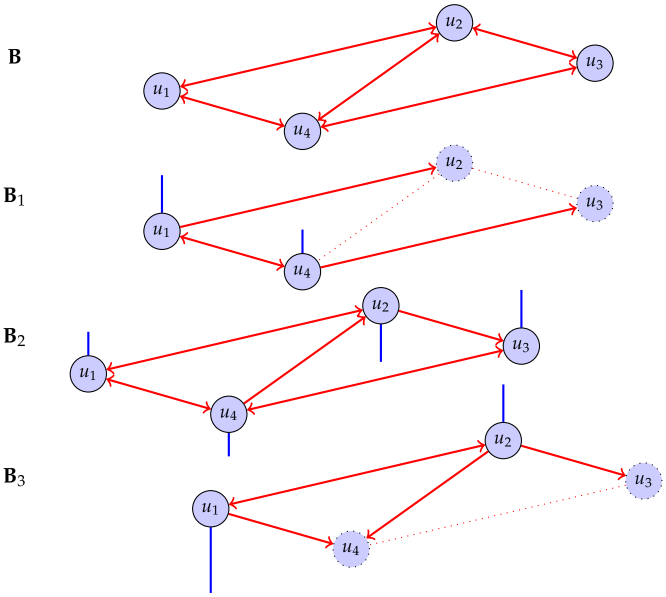


Figure 82: Viewing user-based collaborative filtering as graph filters. The ratings for each item can be considered as graph signals on a user-to-user network $\mathbf{B}_i$ that depends on the specific item i. For each specific item, edges in $\mathbf{B}$ starting from users who have not rated them are removed; e.g. edges from users u_3 and u_4 are not present in network $\mathbf{B}_1$ and edges from users u_2 and u_4 are not present in network $\mathbf{B}_3$. Then, given a specific user u, for all the edges coming into u, only the ones with the k-highest edge weights are kept; e.g. when $k = 2$, in $\mathbf{B}_2$, the edge from u_3 to u_2 is removed because its weight is small compared to the edges from u_4 to u_2 and from u_1 to u_2 (edge weights are proportional to the width of the edges). Proper normalization is then applied to make each $\mathbf{B}_i$ right stochastic. NN are a specific type of band-stop graph filter.

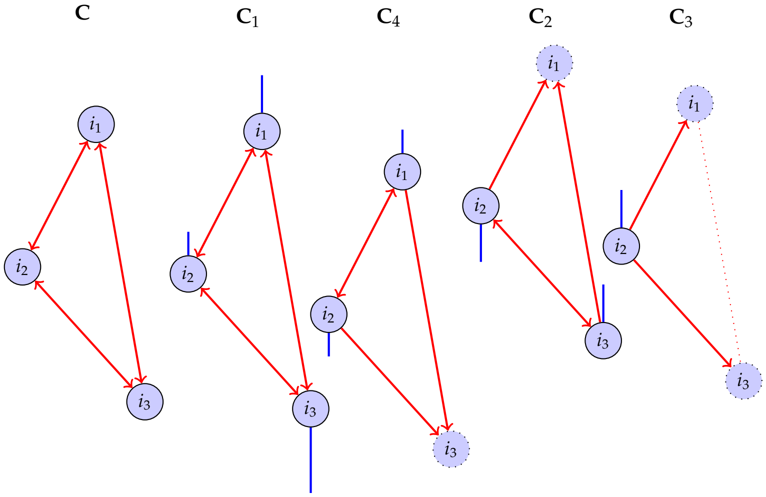
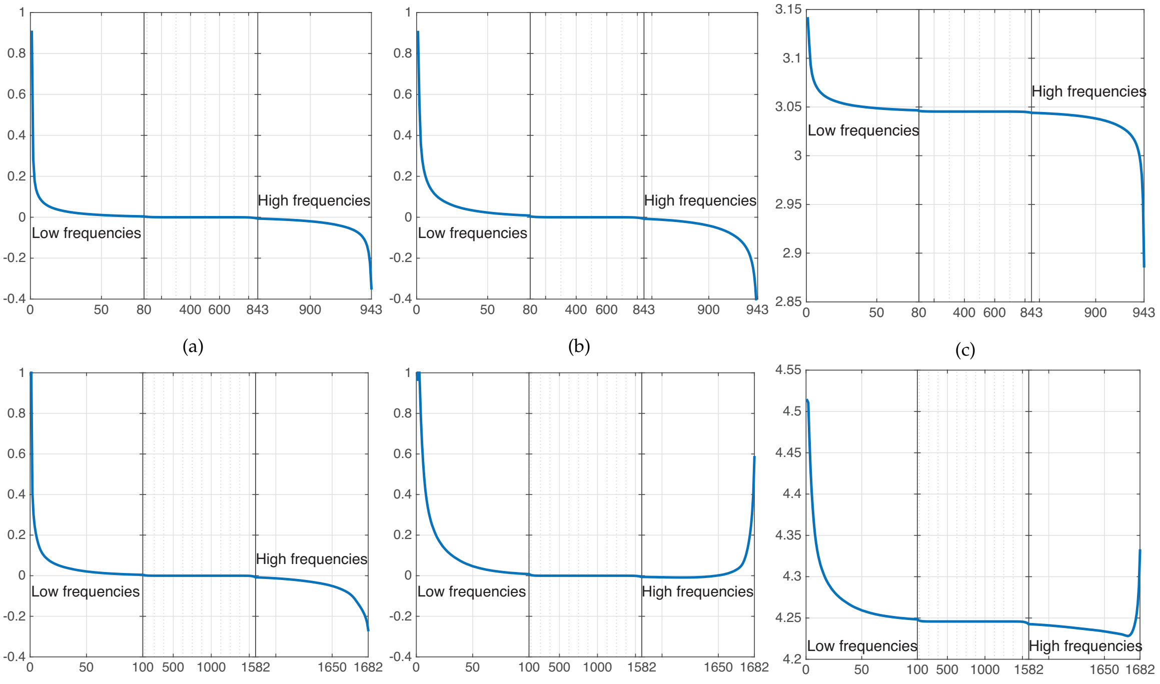


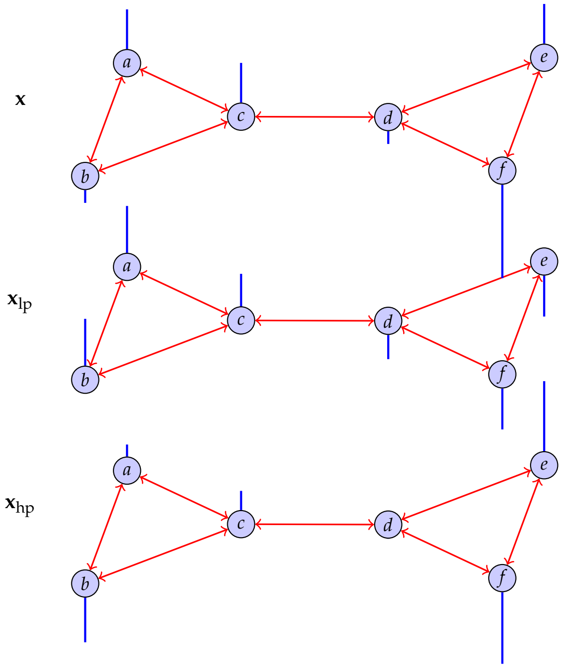
Figure 83: Viewing item-based collaborative filtering as graph filters. As in Figure 82, the ratings for each user can be considered as graph signals on an item-to-item network that depends on the specific user. For each user, edges in $\mathbf{C}$ starting from items not rated by the user are removed. Proper normalization is then applied to make each $\mathbf{C}_u$ right stochastic.



(a) (b) (c)

(d) (e) (f)

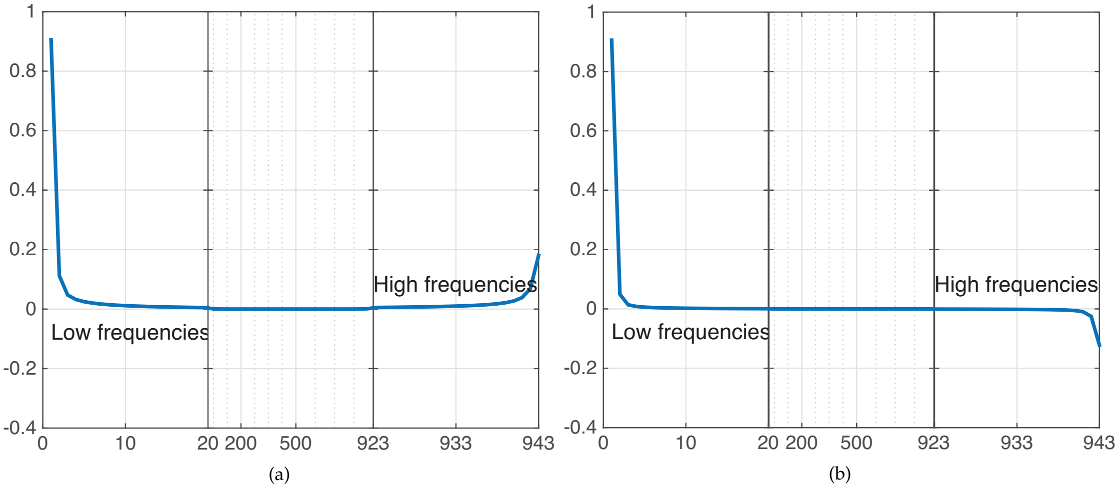
Figure 84: Frequency response for the graph filters used in user-based filtering (Top) and used in item-based filtering (Bottom). The horizontal axis represents the index p of the frequency (lower frequency with smaller p and higher frequency with larger p) and the vertical axis denotes the corresponding frequency response of the examined filters. For each $\mathbf{B}_i$, we order its eigenvalues according to $|\lambda_p - \lambda_{\max}| \leq |\lambda_{p+1} - \lambda_{\max}|$, so that lower frequencies (lower values of p) are those closer to $\lambda_{\max}$. The frequency response for low, middle, and high frequencies is computed for different items and sample sets, and the average behavior of such frequency responses is visualized in the top plots. Frequency responses for item-based filtering are prepared similarly and visualized in the bottom plots; there exist certain $\mathbf{C}_u$ with some rows being entirely zero, making their largest eigenvalues strictly less than 1. (a) Frequency response for user-based NN; (b) Response for the best user-based filter trained on training set using user-based networks $\{\mathbf{B}_i^l\}_{l=0}^{6}$; (c) User-based frequency response in the best MiFi. (d) Frequency response for item-based NN; (e) Response for the best item-based filter trained on training set using item-based networks $\{\mathbf{C}_u^q\}_{q=0}^{3}$; (f) Item-based frequency response in the best MiFi.



