

ESSAYS IN
LEARNING REPRESENTATIONS OF COMPLEX NETWORKS

FURKAN GÜRSOY

BOĞAZIÇI UNIVERSITY

2021

ESSAYS IN
LEARNING REPRESENTATIONS OF COMPLEX NETWORKS

Thesis submitted to the
Institute for Graduate Studies in Social Sciences
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Management Information Systems

by
Furkan Gürsoy

Boğaziçi University

2021

DECLARATION OF ORIGINALITY

I, Furkan Gürsoy, certify that

- I am the sole author of this thesis and that I have fully acknowledged and documented in my thesis all sources of ideas and words, including digital resources, which have been produced or published by another person or institution;
- this thesis contains no material that has been submitted or accepted for a degree or diploma in any other educational institution;
- this is a true copy of the thesis approved by my advisor and thesis committee at Boğaziçi University, including final revisions required by them.

Signature:

Date:

ABSTRACT

Essays in Learning Representations of Complex Networks

This thesis contains three essays in learning representation of complex networks, the first two of which develop new methods and the third utilizes these methods in a real-world application. The first essay provides methods for extracting underlying signed network backbones from intrinsically dense weighted networks. Utilizing a null model based on statistical techniques, we propose significance and vigor filters that enable inferring edge signs and weights. Empirical analysis on four real-world networks reveals that the proposed filters extract meaningful and sparse signed backbones that exhibit characteristics typically associated with signed networks while respecting the multiscale nature of the network. The second essay deals with the misalignment problem in dynamic representation learning. We provide the first formal definitions of alignment and stability, propose novel metrics for measuring them, and show their suitability through a set of synthetic and real-world experiments. We show that, by ensuring alignment, the performance of dynamic network inference tasks improves by a remarkable amount. The third essay applies the novel methods developed in the first two essays as well as other methods from the network analysis literature to investigate the structure and dynamics of internal migration in Turkey. In addition to providing unique and specific insights, we find that most migration links are geographically bounded with exceptions of cities with large economic activity, migration takes place in well-defined routes, counter-streams develop for major migration streams, and the migration system is largely stable over time; which are generally in line with classical migration laws.

ÖZET

Karmaşık Ağların Temsillerini Öğrenme Üzerine Makaleler

Bu tez, karmaşık ağların temsillerini öğrenme üzerine yazılmış, ilk ikisi yeni yöntemler geliştiren ve üçüncüsü bu yöntemleri bir gerçek hayat problemine uygulayan üç makale içermektedir. İlk makale, doğası itibarıyla yoğun ağların temelindeki işaretli ağ omurgalarını ayırıp çıkararak yöntemler sağlamaktadır. İstatistiksel tekniklere dayalı bir sıfır modeli kullanarak, bağ işaretlerini ve ağırlıklarını çıkararak önem ve dinçlik filtrelerini tasarlıyoruz. Dört gerçek hayat ağı üzerindeki deneysel incelemeler, önerilen filtrelerin bir ağın çok ölçekli yapısını da göz önünde bulundurarak, sıklıkla işaretli ağlarla ilişkilendirilen niteliklere sahip, anlamlı ve seyrek işaretli omurgalar elde edebildiğini göstermektedir. İkinci makale, zamana bağlı ağ temsillerini öğrenmedeki hizasızlık problemiyle ilgilenmektedir. Hizalanma ve kararlılık için ilk muntazam tanımları sağlıyoruz, bunları ölçümlemek için özgün ölçüler tasarlıyoruz ve yapay ve gerçek hayat deneyleriyle tüm bunların kullanılabilirliğini gösteriyoruz. Hizalanmayı sağlayarak, zamana bağlı ağ çıkarım problemlerindeki performansın önemli ölçüde artırıldığını gösteriyoruz. Üçüncü makale, ilk iki makalede geliştirilen özgün yöntemleri ve ağ analizi kaynaklarındaki diğer yöntemleri, Türkiye'deki iç göçlerin yapısını ve hareketlerini incelemek için kullanmaktadır. Özgün ve belirli içgörüler sağlamaya ek olarak, büyük ekonomik etkinliğe sahip şehirler haricinde çoğu göç bağının coğrafi olarak kısıtlandığını, göçlerin çizgileri belli güzergahlar üzerinde gerçekleştiğini, önemli göç akışları için karşıt göç akışlarının geliştiğini, göç sisteminin genel olarak kararlı olduğunu ve tüm bunların klasik göç ilkeleriyle uyumlu olduğunu gösteriyoruz.

CURRICULUM VITAE

NAME: Furkan Gürsoy

DEGREES AWARDED

PhD in Management Information Systems, 2021, Boğaziçi University

MS in Data Science, 2017, İstanbul Şehir University

BS in Management Information Systems, 2014, Boğaziçi University

AREAS OF SPECIAL INTEREST

Network Science, Machine Learning, Complex Systems, Simulation, Data Science

AWARDS AND HONORS

Rectorate Award, Boğaziçi University, 2017

First Ranked, School of Applied Disciplines (GPA: 3.91), Boğaziçi University, 2014

GRANTS AND SCHOLARSHIP

French Embassy Research Fellowship, 2020

Sci. and Tech. Res. Council of Turkey (TÜBİTAK) 2211-A Scholarship, 2020-2021

Council of Higher Education, Turkey (YÖK) 100/2000 Scholarship, 2017-2021

Sci. and Tech. Res. Council of Turkey (TÜBİTAK) 2210-A Scholarship, 2014-2017

PUBLICATIONS

Journal Articles

Gürsoy, F., & Badur, B. (in press). Extracting the signed backbone of intrinsically dense weighted networks. *Journal of Complex Networks*.

Gürsoy, F., & Badur, B. (2021). An agent-based modeling approach to brain drain. *IEEE Transactions on Computational Social Systems*, (Early Access).

Altın, M., Gürsoy, F., & Xu, L. (2021). Machine-generated hierarchical structure of human activities to reveal how machines think. *IEEE Access*, 9, 18307–18317.

Özturan, M., Gürsoy, F., & Çeken, B. (2019). An empirical analysis on the effects of investment assessment methods on IS/IT project success. *International Journal of Information Systems and Project Management*, 7(4), 33–52.

Gürsoy, F., & Günneç, D. (2018). Influence maximization in social networks under deterministic linear threshold model. *Knowledge-Based Systems*, 161, 111–123.

Conference Proceedings

Gürsoy, F., & Durahim, A. O. (2018). Predicting diffusion reach probabilities via representation learning on social networks. *Proceedings of the 5th International Management Information Systems Conference, Ankara, Turkey*.

Gürsoy, F., & Badur, B. (2018). A community-aware network growth model for synthetic social network generation. *Proceedings of the 5th International Management Information Systems Conference, Ankara, Turkey*.

ACKNOWLEDGMENTS

First and foremost, I would like to express my gratitude to my advisor, Prof. Bertan Badur, who supported me since my undergraduate studies and inspired me to pursue a path in academia. Without his precious feedback and guidance, this thesis in particular and my research in general would not have become what they are now.

Besides my advisor, for their valuable feedback and suggestions, I am grateful to Prof. Sona Mardikyan and Prof. Nazım Ziya Perdahçı who agreed to be in my thesis progress committee and thesis defense committee, and Prof. Meltem Özturan and Prof. Mehmet Nafiz Aydın who agreed to be in my thesis defense committee. I would like to extend my gratitude to each and every member of the Department of Management Information Systems at Boğaziçi University, for their continuous support and guidance within and outside the curriculum.

I would like to thank Prof. Cécile Bothorel and Mounir Haddad who welcomed with during my research visit at IMT Atlantique and provided valuable guidance and input, which resulted in the work presented in Chapter 3. I thank the French Embassy in Turkey for the fellowship that funded my research visit. I would also like to thank the international research community around complex networks, particularly the participants and organizers of the yrCSS, WWCS, Complexity72h, and CNWW, who became friends and collaborators and widened my horizon.

The author is financially supported by the Scientific and Technological Research Council of Turkey (TUBITAK) under the 2211-A Program (2020-2021) and the Council of Higher Education (YÖK) in Turkey under the 100/2000 Program (2017-2021).

TABLE OF CONTENTS

| | |
|---|-----|
| CHAPTER 1: INTRODUCTION | 1 |
| CHAPTER 2: EXTRACTING THE SIGNED BACKBONE OF INTRINSICALLY DENSE WEIGHTED NETWORKS | 7 |
| 2.1 Introduction..... | 7 |
| 2.2 Backbone extraction methods..... | 11 |
| 2.3 Methods | 16 |
| 2.4 Empirical analysis | 23 |
| 2.5 Conclusion | 34 |
| 2.6 Software package..... | 36 |
| CHAPTER 3: ALIGNMENT AND STABILITY OF EMBEDDINGS: MEASUREMENT AND INFERENCE IMPROVEMENT | 39 |
| 3.1 Introduction..... | 39 |
| 3.2 Background..... | 42 |
| 3.3 Stability and alignment..... | 45 |
| 3.4 Experiments on synthetic networks | 50 |
| 3.5 Experiments on real-world networks..... | 59 |
| 3.6 Conclusion | 69 |
| 3.7 Software package..... | 70 |
| CHAPTER 4: INVESTIGATING INTERNAL MIGRATION WITH NETWORK ANALYSIS AND REPRESENTATION LEARNING: AN APPLICATION TO TURKEY | 71 |
| 4.1 Introduction..... | 71 |
| 4.2 Related work..... | 73 |
| 4.3 Material and methods | 76 |
| 4.4 Results and findings | 85 |
| 4.5 Conclusion | 96 |
| CHAPTER 5: CONCLUSION..... | 98 |
| REFERENCES | 118 |

LIST OF TABLES

| | | |
|----------|---|----|
| Table 1. | Confidence Intervals in the OECD Trade Network | 20 |
| Table 2. | Employed Networks | 23 |
| Table 3. | Reciprocity in Extracted Directed Backbones | 32 |
| Table 4. | Structural Characteristics of Extracted Backbones | 33 |
| Table 5. | Block Memberships | 35 |
| Table 6. | Orthogonal Matrix Canonical Form | 48 |
| Table 7. | Datasets | 61 |
| Table 8. | Prediction Accuracy Before and After Alignment | 66 |

LIST OF FIGURES

| | | |
|------------|---|----|
| Figure 1. | Hypergeometric backbones of the OECD trade network | 20 |
| Figure 2. | Effects of significance filter on the backbone size | 27 |
| Figure 3. | Effects of vigor filter on the backbone size | 28 |
| Figure 4. | Original weights of the retained edges | 30 |
| Figure 5. | Original strength distribution of the retained dyads | 31 |
| Figure 6. | Block models of extracted backbones | 33 |
| Figure 7. | An example of misalignment | 43 |
| Figure 8. | Tangent embeddings regarding their radii | 46 |
| Figure 9. | Magnitude of rotation versus rotation error | 51 |
| Figure 10. | Scale and translation errors versus magnitudes of the respective transformations | 53 |
| Figure 11. | Combined scaling and shifting versus scale and translation errors . . . | 54 |
| Figure 12. | The relationship between noise and stability error | 56 |
| Figure 13. | Combined effect of noise and scaling, shifting, or rotation on stability error | 57 |
| Figure 14. | Combined effect of noise and scaling, shifting, or rotation on scale, translation, and rotation errors | 58 |
| Figure 15. | Alignment and stability errors produced by embedding methods | 63 |
| Figure 16. | Distribution of prediction score improvement after alignment | 65 |
| Figure 17. | Effect of synthetic misalignment on prediction accuracy | 68 |
| Figure 18. | Prediction accuracy versus stability error of the embeddings | 68 |
| Figure 19. | Internal migration trend in Turkey | 86 |
| Figure 20. | Migration flow volume characteristics | 87 |
| Figure 21. | Characteristics of migration backbones | 89 |
| Figure 22. | Maps of Turkey | 90 |
| Figure 23. | Spatial network of migration backbone in 2020 | 91 |
| Figure 24. | Spatial network of migration backbone in 2008 | 91 |

| | | |
|------------|---|----|
| Figure 25. | Link persistence in migration backbones | 94 |
| Figure 26. | Embedding loss | 94 |
| Figure 27. | Stability error over time | 95 |
| Figure 28. | Density-based clusters | 96 |
| Figure 29. | Hierarchical clustering dendrograms | 97 |

CHAPTER 1

INTRODUCTION

A network refers to a system or group of interconnected or interrelated things, which can be visualized as a collection of points and lines that connect them (Menczer, Fortunato, & Davis, 2020; Newman, 2018). The points and links are usually referred to as vertices and edges in some fields such as mathematics, as nodes and links in some fields such as computer science, and as actors and ties in some fields such as sociology. Many systems surrounding our lives can be modeled as networks. Internet is a collection of computers linked via data connections. The web is a collection of web pages linked via hyperlinks. Human societies are a collection of people linked via social interactions. Neural systems are collections of neurons linked via dendrites and axons. World trade can be represented as networks where nodes are the countries and links are the trade between them. Food chains in an ecosystem can be represented by predator-prey relationships between species. All these and countless other systems constitute examples of technological, social, biological, information, or other types of networks (Newman, 2003).

Torres, Blevins, Bassett, and Eliassi-Rad (2020) provide definitions for systems, relations, and dependencies; and discuss the formalisms for representing complex systems. A system is a collection of units (e.g., nodes) and all relations between them such that the collection does not need any other piece to function in general. A relation (e.g., a link) may arise from the correlations or interactions between the units. Dependency in a system refers to the information provided by a relation about units or other relations, including temporal dependencies. A system may be represented via various mathematical or computational encodings. Networks are among the most common formalisms employed to represent systems.

Theoretical roots of network analysis broadly lie in sociology and mathematics. Mathematical roots of network analysis stem from graph theory, which is widely accepted to be founded by Euler (1736). The sociological roots go back to

the hand-drawn diagrams of Moreno (1934), which explore the relationship between psychological well-being and social configurations, and the formation of groups. Today, network analysis (or network science) is viewed more as an interdisciplinary field with many neuroscientists, sociologists, linguists, computer scientists, statistical physicists, and others working on it with methods and problems brought from their respective fields.

The structure of networks mainly refers to the topological patterns of networks. Communities and other group structures in the network, distances between nodes, the position and centrality of nodes, and the patterns of link formations are among many notions that can be studied while analyzing the structure of a network. Networks also evolve over time. Some nodes and links die, new nodes and links emerge. Networks grow and shrink. Nodes interact with each other. Node and link metadata may change. Dynamics of networks refer to the evolution of networks over time whereas dynamics on networks refer to the process that happens on networks. The structure and dynamics of networks are interwoven and cannot be studied in isolation.

Simple networks are those that consist only of nodes and undirected links between them. Directed networks have link direction information available. Weighted networks contain weights associated with links. Attributed networks contain node metadata. Signed networks contain not only positive links but also negative links. Bipartite networks include two sets of nodes where links exist only between the nodes from different sets. Temporal (dynamic) networks contain individual network snapshots at discrete times or the changes happening in the network in continuous time. The exhaustive list of types and kinds of networks goes beyond the ones mentioned here and new types emerge in response to the systems that are more complex and rich in data. Therefore, choosing what level of detail to encode from a system and what type of network to employ to represent a system is a noteworthy decision.

On the other hand, the advances in machine learning brought up a new wave that aims to utilize highly-capable machine learning algorithms on networks or develop new machine learning algorithms that are suitable for network data. Most machine learning algorithms require input data to be in a tabular form in which rows represent objects and columns represent features. As the feasibility and difficulty level of many information processing tasks depend on how the information is represented, representation learning (RL) aims to find good representations of the data that can be used potentially in multiple subsequent tasks (Goodfellow, Bengio, & Courville, 2016). A particular task in RL for networks is called network embedding and it is concerned with designing methods that can embed the network to a latent space in a way that the latent representations contain as much information as possible within a reasonable (i.e., lower) dimensionality.

The growing size and variety of data such as more nodes and links and associated weights, directions, and signs can provide accessory information. Link and weight abundance, on the other hand, results in denser networks with noisy, insignificant, or otherwise redundant data. Moreover, typical network analysis and visualization techniques presuppose sparsity and are not appropriate or scalable for dense and weighted networks. As a remedy, network backbone extraction methods aim to retain only the important links while preserving the useful and elucidative structure of the original networks for further analyses. In Chapter 2, we provide the first methods in the literature for extracting signed network backbones from intrinsically dense unsigned unipartite weighted networks. Utilizing a null model based on statistical techniques, the proposed significance filter and vigor filter allow inferring edge signs. Empirical analysis on migration, voting, temporal interaction, and species similarity networks reveals that the proposed filters extract meaningful and sparse signed backbones while preserving the multiscale nature of the network. The resulting backbones exhibit characteristics typically associated with signed networks such as reciprocity, structural balance, and community structure.

In Chapter 2, we contribute to the literature first by providing an extensive review of existing network backbone extraction methods. Then, we coin the term intrinsically dense networks and characterize such networks for the first time in the literature. We design novel backbone extraction methods to represent intrinsically dense networks via their underlying sparse signed backbones and demonstrate the feasibility and usefulness of our methods through a set of empirical analyses supported by relevant social theories. As tools and techniques from the broad network science literature are increasingly useful in diverse disciplines, we believe our methods have a potential for wide adoption in studies that investigate systems with an intrinsically dense nature.

As network embedding methods (or other RL methods in the same vein) learn objects' latent embeddings where information is preserved by distances and since distances are invariant to certain linear transformations, one may obtain different embeddings while preserving the same information. In dynamic systems, a temporal difference in embeddings may be explained by the stability of the system or by the misalignment of embeddings due to arbitrary transformations. In Chapter 3, we explore the embedding alignment and its parts, provide the first formal definitions, propose novel metrics to measure alignment and stability, and show their suitability through synthetic experiments. Real-world experiments show that both static and dynamic RL methods produce misaligned embeddings and such misalignment worsens the performance of dynamic network inference tasks. By ensuring alignment, the prediction accuracy raises by up to 90% in static and by up to 40% in dynamic RL methods.

In the literature, embedding alignment has not been defined formally, explored theoretically, or analyzed empirically. In Chapter 3, our main contribution is dismantling and exploring the alignment while providing formal definitions and development of appropriate and mathematically justified novel metrics to measure the alignment and stability of dynamic embeddings. We further provide strategies to

realign misaligned embeddings as we show via real-world experiments that even dynamic network embedding methods are prone to generate misaligned embeddings.

The methods and measures developed in Chapter 2 and 3 might be useful in various problems in diverse domains. Therefore, we make them available via free software packages with open-source licenses. The proposed filters used for signed backbone extraction in Chapter 2 are provided as software packages in Python and in R. The metrics to measure alignment and stability errors of given embeddings and the methods to align them, as proposed in Chapter 3, are also provided as a software package in Python.

In Chapter 4, we utilize methods and tools presented in Chapter 2 and 3 as well as other network analysis methods from the literature on the dynamic internal migration system of Turkey from 2008 to 2020. As migration patterns influence the redistribution of population characteristics over the geography and since such distributions are closely related to social and economic outcomes, investigating the structure and dynamics of internal migration plays a crucial role in understanding and designing policies for such systems. We identify a set of classical migration laws and examine them via various methods for signed network analysis, ego network analysis, representation learning, temporal stability analysis, community detection, and network visualization. Overall, our findings indicate that the internal migration system of Turkey agrees with the general migration laws in addition to providing unique insights for the specific case of Turkey.

In Chapter 4, our major contributions are two-fold. Our first contribution lies in the tools and techniques that we utilize. We employ signed network analysis and network embedding techniques on migration systems at an aggregate level for the first time in the literature, which enables us to investigate selected migration laws from a complex networks perspective. Utilization of our proposed signed backbone extraction method from Chapter 2 allows us to capture the relevant and useful information to represent the migration system. Incorporating negative links, which are not available without a signed backbone extraction method, enables us to approach

the migration system using theories from the signed social networks literature, hence, enabling additional insights. Moreover, via the latent low-dimensional representation of the system that is generated with our network embedding strategy based on signed networks, we are able to utilize methods that reveal a density-based and a hierarchical view of the system's structure. In learning latent representation, we ensure alignment between embeddings of different years via the procedure described in Chapter 3. The stability error metric devised in the same chapter provides a method to inspect the overall stability of the system in the latent space over time. In short, our methods proposed in the previous chapters enable us to investigate a dynamic migration system with an innovative toolset; making it possible or easier to uncover unique and often richer and more sophisticated insights while empirically examining the existing migration laws. Our second contribution is more country-specific. The work presented in this chapter is among the first studies to investigate the internal migration system of Turkey with a network analysis approach and reveals the system's dynamic and structural patterns in the context of but not limited to the existing migration laws.

Finally, in Chapter 5, we present a succinct summary of the problems we study and the methods we develop and employ in this thesis. We review our findings and provide our conclusions and final remarks.

CHAPTER 2

EXTRACTING THE SIGNED BACKBONE OF INTRINSICALLY DENSE WEIGHTED NETWORKS

2.1 Introduction

Networks are increasingly useful in modeling and studying many problems in seemingly unrelated domains from social sciences (Borgatti, Mehra, Brass, & Labianca, 2009), natural sciences and engineering (Hoche, Nürnberger, & Flach, 2007), and arts and humanities (Schich & Meirelles, 2017). A very simple network consists of nodes (vertices, actors) and links (edges, ties) connecting them. The network science tools utilize the information from the interdependence between entities given by the network structure. The data that can be modeled in the network form is not limited only to the node and edge structure. More complex networks have attributes associated with edges and nodes, especially with the advent of advanced data collection methods.

In weighted networks, edges have numeric weights indicating their intensity (e.g., coupling strength, amount, similarity, etc.). In directed networks, edges have distinguishable source and target nodes, indicating the direction of the relation (e.g., flow, following, etc.). For instance, the air traffic can be modeled as a very simple network such that a node represents an airfield and an edge between two nodes indicates a flight between them. In this case, the number of connections an airfield has is given by the degree of the corresponding node. The same system can also be modeled as a directed weighted network where an edge carries the take-off and landing airfield data as its direction and passenger capacity data as its weight. In this way, the outgoing and incoming capacity of an airfield is given by the out- and in-strength of the node, that is the total weight of edges leaving and entering the node, respectively. This very simplified example demonstrates the view that as networks get more complex, richer analyses can be made.

Another type of networks, which is explored relatively less in the literature, are signed networks where edges have negative or positive signs respectively indicating antagonism (e.g., dislike, distrust, foes, dissimilarity, voting against, inhibition, etc.) or rapport (e.g., like, trust, friendship, similarity, voting for, activation, etc.). Negative edges are not simply negation of positive edges (Jiliang Tang, Hu, & Liu, 2014), they show distinct behavior in networks (Szell, Lambiotte, & Thurner, 2010), and their inclusion enrich the typical tasks on networks such as link prediction (Gupta & Mishra, 2020; Kunegis, Preusse, & Schwagereit, 2013), recommender systems (Jiliang Tang, Aggarwal, & Liu, 2016b; P. Zhang, Song, Xue, & Gu, 2019), node classification (Mercado, Bosch, & Stoll, 2020; Jiliang Tang, Aggarwal, & Liu, 2016a), node centrality and ranking (Gangal, Narwekar, Ravindran, & Narayanam, 2016; Wan et al., 2019), representation learning (Derr, Ma, & Tang, 2018; Kim, Park, Lee, & Kang, 2018), information diffusion and influence maximization (Hosseini-Pozveh, Zamanifar, & Naghsh-Nilchi, 2019; Ju et al., 2020), finding cliques (Li et al., 2021), community detection, graph partitioning and blockmodels (Esmailian & Jalili, 2015; Xueyan Liu et al., 2020; Ma, Zhu, & Yu, 2019), and polarization (Bonchi, Galimberti, Gionis, Ordozgoiti, & Ruffo, 2019; Xiao, Ordozgoiti, & Gionis, 2020). In addition to the typical and prevalent applications in social media (Jiliang Tang, Chang, Aggarwal, & Liu, 2016), signed networks also find application areas in politics (Arinik, Figueiredo, & Labatut, 2020), international relations (Doreian & Mrvar, 2015), finance (Harary, Lim, & Wunsch, 2002), biology (B. Hu, Wang, & Yu, 2019; Iacono, Ramezani, Soranzo, & Altafini, 2010; Ou-Yang, Dai, & Zhang, 2015), and ecology (Saiz et al., 2017).

The density of a network is usually defined by the ratio of the number of observed links to the number of possible links. Formally, sparse networks are those where density asymptotically goes to zero in the limit of large number of nodes. However, in most empirical networks, it is impossible to evaluate this (Newman, 2018). As a result, in general, an empirical network is called sparse if it has a low density. The majority of real-world networks are sparse (Newman, 2003). For

instance, the number of stable social relations of a human is limited by cognitive constraints (Gonçalves, Perra, & Vespignani, 2011), stations are usually only connected to nearest stations in power grid networks, a webpage points out only so many others in web networks, and so on. On the other hand, some networks might be very dense or even almost completely connected by their own nature. Most countries trade with most other countries, humans move/migrate from many locations to many other locations, many species have many predators and preys (Dunne, Williams, & Martinez, 2002), most people interact with most others in certain social settings, and so on.

The earlier view that additional data allowing richer analyses is restricted by the amount and structure of the data itself, particularly in the case where the additional data is provided by more links between the nodes. In addition to the fact that sparsity is desirable due to the computational complexity of many network algorithms, most of the typical network analysis and visualization methods assume the networks to be sufficiently sparse (Newman, 2018). Apart from computational and methodological concerns, dense networks might have noisy, uninformative, insignificant, or otherwise redundant links. This further worsens the application and interpretation capacity in many network tasks, for instance, those relating to node centrality, cliques and communities, and diffusion. Therefore, the relevant and significant information should be extracted from such dense networks such that the original rich data is reduced into a network that is sparse and simpler but maintains adequate structural information for efficient and effective analyses.

There is a body of literature for extracting backbones of usually weighted and relatively dense networks, which we review in the next section. Such information filtering task is usually referred to as backbone extraction or network sparsification and aims to remove statistically insignificant or otherwise redundant links while maintaining the informative structural properties for further analysis. To the best of our knowledge, this is the first study in the literature that provides methods for extracting signed network backbones from weighted, dense, and originally unipartite

networks. Unlike the earlier studies, our method relies on and requires the input network to be intrinsically dense; which in turn enables inferring the link signs.

We use the term *intrinsically dense* for characterizing networks where all nodes, in a sense, are aware of all other nodes and can interact with them without obvious natural limits such as those mentioned earlier for sparse networks. This definition follows that an edge does not necessarily represent the existence of a positive relationship but might be an artifact of the randomness or even a negative relationship depending on its weight. Such a definition of intrinsically dense networks provides a distinction not only from sparse networks but also from certain dense networks. For instance, a human interaction network in a workshop with two parallel sessions with a single mutual short break is not intrinsically dense because the participants in different sessions are not given sufficient time to meet with others, thus, the absence of a link cannot reasonably indicate if two participants from the different sessions are avoiding each other. As a demonstrating example of intrinsically dense networks, consider a network where link weights denote the similarity of respective nodes. Lower similarity does not only mean a lack of similarity but often indicates dissimilarity, a negative underlying link. In the same vein, voting for a set of candidates indicates support for those candidates while also suggesting opposition for the other candidates especially those who are otherwise popular. Finally, we should highlight that whether a network is intrinsically dense or not is a matter of extent and not a strictly binary decision.

We suggest that positive and negative links are those with weights significantly and substantially deviating from the random expectation under a suitable null model. We develop an appropriate null model based on hypergeometric distribution and a relative entropy minimization procedure and offer significance filter and vigor filter for extracting signed backbones of intrinsically dense networks. The proposed methodology is capable of handling directed or undirected networks as input and producing weighted or unweighted signed backbones.

Our contribution can be summarized as follows.

- An extensive literature review on network backbone extraction methods is provided (Section 2.2).
- The first methods to extract signed backbones of intrinsically dense, weighted, originally unipartite networks are proposed (Section 2.3).
- The methodology is empirically evaluated on real-world networks of different characteristics and its feasibility and usefulness is shown (Section 2.4).
- The proposed signed network backbone extraction tool is provided as an open-source software package (Section 2.6).

2.2 Backbone extraction methods

The simplest yet a popular approach is to apply a global threshold where only those edges with weights satisfying the predetermined threshold are retained. There are two major problems with this approach. First, the choice of the threshold is rather arbitrary and non-impartial unless the physical meaning of the weights in the domain of the particular network allows an explanation. The second and more important problem is the case of multiscale networks where the weights are distributed over a broad range of scales. Such a characteristic is definitely not an exception but an observed phenomenon in many real-world networks (Barrat, Barthélemy, Pastor-Satorras, & Vespignani, 2004). For instance, in many networks where rich club effects (S. Zhou & Mondragon, 2004) are prevalent, nodes with higher degrees/strengths tend to form links with nodes of the same or higher degree/strength which results in a hierarchical and multiscale network structure. Rich club effects and such structures are present in many weighted networks including the global airline traffic network (Alstott, Panzarasa, Rubinov, Bullmore, & Vértés, 2014), worldwide maritime transportation network (Y. Hu & Zhu, 2009), world trade network (Bhattacharya, Mukherjee, Saramäki, Kaski, & Manna, 2008; Schiavo, Reyes, & Fagiolo, 2010), international credit-debt networks (Chinazzi, Fagiolo, Reyes, & Schiavo, 2013), population flow network during a national holiday (Wei, Song, Xiu,

& Zhao, 2018), patient referral networks (C. Tang, Dong, Lian, & Tang, 2020), and communication networks in the brains of humans (Alstott et al., 2014) and rats (X. Liang et al., 2017). Therefore, weights with small magnitudes are not necessarily noise and weights with large magnitudes are not necessarily important but might be the result of such multiscale weight characteristics. In multiscale networks, the application of a global threshold tends to eliminate the edges of low-strength nodes regardless of the local significance of those edges. As a result, the structure of high-strength nodes can be preserved but local regions characterized by relatively low-strength nodes are underestimated, might become disconnected, and might even be completely eliminated. In summary, increasing the value of a global threshold loses local structures at the bottom of the hierarchy whereas decreasing it results in retained noise in the upper levels. Consequently, global thresholding would only work under the often unrealistic assumption that edge weights are independent and identically distributed random variables.

As a remedy, the global threshold method can be improved to work on weights represented not in universal units but as fractions of the node degrees/strengths. Yet, an edge is associated with two nodes and it again becomes relatively arbitrary to choose the normalization factor. Such normalization might not be robust to nodes with very low and high degrees as well since it potentially overestimates the edges of low-degree nodes and underestimates the edges of high-degree nodes. Overall, it could serve as a practical remedy for certain problems but the need for more statistically sound methods is clear. Another approach is to extract the spanning tree with maximal weight (i.e., maximum spanning tree) by appropriately transforming the weights and finding the minimum spanning tree. This method does not require any extrinsic input and ensures that the extracted backbone is connected. However, the backbone will be acyclic and local clustering and the community structure cannot be preserved. Using a similar spanning tree approach, Tumminello, Aste, Di Matteo, and Mantegna (2005) propose a technique for controlling the genus of the backbone which, in turn, enables preservation of some local structure.

Grady, Thiemann, and Brockmann (2012) extract the *high-saliency skeleton* of the network which consists of only the salient edges. For each node, they construct the shortest path tree by merging all shortest paths from the node to all other nodes. Then, the saliency of an edge is defined as the fraction of all shortest-path trees where the edge is a member. They show that edge saliency shows a bimodal distribution near the boundaries. Only the edges with saliency near 1 are retained; effectively eliminating the need for choosing an arbitrary threshold. By design, it ensures the connectivity of the extracted backbone. Edge saliency is different from edge betweenness where the former tends to award the edges in the periphery (e.g., low-degree nodes) whereas the latter tends to award the edges in the core (e.g., high-degree nodes).

The methods described so far do not assume any underlying null model and do not compare observed weights to expected weights for statistical evaluation.

Disparity filter (Serrano, Boguñá, & Vespignani, 2009a) assumes that the normalized weights of the links of a node follow a uniform distribution. Comparing the observed weights to this null model at a desired significance level, the network backbone including only the statistically significant links can be obtained. As mentioned earlier, a weight can be normalized and evaluated separately for the two nodes it connects. Thus, a link can be significant from the viewpoint of one node and not the other. This problem is tackled by retaining a link if it satisfies the significance condition for at least one of the two nodes. Adopting a loosely similar weight normalization scheme, *bistochastic filter* (Slater, 2009) first scales the weight matrix such that its marginals (i.e., the row and column totals, in- and out-strength sequence) are equal to 1 resulting in a doubly stochastic matrix. Then, the links with the highest weights are retained until the network is strongly connected; or until any other stopping criteria which can result in sparser backbones with disconnected components or even denser backbones.

In the same line of research, *locally adaptive network sparsification* (Foti, Hughes, & Rockmore, 2011) does not assume an underlying distribution for the weights of a node and employs the empirical cumulative density function instead. It is

stated that it is more robust to the highly heterogeneous local weight distributions than *disparity filter* and *bistochastic filter* are. Rather than adopting a local approach in developing null models, *global statistical significance filter* (Radicchi, Ramasco, & Fortunato, 2011) utilizes a global null model. The model considers both the strengths and degrees and aims to preserve the global weight distribution and the topology. *globally and locally adaptive network backbone* method (X. Zhang, Zhang, Zhao, Wang, & Zhu, 2014) evaluates the statistical significance of link saliency to retain satisfactory links; loosely combining *disparity filter* and *high-saliency skeleton*.

Dianati (2016) introduces two interrelated filters. Treating an integer-weighted network as a multiedge network, they assume that each unit edge randomly chooses two nodes respecting the degree sequences which results in a binomial distribution for weights. The null model is indicated to blend the approaches of *disparity filter* and *globally and locally adaptive network backbone*. *Marginal likelihood filter* operates by evaluating each edge separately against a chosen significance level whereas *global likelihood filter* incorporates exponential random graph model with a Monte Carlo simulation scheme to consider all links at once. *Enhanced configuration model* (Gemmetto, Cardillo, & Garlaschelli, 2017) enhances the null model of Dianati (2016) with the purpose of retaining the relation between strengths and degrees. *Polya filter* (Marcaccioli & Livan, 2019) assumes an underlying null model with a self-reinforcing mechanism in which the generation of link weight increases the probability of further weight generations for that link.

Tumminello, Miccichè, Lillo, Piilo, and Mantegna (2011) introduce *hypergeometric filtering* originally for bipartite weighted networks. It is employed for unipartite weighted networks (Riccaboni, Rossi, & Schiavo, 2013; Sgrignoli, Metulini, Schiavo, & Riccaboni, 2015) as well. With fixed node in- and out-strengths, assuming integer weights and treating weighted links as multiple links, a unit-weight link from a node chooses its other end randomly among all possible nodes. Then, the edge weight generation process can be described by a hypergeometric distribution which becomes the null model for the filter. Comparing observed values with the null

model, statistically significant links can be extracted at desired levels of confidence. *Noise-corrected Bayesian filter* (Coscia & Neffke, 2017) assumes edge weight generation follows a binomial distribution where a null model based on the hypergeometric distribution is used in determining Bayesian priors. Lift value for each link (i.e., the ratio of the observed value to the expected value) is calculated, transformed to the $[-1, 1]$ range, and the associated variance is estimated with a Bayesian inference schema. Using the appropriate posterior variances, the links which satisfy a desired significance level are retained.

The studies described so far, except for Tumminello et al. (2011), are primarily designed for and concerned with unipartite networks. When backbones are extracted from the one-mode projections of originally bipartite networks, a major concern is the loss of relevant and important information. A binary bipartite network where links represent the relations (usually events) between two different types of nodes (e.g., agents and artifacts) can be projected into a one-mode network where link weights between agents denote the number of shared artifacts. However, such projection loses some information such as the degree distribution of the artifacts and which artifacts are common between any given two agents. Neal (2014) proposes stochastic degree sequence model that generates a reasonably large set of random bipartite networks conditioned on both agent and artifact degrees. Then, expected weight distributions from the projections of generated networks are employed for testing the significance and sign of links in the original projection. Liebig and Rao (2016) show that edge weights in such projections follow a Poisson binomial distribution and identifies the significant links without generating random bipartite networks. Domagalski, Neal, and Sagan (2021) provide a software package for extracting the binary or signed backbones of projections of bipartite networks using hypergeometric model, stochastic degree sequence model, or fixed degree sequence model. Overall, these methods are better suited when the backbone is to be extracted from one-mode projections of bipartite binary networks and the information on the original bipartite

network is available. Comprehensive reviews on backbones of bipartite projections can be found in the studies of Domagalski et al. (2021) and Neal (2014).

There are more specific approaches with different purposes, e.g., to retain the most consistent links in brain networks based on diffusion images (Roberts, Perry, Roberts, Mitchell, & Breakspear, 2017), to preserve functional backbones based on network motifs (J. Cao, Ding, & Shi, 2019), and to extract underlying networks from correlation matrices (Kojaku & Masuda, 2019). Generally, the state-of-the-art concentrates on developing appropriate null models to evaluate link weights against and retain only those satisfying a desired level of statistical significance.

2.3 Methods

2.3.1 Formal problem definition

We denote an intrinsically dense, undirected or directed, and non-negative weighted network without self-loops by $G := (V, W)$. V is the set of nodes with $i, j \in V$ as its general elements and its cardinality is $n = |V|$. W is the corresponding weight matrix. When G is undirected W_{ij} denotes the weight of the link between i and j and $W_{ij} = W_{ji}$. When G is directed, W_{ij} denotes the weight of the link from i to j . Self-loops are not allowed hence $W_{ii} = 0 \forall i$. Total outflow of i is denoted with $W_{i.} = \sum_j W_{ij}$. Total inflow to j is denoted with $W_{.j} = \sum_i W_{ij}$. The sum of all weights is denoted with $W_{..} = \sum_i \sum_j W_{ij}$. We employ the same definitions of $W_{i.}$, $W_{.j}$, and $W_{..}$ for directed and undirected networks, i.e., treating an undirected link as two reciprocal directed links with equal weights.

Given G , the aim is to extract a meaningful signed network $\hat{G} := (\hat{V}, \hat{A}, \hat{W})$ where \hat{A} is a sparse adjacency matrix in general such that $\hat{A}_{ij} \in \{-1, 0, 1\}$ where -1 denotes a negative link, 1 denotes a positive link, and 0 indicates the absence of a link. \hat{W} is the corresponding optional weight matrix. If $\hat{A}_{ij} = 0$, then $\hat{W}_{ij} = 0$; otherwise \hat{W}_{ij} has the same sign as \hat{A}_{ij} .

2.3.2 Proposed solution

To extract the signed edges, it is necessary to have a null model where empirical edge weights are compared against to distinguish edge weights that are due to chance and edge weights that significantly deviate from the expected values in either positive or negative direction. Our null model ensures that node strengths are fixed and edge weight distributions are characterized by a sampling without replacement procedure.

We assume that in- and out- strengths are fixed, thus, each node has a specified number of stubs (i.e., half edges). A unit-link is a connection of two stubs from different nodes. When a stub makes a connection, it chooses the other stub randomly from all available stubs. Hence, for a stub, the probability of connecting to a specific node is proportional to the number of its available stubs. The weight of the link between two nodes is equal to the number of unit-links between them. In this way, the process is reduced to sampling without replacement problem, of which the urn problem is a famous example. We describe the process with a simple analogy. For each node i , there are $W_{..} - W_{.i}$ marbles in the urn (not allowing self-loops), $W_{.i}$ marbles are chosen from the urn without replacement, and there are $W_{.j}$ marbles in the urn for each j . Such a process is well-characterized by hypergeometric distribution and allows us to calculate statistical quantities for links between all i and j .

The mean of the hypergeometric distribution associated with the link from i to j , that is P_{ij} , is given by Equation 2.1. However, this formulation does not preserve the in-strength and out-strength sequences in the system, thus, is inadequate for our purposes. Given that N is the weight matrix under the null model with N_{ij} denoting the expected weight from i to j , the following should be satisfied: $N_{i.} = W_{.i}$, $N_{.j} = W_{.j}$, and $N_{..} = W_{..}$. For this purpose, we utilize the iterative proportional fitting procedure (IPFP) described by Bacharach (1965) and Mazzarisi and Lillo (2017). Given out- and in-strength ($W_{.i}$ and $W_{.j}$) sequences as the marginals and P as the prior matrix with its diagonal elements set to 0, the objective of IPFP is to estimate N that minimizes the relative entropy (known as Kullback–Leibler divergence (Kullback & Leibler, 1951)) as formalized in Equation 2.2. IPFP estimates

the values for non-zero elements of the matrix through a series of row-scaling and column-scaling operations until achieving a desired precision and is shown to converge to an optimal solution (Bregman, 1967). Then, each element of N serves as the expected value of the respective link under our null model.

$$P_{ij} = \frac{W_i \cdot W_j}{W_{..} - W_i} \quad (2.1)$$

$$\min_{N_{ij}} \sum_{i,j \neq i} N_{ij} \log \frac{N_{ij}}{P_{ij}} \quad (2.2)$$

subject to $N_{i.} = W_i, N_{.j} = W_j, N_{ij} \geq 0, N_{ii} = 0; \forall i, j$

In evaluating the statistical significance of an edge weight, only considering the difference of observed and expected values without referencing a dispersion measure is not viable. For each edge, dispersion of its expected weight must be known to produce confidence intervals. Accordingly, we estimate the variance of N_{ij} based on the associated hypergeometric distribution and obtain its well-known standard deviation as given by Equation 2.3.

$$\sigma_{ij} = \sqrt{\left(W_i \cdot \frac{W_j}{W_{..} - W_i} \cdot \frac{(W_{..} - W_i - W_j)}{W_{..} - W_i} \cdot \frac{W_{..} - W_i - W_i}{W_{..} - W_i - 1} \right)} \quad (2.3)$$

Overall, our null model can be viewed as extending the hypergeometric approach employed in the literature to intrinsically dense unipartite networks and to ensure that node strengths are exactly preserved with the inclusion of entropy-based method IPFP. Moreover, distinctly from the existing hypergeometric filtering literature, we do not necessarily assume weights to be integer counts of events hence we do not employ the traditional hypothesis testing procedure associated with the hypergeometric distribution.

The first filter we propose functions to eliminate links whose weights do not significantly deviate from their expected values. Significance filter eliminates those links satisfying the condition $\sigma_{ij} \alpha^- < W_{ij} - N_{ij} < \sigma_{ij} \alpha^+$ where α^- and α^+

respectively take non-positive and non-negative values and are user-defined hyperparameters specifying the desired significance thresholds for negative and positive signed edges.

It should be highlighted that the hypergeometric distribution is a distribution for discrete events. In general, we can slightly abuse it by allowing continuous weights or rounding weights to the nearest integers without much impact on the resulting backbone. However, a larger issue that is not adequately discoursed in the hypergeometric filtering literature is the impact of the magnitude of weights on the statistical significance evaluation. As the magnitude of weights increases, the confidence intervals relative to the weights become narrower, as a consequence of the assumption that the weights are the counts of discrete events. A demonstrating example is provided next.

Testing the statistical significance of edge weights with hypergeometric distribution in a traditional way is not appropriate when edge weights are not counts of discrete events but continuous values with possibly arbitrary units. Here, we employ the trade network between OECD countries in 2019 (Organisation for Economic Co-operation and Development, 2021) where edge weights represent the amount of exports in monetary amounts. We show that when edge weights are represented in units of 1 million USD and 10 million USD, the confidence intervals obtained from hypergeometric distribution change. Table 1 presents actual edge weights and confidence intervals with 99.999% confidence for the two cases. Randomly selected five edges are included and all values are shown in 1000 USDs. Figure 1 visualizes the networks extracted based on the same confidence value. The density of extracted backbones are 0.65 and 0.28 respectively in Figure 1a and 1b. As shown by these results, when edge weight units do not correspond to discrete events, traditional hypothesis testing with such discrete distributions does not generalize well to the continuous edge weights.

As demonstrated, for instance, the same monetary network represented in dollars or cents would result in different network backbones when they are evaluated

desired level of sparsity. One may choose to retain top $x\%$ of edges based on their absolute significance levels (i.e., $|W_{ij} - N_{ij}|/\sigma_{ij}$) or bottom $y\%$ and top $z\%$ of edges based on (non-absolute) significance levels. Ultimately, any rank-based filtering is equivalent to some (α^-, α^+) configuration but may be more intuitive.

The significance filter might retain statistically significant but otherwise very weak links. Yet, an edge can be evaluated also based on whether it is sufficiently strong in terms of intensity. This is useful for multiple reasons: (i) the employed significance filter might be too permissive in certain circumstances, (ii) the signed network backbone is needed to reflect only binary links where opinions are rather strong, or (iii) higher sparsity is desired. We define vigor of an edge from i to j as β_{ij} (pronounced as *víta* in Modern Greek) in Equation 2.4. Vigor values may be considered as edge weights normalized based on their expected values under our null model. It takes values in the range $[-1, 1]$, its magnitude indicates the intensity of the link and $\beta_{ij} = 0$ when $W_{ij} = N_{ij}$. Vigor filter eliminates those links satisfying the condition $\beta^- < \beta_{ij} < \beta^+$ where β^+ and β^- respectively take non-positive and non-negative values and are user-defined hyperparameters specifying the desired vigor thresholds for positive and negative signed edges.

$$\beta_{ij} = \frac{W_{ij}/N_{ij} - 1}{W_{ij}/N_{ij} + 1} \quad (2.4)$$

The signed backbone of a network can be extracted using a combination of significance filter and vigor filter. The former serves as the primary filter and should always be employed since β values do not reflect the statistical significance and might be unreliable in the small-magnitude weight regime characterized by small N_{ij} , relatively large σ_{ij} , and small W_{ij} . Significance filter would function to eliminate such links, therefore, the retained links would have reliable vigor. Vigor values of the remaining edges can be utilized as the signed edge weights in the extracted backbone. In the case of weighted backbone, the use of vigor filter is rather optional since the vigor information is directly carried onto the network.

Alternatively, the weights can be ignored to produce a network with binary opinions. In this case, one may consider utilizing also the vigor filter to retain only the edges that exhibit appropriate levels of intensity. In the high-magnitude regime characterized by large N_{ij} , relatively small σ_{ij} , and large W_{ij} ; statistical significance may be easier to obtain. Accordingly, we may confidently identify a negative or positive relationship without necessarily uncovering whether the relation is intense. This is intuitive since, for instance, a slightly positive or negative relationship between two actors can be identified more confidently when we have more observations on them while higher confidence does not necessarily mean a higher intensity of rapport or antagonism. Since β_{ij} values have interpretable physical meanings as they are comparable weights that were normalized based on null expectations (e.g., $\beta = 0.33$ indicates that the observed value is double the expected value), users may choose appropriate values of β as they see fit for their problem.

Overall, one should always employ significance filter to retain those links where the sentiment of a link can be confidently identified. Additionally, one may utilize vigor filter to retain those links where the sentiment is rather strong. In setting appropriate values for the hyperparameters, one should consider (i) the physical meaning of hyperparameters in the context of the specific problem and (ii) the desired level of sparsity for the extracted backbone. In addition, one may look at the distribution of α_{ij} and β_{ij} values for all edges before deciding on appropriate threshold values.

The described method is defined well on the directed networks. The generalization to the undirected networks is ensured in the following way. An undirected link is replaced with two reciprocal directed links of the same weight; effectively transforming the network into a directed network. The null model and the filters produce the same null expectation and nearly the same variance¹ for such reciprocal links. Finally, the directed backbone can be transformed back into an

¹For sufficiently large networks, $W_{..} - W_{.j} \approx W_{..} - W_{.i}$ in Equation 2.3.

undirected backbone by treating reciprocal directed links as undirected links and removing the redundancy by keeping the link with the highest absolute vigor.

2.4 Empirical analysis

In this section, relevance and diverse characteristics of the employed network datasets are described, and proposed filters are empirically analyzed in terms of resulting backbone sizes, robustness to multiscale networks, and structures of the extracted signed network backbones.²

2.4.1 Datasets

The proposed method for signed network backbone extraction is experimented on four real-world networks from different domains and of varying statistical properties. For each network; the number of nodes n , number of nonzero links m , density, five-point summary of weights (minimum, first quartile, median, third quartile, and maximum values), and three-point summary (minimum, median, and maximum values) of node strengths (out- and in-strengths if networks are directed) are presented in Table 2.

Table 2. Employed Networks

| | n | m | density | 5-point summary of weights | 3-point summary of (out-, in-) strengths |
|------------|-----|------|---------|----------------------------|---|
| Migration | 51 | 2338 | 0.92 | (3, 356, 1079, 3236, 86k) | out:(21k, 101k, 691k), in:(26k, 109k, 587k) |
| Eurovision | 26 | 260 | 0.4 | (1, 3, 6, 8, 12) | out:(58, 58, 58), in:(0, 38, 167) |
| Contact | 113 | 2196 | 0.17 | (1, 1, 2, 5, 1281) | (2, 287, 1483) |
| Species | 62 | 1801 | 0.48 | (1, 26, 51, 76, 194) | (954, 3249, 5448) |

Migration network (United States Census Bureau, 2019) reports interstate migration flows (in terms of people) among the US states³ in 2018. Such internal migration data is explored for its correlation with economic productivity differences, geographical proximity, political preferences, and other cultural and historical factors (Chakrabarti & Sengupta, 2017; Charyyev & Gunes, 2019; Xi Liu, Andris, & Desmarais, 2019). *Eurovision* network (European Broadcasting Union, 2003)

²The datasets and code for producing the analyses in this section are available at https://github.com/furkangursoy/signed_backbones.

³The dataset contains District of Columbia in addition to the 50 states.

represents the votes between participant countries of the song contest in 2003.⁴ Each country, via public votes⁵, awards the set of points $\{1,2,3,4,5,6,7,8,10,12\}$ to 10 other countries. Studies employing a network analysis perspective (Charron, 2013; D’Angelo, Murphy, & Alfò, 2019; Fenn, Suleman, Efstathiou, & Johnson, 2006; Garcia & Tanase, 2013; Mantzaris, Rein, & Hopkins, 2018a, 2018b; Svete & Hostnik, 2020) show that voting behavior is not determined only by the music/performance quality but affected by political factors, geographical and cultural similarity, diasporas, and others. *Contact* network data (Isella et al., 2011) is gathered via wearable sensors at an academic conference⁶ with 113 attendees over 2.5 days. The edge weights are generated by aggregating the number of 20-second intervals the respective two participants spent face-to-face over the course of the conference. The networks of such temporal ties are utilized for temporal backbone extraction, investigation of spreading processes, and other behavioral and structural analysis (Barrat et al., 2013; Kobayashi, Takaguchi, & Barrat, 2019; Kulisiewicz, Kazienko, Szymanski, & Michalski, 2018; Zhan, Hanjalic, & Wang, 2019). *Species* network (Kritzer et al., 2016) is generated based on the cohabitation patterns of 62 marine species in South Florida. Edge weights represent the similarity of species based on the habitats they co-occupy during the same life stages.⁷ Information derived from such cohabitation networks is useful in various ecological studies (Camp, Hobbs, De Brauwer, Dumbrell, & Smith, 2016; Fernandes et al., 2020; Kikkawa, 1968). *Species* network is generated from a bipartite network, thus, backbone extraction methods in the bipartite network projection literature are likely to be more suitable. However, it is also slightly different from the datasets in that literature since the original bipartite matrix here represents heterogeneous suitability scores instead of

⁴2003 was selected because it was the last year that all countries competed in a single round.

⁵Ireland, Russia, and Bosnia and Herzegovina exercised jury voting instead of public voting.

⁶The temporal face-to-face interaction dataset was collected at Hypertext conference in 2009 as part of *SocioPatterns* research collaboration.

⁷Given the species v. habitat-life stage bipartite matrix B , $W_{ij} := \sum_k B_{ik}B_{jk}$ where i, j are the species and k is the habitat-life stage pair, and B_{ik} is the respective species-habitat-life stage score. More information on the original dataset can be found at <https://atlanticfishhabitat.org/species-habitat-matrix/>.

binary event-like relations. Moreover, none of the employed networks have natural self-loops; which makes them particularly suitable for our study.

Our proposed method considers the propensity of nodes for incoming and outgoing edge weights by explicitly including the in-strengths and out-strengths in the null model. For instance, high economic productivity for a state would result in more immigration into it. Similarly, low productivity would cause emigration from that state. The effect of such productivity is homogeneous in the sense that all other nodes perceive it in the same way. Such homogeneous effects are eliminated for the large part by our null model. However, for instance, the effect of political affiliation or geographic location is perceived differently by other nodes. Hence, the extracted backbone is expected to reflect non-economic factors in *Migration* network. With a simple analogy, the quality of the songs and performances in the Eurovision song contest is equivalent to productivity. Hence, the elimination of such an effect by the null model would result in an extracted signed network representing the non-quality factors in voting behavior. Similarly, the null model largely eliminates the effects of participants' tendency to engage in conversations (e.g., popularity, extroversion) and the extent of species' ability to live in many different habitats.

2.4.2 Backbone size

In general, an extracted backbone should be sufficiently sparse such that only the important edges are preserved. At the same time, unlike edges, most nodes should be retained since they are often the subject of the analysis and are largely indispensable in understanding the global network structure.

A few arguments for the non-isolation of nodes in general cases are as follows. First, the very definition of networks includes relationships, thus, we expect nodes to have some underlying relations with other nodes particularly in intrinsically dense networks. Second, most real-world networks have a high local clustering coefficient, which according to Granovetter (1973) and van der Leij and Goyal (2011) are closely related to strong ties. Hence, it is generally safe to say that most nodes

have at least some strong connections which should be retained by the backbone. Third, since we extract signed backbones, negative links also emerge which provide extra connectedness. Moreover, in the literature, Grady et al. (2012), Slater (2009), and Tumminello et al. (2005) and the maximum spanning tree method ensure that backbones are fully connected while Coscia and Neffke (2017), Foti et al. (2011), and Gemmetto et al. (2017) present it as a generally desirable result.

Here, we extract the signed backbones of the four networks under various hyperparameter regimes and observe the size of the resulting backbone in terms of nodes and signed edges.