$$[\tilde{\mathbf{x}}_{\mathrm{lp}}]_p = \begin{cases} [\tilde{\mathbf{x}}]_p, & \lambda_p \geq 0; \\ 0, & \lambda_p < 0. \end{cases}$$

$$[\tilde{\mathbf{x}}_{\mathrm{hp}}]_p = \begin{cases} [\tilde{\mathbf{x}}]_p, & \lambda_p < 0; \\ 0, & \lambda_p \geq 0. \end{cases}$$

Figure 85: Low-pass and high-pass components of a rating signal $\mathbf{x}$. The low-pass components represent signals where similar users tend to have similar ratings, which provide the big picture for the rating profile. The high-pass components represent the differences between users with similar taste for the particular item, which can be considered as the distinguishing features to separate the tastes of individual users.



(a) (b)

Figure 86: (a) Frequency response for the square $\mathbf{B}_i^2$ of graph shifts and (b) frequency response for the cubic $\mathbf{B}_i^3$ of shifts. For each $\mathbf{B}_i$, we order its eigenvalues in an increasing way according to $|\lambda_p - 1|$ and respectively record the frequency response λ_p^2 and λ_p^3 for low, middle, and high frequencies. The average behavior of the frequency response across all graph shifts is visualized.

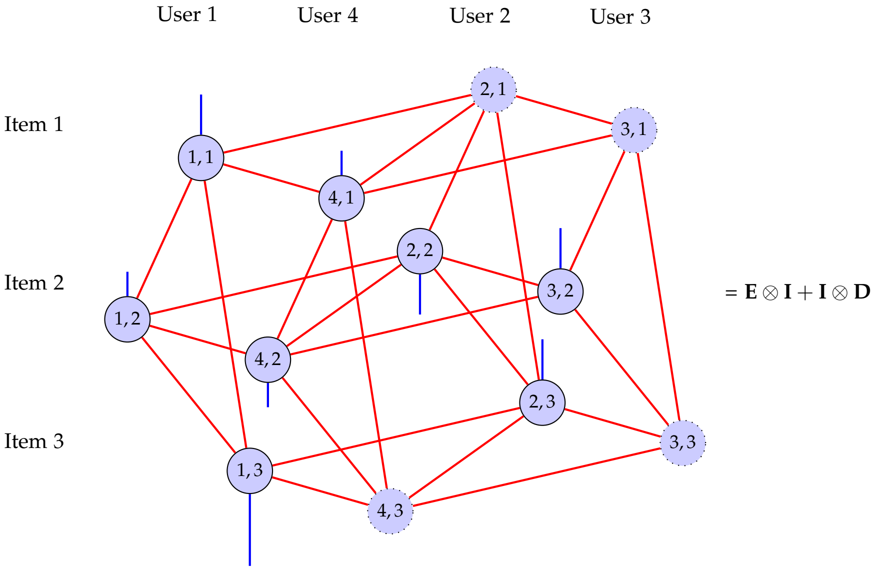
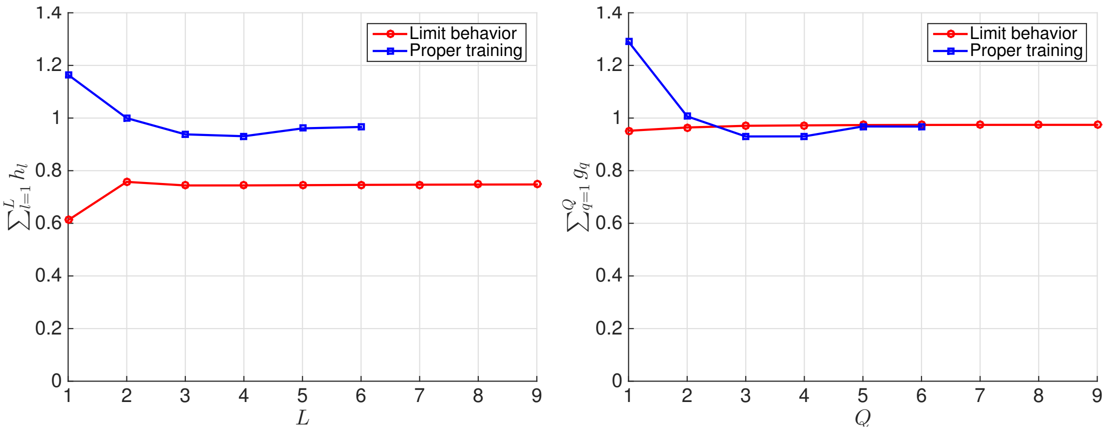


Figure 87: Viewing the entire rating matrix where the entry for X_{ui} is represented as (u, i) as a graph signal on the product graph of covariance shifts **D** and **E**. Rating prediction problem can be solved by using graph filters on the product graphs, or using cascade of graph filters on the respective graphs.



(a)

(b)

Figure 88: (a): Sum of the filter coefficients $\sum_{l=1}^{L} h_l$ for different tap of filters L used in user-based NN. In both the "Limit behavior test" (Red) and the "Proper training test" (Blue), the sum of coefficients is close to 1, indicating that without the design of filter banks, NN based collaborative filtering is indeed the optimal solution. (b): Sum of the filter coefficients $\sum_{q=1}^{Q} g_q$ for different tap of filters Q used in item-based NN. Again, in both the "Limit behavior test" (Red) and the "Proper training test" (Blue), the sum is close to 1.

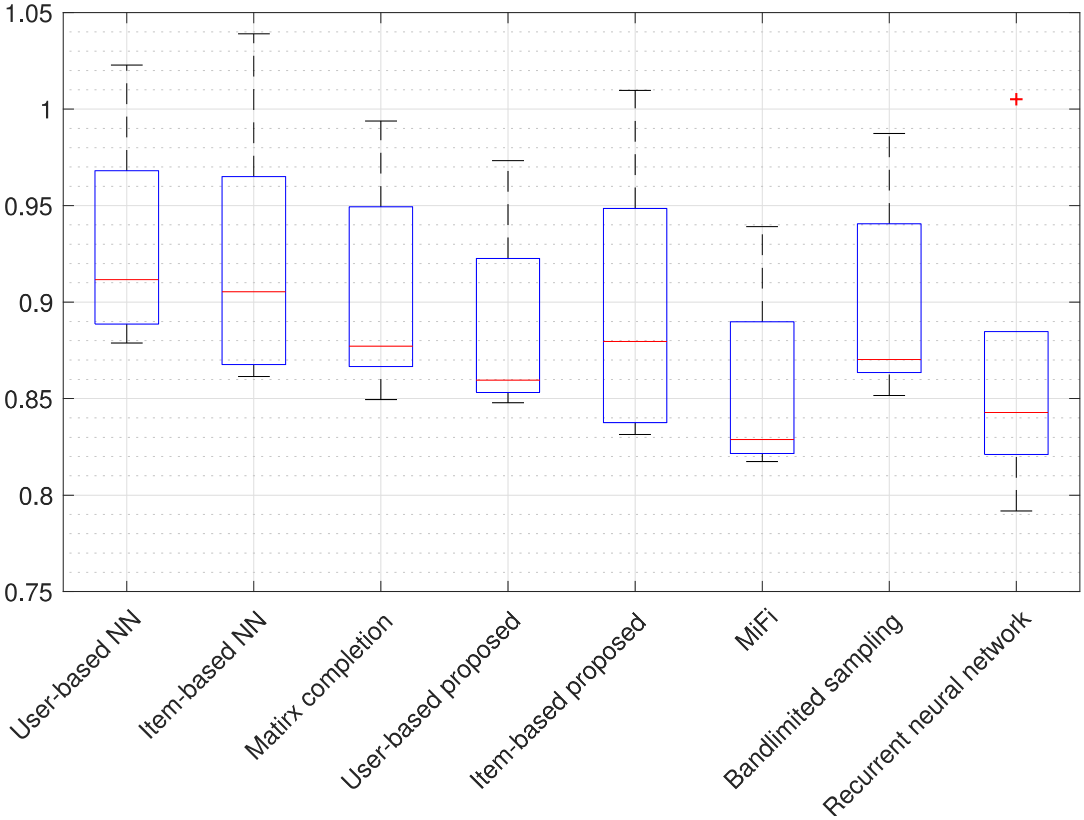


Figure 89: Boxplots of the RMSE for the proposed methods and traditional rating prediction algorithms across 5 rounds of randomizations. The symbol '+' on the top of the right-most boxplot corresponds to the performance of [87] on the third round.

Chapter 12

Conclusion

The first part of the thesis, Part I from Chapter 2 to Chapter 6, focused on the analysis of the network themselves. The problems addressed in this part are the comparison of networked data structures as well as filtering of networks.