Figure 2 visualizes the fraction of links and nodes retained for meaningful continuous ranges of the significance threshold under different vigor filters. The suitable range of α differs between networks mostly due to the varying overall magnitude of edge weights. The significance threshold is represented with $\alpha = \alpha^+ = -\alpha^-$ on x -axes. Three different vigor threshold settings are represented in columns. For edges, percentage values on y -axes are with respect to all possible edges, that is $n^2 - n$ edges for directed networks and half of it for undirected networks.

Except for *Species* network, all nodes are retained in the backbones even under very strict settings where 80% to 85% of all possible edges are eliminated. When edges are inspected, apart from the *Species* network where edges represent similarity, the negative edges are more frequent than the positive edges when $\alpha = 0, (\beta^-, \beta^+) = (0, 0)$. This is due to the case that edges with zero weights (i.e., nonexistent edges) or trivially small values are treated as negative links regardless of their statistical significance. As the values of α increase, the number of negative links in the extracted backbone decreases faster than the positive links since many of the zero or trivially small weights are rather statistically insignificant. On the other hand, utilizing vigor filter without the significance filter does not eliminate such zero-weighted links since those edges have vigor values of -1 (since 0 divided by any null expectation is 0, which maps to vigor of -1). Hence, when $\alpha = 0$, the smaller

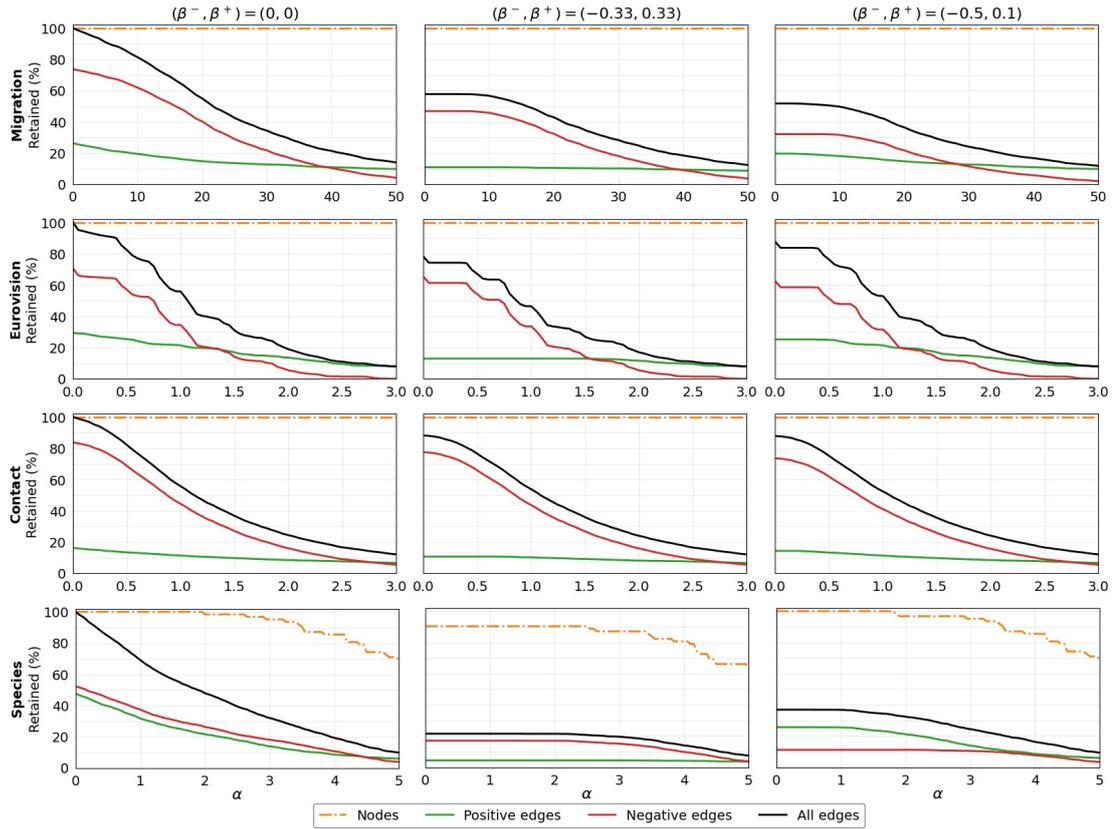


Figure 2. Effects of significance filter on the backbone size

number of negative edges in the second and third columns in comparison to the first column of Figure 2 is not due to the elimination of zero-weighted edges but the elimination of other weak edges. Therefore, significance filter is always necessary when insignificant zero or trivially small edge weights exist.

Employing very large α values eliminates negative links almost completely. This is mainly because the edges with relatively small expected values under the null model have relatively large variances. Increasing α too much, thus, tends to eliminate negative edges first as well as other edges in local regions characterized by edge weights of small magnitude. In the case of high-magnitude regions, an edge between two high-strength nodes is expected to have a large expectation under the null model. It is much harder for such edges to empirically observe much larger weights than the null expectation since the total weight in the network is fixed. This implies an upper boundary on vigor values for edges connecting to high-strength nodes. Therefore,

increasing β too much, especially in its positive range, tends to eliminate edges between central, high-strength nodes.

In a similar fashion, we have also explored the whole continuous range of vigor threshold under different significance threshold filters in Figure 3. The vigor threshold is represented with $\beta = \beta^+ = -\beta^-$ on x -axes. Three different significance threshold settings are represented in columns. Except for *Species* network and except under strict vigor thresholds, all or almost all nodes remain in the backbone. When edges are inspected, in line with the relevant conclusions derived from Figure 2, increasing β does not sufficiently eliminate the negative edges that are otherwise statistically insignificant. In very large values of β , positive links are largely eliminated. When $\beta = 1$, the edges remaining in the backbone are the zero-weighted (i.e., nonexistent) edges. Accordingly, employing very large vigor thresholds should be avoided.

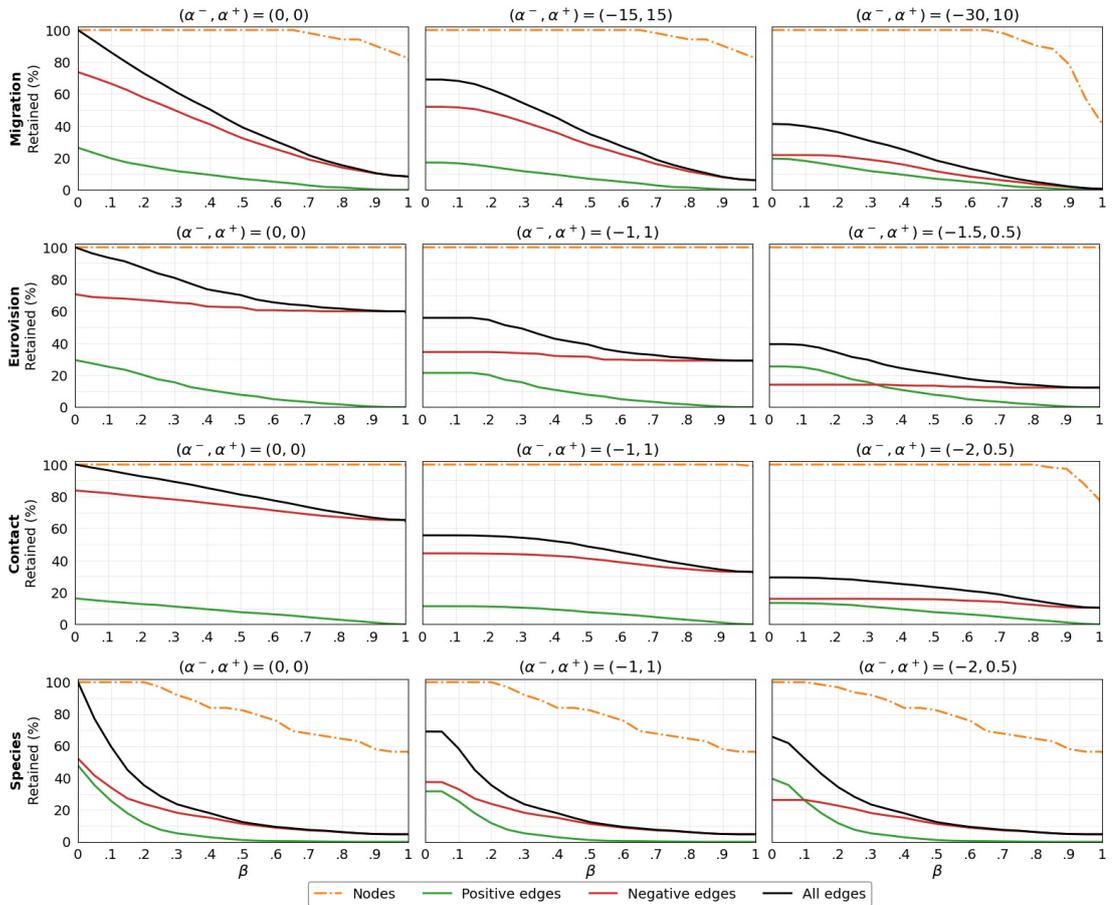


Figure 3. Effects of vigor filter on the backbone size

Overall, we have shown that the proposed method is able to reduce an intrinsically dense network to a signed backbone of the relative size of 10% to 20% such that the resulting backbone contains comparable portions of negative and positive edges. Yet, the hyperparameter selection largely lies with the user and can be changed according to the purpose of analysis and the nature of the network. For instance, *Species* network is constructed with similarity values as edge weights and the edge weights do not follow a skewed distribution, unlike other networks. Therefore, its behavior under the proposed filters partially deviates from the behavior of the other employed networks.

Principally, we can conclude that (i) significance filter should almost always be used with a fairly medium threshold where fairness can be evaluated with respect to the backbone size or by inspecting the range and scale of α_{ij} values in the original network, (ii) very large threshold values for both filters but especially for vigor filter should be avoided, and (iii) a sufficiently balanced, statistically meaningful, and sparse backbone can be extracted with a balanced utilization of the two proposed filters.

2.4.3 Heterogeneity

As discussed in Section 2.2, an enviable backbone extraction method should respect the weight and strength heterogeneity in the original network (i.e., its multiscale, hierarchical nature). That is to say, the retained edges should not be only those originally with large weights or those connecting high-strength nodes.

Figure 4 presents the original weight distribution⁸ of the retained edges in backbones⁹ of different sizes. The figure shows that even when the extracted backbone retains only 5% of all possible edges, the heterogeneity of edge weights are

⁸Small random noise is added on *y-axis* for all networks and *x-axis Eurovision* network for visualization purposes.

⁹The backbones are extracted with the following (α^-, α^+) , (β^-, β^+) settings for the respective backbone sizes:

Migration → 25%: (-33, 33), (-0.33, 0.33); 10%: (-40, 40), (-0.57, 0.57); 5%: (-40, 40), (-0.72, 0.72).

Eurovision → 25%: (-1.5, 2.25), (0, 0); 10%: (-2, 4.5), (0, 0); 5%: (-2.5, 5), (0, 0).

Contact → 25%: (-2, 2), (0, 0); 10%: (-3, 3), (-0.66, 0.66); 5%: (-5, 5), (-0.55, 0.55).

Species → 25%: (-3.5, 3.5), (0.15, 0.15); 10%: (-5, 5), (-0.2, 0.2); 5%: (-5.5, 5.5), (-0.33, 0.33).

respected, i.e., edges with different weights are retained and some other edges with similar weights are eliminated. It also visually depicts that there is no perfect global cutoff value for determining the sign of edges as those cutoff values are established individually for each edge by the null model.

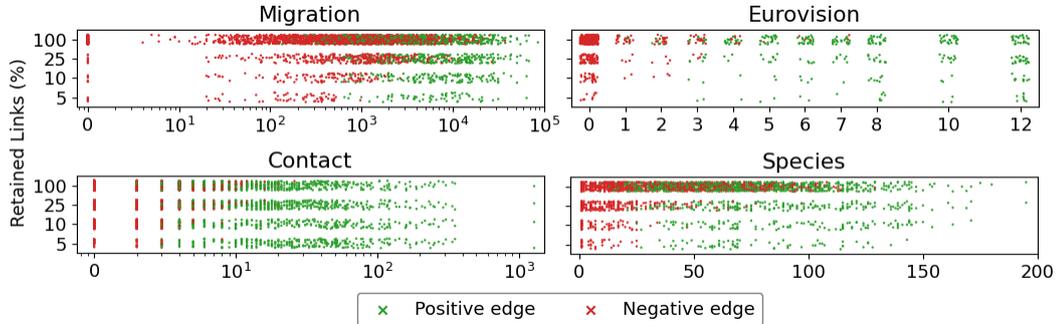


Figure 4. Original weights of the retained edges

Figure 5 shows the strength distribution¹⁰ of the dyads for the original networks in transparent color and the extracted backbones¹¹ with relative size of $\approx 20\%$ in opaque color. Specifically, the edge between nodes i and j is represented with a point colored based on its sign. X -axis denotes the (out-)strength of i and y -axis denotes the (in-)strength of j .¹² As demonstrated by the figure, the retained edges are rather evenly distributed in the plotting space and the original heterogeneity is respected. The retained edges between very low-strength nodes are usually positive since null expectations are usually low in that regime and it is difficult to conclude whether small-weighted edges in low-strength regimes are statistically significant. Likewise, the edges between very high-strength nodes are usually negative due to the upper bound on vigor implied by the large null expectation and the fixed amount of total weights. On the other hand, in the majority of strength-strength regions, the edge signs exhibit a mixed distribution. Therefore, this is generally a desired property since the proposed method, by itself, does not allow deducing strong conclusions when the data is limited.

¹⁰Small random noise is added for *Eurovision* network for visualization purposes.

¹¹The backbones are extracted with the following (α^-, α^+) , (β^-, β^+) settings. *Migration* $\rightarrow (-40, 40)$, $(-0.25, 0.25)$. *Eurovision* $\rightarrow (-1.8, 2.4)$, $(-0.5, 0.3)$. *Contact* $\rightarrow (-2.25, 2.25)$, $(-0.25, 0.25)$. *Species* $\rightarrow (-3.6, 3.6)$, $(-0.25, 0.25)$.

¹²For an undirected network, the order of i and j for an edge is established alphabetically.

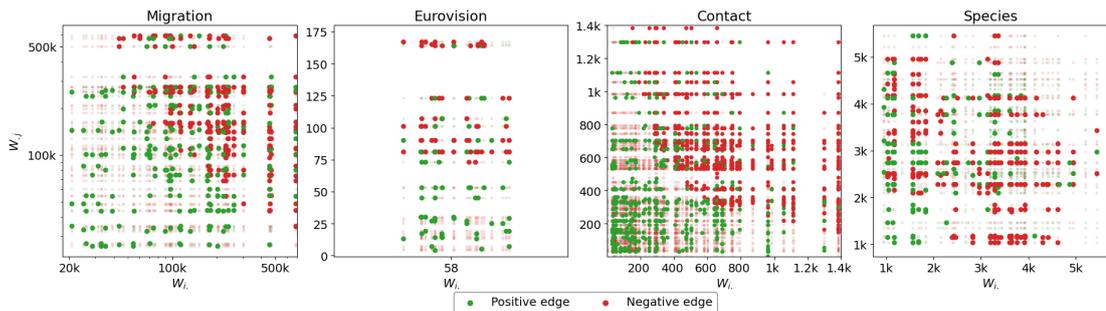


Figure 5. Original strength distribution of the retained dyads

We can conclude that the heterogeneity given by (i) the multiscale nature of edge weights and (ii) the heterogeneous nature of node strengths are respected, and (iii) the proposed filters do not draw strong conclusions when the data is limited.

2.4.4 Structure of the backbones

Generally speaking, we expect real-world networks to exhibit some extent of reciprocity (Doreian & Mrvar, 2009; Jiang, Zhang, & Towsley, 2015), have their reciprocal edges be of the same sign in the case of signed networks (Leskovec, Huttenlocher, & Kleinberg, 2010), and demonstrate sufficient levels of structural balance. Moreover, most real-world networks exhibit a community structure regardless of whether they are signed or not. Therefore, an informative signed backbone should have these characteristics in general and up to a certain extent. We analyze the extracted backbones¹³ of the employed networks in terms of reciprocity, structural balance, and community structure.

Reciprocity in unsigned networks is generally defined as the ratio of the number of reciprocal edges to the number of all edges. In signed networks, however, the edge signs should also be considered in the analyses. Table 3 presents the number of nodes, edges, nonreciprocal edges ($\{\times, \cdot\}$), reciprocated positive edges ($\{+, +\}$), reciprocated negative edges ($\{-, -\}$), and reciprocated edges with sign conflict ($\{+, -\}$) for the directed backbones extracted from directed networks. In *Migration* backbone, there is considerable reciprocity with no conflicting edge signs between any node

¹³The backbones are extracted with the following (α^-, α^+) , (β^-, β^+) settings. *Migration* $\rightarrow (-40, 40)$, $(-0.33, 0.33)$. *Eurovision* $\rightarrow (-1, 0.5)$, $(-0.33, 0.1)$. *Contact* $\rightarrow (-3, 3)$, $(-0.33, 0.33)$. *Species* $\rightarrow (-5, 3)$, $(-0.5, 0.1)$.

pair. In contrast, *Eurovision* backbone discloses a substantial amount of conflicting edge pairs which might be of interest for further investigation into its voting dynamics (e.g., the influence of diasporas).

Table 3. Reciprocity in Extracted Directed Backbones

| | nodes | edges | { \times , \cdot } | {+, +} | {-, -} | {+, -} |
|------------|-------|-------|------------------------|--------|--------|--------|
| Migration | 51 | 471 | 179 | 188 | 104 | 0 |
| Eurovision | 26 | 380 | 158 | 58 | 80 | 84 |

For the rest of the analysis, the directed backbones are transformed into undirected backbones in the following way. For *Migration* backbone, directed edges are transformed into undirected edges of the same sign with positive signs having a priority over negative signs when in conflict. For *Eurovision* backbone, a positive (or negative) edge is created between two nodes when there are exactly two positive (or negative) directed edges between them (i.e., reciprocal edges of the same sign).

As put forward by Heider (1946) and formalized for signed networks by Cartwright and Harary (1956), an undirected triple is said to be balanced if its edges have the signs {+, +, +} or {+, -, -} and unbalanced if its edges have the signs {+, +, -} or {-, -, -}. The balanced triples can be simply described with the following expressions: "friend of my friend is my friend" and "enemy of my friend is my enemy". Davis (1967) defines a weaker notion and proposes that "enemy of my friend is my enemy" is not necessarily required for balance and the only unbalanced triple among the four possible settings is the one with edge signs {+, +, -}. Accordingly, structural balance (SB) and weak structural balance (WSB) of a network can be defined as the ratio of the number of balanced triples to the number of all triples. There is also recent evidence for the existence of (weak) structural balance in social networks and voting networks (Levorato, Figueiredo, Frota, & Drummond, 2017) and spatial ecological networks (Saiz et al., 2017).

Table 4 shows the count of nodes, edges, and four possible triple settings; and SB and WSB measures for the undirected backbones. The backbones for *Migration* and *Species* shows a strong balance for both measures. The other two backbones are

also highly balanced in terms of WSB which is shown to be more appropriate in real-world networks.

Table 4. Structural Characteristics of Extracted Backbones

| | nodes | edges | {+, +, +} | {+, +, -} | {+, -, -} | {-, -, -} | SB | WSB |
|------------|-------|-------|-----------|-----------|-----------|-----------|------|------|
| Migration | 51 | 325 | 117 | 12 | 404 | 98 | 0.83 | 0.98 |
| Eurovision | 24 | 69 | 4 | 6 | 36 | 18 | 0.63 | 0.91 |
| Contact | 113 | 760 | 194 | 98 | 809 | 880 | 0.51 | 0.95 |
| Species | 56 | 328 | 655 | 1 | 295 | 0 | 1.00 | 1.00 |

Based on the partitioning methods for signed networks (Doreian & Mrvar, 2009), the extracted backbones are partitioned into communities where the number of negative edges within communities and number of positive edges between communities is minimized via heuristics.¹⁴ Figure 6 depicts the community structure via a visual block matrix where the order of nodes is the same for its rows and columns. Diagonal blocks are expected to cover the positive edges and non-diagonal blocks are expected to cover the negative edges. Overall, the extracted backbones manifest community structures. The backbone of *Migration* has five clear densely connected communities and two outlier individual nodes whereas the communities in *Eurovision* are not as dense. The backbone of *Contact* consists of one large sparse group and several smaller and denser groups. The backbone of *Species* has a very clear structure with two similar-sized communities, visually confirming the very high SB and WSB measures obtained for it.

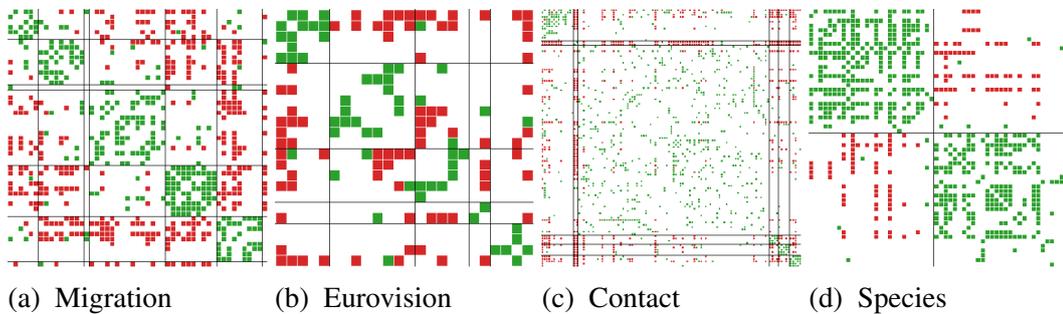


Figure 6. Block models of extracted backbones

¹⁴Communities are found and visualized using *signnet* (Schoch, 2020) package in R.

Table 5 shows members of each group in the order of diagonal blocks where nodes are also listed in the same order as they appear in the rows and columns. In *Migration* backbone, we observe that the states are grouped mostly based on their geographical proximity, which means that there are more positive links between geographically close states and more negative links between geographically distant states. This is in line with the same geographical constraints observed by several other backbone studies (Dianati, 2016; Marcaccioli & Livan, 2019; Serrano et al., 2009a). Florida, one of the two singletons here, is also shown to be the last state to connect to the rest of the network by Maier and Vyborny (2005). In *Eurovision* backbone, for the most part, countries are grouped mainly based on geographic, cultural, linguistic, ethnic, and historical ties. In *Species* backbone, the first group is composed of inshore/nearshore/estuarine species while the second one includes more tropical/Caribbean/reef fish species, agreeing with the expert knowledge in marine biology. The plausibility of the found communities suggests that the proposed backbone extraction method retains the appropriate edges that preserve and reveal the important structural information available in the original network.

In conclusion, we showed that the extracted backbones usually (i) reveal interesting or expected reciprocal structures, (ii) are structurally balanced, and (iii) exhibit clear and intuitive community structure in agreement with the existing knowledge.

2.5 Conclusion

In this chapter, we put forward an initial discussion on intrinsically dense networks and provide significance filter and vigor filter for extracting signed backbones of such networks. Empirical evaluations that utilize the proposed filters on a variety of real-world networks show that sparse backbones can be obtained while maintaining comparable numbers of positive and negative links and respecting the original weight and strength heterogeneity. In general, the obtained backbones exhibits characteristics associated with signed networks such as reciprocity, structural balance, and

Table 5. Block Memberships

| Migration | Eurovision | Species |
|--|--|--|
| Maryland, D. of Columbia, S. Carolina, W. Virginia, Virginia, N. Carolina | Bosnia and Herzegovina, Croatia, Turkey, Slovenia, Austria | Florida Gar, Gray Snapper, Gulf Flounder, Ladyfish, Mangrove Species Rivulus, Oyster Toadfish, Sheepshead, Red Drum, ~Florida Pompano, American Eel, ~Brown Shrimp, Bucktooth Parrotfish, Pink Shrimp, Spanish Sardine, Sheepshead Minnow, Bonefish, Permit, Striped Mullet, Snook, Black Tip Shark, Spotted Sea Trout, Tarpon, Yellowfin Mojarra, Pinfish, Crevalle Jack, Goliath Grouper, Lemon Shark |
| Georgia, Kentucky, Louisiana, Mississippi, Tennessee, Ohio, Alabama, Indiana, Arkansas | Belgium, France, Germany, Portugal, Romania, Poland, | |
| Delaware | Spain, Sweden | |
| New Mexico, Wisconsin, Missouri, Oklahoma, Texas, Nebraska, S. Dakota, Wyoming, Kansas, Colorado, Minnesota, Illinois, Iowa, Michigan, N. Dakota | Ireland, Iceland, Netherlands, Norway, Malta | Bluehead Wrasse, Carribean Spiny Lobster, French Angelfish, French Grunt, Hogfish, ~Jolthead Porgy, Long-Spined Sea Urchin, Queen Conch, Queen Triggerfish, King Mackerel, Atlantic Sailfish, |
| California, Oregon, Hawaii, Nevada, Washington, Arizona, Montana, Idaho, Utah, Alaska | Cyprus, Greece | Black Grouper, Dusky Squirrelfish, Longsnout Butterflyfish, Sergeant Major, Spanish Spotted Lobster, Barred Hamlet, Carribean Reef Squid, ~Green Moray, Reef Croker, Florida Stone Crab, Black Margate, Cero Mackerel, |
| Massachusetts, New York, Connecticut, New Jersey, Pennsylvania, New Hampshire, Maine, Rhode Island, Vermont | Estonia, Israel, Russia, Ukraine | Tripletail Peacock Flounder, Stoplight Parrotfish, Bar Jack, Cobia, Yellowtail Snapper |
| Florida | | |

community structure. The extraction method, by design, does not prematurely arrive at conclusions regarding the existence of signed links between nodes when the data is limited. On the other hand, choosing appropriate hyperparameter values lies with the user and should be guided by empirical analysis and recommendations provided in this chapter as well as the purpose and the nature of the specific problem.

Further studies can improve this work in two major ways. As we presented many examples throughout the chapter, intrinsically dense networks exist in many different domains. First, a stream of its applications in different fields and utilization of the resulting backbones in different tasks would provide important feedback regarding its usefulness, weaknesses, and strengths. Second, hypergeometric distribution characterizes a discrete process and such discrete distributions are being

utilized in the state-of-the-art backbone extraction methods including ours. As we discussed earlier, null models developed based on it can have certain undesirable properties for certain networks. Therefore, a natural avenue is to develop new null models that are more appropriate for the case of link weights that are continuous or that can be equivalently represented in different units.

2.6 Software package

The code for the extraction methods proposed in this chapter is published as an open-source Python package *signed_backbones* and an open-source R package *signed.backbones*.

In Python, the package can be installed using the package manager *pip* as follows.

```
pip install signed_backbones
```

In R, the package can be installed via CRAN as follows.

```
install.packages("signed.backbones")
```

In both packages, the filters are provided through a function named *extract()*. Its parameters are the same for the packages in both languages and are summarized below. Language-specific details are documented within the respective packages.

- *edgelist*: First two columns contain node pairs, and the third column contains the edge weights. If *directed = True*, columns should be in this order: source node, target node, edge weight.
- *directed*: *True* or *False* for indicating whether the input network is directed.
- *significance_threshold*: Threshold for the significance filter. If filtering is based directly on α values: a single nonnegative value, e.g., 1.23; or a couple of nonpositive and nonnegative values, e.g., (-1.23, 4.56). If filtering is based on ranking: a single percentage value in the format such as '10pc'; or a couple of percentage values in the format such as ('5pc', '5pc').

- *vigor_threshold*: Threshold for the vigor filter. A single nonnegative value in the range [0, 1], e.g., 0.33; or a couple of nonpositive and nonnegative values in the ranges [-1, 0] and [0, 1], e.g., (-0.5, 0.3).
- *return_weights*: Whether the returned backbone should contain the signed link weights that show the intensity of the link sentiment.
- *return_significance*: Whether the returned backbone should contain the link significance values that are benchmarked against the *significance_threshold*.
- *max_iteration*: Maximum number of iterations to be used in the IPFP.
- *precision*: A small epsilon value for numerical precision issues. It can be left as default.

The *extract()* function returns a tabular data where each row represents a link in the extracted backbone. First two columns contain node pairs, and the third column contains the edge sign. If *directed = True*, columns are in this order: source node, target node, edge sign. If *return_weights = True*, signed edge weights (β_{ij}) are returned instead of edge sign. If *return_significance = True*, a fourth column containing significance values (α_{ij}) are returned.

The following piece of code in Python extracts the signed backbone of an undirected network using significance filter threshold $-\alpha^- = \alpha^+ = 2.576$ and vigor filter threshold $(\beta^-, \beta^+) = (-0.3, 0.2)$. The variable *sbb* contains the signed edges of the extracted backbone.

```
import signed_backbones as sb
import pandas as pd
net = pd.read_csv('edgelist.txt')
sbb = sb.extract(net, directed = False, significance_threshold =
↪ 2.576, vigor_threshold = (-0.3, 0.2))
```

The following piece of code in R extracts the signed backbone of a directed network using significance filter threshold $(\alpha^- = \alpha^+) = ('10pc', '10pc')$ and vigor filter threshold $-\beta^- = \beta^+ = 0.1$. The variable *sbb* contains the signed and weighted edges of the extracted backbone.

```
net <- read.csv('edgelist.txt')
sbb <- signed.backbones::extract(net, directed = TRUE,
  ↪ significance_threshold = c('10pc', '10pc'), vigor_threshold =
  ↪ 0.1, return_weights = TRUE)
```

CHAPTER 3

ALIGNMENT AND STABILITY OF EMBEDDINGS: MEASUREMENT AND INFERENCE IMPROVEMENT

3.1 Introduction

Complex information about objects and relationships between them can be effectively and efficiently represented in low-dimensional latent spaces instead of their original representations in high-dimensional spaces. Examples of such complex systems include networks, text documents, or images. Representation learning methods are concerned with inferring such low-dimensional representations. These representations are also called embeddings. In the case of networks, one popular research question is to learn low-dimensional node embeddings from high-dimensional adjacency matrices. In natural language processing field, another popular research question is to learn word embeddings. In general, any pair of objects that are similar in the original system should also be closer in the latent space and vice versa. Therefore, embeddings preserve most of the useful information and can be ultimately utilized in various problems such as prediction of metadata in networks and relations, or detection of clusters.

The reference system of a latent space, in general, has no physical meaning itself. The coordinates of a single object in the latent space, i.e., the numbers which make up the embedding vector of an object, have no standalone usefulness. The useful information comes from its relative position to the other objects. In simplified terms, the information is provided by the distances between nodes. It follows that transformations of a latent space that do not change the pairwise distances do not change the available information. Hence, in static systems such as static networks, one can obtain different embeddings that provide exactly the same information, e.g., possibly based on initial conditions provided to a specific representation learning method. This phenomenon and issues associated with it do not arise for tasks on static systems since there is only a single latent space to represent all objects at once.

In contrast to static systems, dynamic systems contain also temporal information. For instance, over time, words might change their meaning or nodes might change their metadata. For the former, the distance between the embedding vectors of a word at different time periods might show its semantic drift. For the latter, the decision boundaries learned in a previous timestep for a classification task can be used to predict the node labels in a later timestep. However, to investigate such temporal evolution using embeddings, it is necessary to have a fixed reference system for the latent space over time. Once the latent space is fixed over time, it also becomes possible to find average embeddings over time or to visualize temporal changes in the latent space.

For a given dynamic system, the difference between the learned representations for different timesteps can be caused either by actual changes in the system or by the transformations that do not alter the pairwise distances in the latent space. The first is related to *stability* of the system over time whereas the second is related to the *alignment* of embeddings which is an artifact of the utilized representation learning method and details of its implementation. For instance, a slightly changing system might produce drastically different embeddings at each timestep when embeddings at each timestep are learned separately by a static representation learning method because such small changes might significantly change the regions in which the learned embeddings converge. Moreover, even a system that does not change over time might result in different embeddings in different timesteps due to the stochasticity in many representation learning methods. In summary, stability characterizes the dynamics of the system itself, whereas misalignment characterizes spurious changes in the latent space. Therefore, dynamic embeddings are said to be *aligned* when the difference between different timesteps' embeddings reflect the actual changes in the system rather than the ineffectual transformations of the latent space.

Overall, the issue of stability and alignment of dynamic embeddings carries utmost importance in tasks that make use of the temporal changes in the latent space

such as future metadata prediction, temporal evolution, dynamic visualization, obtaining average embeddings, etc. Although acknowledged in the literature, to the best of our knowledge, the issue has not been defined formally, explored theoretically, or analyzed empirically in a sufficient manner.

In this chapter, we make the following contributions:

- We dismantle the alignment and explore and explain its parts; and provide the first formal definition for embedding alignment.
- We develop appropriate and mathematically justified metrics to measure the alignment and stability of dynamic embeddings and show their usefulness through a series of extensive synthetic and real-world experiments.
- We employ more than 10 various static and dynamic network representation learning methods and analyze alignment and stability metrics as well as inference task performance on seven real-world datasets with different characteristics.
- We show that when embeddings are forced to be aligned based on provided formal definitions, remarkable improvements in accuracy are obtained in dynamic network inference tasks.

The remainder of this chapter is structured as follows. In the next section, we explore the topic of alignment in more detail, look at its description in the literature, and review the existing approaches. In Section 3.3, we provide formal definitions for embedding stability and embedding alignment and propose a set of metrics to measure them. In Section 3.4, we conduct extensive synthetic experiments to show the appropriateness and behavior of the proposed measures. In Section 3.5, we conduct experiments using several real-world datasets and many embedding methods to test our approach in real-world cases and explore the relationship between alignment, stability, and dynamic inference task performance. The conclusion and final remarks are given in Section 3.6. We also provide the proposed alignment and

stability measures and alignment technique as an open-source software package and describe its use in Section 3.7.

In the rest of this chapter, particularly in the real-world experiments, we focus on the case of dynamic network node embeddings. However, our proposed measures require only the embedding matrices and are agnostic to how and from what type of system these embeddings are created. Therefore, the proposed measures are directly generalizable to other types of embeddings such as subgraph embeddings, entire graph embeddings, word embeddings, document embeddings, image embeddings, and so on. Even more, our measures may be used outside the dynamic system assumption, e.g., to obtain average embedding vectors from different representation learning methods in a static system.

3.2 Background

First inspired by the advances in the natural language processing methods, graph representation learning techniques have seen a surge in the last years regarding their performances and the possibilities they offer. Graph embedding techniques can be classified into different categories. Some of them are based on node similarity matrix factorization (Belkin & Niyogi, 2001; S. Cao, Lu, & Xu, 2015; Ou, Cui, Pei, Zhang, & Zhu, 2016). Another type adapts the notion of random walks to graph data to form sequences of nodes, similar to sentences within word embedding techniques (Grover & Leskovec, 2016; Perozzi, Al-Rfou, & Skiena, 2014). Also, some methods take advantage of the advances in deep learning (S. Cao, Lu, & Xu, 2016; Kipf & Welling, 2016; Wang, Cui, & Zhu, 2016).

Although those methods were ground-breaking, they are static as they were conceived to address graphs that do not evolve over time. When a static method is applied to compute dynamic network representations over its timesteps, the produced embeddings do not abide by any temporal coherence. As a matter of fact, pairwise euclidean distances between node embeddings remain unchanged regarding some operations like translation, rotation, or reflection. This means that, as described in

Figure 7, two consecutive timesteps static methods embeddings can be shifted or rotated one regarding the other as they might be placed in different latent spaces. In such situations, temporal embeddings are said to be misaligned.

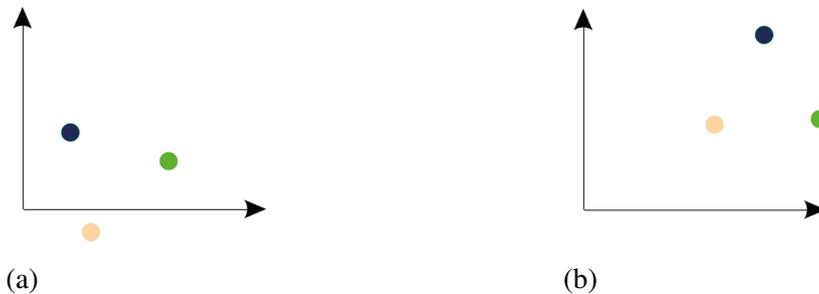


Figure 7. An example of misalignment

(a) Sample embeddings at time t . (b) The embeddings for the same objects at time $t + 1$. Although pairwise distances are preserved between the two timesteps, the embedding vectors are different. It demonstrates a misalignment where the difference is fully induced by a translation and a rotation.

Few dynamic approaches handle time (and alignment consequently) in the conception of the embeddings. Some approaches apply an optimal linear transformation minimizing the distance between the consecutive timesteps' independently learned embeddings (Kulkarni, Al-Rfou, Perozzi, & Skiena, 2015; Y. Zhou et al., 2019). Some others initialize the current timestep representation with the previous embedding vectors within the learning process (Mahdavi, Khoshraftar, & An, 2018). It is also possible to force embeddings continuity over time (Haddad, Bothorel, Lenca, & Bedart, 2020).

The main advantage of the dynamic embedding methods lies in the incorporation of temporal information to conceive more reliable global embeddings. However, alignment is an important issue that most of these methods do not generally address in particular. Some studies in the literature mention the alignment issue without providing an explicit way to handle it (S. Liang, Zhang, Ren, & Kanoulas, 2018; Palmucci, Liao, Napoletano, & Zaccaria, 2020). In word embeddings, Kulkarni et al. (2015) uses locally linear regression to align the neighborhood of a focal word in different timesteps. However, this operation has to be applied for each focal word independently. Singer, Guy, and Radinsky (2019) and Xu, Tao, Yan, and Lin (2018)

employ orthogonal procrustes analysis: the idea is to find and apply the orthogonal matrix (i.e., rotation and/or reflection) that maps the two consecutive timesteps' embeddings most closest to each other using singular value decomposition (Schönemann, 1966). Fang, Kohram, Meng, and Ralescu (2011), Hewapathirana, Lee, Moltchanova, and McLeod (2020), and Passino, Bertiger, Neil, and Heard (2019) employ generalized procrustes analysis where optimal translation and scaling operations are allowed in addition to the rotation (Gower, 1975). It should be noted that orthogonal procrustes analysis does not offer the possibility of finding possible translations between embeddings of different timesteps, which might be a potential drawback when the distance metric to be considered in the embedding latent space is the euclidean distance rather than the dot product or the cosine similarity.

We consider two sets of embeddings as being aligned if there is no global move (shift, rotation, reflection) between the vectors they are composed of. In addition to its practical aspect, this definition flows from an observation one can make: in all its possible forms (graphs, adjacency matrices, edge lists...), the input data of the common embedding methods represent interactions between nodes; consequently, in terms of nodes embeddings dynamics, if the alignment issue is intuitively seen as movements of nodes, then a global move of embeddings should not be allowed as the input data cannot include such information. Based on that, we also define alignment forcing and measuring as the process of finding and quantifying the optimal linear transformations that bring the embeddings of two timesteps to the closest.

Few works in the literature address the problem of measuring embedding alignment. For embeddings, C. Chen, Tao, and Lin (2019) and Goyal, Kamra, He, and Liu (2018) define quantities for temporal stability (or continuity) that are meant to measure the amount of *misalignment* between two timesteps' embeddings. The general idea is to analyze the correlation between the change of a node's neighborhood and the move over time of a node in the latent space. However, such a quantity encompasses what we consider to be two related but different notions:

alignment and stability. Stability reflects the changes that may occur between consecutive timesteps' embeddings that are not attributable to alignment issues. For an ideally aligned embedding method, stability is directly related to the dynamics of the considered network, i.e., its structural evolution at different scales. It should also be noted that instability may also be caused by the used embedding method due to the randomness involved in the process of representation creation.

3.3 Stability and alignment

In this section, we introduce and define the proposed alignment measures, as well as the strategy used to force alignment between different embeddings. As mentioned before, multiple operations that may occur between consecutive timesteps' embeddings can compromise the alignment. So is the case with translations, rotations, and reflections. Then, an appropriate measure of the alignment should quantify the amplitude of each one of these operations. However, trying to measure alignment with a single value may hide different situations: for example, a rotation of some angle θ and a translation of vector \vec{t} may produce the same alignment measure. Such equivalencies do not seem to be relevant. Instead, we choose to design an alignment measure for each of the operations that come into play in alignment.

In general, we denote scalars with lowercase letters (e.g., x), matrices with uppercase letters (e.g., X), timesteps with lowercase superscripts (e.g., X^t), and row vectors of a matrix with lowercase letters followed by a single subscript (e.g., x_i) where $X = \{x_i\}$. $\|X\|$ denotes the Frobenius norm of X .

3.3.1 Translation error

Translation error intends to quantify the average global shift between embeddings in two consecutive timesteps. More specifically, we are interested in the move of the center of gravity across time. Given two consecutive embeddings $\{e_i^t\}$ and $\{e_i^{t+1}\}$ for a node $i \in \llbracket 1, |V| \rrbracket$ where $|V|$ is the number of nodes, we define the global shift t_{glob} in Equation 3.1.

$$t_{glob} = \|o^{t+1} - o^t\| \quad \text{where} \quad o^t = \frac{\sum_{i=1}^{|V|} e_i^t}{|V|} \quad (3.1)$$

As t_{glob} is not representative due to its dependence on the embedding space, one has to normalize it regarding some characteristic distances of the embeddings in the latent space. In our case, we choose the radius, i.e., the average distances to the center of gravity. Equation 3.2 defines the quantity t_{norm} and radius r^t where o^t and t_{glob} respectively refer to the center of gravity (Equation 3.1) and the global shift (Equation 3.1). It is worth noting that, as Figure 8 shows, t_{norm} is equal to 1 when both timesteps' embeddings are *tangent* in terms of their radii.

$$t_{norm} = \frac{t_{glob}}{r^t + r^{t+1}} \quad \text{where} \quad r^t = \frac{\sum_{i=1}^{|V|} \|e_i^t - o^t\|}{|V|} \quad (3.2)$$

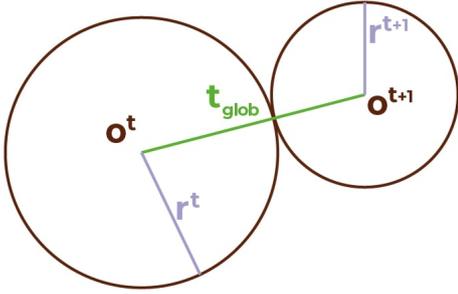


Figure 8. Tangent embeddings regarding their radii

o^t , r^t and t_{glob} respectively refer to the center of gravity of the embeddings of timestep t (defined in Equation 3.1), their radius (Equation 3.2) and the global shift (Equation 3.1). In such a case, t_{norm} is equal to 1 and the translation error ξ_{tr} is equal to 0.5.

The quantity t_{norm} is limitless as it can reach infinite values. To bound our final translation error, we define our proposed translation error ξ_{tr} as in Equation 3.3. Therefore, ξ_{tr} ranges between 0 and 1.

$$\xi_{tr} = \frac{t_{norm}}{t_{norm} + 1} \quad (3.3)$$

Finally, once the translation error is measured, we center the two compared embeddings by shifting them to make their centers of gravity match with the origin of the embedding latent space as shown in Equation 3.4.

$$C^t = \{c_i^t\} \quad \text{with} \quad c_i^t = e_i^t - o^t \quad \text{for} \quad i \in \llbracket 1, |V| \rrbracket \quad (3.4)$$

3.3.2 Rotation error

Given two consecutive timesteps' centered embeddings C^t and C^{t+1} , it is possible to find the optimal global rotation and/or reflection matrix R that maps most closely C^t with C^{t+1} . This can be achieved using orthogonal procrustes as shown in Equation 3.5.

$$R = \operatorname{argmin}_{\Omega} \|C^t \Omega - C^{t+1}\| \quad (3.5)$$

To quantify the rotation/reflection between the embeddings, we compare R to the identity matrix I . In the ideal case where $R = I$, C^t and C^{t+1} are already well oriented one regarding the other. We define our proposed rotation error ξ_{rot} in Equation 3.6 where d is the embedding latent space dimension and $\operatorname{Tr}(R)$ is the trace of the matrix R . The division by $2\sqrt{d}$ ensures that ξ_{rot} ranges from 0 to 1.

$$\xi_{rot} = \frac{\|R - I\|}{2\sqrt{d}} = \frac{\sqrt{2d - 2\operatorname{Tr}(R)}}{2\sqrt{d}} \quad (3.6)$$

One might as well ask the question of the ability of ξ_{rot} to quantify the *amount* of rotation/reflection that R represents. In order to answer this question, we introduce R_{can} , the canonical form of R , in Equation 3.7. Each orthogonal matrix can be decomposed into k elementary rotations and brought into a block diagonal matrix form, e.g., using the Schur decomposition where Q is an orthogonal matrix. More precisely, as Table 6 shows, the form of R_{can} depends on the parity of d and the sign of the determinant of R (i.e., whether R is a rotation-only matrix or it may contain both reflection and rotation).

$$R_{can} = Q^T R Q = \begin{bmatrix} R_1 & & & & & \\ & \ddots & & & & \\ & & R_a & & & \\ & & & \pm 1 & & \\ & & & & \ddots & \\ & & & & & \pm 1 \end{bmatrix} \quad \text{with } R_j = \begin{bmatrix} \cos(\theta_j) & -\sin(\theta_j) \\ \sin(\theta_j) & \cos(\theta_j) \end{bmatrix} \quad (3.7)$$

Table 6. Orthogonal Matrix Canonical Form

| | odd dimension | even dimension |
|---------------------------------------|--|---|
| rotation only $\det(R) = 1$ | $\begin{bmatrix} R_1 & & & \\ & \ddots & & \\ & & R_{\frac{d-1}{2}} & \\ & & & 1 \end{bmatrix}$ | $\begin{bmatrix} R_1 & & & \\ & \ddots & & \\ & & R_{\frac{d}{2}} & \\ & & & 1 \end{bmatrix}$ |
| rotation/reflection $\det(R) = -1$ | $\begin{bmatrix} R_1 & & & \\ & \ddots & & \\ & & R_{\frac{d-1}{2}} & \\ & & & -1 \end{bmatrix}$ | $\begin{bmatrix} R_1 & & & \\ & \ddots & & \\ & & R_{\frac{d-2}{2}} & \\ & & & 1 \\ & & & & -1 \end{bmatrix}$ |

It is possible to look at R_{can} as an expression of R in a basis where it can be decomposed into k elementary 2-dimensional rotations in orthogonal planes (with $\{\theta_j, j \in \llbracket 1, k \rrbracket\}$ as rotation angles), and possibly a reflection. Both matrices, R and R_{can} , have the same rotation error ξ_{rot} as they share the same trace. Also, as the trace of R_{can} is related the θ_j angles, ξ_{rot} can be written as in Equation 3.8. Thus, we can observe that the smaller θ_j angles are, the smaller ξ_{rot} is. Also, reflections increase the rotation error.

$$\xi_{rot} = \frac{\sqrt{2d - 4 \sum_{j=1}^k \cos(\theta_j) - 2\mu}}{2\sqrt{d}} \quad \text{where } \mu = \begin{cases} 1 & \text{if } d \text{ is odd \& } \det(R) = 1 \\ -1 & \text{if } d \text{ is odd \& } \det(R) = -1 \\ 0 & \text{if } d \text{ is even} \end{cases} \quad (3.8)$$

Lastly, after measuring the rotation/reflection error, we rotate C^t embedding matrix according to R matrix in order to obtain centered well-oriented embeddings at

consecutive timesteps, as shown in Equation 3.9. We consider C_r^t and C^{t+1} to be the aligned versions of the input embeddings.

$$C_r^t = C^t R \quad (3.9)$$

3.3.3 Scale error

Scale error intends to examine the change of scale over time of centered well-oriented embeddings. For this purpose, we define our proposed scale error ξ_{sc} in Equation 3.10 based on the radii, i.e., the average distances to the origin. The scale error ranges from 0 to 1. In particular, ξ_{sc} is zero when the embeddings have exactly the same radius. To the contrary, ξ_{sc} approaches 1 when the radii of the embeddings are very different.

$$\xi_{sc} = \frac{|r^t - r^{t+1}|}{r^t + r^{t+1}} \quad (3.10)$$

Once the scale error is measured, we normalize the centered well-oriented embeddings by their respective radii as shown in Equation 3.11.

$$N^t = \frac{C_r^t}{r^t} \quad (3.11)$$

It should be noted that we do not consider the scale error as being part of the alignment. As a matter of fact, contrary to translation, rotation, and reflection, pairwise distance matrices are not invariant under the change of scale. However, the change of scale can indicate the evolution of the density of embeddings and, consequently, the change of the strength of the interactions between the input graph nodes.

3.3.4 Stability error

Given two centered, well-oriented and normalized embeddings, it is interesting to examine the evolution of embeddings over time. Indeed, the difference between the respective characteristics of $N^t = \{n_i^t\}$ and $N^{t+1} = \{n_i^{t+1}\}$ cannot be attributable to alignment issues. Thus, the comparison between N^t and N^{t+1} gives information about the structural dynamics of the embedded network, like the temporal continuity

of the embeddings or the noise involved between consecutive timesteps. We define the stability error as in Equation 3.12. The stability error ξ_{st} ranges from 0 to 1. It is equal to 0 if and only if $N^t = N^{t+1}$.

$$\xi_{st} = \frac{\sum_{i=1}^{|V|} \frac{\|n_i^{t+1} - n_i^t\|}{\|n_i^t\| + \|n_i^{t+1}\|}}{|V|} \quad (3.12)$$

Even though the stability error is computed after the alignment and scale normalization process, it brings additional information about the relevance of the other defined measures. For example, in the extreme case of comparing two randomly generated embedding matrices, one can still obtain the values of ξ_{tr} , ξ_{rot} and ξ_{sc} . However, these values would not be very meaningful. In such a case, the stability error would be relatively high.

3.4 Experiments on synthetic networks

We design and conduct extensive synthetic experiments to demonstrate the suitability and characteristics of the proposed error measures. The behavior of each error measure is explored under interesting combinations of transformations such as scaling, translations, rotations, and noise; by applying them to an initial matrix (e.g., E^t) to create the transformed matrix (e.g., E^{t+1}). The error measures are then computed to investigate the stability and alignment of the two embeddings, where we can trace the differences directly to the applied synthetic transformations. We are also interested in whether the proposed measures are affected by the shape of the embedding matrix, represented by the number of objects and the length of embedding vectors.

Any rotation matrix for an embedding matrix can be represented in its canonical form with $d/2$ angles or $(d-1)/2$ angles and an optional reflection when d is even, or with $(d-1)/2$ angles and a reflection when d is odd. Intuitively, we expect our rotation error to reflect the magnitude of the optimal rotation matrix as measured by its underlying angles. To test this, and to provide a visual demonstration for the cases when $d = 3$ and $d = 4$, we create a large number of rotation matrices with

varying rotation angles. Figure 9a visualizes the obtained rotation error for different angles using two separate series for the cases where reflection is applied or not. The rotation error is at its maximum when the rotation angle is maximized, i.e., when it is equal to π regardless of whether the reflection is applied or not. We also observe that the application of reflection on top of any angle of rotation worsens the error.

As depicted in Figure 9b the rotation error is maximized when both angles are equal to π , and lowest when both angles are 0. Overall, Figure 9 shows the clear relationship between the angles of rotation and reflection and the proposed rotation error, visually confirming the intuitive expectations and the mathematical implications.

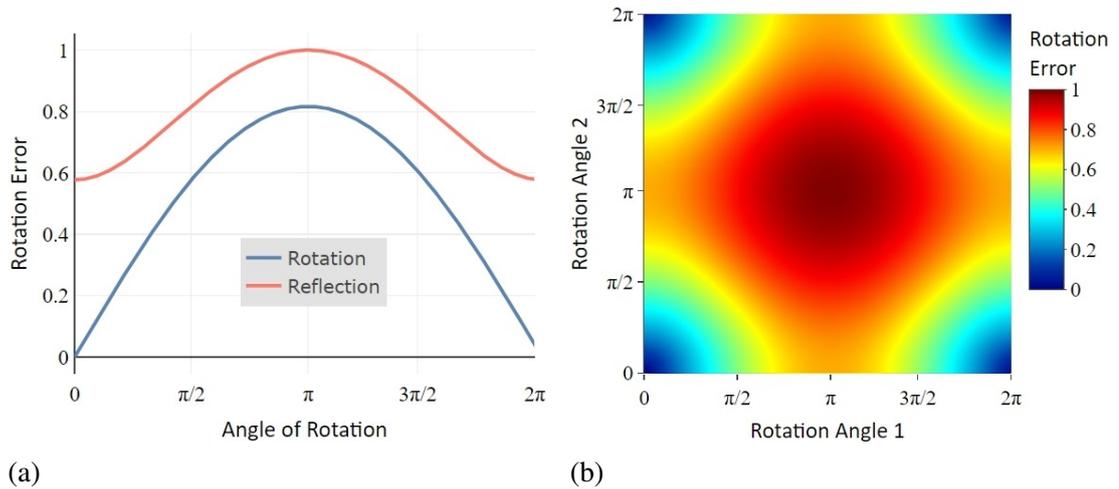


Figure 9. Magnitude of rotation versus rotation error

(a) The angle of rotation, an optional reflection, and corresponding rotation error when $d = 3$.
 (b) The two angles of rotation and corresponding rotation error when $d = 4$. Overall, rotation error reflects magnitudes of rotation angles and the optional reflection.