Chapter 2 presented methods to compare high order networks, defined as weighted complete hypergraphs collecting relationship functions between elements of tuples. They can be considered as generalizations of conventional networks where only relationship functions between pairs are defined. Important properties between relationships of tuples of different lengths were established, particularly when relationships encode dissimilarities or proximities between nodes. Two families of distances were then introduced in the space of high order networks. The distances measure differences between networks. We proved that they are valid metrics in the spaces of high order dissimilarity and proximity networks modulo permutation isomorphisms. Practical implications were explored by comparing the coauthorship networks of two popular signal processing researchers. The metrics succeeded in identifying their respective collaboration patterns.

Chapter 3 built on the results in Chapter 2. The goal in Chapter 3 was to find tractable approximations of these network distances. We did so by mapping high order networks to filtrations of simplicial complexes and showing that the distance between networks can be lower bounded by the difference between the homological features of their respective filtrations. Practical implications were explored by classifying weighted pairwise networks constructed from different generative processes and by comparing the coauthorship networks of engineering and mathematics academic journals. The persistent homology methods succeeded in identifying different generative models, in discriminating engineering and mathematics communities, as well as in differentiating engineering communities with different research interests.

Chapter 4 was based on the some observations found in Chapter 3. The goal of Chapter 4 was to establish definitions of frequencies and filtering operations in the space of networks. Homospectrum of a given network collects all such frequencies. Filtering on networks were then defined by trivializing the relationships of tuples whose generated homological features have certain durations. We illustrated that the original network can be easily recovered from homospectrum, and therefore, homospectrum offers a different representation of the same information described in the network. Besides, we illustrated that the difference between the original network and the filtered network using network filter is bounded and small. Practical implications were explored by showing the ability of homospectrums to discriminate networks generated by different models, and by illustrating how we can capture the different features of a point cloud by applying different types of filter.

Chapter 5 built on results in Chapters 2 and 3. Chapter 5 considered metric space where the exact dissimilarities between pairs of points are not unknown but known to belong to some interval. The goal was to study methods for the determination of hierarchical clusters, i.e., a family of nested partitions indexed by a resolution parameter, induced from the given distance intervals of the dissimilarities. Our construction of hierarchical clustering methods was based on defining admissible methods to be those methods that satisfy the axioms of value – nodes in a metric space with two nodes are clustered together at the convex combination of the upper and lower bounds determined by a parameter – and transformation – when both distance bounds are reduced, the output may become more clustered but not less. Two admissible methods were constructed and are shown to provide universal bounds in the space of admissible methods. Practical implications were explored by clustering moving points via snapshots and by clustering coauthorship networks representing collaboration between researchers from different communities. The proposed clustering methods succeeded in identifying underlying hierarchical clustering structures via the maximum and minimum distances in all snapshots, as well as in differentiating collaboration patterns in journal publications between different research communities based on bounds of network distances.

Chapter 6 started with a different direction from that in Chapter 2. In Chapter 6, we defined such network distance by searching for the optimal method to embed one network into another network, proved that such distance is a valid metric in the space of networks modulo permutation isomorphisms, and examined its relationship with other network metrics. The network distance defined can be approximated via multi-dimensional scaling, however, the lack of structure in networks results in sub-optimal approximations. To alleviate such problem, we considered methods to define the interiors of networks. We showed that comparing interiors induced from a pair of networks yields the same result as the actual network distance between the original networks. Practical implications were explored by showing the ability to discriminate networks generated by different models.

The second part of the thesis, Part II from Chapter 7 to Chapter 11, focused on signals defined on top of the networks. We design domain-specific tools to leverage the network structure to analyze the graph signals, and apply them in cancer subtype classification, brain signal analytics, as well as rating prediction.

After introducing prerequisite definitions of graph signal processing including graph Fourier transform, graph filters, and their interpretations in Chapter 7, in Chapter 8, we defined the diffusion and superposition distances as two metrics to compare signals supported in the nodes of a network. The similarity between the given vectors is determined by the similarity of the respective diffusion profiles. The superposition distance computes the instantaneous difference between the diffused signals and integrates the difference over time. The diffusion distance determines a distance between the integrals of the diffused signals. We proved that both distances define valid metrics and that they are stable to perturbations in the underlying network. We utilized numerical experiments to illustrate their utility in classifying signals in a synthetic network as well as in classifying ovarian cancer histologies using gene mutation profiles of different patients. We also utilized diffusion in label propagation process in semi-supervised learning and demonstrate its benefit through the classification of handwritten digits.

In Chapter 9, we presented methods to analyze functional brain networks and signals from graph spectral perspectives. Brain network frequency enables the decomposition of brain signals into pieces corresponding to smooth or rapid variations. We related graph frequency with principal component analysis when the networks of interest denote functional connectivity. The methods were utilized to analyze brain networks and signals as subjects master a simple motor skill. We observed that brain signals corresponding to different graph frequencies exhibit different levels of adaptability throughout learning. Further, we noticed a strong association between graph spectral properties of brain networks and the level of exposure to tasks performed, and recognize the most contributing and important frequency signatures at different levels of task familiarity.

Chapter 10 builds on Chapter 9 but considers structural brain connectivity as the underlying network. In specific, in Chapter 10, we reviewed graph signal processing for brain imaging data and discussed their potential to integrate brain structure, contained in the graph itself, with brain function, residing in the graph signals. We reviewed how brain activity can be meaningfully filtered based on concepts of spectral modes derived from brain structure. We also derived other operations such as surrogate data generation or decompositions informed by cognitive systems. The combined observations in Chapters 9 and 10 shows how GSP offers a novel framework for the analysis of brain imaging data.

In Chapter 11, we developed new designs for recommendation systems inspired by recent advances in GSP. Recommendation systems aim to predict unknown ratings by exploiting the information revealed in a subset of user-item observed ratings. Leveraging the notions of graph frequency and graph filters, we demonstrated that classical collaborative filtering methods, such as k-nearest neighbors, can be modeled as a specific *band-stop* graph filter on networks describing similarities between users or items. We also demonstrated that linear latent factor models, such as low-rank matrix completion, can be viewed as *bandlimited* interpolation algorithms that operate in a frequency domain given by the spectrum of a joint user and item network. These new interpretations paved the way to new methods for enhanced rating prediction. For nearest-neighbor-based collaborative filtering, we developed more general band-stop graph filters, and present a novel predictor, called Mirror Filtering, that filters jointly across user and item networks. For latent factoring, we proposed a low complexity method by exploiting the eigenvector of correlation matrices constructed from known ratings. The performance of our algorithms was assessed in the MovieLens 100k dataset, showing that our designs reduce the root mean squared error (up to a 9.10% for MiFi) compared to the one incurred by the benchmark approach.

Bibliography

[1] O. Sporns, *Networks Of the Brain*. MIT press, 2011.

[2] C. Wrzus, M. Hänel, J. Wagner, and F. J. Neyer, "Social network changes and life events across the life span: a meta-analysis.," *Psychol. Bull.*, vol. 139, p. 53, Jan. 2013.

[3] R. Sharan, S. Suthram, R. M. Kelley, T. Kuhn, S. McCuine, P. Uetz, T. Sittler, R. M. Karp, and T. Ideker, "Conserved patterns of protein interaction in multiple species," *Proc. Natl. Acad. Sci. U.S.A.*, vol. 102, pp. 1974–1979, Feb. 2005.

[4] B. Zhang and S. Horvath, "A general framework for weighted gene co-expression network analysis," *Stat. Appl. Genet. Mol. Biol.*, vol. 4, Nov. 2005.

[5] D. Weintraub, N. Dietz, J. E. Duda, D. A. Wolk, J. Doshi, S. X. Xie, C. Davatzikos, C. M. Clark, and A. Siderowf, "Alzheimer's disease pattern of brain atrophy predicts cognitive decline in parkinson's disease," *Brain*, vol. 135, pp. 170–180, Nov. 2011.

[6] X. Zhao, Y. Liu, X. Wang, B. Liu, Q. Xi, Q. Guo, H. Jiang, T. Jiang, and P. Wang, "Disrupted small-world brain networks in moderate alzheimer's disease: a resting-state fmri study," *PloS one*, vol. 7, p. e33540, Mar. 2012.

[7] M. A. Binnewijzend, J. P. Kuijer, W. M. van der Flier, M. R. Benedictus, C. M. Möller, Y. A. Pijnenburg, A. W. Lemstra, N. D. Prins, M. P. Wattjes, B. N. van Berckel, *et al.*, "Distinct perfusion patterns in alzheimer's disease, frontotemporal dementia and dementia with lewy bodies," *Eur. Radiol.*, vol. 24, pp. 2326–2333, Sep. 2014.

[8] W. Huang and A. Ribeiro, "Metrics in the space of high order networks," *IEEE Trans. Signal Process.*, vol. 64, pp. 615–629, Feb. 2016.

[9] R. B. Squires, B. E. Pickett, S. Das, and R. H. Scheuermann, "Toward a method for tracking virus evolutionary trajectory applied to the pandemic h1n1 2009 influenza virus," *Infect. Genet. Evol.*, vol. 28, pp. 351–357, Dec. 2014.

[10] J.-P. Onnela, J. Saramäki, J. Hyvönen, G. Szabó, D. Lazer, K. Kaski, J. Kertész, and A.-L. Barabási, "Structure and tie strengths in mobile communication networks.," *Proc. Natl. Acad. Sci. U.S.A.*, vol. 104, pp. 7332–6, May 2007.

[11] G. Kossinets and D. J. Watts, "Empirical analysis of an evolving social network.," *Science*, vol. 311, pp. 88–90, Jan. 2006.

[12] D. Khmelev and F. Tweedie, "Using Markov Chains for Identification of Writer," *Lit. Ling.*

Comput., vol. 16, Sep. 2001.

[13] T. Wang and H. Krim, "Statistical classification of social networks," in *IEEE Int. Conf. Acoust., Speech, Signal Process.*, pp. 3977–3980, Mar. 2012.

[14] R. Singh, J. Xu, and B. Berger, "Global alignment of multiple protein interaction networks with application to functional orthology detection.," *Proc. Natl. Acad. Sci. U.S.A.*, vol. 105, pp. 12763–12768, Sep. 2008.

[15] L. Peng, L. Liu, S. Chen, and Q. Sheng, "A network comparison algorithm for predicting the conservative interaction regions in protein-protein interaction network," in *Int. Conf. Bioinspired Comput. Theor. Appl.*, pp. 34–39, Sept. 2010.

[16] S. Choobdar, P. Ribeiro, S. Bugla, and F. Silva, "Comparison of Co-authorship Networks across Scientific Fields Using Motifs," *IEEE/ACM Int. Conf. Adv. Soc. Networks Anal. Min.*, pp. 147–152, 2012.

[17] L. Yong, Z. Yan, and C. Lei, "Protein-protein interaction network comparison based on wavelet and principal component analysis," in *IEEE Int. Conf. Bioinformatics Biomed.*, pp. 430–437, Dec. 2010.

[18] N. Pržulj, "Biological network comparison using graphlet degree distribution.," *Bioinformatics*, vol. 23, pp. e177–183, Jan. 2007.

[19] T. Milenković and N. Pržulj, "Uncovering biological network function via graphlet degree signatures.," *Cancer Inform.*, vol. 6, p. 257, Jan. 2008.

[20] N. Shervashidze, S. Vishwanathan, T. H. Petri, K. Mehlhorn, and K. M. Borgwardt, "Efficient graphlet kernels for large graph comparison," in *Int. Conf. Artif. Intell. Stat.*, vol. 5, pp. 488–495, Apr. 2009.

[21] G. Carlsson, F. Memoli, A. Ribeiro, and S. Segarra, "Axiomatic construction of hierarchical clustering in asymmetric networks," *arXiv preprint arXiv:1301.7724*, 2014.

[22] S. Fortin, *The Graph Isomorphism Problem*. PhD thesis, The University of Alberta, Edmonton, Alberta, Canada, 1996.

[23] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network Flows: Theory, Algorithms, And Applications*. Prentice-Hall, Inc., 1993.

[24] S. Wasserman and K. Faust, *Social Network Analysis: Methods and Applications*. Structural Analysis in the Social Sciences, Cambridge University Press, 1994.

[25] R. Ghrist and A. Muhammad, "Coverage and hole-detection in sensor networks via homology," in *International Symposium on Information Processing in Sensor Networks* (I. S. I. P. S. Netw., ed.), pp. 254–260, Apr. 2005.

[26] V. de Silva and R. Ghrist, "Coordinate-free Coverage in Sensor Networks with Controlled Boundaries via Homology," *Int. J. Rob. Res.*, vol. 25, pp. 1205–1222, Dec. 2006.

[27] H. Chintakunta and H. Krim, "Divide and Conquer: Localizing Coverage Holes in Sensor Networks," in *Annu. IEEE. Commun. Soc. Conf. Sens., Mesh, Ad Hoc Commun. Netw.*, pp. 1–8, 2010.

[28] B. Zhang, "Hypernetworks: A molecular evolutionary architecture for cognitive learning and memory," *IEEE Comput. Intell. M.*, vol. 3, pp. 49–63, Aug. 2008.

[29] W. Ren, Q. Zhao, R. Ramanathan, J. Gao, A. Swami, A. Bar-Noy, M. P. Johnson, and P. Basu, "Broadcasting in multi-radio multi-channel wireless networks using simplicial complexes," *Wirel. Netw.*, vol. 19, pp. 1121–1133, Nov. 2012.

[30] J. Xu and V. Singh, "Unified Hypergraph for Image Ranking in a Multimodal Context," in *IEEE Int. Conf. Acoust., Speech, Signal Process.*, pp. 2333–2336, Mar. 2012.

[31] Y. Gao, M. Wang, D. Tao, R. Ji, and Q. Dai, "3-D object retrieval and recognition with hypergraph analysis.," *IEEE Trans. Image Process.*, vol. 21, pp. 4290–303, Sept. 2012.

[32] A. Wilkerson, T. Moore, A. Swami, and H. Krim, "Simplifying the Homology Of Networks via Strong Collapses," in *IEEE Int. Conf. Acoust., Speech, Signal Process.*, pp. 5258–5262, May 2013.

[33] S. Biasotti, L. De Floriani, B. Falcidieno, P. Frosini, D. Giorgi, C. Landi, L. Papaleo, and M. Spagnuolo, "Describing shapes by geometrical-topological properties of real functions," *ACM Comput. Surv.*, vol. 40, p. 12, Oct. 2008.

[34] A. Cerri, B. D. Fabio, M. Ferri, P. Frosini, and C. Landi, "Betti numbers in multidimensional persistent homology are stable functions," *Math. Methods Appl. Sci.*, vol. 36, pp. 1543–1557, Aug. 2013.

[35] B. Di Fabio and P. Frosini, "Filtrations induced by continuous functions," *Topol. Appl.*, vol. 160, pp. 1413–1422, Aug. 2013.

[36] H. Edelsbrunner, D. Letscher, and A. Zomorodian, "Topological Persistence and Simplification," in *Proc. Annlu. Symp. Found. Comput. S.*, pp. 454–463, 2000.

[37] A. Zomorodian and G. Carlsson, "Computing persistent homology," *Discrete Comput. Geom.*, vol. 33, pp. 249–274, Feb. 2005.

[38] G. Carlsson, "Topology and data," *Bull. Am. Math. Soc.*, vol. 46, no. 2, pp. 255–308, 2009.

[39] G. Cormode and A. McGregor, "Approximation algorithms for clustering uncertain data," in *ACM Proc. Symp. Princip. Database Syst.*, pp. 191–200, Jun. 2008.

[40] F. Gullo, G. Ponti, A. Tagarelli, and S. Greco, "A hierarchical algorithm for clustering uncertain data via an information-theoretic approach," in *IEEE Int. Conf. Data Min.*, pp. 821–826, Dec. 2008.

[41] E. Schubert, A. Koos, T. Emrich, A. Züfle, K. A. Schmid, and A. Zimek, "A framework for clustering uncertain data," *Proc. VLDB Endowment*, vol. 8, pp. 1976–1979, Aug. 2015.

[42] M. Chavent and Y. Lechevallier, "Dynamical clustering of interval data: Optimization of an adequacy criterion based on hausdorff distance," in *Springer Classif., Cluster., Data Anal.*, pp. 53–60, Springer Berlin Heidelberg, 2002.

[43] R. M. de Souza and F. d. A. De Carvalho, "Clustering of interval data based on city–block distances," *Elsevier Pattern Recognit. Lett.*, vol. 25, pp. 353–365, Feb. 2004.

[44] P. Yu, L. Qinghua, and P. Xiyuan, "Uck-means: A customized k-means for clustering uncer-

tain measurement data," in *Int. Conf. Fuzzy Syst. Knowl. Discov.*, vol. 2, pp. 1196–1200, Jul. 2011.

[45] B. Jiang, J. Pei, Y. Tao, and X. Lin, "Clustering uncertain data based on probability distribution similarity," *IEEE Trans. Knowl. Data Eng.*, vol. 25, pp. 751–763, Apr. 2013.

[46] M. Chau, R. Cheng, B. Kao, and J. Ng, "Uncertain data mining: An example in clustering location data," in *Springer Pac. Asia Conf. Knowl. Discov. Data Min.*, pp. 199–204, Apr. 2006.

[47] B. Kao, S. D. Lee, F. K. Lee, D. W. Cheung, and W.-S. Ho, "Clustering uncertain data using voronoi diagrams and r-tree index," *IEEE Trans. Knowl. Data Eng.*, vol. 22, pp. 1219–1233, Sep. 2010.

[48] G. Carlsson and F. Mémoli, "Characterization, stability and convergence of hierarchical clustering methods," *J. Mach. Learn. Res*, vol. 11, pp. 1425–1470, Apr. 2010.

[49] A. K. Jain and R. C. Dubes, *Algorithms For Clustering Data*. Prentice-Hall, Inc., 1988.

[50] A. M. Bronstein, M. M. Bronstein, and R. Kimmel, "Efficient computation of isometry-invariant distances between surfaces," *SIAM J. Sci. Comput.*, vol. 28, pp. 1812–1836, Oct. 2006.

[51] A. M. Bronstein, M. M. Bronstein, and R. Kimmel, "Generalized multidimensional scaling: a framework for isometry-invariant partial surface matching," *Proc. Nat. Acad. Sci.*, vol. 103, pp. 1168–1172, Jan. 2006.

[52] A. M. Bronstein, M. M. Bronstein, and R. Kimmel, "Robust expression-invariant face recognition from partially missing data," in *Eur. Conf. Comput. Vis.*, pp. 396–408, Springer, May 2006.

[53] E. Lieberman, C. Hauert, and M. A. Nowak, "Evolutionary dynamics on graphs," *Nature*, vol. 433, p. 312, Jan. 2005.

[54] M. E. Newman, "Finding community structure in networks using the eigenvectors of matrices," *Phys. Rev. E*, vol. 74, no. 3, p. 036104, 2006.

[55] J.-F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM J. Optim*, vol. 20, pp. 1956–1982, Mar. 2010.

[56] J. B. Schafer, D. Frankowski, J. Herlocker, and S. Sen, "Collaborative filtering recommender systems," in *The Adaptive Web*, pp. 291–324, Springer, 2007.

[57] A. Sandryhaila and J. Moura, "Discrete signal processing on graphs," *IEEE Trans. Signal Process.*, vol. 61, pp. 1644–1656, Apr. 2013.

[58] A. Sandryhaila and J. Moura, "Discrete signal processing on graphs: Frequency analysis," *IEEE Trans. Signal Process.*, vol. 62, pp. 3042–3054, Jun. 2014.