When two embeddings are the same except that one is a scaled-up (i.e., dilated) or a scaled-down (i.e., contracted) version of the other, we expect the scaling error to reflect the magnitude of scaling. To demonstrate this, we create a large number of initial embedding matrices, apply scale transformations with different magnitudes of scaling factors, and report the results. To create an initial embedding matrix, the number of objects n is uniformly sampled from the range $[10, 10000]$, the length of the embedding vectors d is uniformly sampled from the range $[2, 32]$, and

the matrix is filled with $n \times d$ values uniformly sampled from the range $[0, 1]$. Each initial embedding matrix is then scaled with different scaling factors such that the center of gravity is preserved and the ratio between radiuses of the transformed and initial embedding matrices is equal to the scaling factor. Figure 10a reports the results of this experiment; and demonstrates that the scale error is 0 when two embeddings have the same scale (i.e., when the scaling factor is 1 hence the radiuses are equal) and approaches 1 as the scaling factor moves away from 1 in either direction (e.g., dilation or contraction).

In a similar fashion, when two embeddings are the same except that one is shifted in some direction in the space, we expect the translation error to reflect the magnitude of the shift. To demonstrate this, again, we create a large number of initial embedding matrices with the same sampling procedure. These matrices are then shifted in random directions such that the ratio of euclidean norm of the shift vector to the radius of the initial matrix is equal to the shifting factor. Figure 10b reports the result of this experiment; and demonstrates that the translation error is 0 when the shifting factor is 0 (i.e., the case of no translation) and approaches 1 as the shifting factor increases. Overall, Figure 10 shows graphically that the bounds of scale and translation errors are meaningful, the values within their range directly reflect the magnitudes of respective transformations as they should, and the measures are independent of the shape of the embeddings; providing the necessary properties we expect from such measures.

Next, we investigate the behavior of our error measures when the difference between the two embeddings can be explained only by scaling and shifting together. The initial embeddings are created and shifting and scaling are applied, all following the procedure employed for the previous experiment. In the case of scale error, we expect it to be independent of shifting since it is concerned with their radius of the embeddings and not where the embeddings are in the space. Confirming this, Figure 11a depicts that the scale error increases if and only when the scaling factor moves away from 1. However, the translation error does not exclusively depend only on how

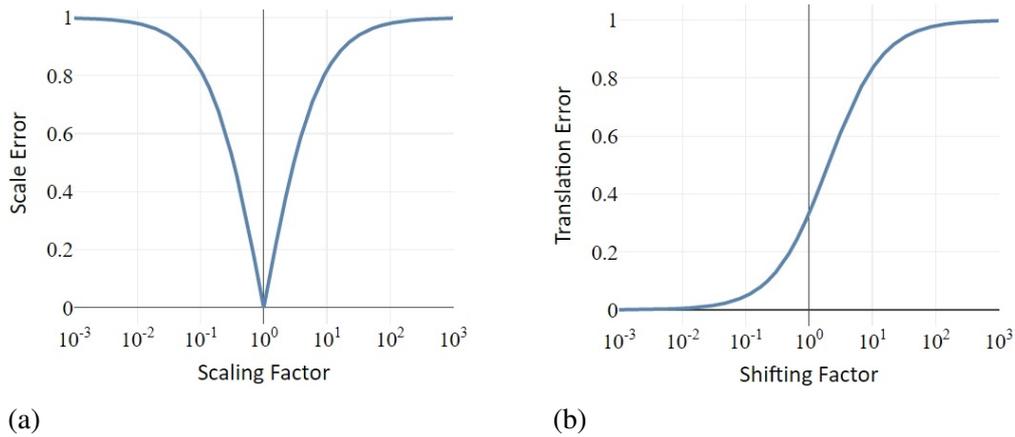


Figure 10. Scale and translation errors versus magnitudes of the respective transformations

(a) Scaling factor and corresponding scale error. (b) Shifting factor and corresponding translation error. Overall, scale and translation errors reflect the magnitudes of respective transformations.

distant two embeddings are in the space since such distance is meaningful and comparable with respect to the radiuses of the embeddings in question. Figure 11b visualizes the experiment where initial embedding matrices shifted first and scaled later with the respective factors. Confirming the expected behavior, the figure shows that the translation error is near 0 when the shifting factor is near 0. It increases as the shifting factor increases; and decreases at the same time when a large scaling factor increases the radius of the transformed matrix. In general, the smoothness of (i.e., lack of randomness in) patterns observed in Figure 11 visually supports the earlier finding that two measures are independent of the shape of embeddings, hence generalizable across systems with different numbers of objects and different embedding vector lengths.

The experimental results along with the mathematical definitions provided earlier show that the effects of the three linear transformations are captured by the error measures specifically designed for them and the resulting error values are directly related to the magnitude of respective transformations. When the only difference between two embeddings is a rotation, the radius and the center of gravity do not change; hence only the rotation error produces a nonzero value. Likewise, if the only change is the radius or the center of gravity, only the respective scale or

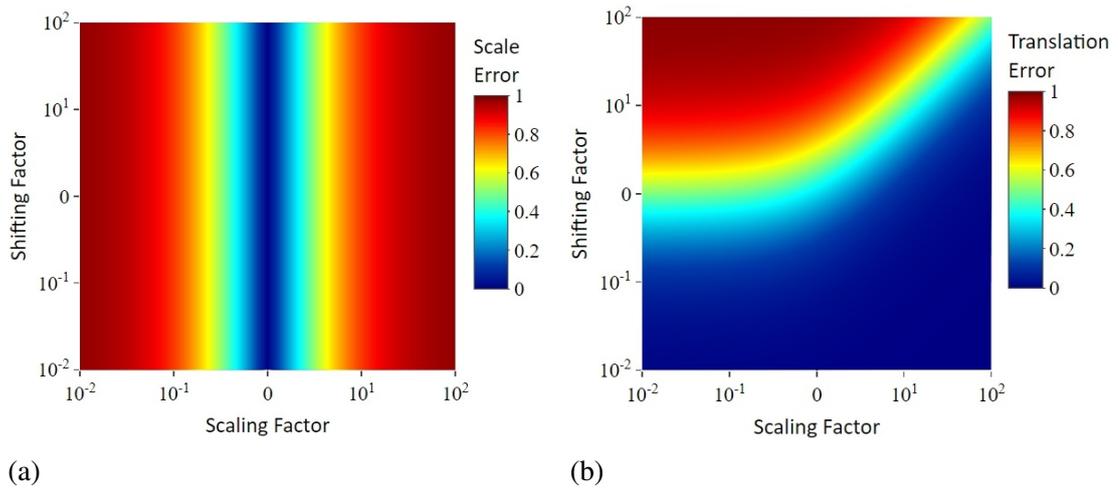


Figure 11. Combined scaling and shifting versus scale and translation errors

(a) Scaling and shifting factors, and corresponding scale errors. (b) Scaling and shifting factors, and corresponding translation errors. Overall, effects of combined scaling and shifting operations with different magnitudes on the scale and translation errors show that scale error reflects only the magnitude of scaling whereas translation error reflects the scale in addition to the magnitude of shifting due to the normalization by average radius.

translation error produces a nonzero value. Accordingly, we can conclude that each of these measures only monitors the transformations relevant to them. The same is true when the difference between embeddings can be explained by a combination of these transformations since these well-separated transformations do not change the aspects of data monitored by the other error measures. The only exception to that is the translation error since it employs the average of the radiuses as its unit to be comparable across different settings. In particular, the value in its numerator still stays the same under non-shift transformations but its denominator changes with scaling. In general, these properties allow us to treat different error measures independently from each other. Specifically, we can reduce a selected nonzero error to zero by applying the respective reverse transformation and other error measures will not behave unexpectedly, i.e., their values will be exactly the same except for the translation error which will change in a predictable manner if the applied reverse transformation is scaling.

Unlike the previous three measures, the stability error evaluates and captures the changes in structure, more specifically the relative structural changes. It corrects for the rotation, translation, and scaling since the relative structure does not change

under these transformations. A difference that can be explained fully by any combination of these transformations cannot produce a nonzero stability error. To investigate the behavior of stability with synthetic experiments, we introduce structural changes to the embeddings by adding noise. Specifically, the addition of noise will change the distances between objects in the latent space.

The noise addition procedure is modeled as a random walk in d -dimensional space where the number of random walk steps corresponds to the noise factor. At each random walk step, the embedding matrix $E^{n \times d}$ is updated following Equation 3.13 where $U(a, b)^{n \times d}$ is a noise matrix with the same shape as embeddings whose values uniformly sampled from the range (a, b) . This ensures that all steps are random across directions and across objects. The term \sqrt{d} ensures that the noise factor is equivalent across different embedding shapes. $r/2$ is a practical choice that ensures that step size is neither too small nor too large. It follows that 25 random walk steps result in added noise with approximately the same radius as the original embedding due to the result given in Equation 3.14 where $u(-1, 1)$ is a scalar uniformly sampled from the respective range. Accordingly, we set 25 steps as the unit for the noise factor, i.e., noise factor of 1 corresponds to noise with the same radius as the original embedding on average.

$$E \leftarrow E + \frac{U(-1, 1)r}{2\sqrt{d}} \quad (3.13)$$

$$P\left(\sum_{1 \text{ to } 25} \frac{u(-1, 1)}{2} > 1\right) \simeq P\left(\sum_{1 \text{ to } 25} \frac{u(-1, 1)}{2} < 1\right) \simeq 50\% \quad (3.14)$$

As is the case with other measures, we expect stability error to be robust against different embedding shapes. To test this, we create initial matrices following the procedure we employed earlier, change their structure via the introduction of noise with the noise factor fixed at 1, and observe the stability error between the pairs of embeddings. Figure 12a visualizes the result of this experiment. Except for the regime characterized by very low n/d values, the stability error is steady across the number of objects and embedding length. This low- n/d regime is not very relevant in

real-world cases since representation learning aims to learn embeddings satisfying $d \ll n$. Moreover, as soon as the number of nodes modestly increases to above 100, it already reaches the robust region even if d is relatively large at 32. In light of these results, we see that our stability measure is robust to the shape of embeddings under realistic scenarios. Following this, using a large number of initial matrices of different shapes as usual, we plot stability error against the noise factor in Figure 12b. The lack of large fluctuations also confirms the robustness of stability error to embedding shapes. On the other hand, we observe that stability error does not reach its maximum value but diminishes around 0.7. This is because the upper bound of stability error is produced only under very extreme cases that are practically improbable to replicate with random embeddings and noise. In addition to the robustness result, Figure 12b demonstrates the direct relationship between the extent of change in the embedding structure and values of the stability error.

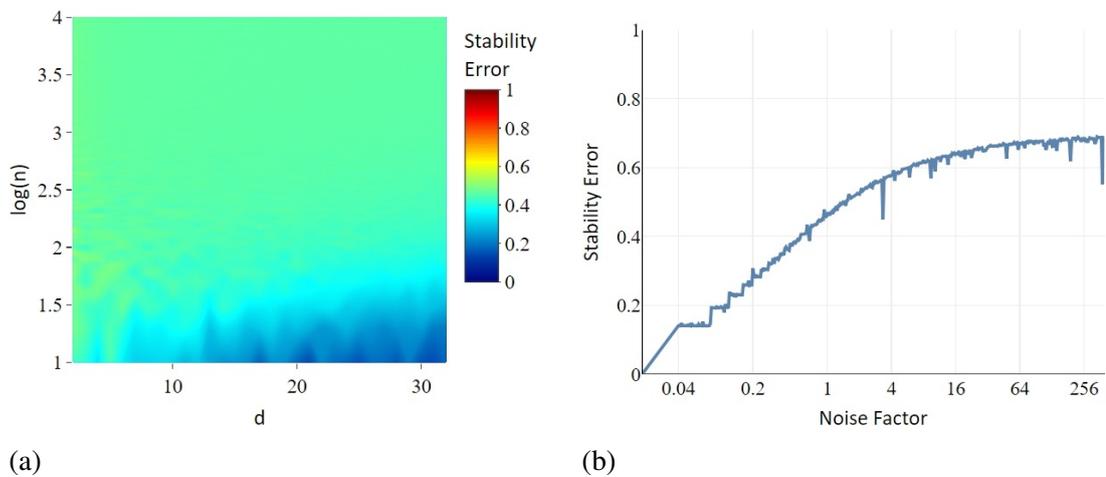


Figure 12. The relationship between noise and stability error

(a) When a noise of the same magnitude is introduced to embeddings of varying shapes ($n \times d$), stability error values are quite stable; demonstrating the measure's robustness against number of objects and embedding size. (b) Stability error reflects the magnitude of the introduced noise as more noise is assumed to reflect more dramatic structural changes in the dynamic system.

In the presence of both structural changes achieved by random noise walks and other transformations that do not alter the relative structure, we expect stability error to be independent of such linear transformations. To test and demonstrate this, a large number of initial matrices are created with the usual procedure, and the linear

transformations and noise are introduced with varying magnitudes. Figure 13 presents the results of these experiments for each noise and linear transformation pair. As shown by the figure, stability error is linked to the extent of changes introduced to the relative structure of objects and completely independent from transformations that do not impact the relative structure.

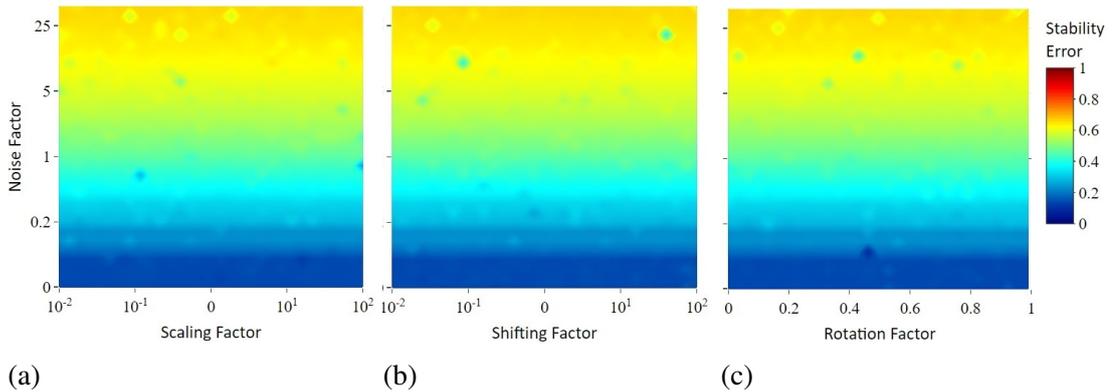


Figure 13. Combined effect of noise and scaling, shifting, or rotation on stability error. The subfigures share the y-axis and the colorbar. (a) Scaling and noise factors, and corresponding stability errors. (b) Shifting and noise factors, and corresponding stability errors. (c) Rotation and noise factors, and corresponding stability errors. Overall, stability error reflects the magnitude of noise; and scaling, shifting, and rotation have no effect on it.

Finally, we investigate behaviors of the scale, translation, and rotation errors against their respective linear transformations and the noise added by the random walk procedure. Figure 14a shows that the scale error, in addition to the magnitude of the applied scaling, is affected also by the magnitude of the noise. Specifically, as we contract the embedding (i.e., scaling factor < 1), the scale error does not increase as fast if a large magnitude of the noise is introduced as well. The reason for this is rather simple: as objects take more random steps in lengthier walks, they spread further than their original boundaries and the radius of the embeddings slowly increases. Such an increase in the radius plays a balancing role against contraction, resulting in the observed behavior in the figure. The tendency of random walks to increase the radius also causes the behavior observed in Figure 14b. The increasing radius increases the normalization term that is used in the translation error computation (i.e., $r^t + r^{t+1}$ in Equation 3.2). As a direct consequence, under very

large random walks, translation error gets reduced. Overall, such behavior resulting from the increasing radius is due to the way we model noise in our synthetic experiments rather than the characteristics or properties of our measure and is visible conspicuously only in very lengthy random noise walks.

Very lengthy random noise walks, as shown in Figure 14c, transform any embedding matrix into a random-like matrix. In the regime characterized by large noise factors, the rotation error converges to a large value even if the initial rotation factors are different. This value is not the upper bound of the rotation error since its worst value is obtained with 180° rotations and an additional reflection for odd dimensions (i.e., with the negative identity matrix) instead of a random matrix. In the regime characterized by moderate and more realistic noise levels, rotation error is able to distinguish between the noise and original structure and produces values in line with the respective values of the rotation factor. Keeping in mind that a noise factor of 1 produces a random noise with the same magnitude as the original embedding, the result also demonstrates the capability of our rotation error measure to capture the rotation differences even under considerable amounts of noise.

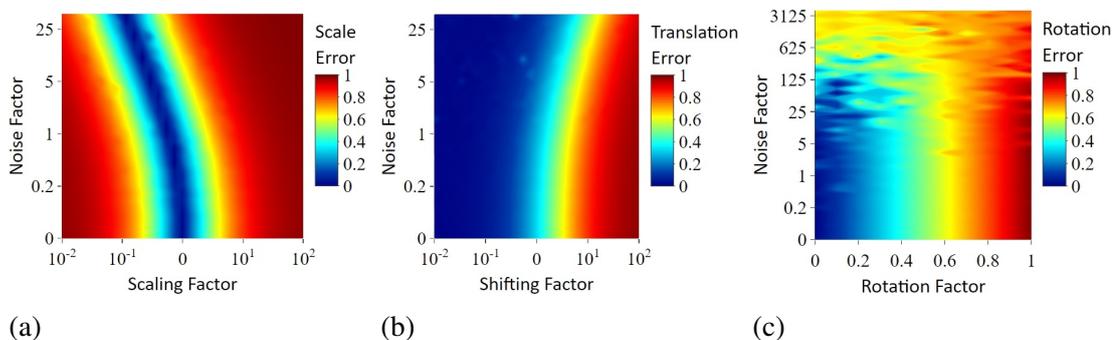


Figure 14. Combined effect of noise and scaling, shifting, or rotation on scale, translation, and rotation errors

(a) Scale error reflects the magnitude of scaling. It is also affected by the magnitude of noise of due to the increasing radius. (b) Translation error reflects the magnitude of shifting. It is also affected by the magnitude of noise of due to the increasing radius. (c) Rotation error reflects the magnitude of rotation. It is also affected by the magnitude of noise since very long random noise walks loses the information in the original embedding.

Here, we should note again that alignment is relevant for the cases where it is possible to find a matching structural pattern between two embeddings. A large stability error indicates that two embeddings are structurally so dissimilar that it is not meaningful to investigate alignment and interpret the results of other error measures. Therefore, the behavior of scale, translation, and rotation errors are not relevant, significant, or practically useful when the length of random noise walk is too large that the introduced noise suppresses the original structure.

In summary, the proposed error measures for stability, scale, translation, and rotation are able to capture the existence and extent of differences introduced by respective structural changes or transformations. They are robust to the different number of objects and embedding vector lengths, and each of them directly and only reflects the changes that it is supposed to measure. These four measures together are fit for use in real-world cases to investigate the existence and degree of stability and alignment and realign embeddings that are stable but misaligned.

3.5 Experiments on real-world networks

In this section, we conduct real-world experiments on embeddings generated for seven real-world datasets using 13 network representation learning methods. First, we measure alignment and stability errors in these embeddings. Then, using the original and aligned versions of these embeddings as input features, we build node classification models and observe their performance on the unseen embedding of the next timestep. In another experiment, we introduce different degrees of misalignment and observe the change in node classification task performance.

3.5.1 Datasets

Using network dataset repositories (Peixoto, 2020; SocioPatterns, 2021) and other resources, we identified seven real-world dynamic networks with fixed or evolving metadata for nodes. For each network, we take appropriate preprocessing steps to ensure that the node set is fixed and all networks are connected over all timesteps.

Information on the datasets is provided below and a summary of the final network datasets is available in Table 7.

- *dutch-school* network (Snijders, van de Bunt, & Steglich, 2010) denotes the friendship network among freshmen students at a secondary school in Netherlands. The data were collected at four times with three-month intervals in 2003 and 2004. Each directed link denotes whether a student considers the other one as a friend or not. Fixed node labels are sex, age, ethnicity, and religion. The evolving node label is delinquency count which we transform into a binary variable showing whether a student has more than one delinquency for the last interval.
- *freshmen* network (van de Bunt, Van Duijn, & Snijders, 1999) denotes the friendship network among freshmen students at the Sociology department of a university in the Netherlands. The data were collected at five times in 1998 and 1999. Each directed link weight denotes the strength of the friendship from the perspective of one student. Fixed node labels are gender, program, and smoking level. Smoking level is transformed into whether a student is a regular smoker.
- *sp-hospital* network (Vanhems et al., 2013) denotes the face-to-face contact data between patients and healthcare workers at a hospital in France from a Monday noon to the next Friday noon in 2010. We process the network such that each link weight represents the total duration of interaction between a pair of people in 24 hours; obtaining four timesteps. The fixed node label is the job title or being a patient.
- *sp-primary-school* network (Stehlé et al., 2011) denotes the face-to-face contact data on two consecutive days at a primary school in France in 2009. Link weights represent the duration of respective face-to-face interaction over a single day. Fixed node labels are class and gender.
- *sp-high-school* network (Mastrandrea, Fournet, & Barrat, 2015) denotes the face-to-face contact data between students in a high school in France over five

days in 2013. We process the network such that each link weight represents the total duration of interaction between a pair of students in one day; obtaining five timesteps. Fixed node labels are class and gender.

- *aminer* network (Jie Tang et al., 2008) denotes the coauthorship relations between researcher. We processed the network such that each edge weight represents the number of coauthored works for a pair of researchers over a 3-year period. There is a total of four three-year periods, spanning 12 years in total. The evolving node label is the research field.
- *yahoo* network (Yahoo!, 2021) denotes the communication between a sample of users of the Yahoo! Messenger over four weeks. The dataset is provided to us through The Yahoo Webscope Program. We process the data such that each directed link weight represents the number of days a user sent a message to the other in a given week; obtaining four timesteps. The evolving node labels are relatively general and relatively specific location areas of the user, obtained respectively from the first one and two letters of the most frequent postcodes in a given week.

Table 7. Datasets

| Name | t | n | Directed | Weighted | Fixed metadata | Evolving metadata |
|-------------------|---|-------|----------|----------|-------------------------------|-------------------|
| dutch-school | 4 | 25 | TRUE | FALSE | sex, age, ethnicity, religion | delinquency count |
| freshmen | 4 | 30 | TRUE | TRUE | gender, age, program, smoking | n/a |
| sp-hospital | 2 | 51 | FALSE | TRUE | job | n/a |
| sp-primary-school | 2 | 232 | FALSE | TRUE | class, gender | n/a |
| sp-high-school | 5 | 244 | FALSE | TRUE | gender, class | n/a |
| aminer | 4 | 10565 | FALSE | TRUE | n/a | research field |
| yahoo | 4 | 46049 | TRUE | TRUE | n/a | postcode |

3.5.2 Embedding methods

We utilize a variety of static and dynamic network representation learning methods. As static methods, we employ random-walk-based methods such as *node2vec* (Grover & Leskovec, 2016), *DeepWalk* (Perozzi et al., 2014); matrix-factorization-based methods such as *GF* (Ahmed, Shervashidze, Narayanamurthy, Josifovski, & Smola, 2013), *LAP* (Belkin & Niyogi, 2001), *HOPE* (Ou et al., 2016); dimension

reduction-based method such as *LLE* (Roweis & Saul, 2000); neural-network-based method such as *SDNE* (Wang et al., 2016); and large graph embedding methods such as *LINE* (Jian Tang et al., 2015). These methods which are originally designed for static networks are applied to generate embeddings for static snapshots of networks at each timestep without any information sharing between different timesteps. As dynamic methods, we employ *DynamicTriad* (L. Zhou, Yang, Ren, Wu, & Zhuang, 2018), *GloDyNE* (Hou, Zhang, He, & Tang, 2020), and *temporalnode2vec* (Haddad et al., 2020).

In addition, we employ the temporalized versions of *DeepWalk* and *node2vec* using the following process. We create t versions of each node for each timestep and concatenate the edge list. In this way, we obtain t disconnected components in a single graph where each component corresponds to a timestep. Then, to ensure information sharing between different timesteps, we create links between the different versions of the same nodes in consecutive timesteps. On this network, we can apply any static representation learning method and obtain a separate embedding vector for each node at each timestep. Lastly, for embeddings methods that require hyperparameters, we choose values based on popular default values available in the relevant literature.¹⁵

3.5.3 Experimental results

Figure 15 depicts rotation, translation, scale, and stability errors produced by each method over all networks and consecutive timesteps. With respect to rotation, we observe that the static methods produce larger errors than the dynamic methods. However, the rotation errors produced by dynamic methods are not insignificant either, with a possible exception of *GloDyNE* which produce only very small rotation errors. With respect to the translation error, random-walk-based methods produce larger errors as well as *SDNE* and *LINE*. Except for the temporalized versions, dynamic methods usually produce lower translation errors than the static ones with a

¹⁵Since our objective does not lie in methods comparison, we do not test different hyperparameters.

very notable exception of *LLE*. With respect to scale error, we observe that most methods usually produce very small errors whereas temporalized methods produce relatively larger errors. With respect to stability error, as expected, we do not observe very meaningful differences between static and dynamic methods. Still, dynamic methods usually produce smaller stability errors possibly due to the information sharing between timesteps which explicitly or implicitly drives embeddings to be closer between consecutive timesteps.

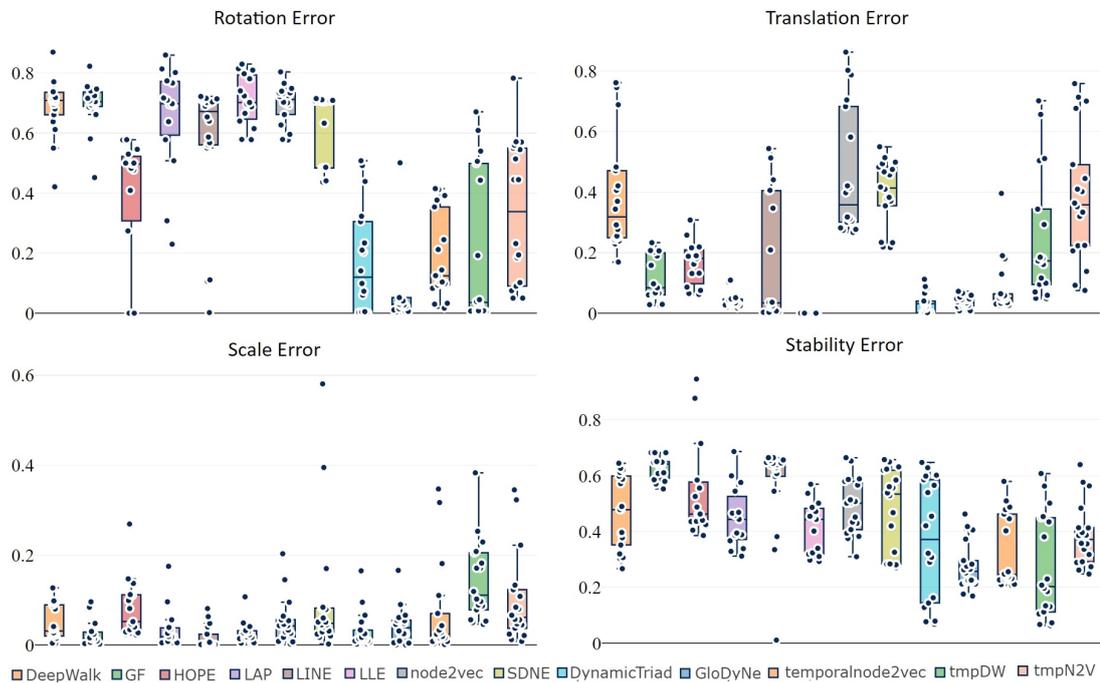


Figure 15. Alignment and stability errors produced by embedding methods

Methods are ordered based on whether they are static or dynamic. Boxplot whiskers are up to 1.5 interquartile range away from the first quartile and third quartile. Individual data points are shown with circle markers.