[59] D. Shuman, S. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Process. Mag.*, vol. 30, pp. 83–98, May 2013.

[60] D. D. Garrett, N. Kovacevic, A. R. McIntosh, and C. L. Grady, "The modulation of bold variability between cognitive states varies by age and processing speed," *Cereb. Cortex.*, vol. 23,

pp. 684–693, Mar. 2012.

[61] J. J. Heisz, J. M. Shedden, and A. R. McIntosh, "Relating brain signal variability to knowledge representation," *Neuroimage*, vol. 63, pp. 1384–1392, Nov. 2012.

[62] J. D. Medaglia, W. Huang, E. A. Karuza, S. L. Thompson-Schill, A. Ribeiro, and D. S. Bassett, "Functional alignment with anatomical networks is associated with cognitive flexibility," *Nat. Hum. Behav.*, vol. 2, p. 156, Feb. 2018.

[63] A. G. Marques, S. Segarra, G. Leus, and A. Ribeiro, "Sampling of graph signals with successive local aggregations," *IEEE Trans. Signal Process.*, vol. 64, pp. 1832–1843, Apr. 2016.

[64] S. Chen, R. Varma, A. Sandryhaila, and J. Kovačević, "Discrete signal processing on graphs: Sampling theory," *IEEE Trans. Signal Process.*, vol. 63, pp. 6510–6523, Dec. 2015.

[65] N. Perraudin, A. Loukas, F. Grassi, and P. Vandergheynst, "Towards stationary time-vertex signal processing," in *IEEE Int. Conf. Acoust., Speech, Signal Process.*, pp. 3914–3918, Mar. 2017.

[66] A. G. Marques, S. Segarra, G. Leus, and A. Ribeiro, "Stationary graph processes and spectral estimation," *IEEE Trans. Signal Process.*, pp. 5911–5926, Aug. 2017.

[67] A. Agaskar and Y. M. Lu, "A spectral graph uncertainty principle," *IEEE Trans. Inf. Theory*, vol. 59, pp. 4338–4356, Jul. 2013.

[68] B. Pasdeloup, R. Alami, V. Gripon, and M. Rabbat, "Toward an uncertainty principle for weighted graphs," in *IEEE Euro. Signal Process. Conf.*, pp. 1496–1500, Aug. 2015.

[69] M. Tsitsvero, S. Barbarossa, and P. Di Lorenzo, "Signals on graphs: uncertainty principle and sampling," *IEEE Trans. Signal Process.*, vol. 64, pp. 4845–4860, Sep. 2016.

[70] O. Teke and P. P. Vaidyanathan, "Uncertainty principles and sparse eigenvectors of graphs," *IEEE Trans. Signal Process.*, vol. 65, pp. 5406–5420, Oct. 2017.

[71] M. Rabbat, M. Coates, and S. Blouin, "Graph Laplacian distributed particle filtering," in *IEEE Euro. Signal Process. Conf.*, pp. 1493–1497, Aug. 2016.

[72] N. Tremblay and P. Borgnat, "Subgraph-based filterbanks for graph signals," *IEEE Trans. Signal Process.*, vol. 64, pp. 3827–3840, Aug. 2016.

[73] M. S. Kotzagiannidis and P. L. Dragotti, "Sampling and reconstruction of sparse signals on circulant graphs-an introduction to graph-FRI," *arXiv preprint arXiv:1606.08085*, 2016.

[74] R. Shafipour, A. Khodabakhsh, G. Mateos, and E. Nikolova, "A digraph Fourier transform with spread frequency components," *arXiv preprint arXiv:1705.10821*, 2017.

[75] S. Chen, Y. Yang, J. Moura, J. Kovačević, *et al.*, "Signal localization, decomposition and dictionary learning on graphs," *arXiv preprint arXiv:1607.01100*, 2016.

[76] W. Huang and A. Ribeiro, "Persistent homology lower bounds on high order network distances," *IEEE Trans. Signal Process.*, vol. 64, pp. 615–629, Jan. 2017.

[77] R. Liu, H. Nejati, and N.-M. Cheung, "Simultaneous low-rank component and graph estimation for high-dimensional graph signals: application to brain imaging," *arXiv preprint*

arXiv:1609.08221, Sep. 2016.

[78] J. Pang and G. Cheung, "Graph laplacian regularization for inverse imaging: Analysis in the continuous domain," *arXiv preprint arXiv:1604.07948*, Apr. 2016.

[79] D. Thanou, P. A. Chou, and P. Frossard, "Graph-based compression of dynamic 3D point cloud sequences," *IEEE Trans. Image Process.*, vol. 25, pp. 1765–1778, Apr. 2016.

[80] M. S. Kotzagiannidis and P. L. Dragotti, "The graph FRI framework-spline wavelet theory and sampling on circulant graphs," in *IEEE Int. Conf. Acoust., Speech, Signal Process.*, pp. 6375–6379, Mar. 2016.

[81] Y. Wang, A. Ortega, D. Tian, and A. Vetro, "A graph-based joint bilateral approach for depth enhancement," in *IEEE Int. Conf. Acoust., Speech, Signal Process.*, pp. 885–889, May 2014.

[82] R. Shafipour, R. A. Baten, M. K. Hasan, G. Ghoshal, G. Mateos, *et al.*, "Closing the knowledge gap in an online learning community: Network-analytic discoveries, simulation and prediction," *arXiv preprint arXiv:1707.01886*, 2017.

[83] V. Kalofolias, X. Bresson, M. Bronstein, and P. Vandergheynst, "Matrix completion on graphs," *NIPS Workshop*, Dec. 2014.

[84] W. Huang and A. Ribeiro, "Axiomatic hierarchical clustering given intervals of metric distances," in *Proc. Int. Conf. Accoustics Speech Signal Process.*, pp. 4227–4231, Mar. 2017.

[85] J. Ma, W. Huang, S. Segarra, and A. Ribeiro, "Diffusion filtering of graph signals and its use in recommendation systems," in *IEEE Int. Conf. Acoust., Speech, Signal Process.*, pp. 4563–4567, Mar. 2016.

[86] S. K. Narang, A. Gadde, E. Sanou, and A. Ortega, "Localized iterative methods for interpolation in graph structured data," in *IEEE Global Conf. Signal Inf. Process.*, pp. 491–494, Dec. 2013.

[87] F. Monti, M. Bronstein, and X. Bresson, "Geometric matrix completion with recurrent multi-graph neural networks," in *Adv. Neural Inf. Process. Syst.*, pp. 3700–3710, Dec. 2017.

[88] M. Mather, J. T. Cacioppo, and N. Kanwisher, "Introduction to the special section: 20 years of fMRI-what has it done for understanding cognition?," *Perspect. Psychol. Sci.*, vol. 8, pp. 41–43, Jan. 2013.

[89] E. Bullmore and O. Sporns, "Complex brain networks: graph theoretical analysis of structural and functional systems," *Nat. Rev. Neurosci.*, vol. 10, pp. 186–198, Mar. 2009.

[90] V. D. Calhoun, J. Liu, and T. Adalı, "A review of group ica for fmri data and ica for joint inference of imaging, genetic, and erp data," *Neuroimage*, vol. 45, pp. S163–S172, Mar. 2009.

[91] D. S. Bassett and O. Sporns, "Network neuroscience," *Nat. Neurosci.*, vol. 20, pp. 353–364, Feb. 2017.

[92] M. Newman, *Networks: An Introduction.* New York, NY, USA: Oxford Univ. Press, 2010.

[93] O. Sporns and R. F. Betzel, "Modular brain networks," *Annu. Rev. Psychol.*, vol. 67, pp. 613–640, Jan. 2016.

[94] M. P. Van Den Heuvel and O. Sporns, "Rich-club organization of the human connectome," *J. Neurosci.*, vol. 31, pp. 15775–15786, Nov. 2011.

[95] J. Richiardi, S. Achard, H. Bunke, and D. Van De Ville, "Machine learning with brain graphs," *IEEE Signal Process. Mag.*, vol. 30, pp. 58–70, May 2013.

[96] T. Adalı, M. Anderson, and G.-S. Fu, "Diversity in independent component and vector analyses: Identifiability, algorithms, and applications in medical imaging," *IEEE Signal Process. Mag.*, vol. 31, pp. 18–33, May 2014.

[97] D. S. Bassett and M. G. Mattar, "A network neuroscience of human learning: Potential to inform quantitative theories of brain and behavior," *Trends Cogn. Sci. (Regul. Ed.)*, vol. 21, pp. 250–264, Apr. 2017.

[98] N. U. F. Dosenbach, B. Nardos, A. L. Cohen, D. A. Fair, J. D. Power, J. A. Church, S. M. Nelson, G. S. Wig, A. C. Vogel, C. N. Lessov-Schlaggar, K. A. Barnes, J. W. Dubis, E. Feczko, R. S. Coalson, J. R. Pruett, D. M. Barch, S. E. Petersen, and B. L. Schlaggar, "Prediction of individual brain maturity using fMRI," *Science*, vol. 329, pp. 1358–1361, Sep. 2010.

[99] M. G. Preti, T. A. W. Bolton, and D. Van De Ville, "The dynamic functional connectome: State-of-the-art and perspectives," *Neuroimage*, vol. 160, pp. 41–54, Oct. 2017.

[100] A. E. Sizemore and D. S. Bassett, "Dynamic graph metrics: Tutorial, toolbox, and tale," *NeuroImage*, in press.

[101] V. D. Calhoun, R. Miller, G. Pearlson, and T. Adalı, "The chronnectome: time-varying connectivity networks as the next frontier in fMRI data discovery," *Neuron*, vol. 84, pp. 262–274, Oct. 2014.

[102] S. D. Keilholz, C. Caballero-Gaudes, P. Bandettini, G. Deco, and V. D. Calhoun, "Time-resolved resting state fMRI analysis: current status, challenges, and new directions," *Brain Connect.*, vol. 7, pp. 465–481, Oct. 2017.

[103] F. I. Karahanoglu and D. Van De Ville, "Dynamics of large-scale fMRI networks: Deconstruct brain activity to build better models of brain function," *Curr. Opin. Biom. Eng.*, vol. 3, pp. 28–36, Sep. 2017.

[104] F. Ricci, L. Rokach, and B. Shapira, *Introduction To Recommender Systems Handbook*. Springer, 2011.

[105] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Comput.*, vol. 42, pp. 30–37, Aug. 2009.

[106] B. M. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Recommender systems for large-scale e-commerce: Scalable neighborhood formation using clustering," in *Int. Conf. Comput. Inf. Technol.*, vol. 1, Dec. 2002.

[107] E. J. Candès and B. Recht, "Exact matrix completion via convex optimization," *Found. Comput. Math.*, vol. 9, pp. 717–772, Jun. 2009.

[108] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "Grouplens: An open architecture for collaborative filtering of netnews," in *ACM Conf. Comput. Support. Coop. Work*, pp. 175–186, Oct. 1994.

[109] J. Ma, W. Huang, S. Segarra, and A. Ribeiro, "Diffusion filtering for graph signals and its use in recommendation systems," in *Proc. Int. Conf. Acoustics Speech Signal Process*, (Shanghai, China), pp. 4563 – 4567, 2016.

[110] W. Huang and A. Ribeiro, "Hierarchical clustering given confidence intervals of metric distances," *IEEE Trans. Signal Process.*, vol. (to appear), Feb. 2018.

[111] W. Huang and A. Ribeiro, "Network comparison: embeddings and interiors," *IEEE Trans. Signal Process.*, vol. 66, pp. 412–427, Feb. 2018.

[112] W. Huang, A. G. Marques, and A. Ribeiro, "Rating prediction via graph signal processing," *IEEE Trans. Signal Process.*, p. (revised), Apr. 2018.

[113] W. Huang, R. Ghrist, and A. Ribeiro, "Frequency representation of networks using persistent homology," *IEEE Trans. Signal Process.*, p. (submitted), Apr. 2018.

[114] W. Huang, L. Goldsberry, N. F. Wymbs, S. T. Grafton, D. S. Bassett, and A. Ribeiro, "Graph frequency analysis of brain signals," *J. Sel. Topics Signal Process.*, vol. 10, pp. 1189–1203, Oct. 2016.

[115] S. Segarra, W. Huang, and A. Ribeiro, "Diffusion and superposition distances for signals supported on networks," *IEEE Trans. Signal Inform. Process. Netw.*, vol. 1, pp. 20–32, Mar. 2015.

[116] J. D. Medaglia, W. Huang, S. Segarra, C. Olm, J. Gee, M. Grossman, A. Ribeiro, C. T. McMillan, and D. S. Bassett, "Brain network efficiency is influenced by the pathologic source of corticobasal syndrome," *Neurol.*, vol. 89, pp. 1373–1381, Aug. 2017.

[117] W. Huang*, T. A. W. Bolton*, J. D. Medaglia, D. S. Bassett, D. Van De Ville, and A. Ribeiro, "A graph signal processing perspective on functional brain imaging," *Proc. IEEE*, p. (to appear), Feb. 2018.

[118] W. Huang and A. Ribeiro, "Metrics in the space of high order proximity networks," in *IEEE Int. Conf. Acoust., Speech, Signal Process.*, pp. 4135–4139, Apr. 2015.

[119] W. Huang and A. Ribeiro, "Persistent homology approximations of network distances," in *Proc. Global Conf. Signal Info. Process.*, pp. 1002–1006, Dec. 2015.

[120] W. Huang, S. Segarra, and A. Ribeiro, "Diffusion distance for signals supported on networks," in *Proc. Asilomar Conf. on Signal. Syst. Comput.*, pp. 1219–1223, Nov. 2015.

[121] W. Huang and A. Ribeiro, "Persistent homology lower bounds on network distances," in *IEEE Int. Conf. Acoust., Speech, Signal Process.*, pp. 4845–4849, Mar. 2016.

[122] W. Huang and A. Ribeiro, "Persistent homology lower bounds on distances in the space of networks," in *Proc. Asilomar Conf. on Signal. Syst. Comput.*, pp. 72–76, Nov. 2016.

[123] W. Huang and A. Ribeiro, "Axiomatic hierarchical clustering for intervals of metric distances," in *Proc. Global Conf. Signal Info. Process.*, pp. 217–221, Dec. 2016.

[124] L. Goldsberry, W. Huang, N. F. Wymbs, S. T. Grafton, D. S. Bassett, and A. Ribeiro, "Brain signal analytics from graph signal processing perspective," in *Proc. Int. Conf. Accoustics Speech Signal Process.*, pp. 851–855, Mar. 2017.

[125] W. Huang, A. G. Marques, and A. Ribeiro, "Collaborative filtering via graph signal processing," in *IEEE Euro. Signal Process. Conf.*, pp. 1694–1698, Aug. 2017.

[126] W. Huang and A. Ribeiro, "Partial embedding distance for networks," in *Proc. Asilomar Conf. on Signals Syst. Comp.*, p. (to appear), Nov. 2017.

[127] W. Huang, A. G. Marques, and A. Ribeiro, "Matrix completion via graph signal processing," in *Proc. Int. Conf. Acoustics Speech Signal Process.*, p. (accepted), Apr. 2018.

[128] W. Huang, T. A. W. Bolton, J. D. Medaglia, D. S. Bassett, A. Ribeiro, and D. Van De Ville, "Graph signal processing of human brain imaging data," in *Proc. Int. Conf. Acoustics Speech Signal Process.*, p. (accepted), Apr. 2018.

[129] M. Gromov, *Metric structures for Riemannian and non- Riemannian spaces*. Birkha user Boston Inc., Boston, MA., 2007.

[130] F. Memoli, "Gromov-Hausdorff distances in Euclidean spaces," in *IEEE Conf. Comput. Vis. Pattern Recognit. Workshop*, pp. 1–8, 2008.

[131] D. Burago, Y. Burago, and S. Ivanov, *A Course in Metric Geometry*, vol. 33. American Mathematical Soc., 2001.

[132] C. Berge, *Graphs And Hypergraphs*. North-Holland Publishing Company, 1976.

[133] A. Bretto, *Hypergraph Theory: An Introduction*. Springer, 2013.

[134] A. N. Kolmogorov, S. V. Fomine, and R. A. Silverman, *Introductory Real Analysis*. Dover, 1975.

[135] "Engineering Village: the place to find answers to engineering questions."

[136] M. Lesnick, "The theory of the interleaving distance on multidimensional persistence modules," *Found. Comut. Math.*, vol. 15, pp. 613–650, Jun. 2015.

[137] K. Mischaikow and V. Nanda, "Morse theory for filtrations and efficient computation of persistent homology," *Discrete Comput. Geom.*, vol. 50, pp. 330–353, Sep. 2013.

[138] H. Edelsbrunner and J. L. Harer, *Computational Topology: An Introduction*. American Mathematical Soc., 2010.

[139] S. Y. Oudot, *Persistence Theory: From Quiver Representations To Data Analysis*, vol. 209. American Mathematical Society, 2015.

[140] R. Burkard, M. Dell'Amico, and S. Martello, *Assignment Problems*. Society for Industrial and Applied Mathematics, 2009.

[141] F. Chazal, D. Cohen-Steiner, L. J. Guibas, F. Mémoli, and S. Y. Oudot, "GromovHausdorff Stable Signatures for Shapes using Persistence," *Eurographics Symp. Geom. Process.*, vol. 28, no. 5, pp. 1393–1403, 2009.

[142] F. Chazal, V. De Silva, and S. Oudot, "Persistence stability for geometric complexes," *Geometriae Dedicata*, vol. 173, pp. 193–214, Dec. 2014.

[143] A. M. Davie and A. J. Stothers, "Improved bound for complexity of matrix multiplication,"

Proc. R. Soc. Edinb. Math., vol. 143, pp. 351–369, Apr. 2013.

[144] P. Erdős and A. Rényi, "On the evolution of random graphs," *Publ. Math. Inst. Hungar. Acad. Sci*, vol. 5, pp. 17–61, 1960.

[145] A. Tausz, M. Vejdemo-Johansson, and H. Adams, "JavaPlex: A research software package for persistent (co)homology," in *Proc. ICMS*, pp. 129–136, Feb. 2014.

[146] M. A. A. Cox and T. F. Cox, "Multidimensional scaling," in *Handbook of Data Visualization*, Springer Handbooks Comp. Statistics, pp. 315–347, Springer Berlin Heidelberg, 2008.

[147] E. Carlsson, G. Carlsson, and V. De Silva, "An algebraic topological method for feature identification," *Int. J. Comput. Geom. Appl.*, vol. 16, pp. 291–314, Aug. 2006.

[148] J.-C. Hausmann, "On the vietoris-rips complexes and a cohomology theory for metric spaces," *Ann. Math Stud.*, vol. 138, pp. 175–188, 1995.

[149] G. Carlsson, F. Mémoli, A. Ribeiro, and S. Segarra, "Axiomatic construction of hierarchical clustering in asymmetric networks," in *IEEE Int. Conf. Acoust., Speech, Signal Process.*, pp. 5219–5223, 2013.

[150] R. Klimberg, *Fundamentals Of Predictive Analytics With JMP*. SAS Institute, 2016.

[151] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, *et al.*, "A density-based algorithm for discovering clusters in large spatial databases with noise.," *Kdd*, vol. 96, pp. 226–231, Aug. 1996.