Overall, our empirical analysis using popular default hyperparameter values for all methods shows that static methods almost always produce large rotation errors and often produce large translation errors. This is expected since the embeddings are learned independently in different timesteps. However, the alignment errors (i.e., rotation and translation errors) produced by dynamic methods are often not trivial either. This result shows that even dynamic methods in the literature do not always guarantee embedding alignment between timesteps.

Next, we perform a network inference task using the available fixed or evolving label information. For each network embedding and node label, we train a Support Vector Machine using the embedding matrix at time t as input features and labels as the output variable. Then, using the model, we predict the labels using the embedding matrix at time $t + 1$. Afterward, we find optimal transformations to ensure alignment and align all embedding matrices over timesteps. After ensuring alignment (i.e., with rotation and translation errors now equal to zero), we perform the same experiment. As a result of this experiment, we obtain classification accuracy scores separately for the original and aligned versions of the embeddings on the unseen test datasets of embeddings at $t + 1$.

Not all labels can be predicted by the network data. For instance, in a setting with perfect gender mixing, there is no information available in the network structure to predict gender. In order to filter out cases with no meaningful learning for the classification tasks, we compare the accuracy scores obtained using aligned embeddings to a dummy classifier. The dummy classifier always outputs the most frequent label as the prediction for all nodes. With a and b respectively denoting the accuracy scores of the learned model and the dummy classifier, we filter out the cases if the following condition is not satisfied: $a > 0.25(1 - b) + b$. In other words, the model should capture the 25% of the information that could not be captured by the dummy classifier.

Figure 16 demonstrates the absolute difference in classification accuracy scores before and after aligning the embeddings, separately for dynamic and static network embedding methods. Figure 16a shows the cases that are filtered out. Confirming our intuitions, there are no significant improvements in classification accuracy in filtered out cases and the values are approximately normally distributed around zero. This is in line with our expectation that if the model is not better at the classification task than a dummy classifier, it is not appropriate to expect improvements after alignment. In the rest of the analysis in this section, we only consider models that are sufficiently better than the dummy classifier. Figure 16b

contains the cases satisfying the above-stated condition. We observe that, in most cases, there is an increase in the classification accuracy after alignment. The small number of negative values are also small in magnitude and are exceptions that may be attributed to some random effects. When alignment is applied as post-processing to the embeddings from static methods, we observe consistent improvement in classification accuracy reaching to 90%. In the case of dynamic networks, we still observe very meaningful and consistent performance increases in classification accuracy, with the difference reaching to more than 40%. Hence, we can conclude that not only the static but also the dynamic embedding methods benefit from alignment.

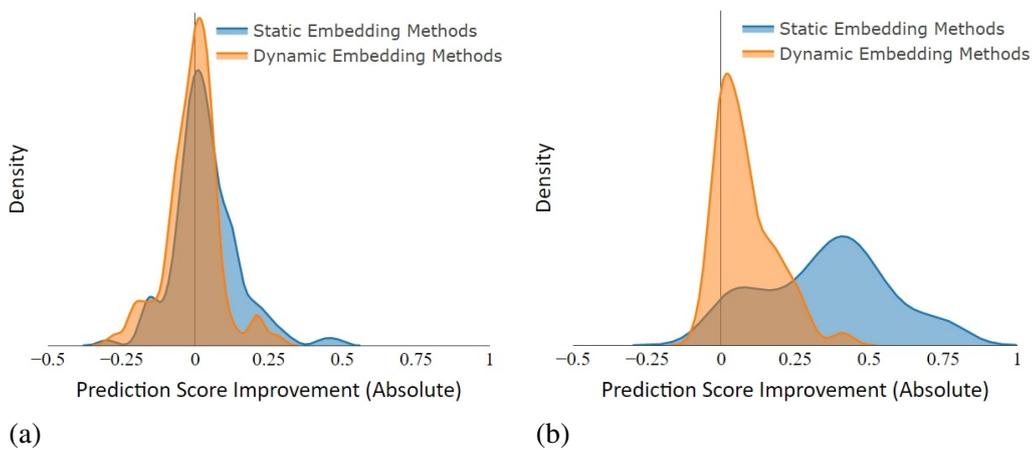


Figure 16. Distribution of prediction score improvement after alignment

Subfigures share the same y-axis. (a) Cases that are filtered out based on the comparison with the dummy classifier. The improvement values are roughly distributed normally around 0, both for the static and dynamic methods. (b) Cases that are retained based on the comparison with the dummy classifier. The improvement values are mostly positive; and are greater on average for the static methods in comparison to the dynamic methods.

After obtaining the result that alignment improves the accuracy in the network inference task, we look at the relationship between the alignment errors and the amount of improvement in classification accuracy. To this end, we calculate partial Spearman's rank correlations between the absolute improvement and alignment errors (rotation and translation errors). First, we control for the translation error and look at the correlation between the improvement and rotation error. Then, we do the same by switching the control variable. The partial correlation of improvement to rotation error is 0.65 ($p < .000001$) and to translation error is 0.42 ($p < .000001$). This result

formally shows that there is a positive relationship between the alignment errors and the amount of improvement that can be obtained by ensuring alignment.

Table 8 provides a closer look into the classification accuracy improvements on the three largest networks we use in this chapter. On the columns, we denote the classification task (dataset, target variable, and consecutive timesteps) and whether the model is tested on the original (O) or aligned (A) embeddings. On the rows, we list the static and dynamic embedding methods separately. We provide average improvements (O-A) over methods (the rightmost column) or over specific classification tasks (in the bottom rows after static and dynamic embedding methods). We observe that the improvement obtained for static embedding methods are larger than the improvement obtained for the dynamic embedding cases. This is in line with the earlier finding that static methods naturally produce more alignment errors hence more room for improvement. In specific cases, we observe that when the embeddings are misaligned, the classification accuracy is as low as 1% but can jump to more than 70% once the same embeddings are aligned.

Table 8. Prediction Accuracy Before and After Alignment

| | sp-high-school | | | | | | | | aminer | | | | | | yahoo | | | | | | | | | O-A | | | |
|------------|----------------|-----|-----|------|-----|-----|-----|-----|--------|-----|-----|-----|-----|-----|--------------|-----|-----|-----|-----|-----|---------------|-----|-----|-----|-----|-----|-----|
| | class | | | | | | | | field | | | | | | general area | | | | | | specific area | | | | | | |
| | 0-1 | | 1-2 | | 2-3 | | 3-4 | | 0-1 | | 1-2 | | 2-3 | | 0-1 | | 1-2 | | 2-3 | | 0-1 | 1-2 | 2-3 | | | | |
| | O | A | O | A | O | A | O | A | O | A | O | A | O | A | O | A | O | A | O | A | O | A | O | | A | | |
| DeepWalk | .01 | .55 | .13 | .66 | .12 | .74 | .09 | .90 | .34 | .68 | .24 | .69 | .35 | .69 | .17 | .63 | .17 | .64 | .18 | .64 | .03 | .43 | .03 | .44 | .04 | .44 | .48 |
| GF | .12 | .16 | .09 | .11 | .12 | .13 | .11 | .13 | .26 | .58 | .27 | .56 | .26 | .60 | | | | | | | | | | | | | .15 |
| HOPE | .12 | .11 | .14 | .13 | .13 | .15 | .09 | .15 | .36 | .36 | .26 | .27 | .13 | .21 | | | | | | | | | | | | | .02 |
| LAP | .14 | .80 | .24 | .65 | .01 | .73 | .07 | .80 | .21 | .46 | .26 | .37 | .06 | .26 | | | | | | | | | | | | | .44 |
| LINE | .11 | .14 | .22 | .17 | .14 | .12 | .16 | .13 | .33 | .51 | .27 | .48 | .15 | .52 | .16 | .59 | .17 | .62 | .21 | .59 | .03 | .51 | .03 | .54 | .05 | .50 | .26 |
| LLE | .23 | .78 | .01 | .65 | .24 | .74 | .02 | .81 | .31 | .51 | .29 | .43 | .25 | .39 | | | | | | | | | | | | | .42 |
| node2vec | .08 | .86 | .07 | .52 | .02 | .74 | .01 | .60 | .24 | .70 | .25 | .70 | .22 | .71 | .16 | .69 | .21 | .70 | .18 | .70 | .02 | .56 | .02 | .59 | .03 | .58 | .55 |
| SDNE | .12 | .46 | .18 | .55 | .08 | .50 | .10 | .51 | .33 | .38 | .19 | .31 | .16 | .28 | .20 | .31 | .21 | .35 | .19 | .39 | .05 | .16 | .04 | .21 | .04 | .26 | .21 |
| O-A | .37 | .30 | .37 | .42 | .23 | .22 | .26 | .38 | .39 | .39 | .38 | .41 | .41 | | | | | | | | | | | | | | |
| Dyn'Triad | .18 | .24 | .14 | .14 | .17 | .21 | .19 | .19 | .52 | .55 | .49 | .53 | .53 | .57 | .68 | .68 | .75 | .75 | .72 | .72 | .61 | .61 | .69 | .69 | .65 | .65 | .02 |
| GloDyNE | .96 | .96 | .92 | .91 | .92 | .91 | .93 | .95 | .65 | .70 | .66 | .69 | .62 | .70 | .62 | .71 | .63 | .74 | .63 | .73 | .42 | .60 | .45 | .65 | .45 | .63 | .08 |
| t'node2vec | .89 | .93 | .56 | .75 | .77 | .85 | .90 | .87 | .49 | .58 | .39 | .54 | .34 | .44 | .33 | .36 | .33 | .37 | .33 | .37 | .16 | .20 | .17 | .21 | .17 | .20 | .06 |
| tmpDW | .59 | .83 | .48 | .89 | .52 | .72 | .52 | .31 | .66 | .70 | .64 | .69 | .44 | .69 | .57 | .71 | .57 | .73 | .48 | .66 | .34 | .58 | .37 | .62 | .27 | .51 | .17 |
| tmpN2V | .99 | .99 | .99 | .99 | .99 | .99 | .99 | .99 | .64 | .66 | .66 | .66 | .45 | .70 | .63 | .70 | .66 | .73 | .55 | .71 | .46 | .57 | .50 | .62 | .32 | .60 | .08 |
| O-A | .07 | .12 | .06 | -.04 | .04 | .05 | .14 | .07 | .08 | .09 | .11 | .12 | .15 | | | | | | | | | | | | | | |

Next, we take interest in the relationship between controlled forced misalignment and prediction accuracy. For the sake of simplicity, we consider only one dataset and embedding method. For each pair of consecutive timesteps of the

sp-high-school dataset, we align the corresponding *node2vec* embeddings. Then, in a similar fashion to the experiments in Section 3.4, we misalign them by applying different rotations and translations with regards to rotation and translation factors. Finally, we predict the labels of the second embeddings set using different models trained on the first set. Figure 17 depicts the obtained results. Overall, we observe that the larger are the rotations/translations applied, the worse is getting the classification accuracy. Also, we remark that the best prediction accuracy is obtained when the rotation and translation are small. This means that, beyond helping improving performances, the way we align is ideal regarding the classification task we address for this dataset. Another interesting point lies in the similarity between the different classifier results. Thus, this results do not depend on what classification model is used. It also justifies our prior choice of employing Support Vector Machines.

We also look at the relationship between embedding stability and prediction accuracy in the inference task. By definition, stability error is directly related to the changes between nodes' consecutive embeddings that are not attributable to misalignment. This means that, when the stability error is high, the boundaries discriminating between the different labels of nodes would also change significantly. Consequently, the score of the prediction of node labels knowing their embeddings at a timestep using a model trained on the previous timestep's data should be correlated to the stability error. Figure 18 shows the results of this experiment on datasets that have more than 50 nodes. Overall, we can see that the higher the stability error, the less accurate the prediction task is, thus confirming our previous intuition. However, we also observe that the points are somehow scattered around the linear regression line. One explanation lies in the differences between the datasets and the embedding algorithms. For example, there are cases where the classifier is performing relatively well even though the stability error is relatively high. This is possibly due to the fact that the different clusters representing the labels are separate enough to compensate for the moves of node embeddings. On the other hand, when the clusters are too close or even overlapping, it may cause misclassifications although stability error is low.

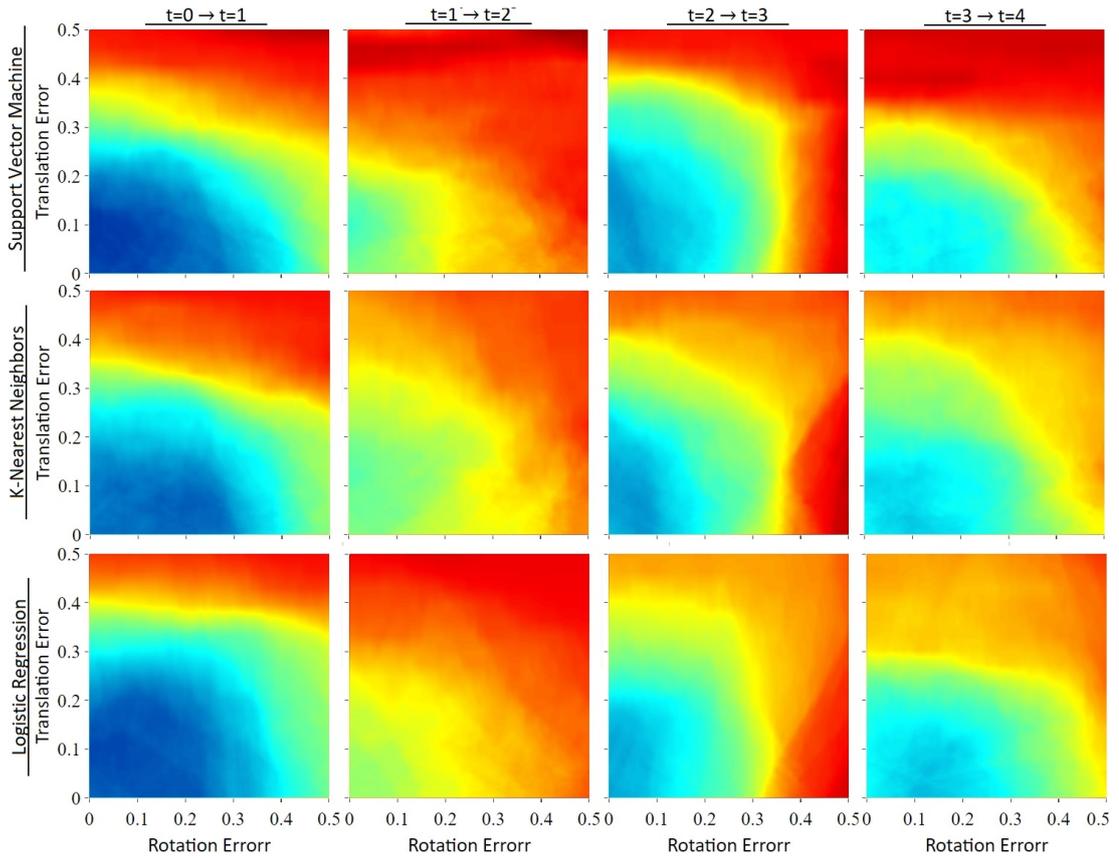


Figure 17. Effect of synthetic misalignment on prediction accuracy

Each row presents a different classifier. Each column presents different consecutive time pairs for the temporal inference task. On the *sp-high-school* dataset, the classifier is trained at time t using the *node2vec* embeddings as input features and students' class as the target variable. Accuracy is calculated on the predictions made at $t + 1$ using $t + 1$ embeddings.

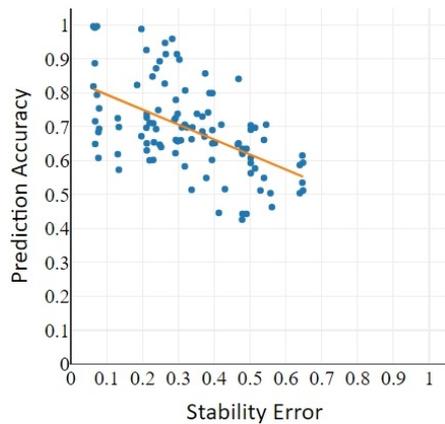


Figure 18. Prediction accuracy versus stability error of the embeddings

For all the considered datasets and embedding methods, the stability measure and the prediction accuracy are reported for each pair of consecutive timesteps' embeddings. Globally, low stability error coincides with good prediction scores.

3.6 Conclusion

In this chapter, we proposed a method to force alignment as well as a measuring procedure able to describe the alignment and stability of dynamic network embeddings.¹⁶ In order to demonstrate the validity of the developed measures, we conducted different experiments using synthetic data, proving that each one of the measures captures the type of misalignment it is designed for. Then we confronted our procedure to real-world datasets, using several static and dynamic embedding methods. Overall, we confirm that, as expected, dynamic methods produce more aligned embeddings comparing to static ones. Also, the performed experiments show that aligning embeddings generally improves prediction performances for the inference tasks that are sensitive to embeddings temporal evolution. In addition, the method we employ to align embeddings seems to be generally ideal regarding the considered transformations and inference task. Finally, we demonstrated that the stability measure we propose is correlated to the accuracy of the considered inference task.

In this chapter, we focused on temporal network embeddings. However, our work can be easily adapted to other types of data. It could be any sequence of matrices of the same shape, each one representing a set of vectors in an d -dimensional space, such as dynamic word embeddings or sequence of images. On a global note, as shown in this chapter, aligning these data sequences can help improving their quality for different purposes. Possible future research directions are incorporating alignment techniques within dynamic embedding methods and ensuring alignment in their outputs, adapting the proposed measures to different distance metrics and geometrical spaces, and employing the proposed stability measure to investigate dynamic characters of systems that are represented in latent spaces.

¹⁶The code and examples are publicly available at <https://github.com/furkangursoy/embassy>.

3.7 Software package

The code to measure alignment and stability errors as well as to align given embeddings, as proposed and discussed in in this chapter, is published as an open-source Python package *embassy*. The package can be installed using the package manager *pip* as follows.

```
pip install embassy
```

In the *embassy* package, all functionality is provided through a single function named *align_and_measure()*. The function accepts two embedding matrices to be aligned as its parameters and returns a single list containing translation error, rotation error, scale error, stability error, and the aligned versions the embedding matrices. The following piece of code in Python demonstrates its use.

```
from embassy import align_and_measure
import numpy as np

X = np.array([[1.8, 2.0], [2.3, 2.5], [ 1.8, 4.2], [4.1, 3.1]])
Y = np.array([[0.0, 1.2], [0.3, 1.6], [-0.4, 3.5 ], [1.6, 2.5]])

output = align_and_measure(X, Y)

print("\n Translation Error :", output['translation_error'],
      "\n Rotation Error   :", output['rotation_error'],
      "\n Scale Error       :", output['scale_error'],
      "\n Stability Error    :", output['stability_error'],
      "\n",
      "\n X_aligned:\n",      output['emb1'],
      "\n",
      "\n Y_aligned:\n",      output['emb2'])
```

CHAPTER 4
INVESTIGATING INTERNAL MIGRATION
WITH NETWORK ANALYSIS AND REPRESENTATION LEARNING:
AN APPLICATION TO TURKEY

4.1 Introduction

Human migration takes place for a large variety of reasons including those that are economic, social, educational, cultural, religious, safety-related, and so on. Most generally, migration can be defined as a relatively permanent move from one migration-defining area to another (Kok, 1997). A distinction between two types of migration can be made based on whether they are international or between settlements within the same country. McAuliffe and Khadria (2019) estimates that 272 million people are international migrants in 2020, which corresponds to 3.5% of the world population with an accelerating increase from 2.3% in 1970 and from 2.8% in 2000. Still, the vast majority of people live in the countries they were born. On the other hand, internal migrants are estimated at 740 million people in 2009 (Klugman, 2009), corresponding to more than 10% of the world population.

The research literature on migration is concerned with questions on *who*, *why* (e.g., the determinants), *where*, and *when* of migration and its consequences (Greenwood, 1997). Another distinction in the literature can be made based on whether the migration is studied at a personal (i.e., microscale) or aggregate level (i.e., macroscale) (Danchev & Porter, 2020). The migration inflows and outflows affect the (re)distribution of population and its characteristics over the geography. As such distributions are closely related to social and economic outcomes, investigating the structure and dynamics of migration are of (or should be of) utmost importance to demographic and developmental policymakers.

Lately, the application of network analysis tools and concepts to migration data has become more popular as they can viably analyze *where* of migration with an overview of the whole system while accounting for the interdependencies in its

structure. Methods from the representation learning literature have not been employed in the migration literature so far.

In this chapter, we focus on internal migration at an aggregate level focusing mainly on the *where* and *when* question with a temporal network analysis approach and a representation learning-based latent space model. We extend the frequently employed network analysis methods by employing a filtering method that extracts not only positive migration ties but the negative ties (i.e., a signed network) and by employing methods from the machine learning literature to learn and analyze the latent low dimensional representation of the migration system. We provide our analysis on the dynamic internal migration flows in Turkey from 2008 to 2020 although the same methods apply to all such internal migration systems. To the best of our knowledge, this study is the first application of signed networks and representation learning methods to migration networks, and the first extensive network analysis study for the internal migration system in Turkey.

In the seminal works of Ravenstein (1885, 1889) and Lee (1966), several laws are proposed as facts and governing rules of migration. Below, we extract and discuss some of those laws that can be addressed with the data that we have.

- Migration tends to show an increasing trend in terms of volume and rate unless stringent counter-policies are implemented. In contrast, the neoclassical economic view suggests that migration tends to neutralize the economic pull and push factors; and as such differences diminish, the migration tends to decrease over time (Massey et al., 1993).
- Migration takes place in well-defined routes from specific source locations to specific destinations. Popular examples include large numbers of Southern Italian immigrants in South Philadelphia (Dickinson, 1966), Vietnamese immigrants in Prague (Drbohlav & Džúrová, 2007), Japanese immigrants in Brazil (Smith, 1979), and the large number of immigrants from particular small towns of Turkey such as from Kulu, Emirdağ, and Ünlüpinar to Sweden, Belgium, and London, respectively (Dağdelen & Karakılıç, 2011; Guveli et al.,

2016; Reniers, 1999). Such highly specific routes gain strength due to the knowledge and support reaching back from the destination due to personal social networks and other factors alike. In addition to the specific routes, this law also states that some specific destinations are highly attractive, therefore, attract much larger migration streams, which indicates a power law in migration flow volumes.

- For every major migration flow, another stream develops in the opposite direction. Such counter-streams exist due to the contact created between the origin and destination as well as return migration. Rogers (1983) enumerate eight types of reasons for return migration based on events related to origin or destination, at the aggregate or individual level, and related or unrelated to the original motive for migration.
- Most migrants only move smaller distances while those who migrate longer distances usually do so towards centers of large economic activity.

The remainder of this chapter is structured as follows. In Section 4.2, we provide an overview of migration research with a specific focus on network analysis approaches at an aggregate level for internal migration systems. In Section 4.3, we describe our dataset, introduce the mathematical notation we employ, and explain network analysis, information filtering, representation learning, and community detection methods that we utilize in this study. In Section 4.4, we present the results of our analysis in line with but not limited to the above-mentioned laws and discuss our findings. We provide brief conclusions and final remarks in Section 4.5.

4.2 Related work

The earliest study on internal migration that we have identified is the seminal study of Ravenstein (1885) that investigates the migration patterns within the United Kingdom and provides extensive hand-drawn cartographies detailing the migration patterns.

Early works of Slater (1974, 1975a, 1975b, 1976a, 1976b) may be considered the first

studies in the migration literature that employs an explicit networks perspective. However, a wide range of network analysis tools are relatively recent and have not found much use in the migration literature. Bilecen, Gamper, and Lubbers (2018) remark that although network characteristics of migration are widely acknowledged, network analysis methods have not received the deserved attention. Caudillo-Cos and Tapia-McClung (2014) also highlight the lack of a network analysis approach in migration research apart from few studies.

Although internal migration data is available at a more fine-grained and reliable level, it has not attracted as much attention as international migration (Carvalho & Charles-Edwards, 2020), particularly in geographies outside the United States (Pitoski, Lampoltshammer, & Parycek, 2021). An early work to employ network analysis perspective studies the migration patterns at the state level in the US (Maier & Vyborny, 2005). They highlight the problem of having very dense networks, and accordingly, the need for filtering methods. The study identifies the hierarchical community structure and finds that the communities are geographically bounded. Slater (2008) investigates the hub and cluster structure in the US at the county level for two time periods. The study also identifies the hierarchical community structure and finds that cosmopolitan areas serve as hubs, have longer-range links, and tend to connect to other geographically bounded clusters at upper levels rather than forming immediate close-knit clusters. Goldade, Charyyev, and Gunes (2018) compare the internal migration in the US at the county level during the housing boom (2004-2007) and the following recession (2008-2011) in terms of their network structure, geographical bounds and political affiliation of communities, and its dynamic behavior. The study finds that high degree nodes do not have high clustering whereas low degree nodes generally show high clustering, migration dynamics are largely stable over time, and the communities are geographically bounded. Charyyev and Gunes (2019) also provide a similar analysis from 2000 to 2015 at the county level with comparisons of periods of economic prosperity and recession. They investigate the dynamic behavior by monitoring the existence of links

over time and find considerable instability. However, it might be due to the very granular view of the network as they also found that the migration network maintained its characteristics over the investigated period, in contrast to their instability result.

Outside the US, the studies on internal migration with a network analysis approach are usually more recent. Caudillo-Cos and Tapia-McClung (2014) compare the migration patterns of the highly qualified and general population between 2005 and 2010 in Mexico at the level of metropolitan areas and provide an online visualization tool for further analyses. Pitoski et al. (2021) study the internal migration in Croatia and find that both degrees and strengths follow a power-law distribution. They also find that migration flows show a reciprocity behavior and geographically bounded communities arise. Y. W. Chen, Ni, and Ospina-Forero (2021) provide a visual network analysis of internal migration within England and Wales in 2019 and find that people tend to move within groups of geographically close regions. They further indicate that such visualizations can support informed planning and decision-making on policies regarding public services, economic and social development, and alike. Carvalho and Charles-Edwards (2020) study dynamics of migration networks from 1980 to 2010 in Brazil and find high reciprocity and hierarchical network topology. In line with other studies, they also find that long-range links generally belong to larger cities, new migration flows tend to be between pairs with previously established migration flows.