[152] R. J. Campello, D. Moulavi, and J. Sander, "Density-based clustering based on hierarchical density estimates," in *Pac.-Asia Conf. Knowl. Discov. Data Min.*, pp. 160–172, Apr. 2013.

[153] W. Ali, T. Rito, G. Reinert, F. Sun, and C. M. Deane, "Alignment-free protein interaction network comparison," *Bioinformatics*, vol. 30, no. 17, pp. 430–437, 2014.

[154] N. J. Kalton and M. I. Ostrovskii, "Distances between banach spaces," *Forum Math.*, vol. 11, no. 17-48, 1997.

[155] N. Linial, E. London, and Y. Rabinovich, "The geometry of graphs and some of its algorithmic applications," *Combinatorica*, vol. 15, pp. 215–245, Jun 1995.

[156] J. H. Ward Jr, "Hierarchical grouping to optimize an objective function," *J. Am. Stat. Assoc.*, vol. 58, pp. 236–244, Mar. 1963.

[157] S. Segarra, A. G. Marques, G. Leus, and A. Ribeiro, "Reconstruction of graph signals through percolation from seeding nodes," *IEEE Trans. Signal Process.*, vol. 64, pp. 4364–4378, Aug. 2016.

[158] G. Golub and C. V. Loan, *Matrix Computations*. Johns Hopkins University Press, 1989.

[159] R. I. Kondor and J. Lafferty, "Diffusion kernels on graphs and other discrete input spaces," in *Int. Conf. Mach. Learn.*, vol. 2, pp. 315–322, Jul. 2002.

[160] R. Bellman, *Introduction To Matrix Analysis*, vol. 960. SIAM, 1970.

[161] R. Varga, "Matrix iterative analysis," *Springer series in computational mathematics*, 2000.

[162] C. V. Loan, "The sensitivity of the matrix exponential," *SIAM J. Numer. Anal.*, vol. 14,

pp. 971–981, Dec. 1977.

[163] D. M. Young, *Iterative Solution Of Large Linear Systems*. New York, Academic Press, 1971.

[164] T. Fujimoto and R. Ranade, "Two characterizations of inverse-positive matrices: the hawkins-simon condition and the le chatelier-braun principle," *Electron. J. Linear Al.*, vol. 11, pp. 59–65, 2004.

[165] M. Hofree, J. Shen, H. Carter, A. Gross, and T. Ideker, "Network-based stratification of tumor mutations," *Nat. Methods*, Nov. 2013.

[166] E. G. Cerami, B. E. Gross, E. Demir, I. Rodchenkov, O. Babur, N. Anwar, N. Schultz, G. Bader, and C. Sander, "Pathway commons, a web resource for biological pathway data," *Nucleic. Acids. Res.*, vol. 39, pp. D685–D690, Nov. 2011.

[167] Y. Lecun and C. Cortes, "The MNIST database of handwritten digits."

[168] J. A. Kleim, S. Barbay, N. R. Cooper, T. M. Hogg, C. N. Reidel, M. S. Remple, and R. J. Nudo, "Motor learning-dependent synaptogenesis is localized to functionally reorganized motor cortex," *Neurobiol. Learn. Mem.*, vol. 77, pp. 63–77, Jan. 2002.

[169] D. S. Bassett, M. Yang, N. F. Wymbs, and S. T. Grafton, "Learning-induced autonomy of sensorimotor systems," *Nat. Neurosci.*, vol. 18, pp. 744–751, May 2015.

[170] D. S. Bassett, N. F. Wymbs, M. P. Rombach, M. A. Porter, P. J. Mucha, and S. T. Grafton, "Task-based core-periphery organization of human brain dynamics," *PLoS Comput. Biol.*, vol. 9, p. e1003171, Sep. 2013.

[171] D. S. Bassett, N. F. Wymbs, M. A. Porter, P. J. Mucha, J. M. Carlson, and S. T. Grafton, "Dynamic reconfiguration of human brain networks during learning," *Proc. Natl. Acad. Sci. USA*, vol. 108, pp. 7641–7646, May 2011.

[172] N. F. Wymbs, D. S. Bassett, P. J. Mucha, M. A. Porter, and S. T. Grafton, "Differential recruitment of the sensorimotor putamen and frontoparietal cortex during motor chunking in humans," *Neuron.*, vol. 74, pp. 936–946, Jun. 2012.

[173] J. Wang, L. Wang, Y. Zang, H. Yang, H. Tang, Q. Gong, Z. Chen, C. Zhu, and Y. He, "Parcellation-dependent small-world brain functional networks: A resting-state fmri study," *Hum. Brain. Mapp.*, vol. 30, pp. 1511–1523, May 2009.

[174] J. D. Power, A. L. Cohen, S. M. Nelson, G. S. Wig, K. A. Barnes, J. A. Church, A. C. Vogel, T. O. Laumann, F. M. Miezin, B. L. Schlaggar, *et al.*, "Functional network organization of the human brain," *Neuron.*, vol. 72, pp. 665–678, Nov. 2011.

[175] F. T. Sun, L. M. Miller, and M. D'Esposito, "Measuring interregional functional connectivity using coherence and partial coherence analyses of fmri data," *Neuroimage*, vol. 21, pp. 647–658, Feb. 2004.

[176] Y. He, Z. J. Chen, and A. C. Evans, "Small-world anatomical networks in the human brain revealed by cortical thickness from mri," *Cereb. cortex.*, vol. 17, pp. 2407–2419, Jan. 2007.

[177] C. R. Genovese, N. A. Lazar, and T. Nichols, "Thresholding of statistical maps in functional neuroimaging using the false discovery rate," *Neuroimage*, vol. 15, pp. 870 – 878, Apr. 2002.

[178] M. Xia, J. Wang, Y. He, *et al.*, “Brainnet viewer: a network visualization tool for human brain connectomics,” *PloS one*, vol. 8, p. e68910, Jul. 2013.

[179] F. Klimm, D. S. Bassett, J. M. Carlson, and P. J. Mucha, “Resolving structural variability in network models and the brain,” *PLoS Comput. Biol.*, vol. 10, p. e1003491, Mar. 2014.

[180] C. Lohse, D. S. Bassett, K. O. Lim, and J. M. Carlson, “Resolving anatomical and functional structure in human brain organization: Identifying mesoscale organization in weighted network representations,” *PLoS Comput. Biol.*, vol. 10, p. e1003712, Oct. 2014.

[181] R. Albert and A.-L. Barabási, “Statistical mechanics of complex networks,” *Rev. Mod. Phys.*, vol. 74, p. 47, Jan. 2002.

[182] C. Aicher, A. Z. Jacobs, and A. Clauset, “Learning latent block structure in weighted networks,” *J. Complex Netw.*, vol. 3, pp. 221–248, Jun. 2014.

[183] V. Braitenberg and A. Schüz, *Cortex: Statistics And Geometry Of Neuronal Connectivity*. Springer Science & Business Media, 2013.

[184] D. A. Rosenbaum, *Human Motor Control*. Academic press, 2009.

[185] E. Dayan and L. G. Cohen, “Neuroplasticity subserving motor skill learning,” *Neuron.*, vol. 72, pp. 443–454, Nov. 2011.

[186] A. Heathcote, S. Brown, and D. Mewhort, “The power law repealed: The case for an exponential law of practice,” *Psychon. Bull. Rev.*, vol. 7, pp. 185–207, Jun. 2000.

[187] P. Hagmann, L. Cammoun, X. Gigandet, R. Meuli, C. J. Honey, V. J. Wedeen, and O. Sporns, “Mapping the structural core of human cerebral cortex,” *PLoS Biol.*, vol. 6, p. e159, Jul. 2008.

[188] A. Zalesky, A. Fornito, I. H. Harding, L. Cocchi, M. Yücel, C. Pantelis, and E. T. Bullmore, “Whole-brain anatomical networks: does the choice of nodes matter?,” *Neuroimage*, vol. 50, pp. 970–983, Apr. 2010.

[189] R. S. Desikan, F. Ségonne, B. Fischl, B. T. Quinn, B. C. Dickerson, D. Blacker, R. L. Buckner, A. M. Dale, R. P. Maguire, B. T. Hyman, *et al.*, “An automated labeling system for subdividing the human cerebral cortex on MRI scans into gyral based regions of interest,” *Neuroimage*, vol. 31, pp. 968–980, Jul. 2006.

[190] D. Kennedy, N. Lange, N. Makris, J. Bates, J. Meyer, and V. Caviness, “Gyri of the human neocortex: an MRI-based analysis of volume and variance.,” *Cereb. Cortex.*, vol. 8, pp. 372–384, Jun. 1998.

[191] S. Gu, F. Pasqualetti, M. Cieslak, Q. K. Telesford, A. B. Yu, A. E. Kahn, J. D. Medaglia, J. M. Vettel, M. B. Miller, S. T. Grafton, and D. S. Bassett, “Controllability of structural brain networks,” *Nat. Commun.*, vol. 6, p. 8414, Oct. 2015.

[192] F.-C. Yeh, V. J. Wedeen, and W.-Y. I. Tseng, “Estimation of fiber orientation and spin density distribution by diffusion deconvolution,” *Neuroimage*, vol. 55, pp. 1054–1062, Apr. 2011.

[193] B. Fischl, “FreeSurfer,” *Neuroimage*, vol. 62, pp. 774–781, Aug. 2012.

[194] L. Cammoun, X. Gigandet, D. Meskaldji, J. P. Thiran, O. Sporns, K. Q. Do, P. Maeder, R. Meuli, and P. Hagmann, “Mapping the human connectome at multiple scales with diffu-

sion spectrum MRI," *J. Neurosci. Methods.*, vol. 203, pp. 386–397, Jan. 2012.

[195] M. Cieslak and S. Grafton, "Local termination pattern analysis: a tool for comparing white matter morphology," *Brain Imaging Behav.*, vol. 8, pp. 292–299, Jun. 2014.