The earliest study that we identified which employs a network analysis approach to the internal migration in Turkey is of Slater (1975b). The study identifies migration flows between then 67 cities by looking at the discrepancies in the cities people were born and live. The study finds the following clusters of cities: Amasya, Tokat, Çorum, Sivas; Kayseri, Kırşehir, Nevşehir, Yozgat; Kocaeli, Sakarya, Bolu; Bilecik, Bursa, Eskişehir; Aydın, Muğla, Denizli; Kütahya, Uşak; Antalya, Burdur, Afyonkarahisar, Isparta; Edirne, Kırklareli, Tekirdağ; Hakkari, Van; Ağrı, Bitlis, Kars, Muş; Diyarbakır, Mardin, Siirt; Bingöl, Tunceli, Elazığ; Artvin, Rize; Adana, Adıyaman, Gaziantep, Hatay, Mersin, Malatya, Kahramanmaraş, Şanlıurfa; Giresun,

Kastamaonu, Ordu, Samsun, Sinop, Zonguldak, Çankırı; and Trabzon, Gümüşhane. It is noted that Istanbul and Ankara are not strongly tied to any community. Apart from this, to the best of our knowledge, the work of Yakar and Eteman (2017) is the only study that takes a network analysis approach at an aggregate level to study internal migration in Turkey. However, their data is based on the birth place, hence providing a snapshot rather than dynamic flows, as is the case with the study of Slater (1975b). They investigate statistics like degree and centrality measures under different levels of global thresholds for link weight. Their findings show that the links towards Istanbul are not affected much by distance whereas other links carry a more regional/local characteristic.

Overall, our review of the literature finds that the studies that employ a network analysis approach to investigate internal migration are limited, usually very recent, and particularly focused on the US. Findings indicate that geographically bounded communities form a hierarchical structure with anomalous links involving areas with large social and economic activity. Migration flows tend to be reciprocated and well-defined streams exist. The migration networks tend to be stable over time even during times of economic prosperity and recession. We also note that even very recent studies employ global thresholds than more appropriate network sparsification methods, methods from the machine learning literature have not yet found a use, and network analysis studies on internal migration at the aggregate level are not only rare in Turkey but also in other countries outside the US.

4.3 Material and methods

4.3.1 Dataset

The dataset in this study is sourced from publicly available migration data published by the Turkish Statistical Institute (2021). The migration figures are calculated based on change of official residences in the Adress Based Population Registration System (ABPRS) which typically indicates a relatively permanent relocation. Specifically, a migration is said to have occurred if the city of official residence is changed as

compared to the previous year. As information in ABPRS is utilized extensively in most tasks of the citizens including voting, utility services and billing, and so on, we consider the dataset to be sufficiently complete.

The data is available at the level of cities at the most granular level. There are 81 cities in Turkey with the minimum, median, mean, and maximum populations of respectively about eighty thousand, half a million, one million, and sixteen million out of a total population of 83.6 million as of 2020. At more general levels than the city level, there are geographic regions and divisions in addition to more specific groupings that are created exclusively for census/statistical purposes. Being a strictly unitary state, there are no administrative divisions above the city level. Below the city level, there are districts and neighborhoods/villages, for which migration flows are not available. We further believe that investigating the internal migration at the city level is appropriate as districts and neighborhoods are not well-separated in terms of economic and even social activities. As migration usually requires a minimum distance of movement, we believe it is an appropriate choice of migration-defining area.

ABPRS became effective in 2007, therefore, the migration data is available from 2008 to 2020. 81 cities remain the same throughout the investigated period. ABPRS covers all Turkish citizens and foreign nationals residing in the country. However, the internal migration data excludes foreign nationals as well as those who are relocated due to mandatory military service or serving their prison sentences, and citizens who spend less than six months in Turkey over a year.

We model the migration in-flows and out-flows between cities as a directed and weighted network. For each year, we generate a separate network. Thus, in each network, a link denotes a migration flow from an origin city (i.e., a source node) to a destination city (i.e., a target node) for that year. Its weight corresponds to the number of people that migrate from the source to the target node. Consequently, we obtain 13 networks and each network has 81 nodes and the migration links between those nodes.

4.3.2 Notation

Formally, we define a directed and weighted network with $G(V, E, W)$ where V is the set of nodes with its general members i, j, \dots and E is the set of directed edges. $|V| = n$ is the number of nodes and $|E| = m$ is the number of edges. W is the weighted adjacency matrix where W_{ij} is the number of migrants from i to j and $W_{ii} = 0$. In-degree and out-degree of a node correspond to the number of cities that the city receives migration from and sends migration to, respectively. In-strength and out-strength of a node correspond to the number of migrants who arrived at the city and the number of migrants who left the city, respectively, i.e., the total in- and out-flow of the city.

In general, we denote scalars with lowercase letters (e.g., x), matrices with uppercase letters (e.g., X), an element of a matrix by two subscripts respectively corresponding to its row and column indices (e.g., X_{ij}), and row vectors of a matrix with lowercase letters followed by a single subscript (e.g., x_i). $\|x_i\|$ denotes the norm of x_i .

4.3.3 Information filtering

Density is defined as the proportion of observed links to the possible links. Here, as we deal with a directed network without self-loops, the density can be calculated as shown in Equation 4.1. A density value of 1 indicates a complete network where all possible links are observed. Too dense networks are undesired for various interrelated reasons. First, some of these links may not be significant or important hence may distort our understanding of them, e.g., when only a few people migrate from a city to another, can we faithfully consider it as a link between the two cities? Second, many network analysis methods desire or require sparsity for accurate or effective analysis. Third, insightful and apprehensible visualizations cannot be devised when there are too many and possibly insignificant and uninformative links. To this end, the literature, including those that study migration networks, utilize various filtering methods.

$$\frac{m}{n(n-1)} \quad (4.1)$$

A filtering method is to establish a global threshold and eliminate links whose weights are below this threshold. This has two major problems. First, the choice of the threshold may be arbitrary and not justified by a theory or empirical evidence. Second, such global threshold methods are inappropriate for multiscale networks. The population of cities is distributed unevenly. Some cities are much more attractive and other cities are much more unattractive. As a result, few cities have much larger in- and/or out-flows and many cities have much smaller in- and out-flows in terms of absolute numbers. When a global threshold is applied, some informative links that connect, for instance, less populated cities may be eliminated while some uninformative links that connect well-populated cities may be retained. These limitations are discussed in depth in Chapter 2. Despite such limitations, the global threshold method is applied by many studies in the literature including those by Carvalho and Charles-Edwards (2020), Y. W. Chen et al. (2021), Pitoski et al. (2021), and Yakar and Eteman (2017).

The multiscale nature (i.e., the heterogeneous scale of node strengths and edge weights) of migration networks is a widely observed phenomenon. Many empirical networks in various domains exhibit a multiscale nature in their edge weights and node strengths. Hence, the information filtering methods to extract network backbones are widely studied in the literature. Accordingly, Charyyev and Gunes (2019), Maier and Vyborny (2005), and Slater (2008) use more sophisticated thresholding methods, e.g., bistochastic filter (Slater, 1976a) or disparity filter (Serrano et al., 2009a).

The null model of the disparity filter assumes that the normalized link weights of a node follow a uniform distribution. Hence, it is possible to extract statistically significant links by comparing the observed weights to the null model. The bistochastic filter also employs a normalization scheme but at the global level, and then identifies the links with the largest normalized values. The similarities and

differences between the two methods are discussed by Slater (2009) and Serrano, Boguñá, and Vespignani (2009b). A common point of these methods and most other methods in the literature, as discussed extensively in Chapter 2 and by Gürsoy and Badur (in press), is the fact that only the links that are significant in the positive direction are extracted. However, nodes may have negative links between them which indicate dissimilarity, animosity, and alike. In addition to the positive links, migration networks may have underlying negative links, e.g., a usually low number of migrants between two cities with relatively large numbers of inflows and outflows.

Extracting negative links from networks with positive edge weights requires an assumption that links with too small edge weights or lack of links may actually correspond to negative links. Such an assumption does not hold in many networks but is acceptable in intrinsically dense networks. Intrinsically dense networks refer to those networks where all nodes are aware of all other nodes and can interact with them without obvious natural limits as discussed in Chapter 2 and by Gürsoy and Badur (in press). As we are investigating a migration system that is empirically very dense (almost complete, in fact) and within a single country with free movement between its cities without limits like a different language, we may safely assume that it is an intrinsically dense network.

After establishing the intrinsically dense nature of the studied network, we use a recent method from Chapter 2 and by Gürsoy and Badur (in press) that extracts not only the positive links but also negative links. The method works by building a null model that controls for the in-strength and out-strength of nodes, estimates the expected values of link weights under this model, and identifies links whose weights are sufficiently distant from their null expectations as positive or negative links. The method can extract the signed backbone at a desired level of sparsity based on the statistical significance of the links (via its significance filter) as well as based on the intensity of the links (via its vigor filter). The intensity (vigor) of the links is a lift-based measure that ranges within $[-1, 1]$ and may also be used as signed edge weights.

Signed networks, to an extent, usually manifest a specific type of reciprocity where positive links are reciprocated with positive links, and negative links with negative links, as discussed in Chapter 2 and by Gürsoy and Badur (in press). Conflicting links where a positive link is reciprocated with negative links are usually rare. Reciprocity in directed networks can be defined as the proportion of directed links that are reciprocated. For signed networks, we can extend the definition with a further restriction that requires reciprocated links to be of the same sign, i.e., the ratio of reciprocated links that are of the same sign over all links. Similarly, one can calculate the ratio of conflicting links over all links.

Most empirical signed backbones also exhibit structural balance (SB) to some extent. Structural balance can be intuitively summarized with the phrases "friend of a friend is a friend" and "enemy of a friend is an enemy". This phenomenon can be thought of in relation to the local clustering of nodes in binary networks where a node tends to be linked with its neighbors' neighbors. Cartwright and Harary (1956) formalized structural balance for an undirected triple of nodes to be balanced if edge signs between them are +, +, + or +, -, - and unbalanced if the edge signs are +, +, - or -, -, -. Davis (1967) suggested that an enemy of a friend is not necessarily an enemy and proposed weak structural balance (WSB) where the only unbalanced triple is the one with edge signs +, +, -. Consequently, SB and WSB of a signed network are defined as the proportion of balanced triples over all triples.

4.3.4 Dynamic representation learning

The information that networks present are usually encoded in adjacency matrices with n rows and n columns. Such representations may suffer from high dimensionality. To overcome such problems, the goal of representation learning (RL) is to learn low dimensional latent representations (i.e., embedding vectors) of nodes such that most of the information in the observed network is preserved in lower dimensionality. To this end, the similarity in the embedding space should approximate the similarity in the observed network. Alternatively, the representations in the embedding space can

be seen as the underlying true data that generates the observed network. Formally, the problem is to estimate embedding matrix Z with shape $(n \times d)$ where n is the number of nodes, d is the embedding size, and z_i is the embedding vector of the node i such that $f(i, j) \sim g(z_i, z_j)$ where f is a similarity function in the observed network and g is a similarity function in the latent space.

Human migration, as modeled by networks, is well-suited for RL since the low-dimensional representation of nodes (e.g., cities in this case) enables further analysis, for instance, via methods for traditional clustering algorithms. Furthermore, it may be more tractable to monitor the dynamics of the system in lower dimensions.

We describe our representation learning method as follows. Given G , we first calculate the underlying signed backbone network $\hat{G} = (\hat{V}, \hat{W})$ by setting both significance filter and vigor filter to 0, i.e., retaining all possible edges. \hat{W} is the weight matrix whose elements are in the range $[-1, 1]$. We employ \hat{W} as the similarity in the observed network where \hat{W}_{ij} denotes the similarity of nodes i and j , i.e., $f(i, j) = \hat{W}_{ij}$. The network is directed thus $W_{ij} \neq W_{ji}$, however, our optimization process will find embeddings that are optimized for an appropriate trade-off between the two as will be obvious shortly.

To calculate similarity in the latent space, there are multiple alternatives such as euclidean distance, dot product, and cosine of the angle between two embedding vectors. Cosine similarity is defined in Equation 4.2. Since the range of similarity in the input data (i.e., edge weights in the signed backbone) is $[-1, 1]$, we choose cosine similarity whose natural range is also $[-1, 1]$, i.e., $g(z_i, z_j) = \text{cossim}(z_i, z_j)$. As the angle between two points in the latent space increases towards 180° , its value approaches -1 . As the angle decreases towards 0° , its value approaches 1. When they are orthogonal, its value is 0. When $d = 2$, the embedding vectors are located on a circle in the latent space. When $d = 3$, the embedding vectors are located on the surface of a sphere. Similarly, when $d > 3$, the embedding space is the surface of a higher-order sphere.

$$\text{cossim}(z_i, z_j) = \frac{z_i z_j}{\|z_i\| \cdot \|z_j\|} \quad (4.2)$$

After we establish the similarity functions in the observed space and latent space, we define the loss function \mathcal{L} as in Equation 4.3. We optimize Z using stochastic gradient descent (Robbins & Monro, 1951). We use stochastic gradient descent to find Z which minimizes \mathcal{L} . After initiating Z with random values, we iterate over each non-diagonal element of W in random order, calculate respective gradients for z_i and z_j , and update them using a learning rate $\alpha = 0.01$. Each iteration over all elements of W is called an epoch. For each year's network, this process is repeated for 100 epochs.

$$\mathcal{L} = \sum |cossim(i, j) - W_{ij}| \quad (4.3)$$

More specifically, we need to find the gradient vector in Equation 4.4 to update the embedding vector z_i for each iteration over all instances of W_{ji} or W_{ij} where $j \in V - \{i\}$. For simple presentation, we define the quantity y as in Equation 4.5. Then, the solution takes the form shown in Equation 4.6 and 4.7.

$$\frac{\partial}{\partial z_i} |W_{ij} - cossim(z_i, z_j)| \quad (4.4)$$

$$y = W_{ij} - cossim(z_i, z_j) \quad (4.5)$$

$$\frac{\partial}{\partial z_i} |y| = \frac{y}{|y|} \frac{\partial y}{\partial z_i} \quad (4.6)$$

$$\frac{\partial y}{\partial z_i} = -\frac{\partial}{\partial z_i} cossim(z_i, z_j) = -\frac{z_j}{\|z_i\| \cdot \|z_j\|} + \frac{z_i}{\|z_i\|^2} cossim(i, j) \quad (4.7)$$

We update z_i as shown in Equation 4.8 and after each epoch, we ensure that the norm of z_i is equal to 1 via the update shown in Equation 4.9. Note that such normalization does not change the value of $cossim(z_i, z_j)$ function.

$$z_i \leftarrow z_i - \alpha \frac{\partial}{\partial z_i} |W_{ij} - cossim(z_i, z_j)| \quad (4.8)$$

$$z_i \leftarrow \frac{z_i}{\|z_i\|} \quad (4.9)$$

As implied by the loss function \mathcal{L} , the information in the latent space is stored in the relative positions of embedding vectors rather than their absolute positions. The optimization may converge to different but equally plausible solutions due to the random initialization of Z or inherent stochasticity of the optimization procedure. For instance, any rotation and/or reflection of the found solution provides exactly the same information. It naturally follows that we cannot directly compare the embedding vectors between different time steps. There might be optimal rotations and/or reflection between different years' embeddings that preserve the cosine similarity while transforming the space. This phenomenon is known as procrustes problem (Schönemann, 1966) and discussed in-depth in Chapter 3 and by Gürsoy, Haddad, and Bothorel (2021). For each consecutive year, we find the optimal rotation matrix R (as in Equation 4.10) and update the embedding matrix as $Z^{t+1} \leftarrow Z^{t+1}R$ where Z^t is the embedding matrix at year t . After this procedure, all Z matrices are considered to be *aligned* across all time steps, i.e., they are brought into the same latent space and are comparable.

$$R = \operatorname{argmin}_{\Omega} \|Z^{t+1}\Omega - Z^t\| \quad (4.10)$$

Stability reflects the changes between embeddings in two time steps that are not attributable to the above-described misalignment issues. Therefore, stability error reflects the extent of structural change in dynamic embeddings, as discussed in Chapter 3 and by Gürsoy et al. (2021). Its range is $[0, 1]$ where 0 indicates that the preserved information is exactly the same between two embeddings and 1 indicates the worst case. Therefore, lower values indicate a largely stable system.

4.3.5 Community detection

We identify community structures in the migration networks using two different clustering approaches: density-based and hierarchical. For the former, we employ

Density Based Spatial Clustering of Applications with Noise (DBSCAN) (Ester, Kriegel, Sander, & Xu, 1996) which works by identifying dense regions in the space by checking for the existence of a desired number of minimum objects (*min_samples*) within a desired distance (*eps*) from a core point, which indicates a dense region. If they are found, it expands the cluster and repeats the same process until it cannot expand the dense region. Each dense region becomes a cluster, hence, the number of clusters is not known a priori. Points outside the dense regions do not belong to any cluster. For the latter, we employ agglomerative hierarchical clustering with complete linkage (Rokach & Maimon, 2005). It works by treating each point as a cluster of its own and iteratively merging closest clusters until it ends up at a single cluster. The distance between two clusters is the distance between their furthest-apart members. The whole process reveals a full hierarchical structure. A desired number of clusters can be extracted from this hierarchy at corresponding cut levels. Both methods are previously employed for embeddings that have angular separation (Bruno et al., 2019) such as ours since our loss function \mathcal{L} is based on *cosim*() function which provides angular separations.

Both methods require the distance between two points but do not require their individual coordinates. We calculate the distance between two cities as $1 - \text{cosim}(z_i, z_j)$ since cosine similarity is the function that gives the similarity in the latent space. In this way, we obtain distances between all city pairs within the range $[0, 2]$. Using the calculated distance values as input, we perform our clustering analysis. For DBSCAN, we use hyperparameter values 0.14 for *eps* and 3 for *min_samples* as initial experiments showed that this configuration (and also other configurations with nearby values) provides plausible clustering results.

4.4 Results and findings

The number of migrants over the years is presented in Figure 19. Each year, 2.2 million to 3 million migrate internally within Turkey, a country whose population increased from 71.5 million in 2008 to 83.6 million in 2020 (Turkish Statistical

Institute, 2021). In terms of the total number of migrants, we observe a slow but often steady increase. However, this increasing trend is associated with the increasing population as the migration rate is mostly steady. Each year, 3% to 3.5% of the population migrate. The sudden decrease after 2019 is likely due to the COVID-19 pandemic as it limited mobility, e.g., almost all universities switched to distant learning, therefore many freshmen students have not relocated. Overall, while the results presented in the figure show an increase in volume, the migration rate does not show a notable increase. Therefore, these results support neither the law stating that migration rate increases nor the neoclassical argument suggesting that migration causes convergence of economic differences and hence the migration itself diminishes over time.

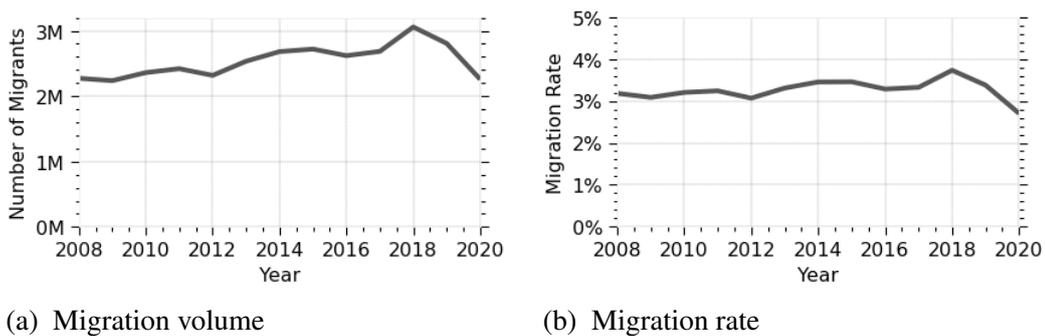


Figure 19. Internal migration trend in Turkey

Figure 20 shows the distribution of edge weights, node in-strengths, and node out-strengths of the original network. The solid lines denote the mean values over 13 years and the transparently colored regions around them indicate the area within the 2.58 standard deviation. Figure 20a reveals that edge weights follow a power law in the tail where a much smaller number of links have much larger weights. Figure 20b and 20c reveal that node strengths are also very heterogeneous where a smaller number of cities receive and send larger migration. We also note that the maximum values on the x-axes originally reach to 24884, 328632, and 381654 respectively in Figure 20a, 20b, and 20c but are cut at lower values for visualization purposes. In other words, the tails of the distributions are actually longer than they appear in the

figures. These findings are also in line with the observation that most empirical networks exhibit a multiscale nature in their edge weights and node strengths. Moreover, it confirms the migration law that suggests sending and receiving locations in migration systems tend to follow a power law in their migration flow sizes.

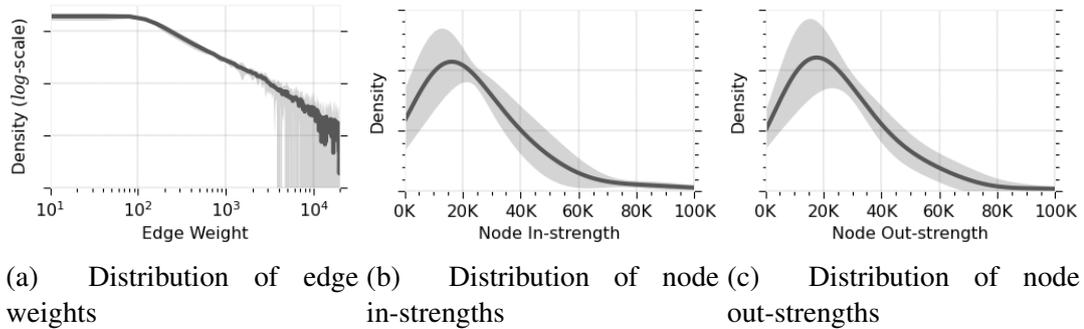


Figure 20. Migration flow volume characteristics

For internal migration networks of Turkey, density values range from 0.999 to 1 over the years, which indicates that there are migration flows between all pairs of cities in both directions barring rare exceptions. Such networks are not suitable for most network analysis methods as many methods desire or require sparsity. Hence, the need for filtering out the insignificant links is clear. Following the need for information filtering methods to extract the sparse backbone of the network while respecting the multiscale nature of the network, we extract the signed backbone of networks using significance filter and vigor filter which are shown to extract statistically sound and meaningful backbones of intrinsically dense networks

We extract the signed backbones via the significance filter retaining 5% to 100% of all possible links and evaluate the structure of backbones in terms of reciprocity and structural balance. Figure 21a visualizes the proportion of positive-positive and negative-negative reciprocated links as well as the proportion of conflicting links over all links. It shows that most signed links are reciprocated with links of the same sign. Conflicting links are rare even when the backbone size is considerably large. These results indicate that the link signs in our migration networks are consistent in terms of reciprocity. This is further in coherence with the law on counter-streams which states that for every major migration stream, there is a

counter-stream. In a sense, our backbone extraction method specifically retains the major streams only, hence the analysis on the extracted backbone is particularly appropriate for this test. Going beyond the explicit statement of this law, it also shows that for major negative streams (i.e., negative links where migrants from a particular city avoid moving to a particular city), there are negative streams in the opposite direction. Also, the lack of conflicting links shows that for every major stream, no negative streams exist in the opposite direction; if otherwise, it would invalidate the law.

Next, we investigate the structural balance and weak structural balance of the extracted backbones. As these measures are defined for undirected networks, we convert our directed migration networks to undirected versions by assuming a positive link when all directed links between a pair are positive, a negative when all directed links between a pair are negative; and assuming no link if there is one positive and one negative directed link between a pair. When there is only a single link between a pair, we assume an undirected link of the same sign. Figure 21b reports the SB and WSB results. It demonstrates great levels of SB when the backbone is sufficiently sparse and great levels of WSB even if the network very dense. These results indicate that the nodes in our networks have structural balance. Such structural balance may arise due to positive relationships among geographically close cities as well as their common destinations in longer distances, e.g., three neighboring cities send migration to each other and they also send migration to a specific industrial city in long distance. Visualizations that are provided later in this chapter provide support for this statement.

Overall, Figure 21 shows that extracted backbones have characteristics usually observed in other empirical signed networks; the findings are intuitive and are coherent with the relevant findings in the migration network literature. This conclusion also justifies our choice in selecting an appropriate information filtering (i.e., backbone extraction) method.

Figure 22 presents the map of Turkey with city borders as well as the distribution of GDP per capita and net migration rate in 2019. Observing this

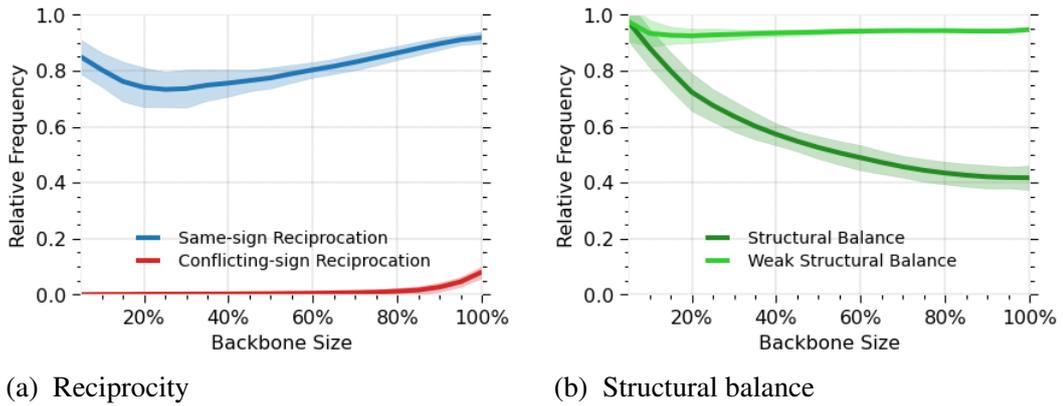


Figure 21. Characteristics of migration backbones

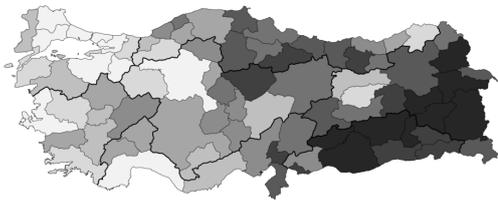
information will also help readers that are not familiar with socioeconomic differences in Turkey by providing a context. In Figure 22b, the increasing values are shown with gray colors transitioning from dark to light with each color denoting an equal-size group. In a similar fashion, in Figure 22a, increasing net migration rates for cities (i.e., in-migrants minus out-migrants divided by the city population) are denoted with colors transitioning from dark gray to light gray. Overall, the figure demonstrates that GDP per capita and net migration rate increase in general as we move towards the west. This supports the view that migration is tied to economical differences between locations.

For the rest of the analysis, before we proceed with the representation learning methods, we extract the backbones for each year in the following way. We first retain 7.5% of links that are most significant and further eliminate links that have absolute vigor values less than 0.33. Vigor threshold is selected as such so that the weight of a positive link is at least more than double the random expectation and the weight of a negative link is at most the half of random expectation. Therefore, the backbones contain the significant and intense links.