[196] A. M. Hermundstad, D. S. Bassett, K. S. Brown, E. M. Aminoff, D. Clewett, S. Freeman, A. Frithsen, A. Johnson, C. M. Tipper, M. B. Miller, S. T. Grafton, and J. M. Carlson, "Structural foundations of resting-state and task-based functional connectivity in the human brain," *Proc. Natl. Acad. Sci. U.S.A.*, vol. 110, pp. 6169–6174, Apr. 2013.

[197] U. Braun, S. F. Muldoon, and D. S. Bassett, "On human brain networks in health and disease," *eLS*, Feb. 2015.

[198] W. Gaetz, L. Bloy, D. Wang, R. Port, L. Blaskey, S. Levy, and T. P. Roberts, "GABA estimation in the brains of children on the autism spectrum: measurement precision and regional cortical variation," *Neuroimage*, vol. 86, pp. 1–9, Feb. 2014.

[199] M. Jenkinson, C. F. Beckmann, T. E. Behrens, M. W. Woolrich, and S. M. Smith, "Fsl," *Neuroimage*, vol. 62, pp. 782–790, Aug. 2012.

[200] S. M. Smith, "BET: Brain extraction tool," *FMRIB TR00SMS2b, Oxford Centre for Functional Magnetic Resonance Imaging of the Brain), Department of Clinical Neurology, Oxford University, John Radcliffe Hospital, Headington, UK*, 2000.

[201] D. N. Greve and B. Fischl, "Accurate and robust brain image alignment using boundary-based registration," *Neuroimage*, vol. 48, pp. 63–72, Oct. 2009.

[202] E. Tagliazucchi, P. Balenzuela, D. Fraiman, and D. R. Chialvo, "Brain resting state is disrupted in chronic back pain patients," *Neurosci. Lett.*, vol. 485, pp. 26–31, Nov. 2010.

[203] K. Christoff, Z. C. Irving, K. C. Fox, R. N. Spreng, and J. R. Andrews-Hanna, "Mind-wandering as spontaneous thought: a dynamic framework," *Nat. Rev. Neurosci.*, vol. 17, pp. 718–731, Sep. 2016.

[204] M. P. van den Heuvel, C. J. Stam, R. S. Kahn, and H. E. H. Pol, "Efficiency of functional brain networks and intellectual performance," *J. Neurosci.*, vol. 29, pp. 7619–7624, Jun. 2009.

[205] H. Haken, *Principles Of Brain Functioning: A synergetic Approach To Brain Activity, Behavior And Cognition*, vol. 67. Springer Science & Business Media, 2013.

[206] D. D. Garrett, G. R. Samanez-Larkin, S. W. MacDonald, U. Lindenberger, A. R. McIntosh, and C. L. Grady, "Moment-to-moment brain signal variability: A next frontier in human brain mapping?," *Neurosci. Biobehav. Rev.*, vol. 37, pp. 610–624, May 2013.

[207] D. S. Bassett, N. F. Wymbs, M. A. Porter, P. J. Mucha, J. M. Carlson, and S. T. Grafton, "Dynamic reconfiguration of human brain networks during learning," *Proc. Natl. Acad. Sci. U.S.A.*, vol. 108, pp. 7641–7646, May 2011.

[208] G. J. Thompson, M. E. Magnuson, M. D. Merritt, H. Schwarb, W.-J. Pan, A. McKinley, L. D. Tripp, E. H. Schumacher, and S. D. Keilholz, "Short-time windows of correlation between large-scale functional brain networks predict vigilance intraindividually and interindividually," *Hum. Brain Mapp.*, vol. 34, pp. 3280–3298, Jun. 2012.

[209] O. Sporns, "Contributions and challenges for network models in cognitive neuroscience,"

Nature Neurosci., vol. 17, pp. 652–660, May 2014.

[210] J. Theiler, S. Eubank, A. Longtin, B. Galdrikian, and J. Doyne Farmer, "Testing for nonlinearity in time series: the method of surrogate data," *Physica D*, vol. 58, pp. 77–94, Sep. 1992.

[211] E. Pirondini, A. Vybornova, M. Coscia, and D. Van De Ville, "Spectral method for generating surrogate graph signals," *IEEE Signal Process. Lett.*, vol. 23, pp. 1275–1278, Sep. 2016.

[212] S. Mallat, *A Wavelet Tour Of Signal Processing*. Academic Press, 2009.

[213] M. Crovella and E. Kolaczyk, "Graph wavelets for spatial traffic analysis," in *IEEE Joint Conf. Comput. Commun.*, vol. 3, pp. 1848–1857, Mar. 2003.

[214] M. Jansen, G. P. Nason, and B. W. Silverman, "Multiscale methods for data on graphs and irregular multidimensional situations," *J. R. Stat. Soc. Ser. B Stat. Methodol.*, vol. 71, pp. 97–125, Sep. 2008.

[215] S. K. Narang and A. Ortega, "Perfect reconstruction two-channel wavelet filter banks for graph structured data," *IEEE Trans. Signal Process.*, vol. 60, pp. 2786–2799, May 2012.

[216] R. R. Coifman and M. Maggioni, "Diffusion wavelets," *Appl. Comput. Harmon. Anal.*, vol. 21, pp. 53–94, Jul. 2006.

[217] R. Talmon, I. Cohen, S. Gannot, and R. R. Coifman, "Diffusion maps for signal processing: A deeper look at manifold-learning techniques based on kernels and graphs," *IEEE Signal Process. Mag.*, vol. 30, pp. 75–86, Jun. 2013.

[218] D. K. Hammond, P. Vandergheynst, and R. Gribonval, "Wavelets on graphs via spectral graph theory," *Appl. Comput. Harmon. Anal.*, vol. 30, pp. 129–150, Mar. 2011.

[219] N. Leonardi, J. Richiardi, M. Gschwind, S. Simioni, J.-M. Annoni, M. Schluep, P. Vuilleumier, and D. Van De Ville, "Principal components of functional connectivity: a new approach to study dynamic brain connectivity during rest," *NeuroImage*, vol. 83, pp. 937–950, Dec. 2013.

[220] H. Behjat, U. Richter, D. Van De Ville, and L. Sornmo, "Signal-adapted tight frames on graphs," *IEEE Trans. Signal Process.*, vol. 64, pp. 6017–6029, Nov. 2016.

[221] N. Tremblay and P. Borgnat, "Graph wavelets for multiscale community mining," *IEEE Trans. Signal Process.*, vol. 62, pp. 5227–5239, Oct. 2014.

[222] D. Van De Ville, R. Demesmaeker, and M. G. Preti, "When slepian meets fiedler: Putting a focus on the graph spectrum," *IEEE Signal Process. Lett.*, vol. 24, pp. 1001–1004, Jul. 2017.

[223] D. Navon, "Forest before trees: The precedence of global features in visual perception," *Cogn. Psychol.*, vol. 9, pp. 353–383, Jul. 1977.

[224] U. Braun, A. Schäfer, H. Walter, S. Erk, N. Romanczuk-Seiferth, L. Haddad, J. I. Schweiger, O. Grimm, A. Heinz, H. Tost, *et al.*, "Dynamic reconfiguration of frontal brain networks during executive cognition in humans," *Proc. Natl. Acad. Sci. U.S.A.*, vol. 112, pp. 11678–11683, Sep. 2015.

[225] I. Leunissen, J. P. Coxon, K. Caeyenberghs, K. Michiels, S. Sunaert, and S. P. Swinnen, "Subcortical volume analysis in traumatic brain injury: the importance of the fronto-striato-

thalamic circuit in task switching," *Cortex*, vol. 51, pp. 67–81, Feb. 2014.

[226] R. F. Betzel, M. Fukushima, Y. He, X.-N. Zuo, and O. Sporns, "Dynamic fluctuations coincide with periods of high and low modularity in resting-state functional brain networks," *Neuroimage*, vol. 127, pp. 287–297, Feb. 2016.

[227] R. Liégeois, E. Ziegler, C. Phillips, P. Geurts, F. Gómez, M. A. Bahri, B. T. Yeo, A. Soddu, A. Vanhaudenhuyse, S. Laureys, *et al.*, "Cerebral functional connectivity periodically (de) synchronizes with anatomical constraints," *Brain Struct. Funct.*, vol. 221, pp. 2985–2997, Jul. 2016.

[228] M. D. Fox, A. Z. Snyder, J. L. Vincent, M. Corbetta, D. C. Van Essen, and M. E. Raichle, "The human brain is intrinsically organized into dynamic, anticorrelated functional networks," *Proc. Natl. Acad. Sci. U.S.A.*, vol. 102, pp. 9673–9678, May 2005.

[229] J. S. Breese, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," in *Conf. Uncertain. Artif. Intell.*, pp. 43–52, Jul. 1998.

[230] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms," in *ACM Int. Conf. World Wide Web*, pp. 285–295, Apr. 2001.

[231] A. Kheradmand and P. Milanfar, "A general framework for regularized, similarity-based image restoration," *IEEE Trans. Image Process.*, vol. 23, pp. 5136–5151, Dec. 2014.

[232] S. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky, "A method for large-scale l1-regularised least squares problems with applications in signal processing and statistics," *IEEE J. Select. Topics Signal Process*, vol. 1, pp. 606–617, Dec. 2007.

[233] K. P. Burnham and D. R. Anderson, *Model Selection And Multimodel Inference: A Practical Information-Theoretic Approach*. Springer Science and Business Media, 2003.

[234] J. M. Duarte-Carvajalino and G. Sapiro, "Learning to sense sparse signals: Simultaneous sensing matrix and sparsifying dictionary optimization," *IEEE Trans. Image Process.*, vol. 18, pp. 1395–1408, Jul. 2009.

[235] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge university press, 2004.