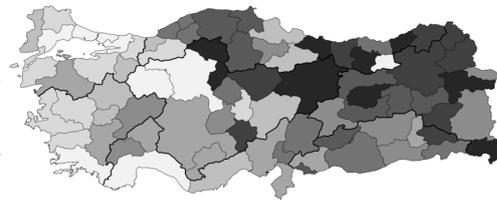
Figure 23a and 24a visualizes the extracted backbones in a spatial geographic layout, respectively for 2008 and 2020. The node sizes are proportional to their population in respective years. In both figures, most positive links are local within close spatial proximity. Most of these local links are reciprocated. There are no local negative links. The non-local links that connect distant locations are relatively few



(a) Map of cities in Turkey



(b) GDP per capita

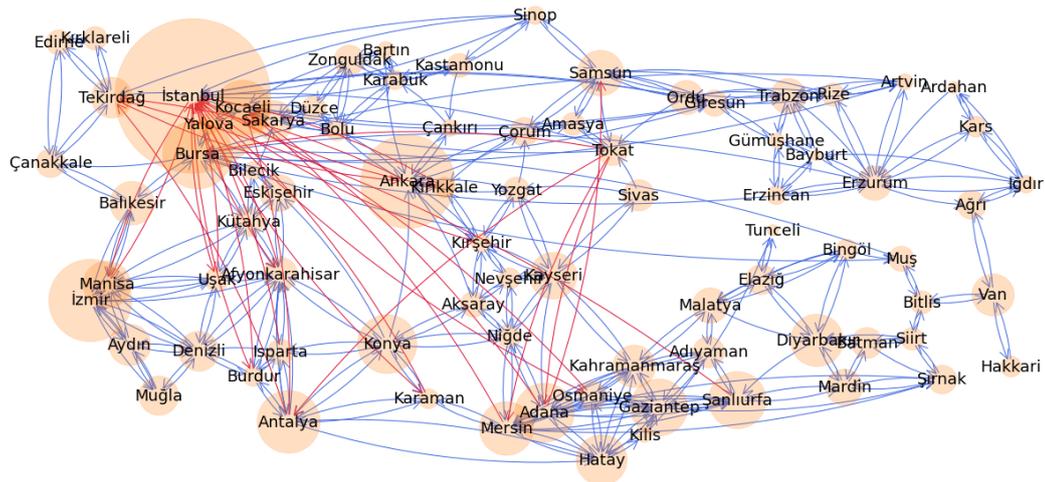


(c) Net migration rate

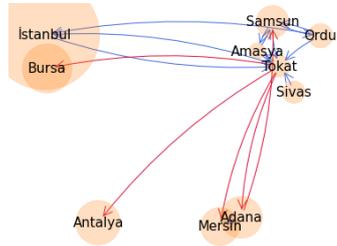
Figure 22. Maps of Turkey

and usually towards cities with great economic and social activity such as Istanbul. Most of the less local positive links are in the east-west direction. This may be explained by the fact that historically large and economically more active cities are concentrated in the western parts. Also, the mountain ranges in northern and southern regions in Turkey lay in the east-west direction which means that historically, most communication and transportation are developed in a horizontal direction. In addition to the positive examples, negative examples also support this finding as the negative links are usually and loosely in the north-south direction. These overall results support some of the previously described rules, e.g., migration takes place in well-defined routes, most migration happens at a local level, and long-range migrations usually involve centers of large economic activity.

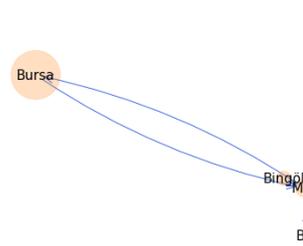
We also investigate ego networks for selected cities for a more particular analysis. For 2020, we present the ego networks for Tokat, Muş, and Edirne respectively in Figure 23b, 23c, and 23d by visualizing both ego-alter links as well as alter-alter links. Tokat has positive links with most of its geographically close



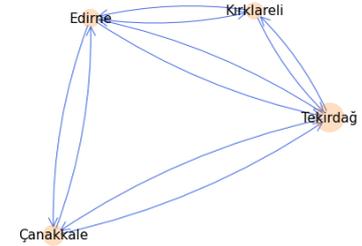
(a) Overall network



(b) Tokat ego network

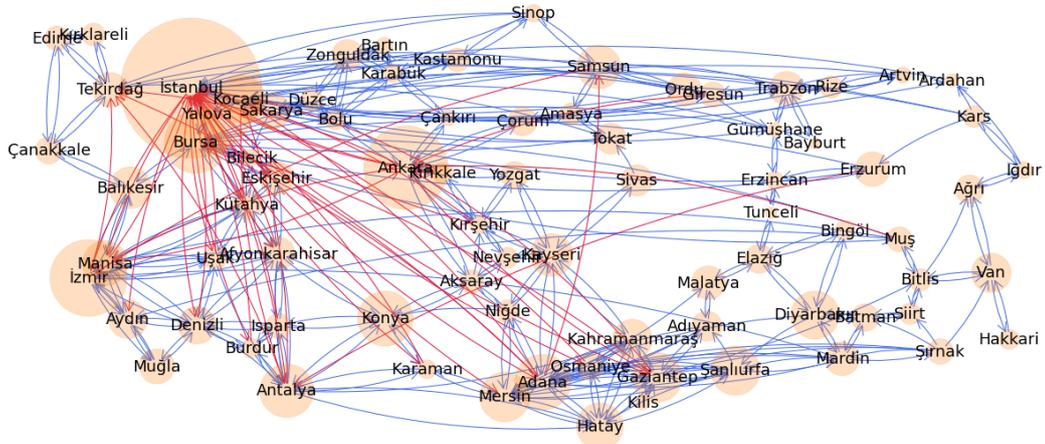


(c) Muş ego network

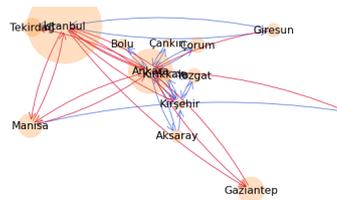


(d) Edirne ego network

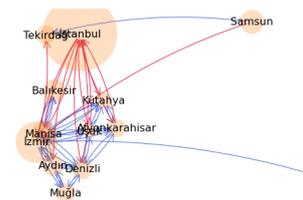
Figure 23. Spatial network of migration backbone in 2020



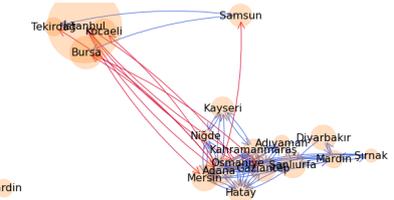
(a) Overall network



(b) Ankara ego network



(c) İzmir ego network



(d) Adana ego network

Figure 24. Spatial network of migration backbone in 2008

neighbors in addition to a positive link with Istanbul, a distant city with a traditionally large socio-economical activity. In contrast, it has negative links to few southern cities that otherwise attract relatively large emigration. Muş shows a similar behavior with positive connections to its geographically closer neighbors in addition to a positive link with a distant but economically more active larger city, Bursa. In contrast to these two cities with low economic activity, Edirne has a relatively good economic situation. Accordingly, Edirne does not have long-range links and all of its links are positive and connected to its immediate geographical neighbors. The specific analyses on the three cities and the general picture provided earlier indicate that certain smaller or economically less active cities have migration outflows towards different and specific large and economically active cities rather than homogeneous outflows.

Such preference of Tokat towards Istanbul and Muş towards Bursa may be explained by the previous strong social ties between the respective pairs. A proxy for such social ties is the number of people who live in the city with greater economic activity but who were born in the city with lesser economic activity. In 2020, based on the data from Turkish Statistical Institute (2021), among those who were born in Tokat but live elsewhere, 52% live in Istanbul followed by Ankara with only 8%. The same figure for Muş and Bursa is 16% with Bursa being the second most popular destination after Istanbul. Among those who live in Istanbul but were born in other cities of Turkey, 4% is born in Tokat with Tokat being the second most popular origin. The same figure for Bursa and Muş is 5% with Muş being the 4th most popular origin. Therefore, the claim about the effects of existing social ties is supported. Furthermore, in both city pairs, the links are reciprocated which supports the migration law that counter-streams develop for every major stream.

In Figure 24b, 24c, and 24d, for 2008, we visualize the ego networks of second, third, and fifth-most populous cities in Turkey: Ankara, İzmir, and Adana. We observe that Ankara has its positive links exclusively with relatively nearby cities. As the capital city of Turkey, it seems to attract migration relatively homogeneously, a deviation from the behavior of other large cities which is worthy of further

investigation. It also has negative links with specific smaller cities, such as Muş and Giresun, which have well-defined streams to other large cities. Other negative links are with relatively larger-economy cities such as Istanbul, Manisa, and Gaziantep. This indicates that the number of migrants between populous and economically well cities is lower than randomly expected, which further hints that the migrants seek to increase their economic utility. The ego network of İzmir indicates mostly local positive links with very high mobility among the cities in the region. The only long-range positive link is from Mardin. Following the social ties hypothesis, we find that among those who were born in Mardin but live elsewhere, 16% live in İzmir with İzmir being the second most popular destination after Istanbul. Among those who live in İzmir but were born in another city, Mardin is the third most popular origin with 4%. Ego network for Adana shows that it plays the role of a regional hub, as is the case with İzmir. It has negative links to the northwestern region, another and larger hub which attracts many long-range links from northern parts of the country.

Due to space limitations, we cannot visually investigate overall networks or ego networks for all years or all cities. Our unreported investigations and the stability results presented earlier indicate that major findings stay the same over the years. Interested readers may conduct their own analyses easily for other years following the same techniques.

Following the observation that the overall migration backbone has not notably changed from 2008 to 2020, we also look at the persistence of individual links over time. Over 13 time steps, we look at how many times each directed link is repeated. 606 links appear at least once. For those links appearing at least once, Figure 25a shows that most of the links persist in all years with a much smaller but notable portion of links appearing only once or twice. We also compare the number of links during two equal periods, 2008-2013 and 2014-2019. We exclude 2020 since it shows a distinct behavior, most likely due to the COVID-19 pandemic. 597 links appear at least once in at least one period. Then, we calculate the difference between frequencies observed in two time periods for those 597 links. Figure 25b shows that

most of the links have equal frequencies in both years where a very small number of cities appear frequently in one period and appear only a few times (or does not appear at all) in the other period.

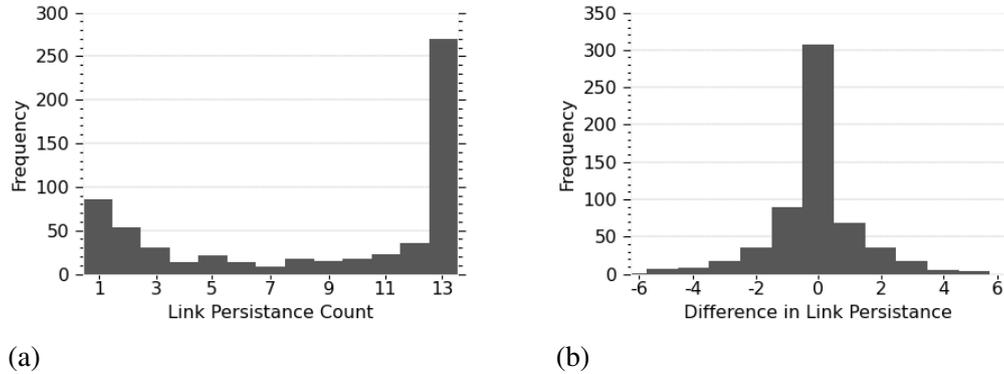


Figure 25. Link persistence in migration backbones

Next, we move to our analysis of the underlying latent low-dimensional space. We generate embeddings for each d in $\{2, 3, 4, \dots, 12\}$. Figure 26 shows the average embedding loss over all links, i.e., all nondiagonal elements of W . The solid line denotes the mean value over 13 years and the transparently colored region indicates the area within its 2.58 standard deviation. As d increases, the marginal gain in preserved information quickly disappears. We choose $d = 8$ for the rest of the analysis.

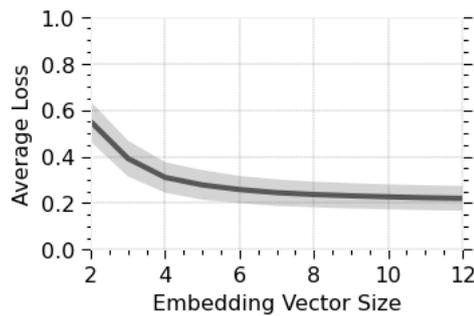


Figure 26. Embedding loss

To test our earlier findings on the dynamics of the migration system, i.e., the general stability observed in the overall structure of the network and most of the links, we investigate the overall changes in the embedding space over time using stability

error. Figure 27 presents the stability error over time by comparing each Z^t with Z^{t-1} , Z^{t-2} , Z^{t-3} , and Z^{t-4} . The results demonstrate that the stability error is generally low and does not fluctuate considerably over time - even when compared to the system from four years earlier. It can be concluded that the structure of the migration network is largely stable over the investigated period and its overall dynamics seem to be modest without any critical transition points. This finding is also in line with the migration law stating that migration streams are well-defined.

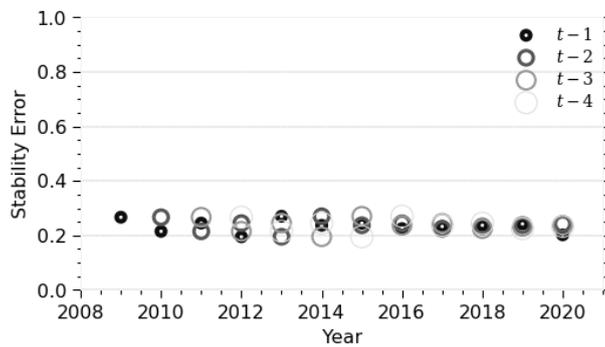


Figure 27. Stability error over time

Figure 28 visualizes the clustering results in 2008, 2014, and 2020 using different colors for each cluster. Cities that do not belong to any cluster are colored in gray with a horizontal grid pattern. In line with previous results, it shows that geographical proximity plays an important role. However, such geographical proximity is not strictly bounded by the seven geographical regions whose borders are separated with thicker lines. Density-based clustering results demonstrate that dense clusters are in line with geographical proximity. However, some cities do not belong to any cluster, which indicates that the positions of these cities are not bounded by their geographic positions. The most prominent example is Istanbul that has many long-range links to different geographies. Another example is Ankara, the capital city. We also note that the found communities and the particular cases of Istanbul and Ankara were recognized in the findings of Slater (1975b), which further suggests that the migration dynamics have not significantly changed in the last several decades. We also observe that cluster boundaries are fluid with some cities in their borders

belonging to different clusters across time. This may be explained not only by their dynamically changing behavior but also by the fact that they are not strongly attached to their corresponding clusters. On the other hand, particularly the cluster in the west denoted with blue color, have its most members same across time. We should also note that it is possible to obtain different but congruent communities, i.e., more or less granular communities, using different hyperparameters for the clustering method.



Figure 28. Density-based clusters

We also investigate hierarchical clusters in Figure 29 for 2008, 2014, and 2020. As we employ the complete-linkage method which requires most distant members to be sufficiently similar, we do not produce the full dendrogram but only present the smaller communities at the bottom of the hierarchy. The figures, in all years, demonstrate that geographical proximity plays a major role in defining communities. However, Istanbul, which does not belong to any cluster in the density-based result and has many long-range links, is often clustered together with cities that it does not share borders with. Overall, the results from both clustering methods tend to agree, which shows that the findings from the clustering methods are robust.

4.5 Conclusion

In this chapter, we have investigated the internal migration patterns of Turkey from 2008 to 2020 via signed network analysis, ego network analysis, representation learning, temporal stability analysis, community detection, and network visualization. We find that most migration links are geographically bounded with exceptional long-range links involving centers of large economic activity. Major migration flows are reciprocated with migration flows in the opposite direction. The migration flows

take place on well-defined routes and the migration system is largely stable over the investigated period. Overall, our analysis and presented findings indicate that the internal migration system of Turkey agrees with the general migration laws and provide unique insights into the internal migration patterns of Turkey. Future research may extend the investigation to a longer time period if the data becomes available and may also extend the analysis to the questions on *who*, *why*, and consequences of migration via collecting further qualitative and quantitative data.

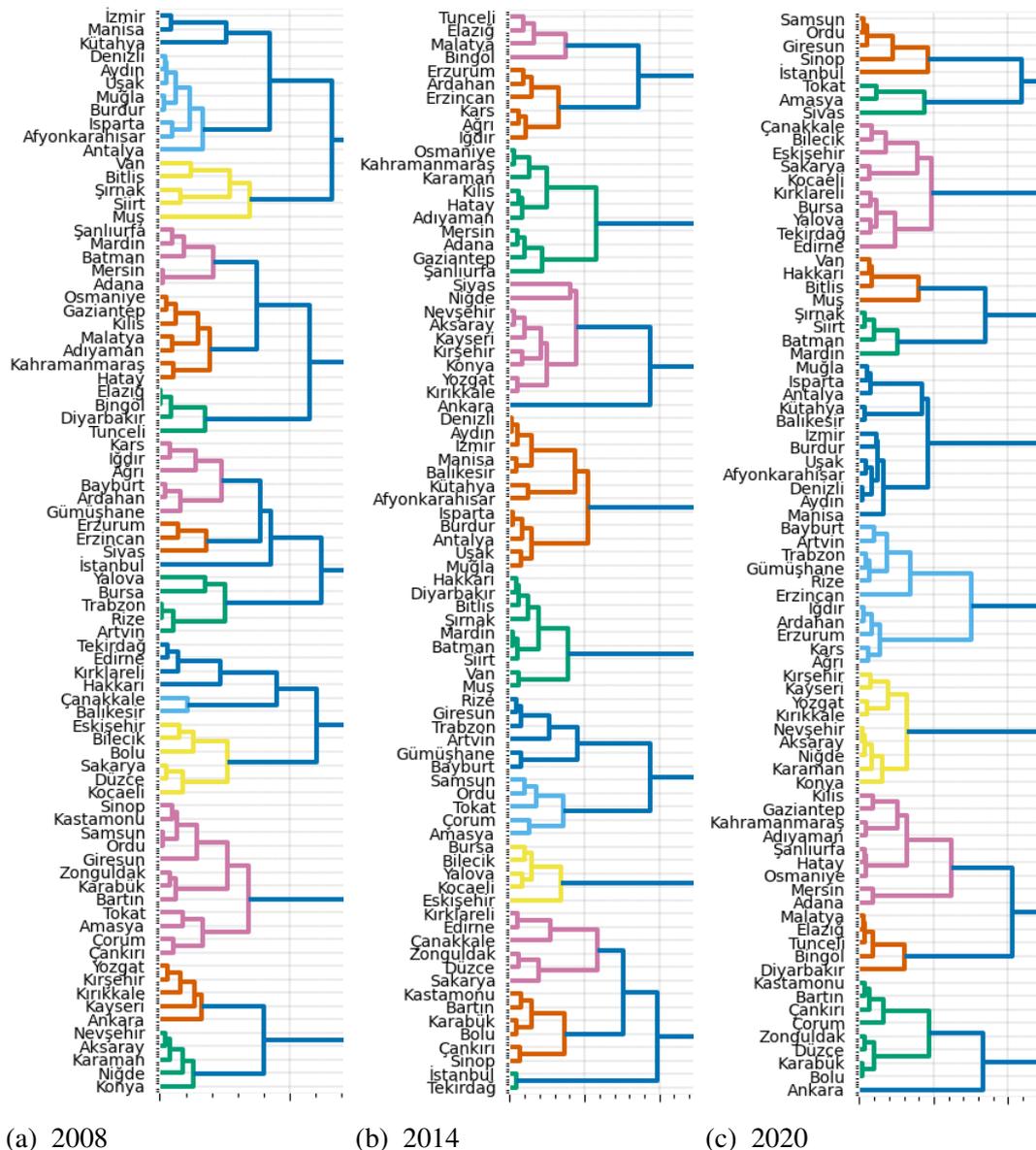


Figure 29. Hierarchical clustering dendrograms

CHAPTER 5

CONCLUSION

In this thesis, we have developed novel methods for problems related to learning representations of complex networks and demonstrated their feasibility through a set of synthetic and real-world experiments. Furthermore, we provided a thorough application of the proposed methods as well as other methods from the literature to a real-world problem. In the spirit of open science, we published open-source Python and R packages for the developed methods.

Tools and techniques from the network analysis literature are shown to be useful for analyzing diverse complex systems from natural, social, and technological domains. As volume and diversity of network data (e.g., more nodes and links and associated weights, directions, and signs) increases with the growing interest and more sophisticated data collection methods, more information is seemingly available. However, the abundance of links and weights results in denser networks with noisy, insignificant, or otherwise redundant data. Furthermore, many network analysis and visualization techniques require or desire sparsity and are not appropriate or scalable for dense and weighted networks. Network backbone extraction methods are concerned with this problem and aim to identify the useful and elucidative structure of the original networks while eliminating statistically insignificant or otherwise unimportant links.

The existing literature exclusively focuses on identifying positive links apart from few studies that extract signed backbones from projections of bipartite networks. The first essay presented in this thesis contributes to the literature on network backbone extraction by introducing the first methods to extract signed backbones of intrinsically dense, weighted, originally unipartite networks. We develop a suitable null model based on hypergeometric distribution and relative entropy minimization via the iterative proportional fitting procedure and compare observed link weights to the expected values under this null model. The proposed significance filter eliminates

the links whose weights do not significantly deviate from their expected values, providing a view into the statistical significance of observed links. The proposed vigor filter eliminates the links whose lift-based intensities are not at a desired level, providing a further sparsification of the backbone or serving as signed edge weights of the extracted backbone.

We demonstrate the feasibility of our filtering methods via empirical analyses on real-world networks from different domains such as the US interstate migration network, the voting network of Eurovision song contest, the human contact network during an academic conference, and a cohabitation similarity network of South Florida marine species. Our findings show that the proposed filters are able to extract meaningful and sparse signed backbones underlying the original intrinsically dense weighted networks. The signed backbones exhibit reciprocity, structural balance, community structure which are typically observed in most real-world networks. These results together with the methodological soundness of our null model show that the proposed filters are accurate and effective in identifying the signed backbones.

A future research direction is related to the empirical evaluation of the proposed methods. As intrinsically dense networks exist in diverse domains, a natural avenue is the utilization of the proposed methods in solving problems from different domains and providing feedback on their usefulness, weaknesses, and strengths. Another research direction is related to the improvement of the proposed filters. As the hypergeometric distribution underlying our null model originally assumes a discrete distribution, such null models may be improved to better suit the networks whose link weights are not counts of discrete events.

With the advancement of tools and techniques and the increasing computational capabilities, machine learning strategies have become very effective in attacking diverse problems in a range of domains. However, most machine learning methods require its input data to be in a tabular format with objects of interest are represented with a relatively small number of efficient features. In this context, representation learning methods aim to infer suitable latent representations of the

objects where the information is preserved by distances between objects in the new low dimensional latent space. However, as distances are invariant to certain linear transformations, the preservation of exactly the same information can be attained with different solutions. When learning representations for dynamic systems such as dynamic networks, one may obtain different embeddings at different time steps due to the inherent stochasticity of many representation learning methods and possibly not due to the actual changes happening in the system. In dynamic systems, such temporal differences in embeddings may be explained by the stability of the system or by the misalignment of embeddings due to arbitrary transformations.

The literature has not yet focused on embedding misalignment in a way that it deserves. In the second essay presented in this thesis, we provided formal definitions for the embedding alignment and its parts, embedding stability, and developed concordant metrics to measure alignment and stability. In our empirical analysis on the synthetically created data, we show that proposed metrics are able to identify, distinguish, and measure different types of alignment and stability errors that are synthetically introduced, and are robust to the number of objects and the size of embedding vectors. We further provide extensive experiments by comparing embeddings learned via several static and dynamic network embedding methods, and their aligned versions according to our measures on dynamic network inference tasks. The findings indicate that ensuring alignment according to the definitions and measurement provided in this study improves prediction accuracy by up to 90% in static network embedding methods and up to 40% in dynamic network embedding methods.

Overall, our findings indicate that our measures are effective in investigating the alignment and stability of embeddings and realigning the embeddings that are originally misaligned. These methods have the potential to be widely adopted in the dynamic representation learning literature to measure the embeddings' alignment and the system's stability, as well as to ensure alignment in supervised or unsupervised machine learning tasks to assure accuracy and improve prediction performance.

Future research may develop dynamic network representation learning methods that incorporate alignment principles explicitly in their design and ensure aligned embeddings in their output. Further work may also adapt the alignment and stability measures to work with various distance metrics and geometric spaces. Moreover, another research direction is to evaluate the effectiveness and usefulness of the proposed stability measure in investigating dynamic structural changes of systems that can be represented in latent spaces.

The third and final essay presented in this thesis provides an in-depth investigation into the structure and dynamics of the internal migration in Turkey from 2008 to 2020 by employing the novel methods presented in the first two essays as well as other complementary methods from the network analysis literature. We identify several classical migration laws in the literature and interpret our findings with respect to these laws. Using the migration networks between cities, we first extract underlying signed backbones and analyze the structure of these backbones via appropriate signed network analysis methods such as reciprocity and structural balance as well as by visualizations of overall networks and particular ego networks. We then proceed to learn the latent representation of this internal migration system and force them to be aligned over time. In this latent space, using hierarchical and density-based clustering methods, we identify communities of cities. We also investigate the stability of the system by measuring link persistence over time in the backbones as well as the stability measurement on the embeddings via the methods provided in the second essay.

The analysis on the internal migration system shows that, in line with the classical migration laws, most migration links are geographically bounded with several exceptions involving cities with large economic activity, major migration flows are countered with migration flows in the opposite direction, there are well-defined migration routes, and the migration system is generally stable over the investigated period. Apart from these general results, we also provided unique and specific insights into Turkey. To the best of our knowledge, our work is the first study

that employs signed network analysis and representation learning methods to migration networks and among the first studies that utilize a networks perspective in analyzing the internal migration system of Turkey.

Future research may extend our analysis to earlier time periods if the required migration data becomes available. Further work may also concentrate on the questions on *who*, *why*, and consequences of migration via collecting further qualitative and quantitative data. An in-depth investigation of the internal migration system for Turkey, as well as any other country, can provide a helpful understanding which may help in designing relevant social and economic policies and may assist with decision making.

REFERENCES

- Ahmed, A., Shervashidze, N., Narayanamurthy, S., Josifovski, V., & Smola, A. J. (2013). Distributed large-scale natural graph factorization. In D. Schwabe, V. Almeida, H. Glaser, R. Baeza-Yates, & S. Moon (Eds.), *Proceedings of the 22nd International Conference on World Wide Web* (pp. 37–48). New York, NY: ACM Press.
- Alstott, J., Panzarasa, P., Rubinov, M., Bullmore, E. T., & Vertes, P. E. (2014). A unifying framework for measuring weighted rich clubs. *Scientific Reports*, 4, 7258. doi:10.1038/srep07258
- Arinik, N., Figueiredo, R., & Labatut, V. (2020). Multiple partitioning of multiplex signed networks: Application to European parliament votes. *Social Networks*, 60, 83–102. doi:10.1016/j.socnet.2019.02.001
- Bacharach, M. (1965). Estimating nonnegative matrices from marginal data. *International Economic Review*, 6(3), 294–310. doi:10.2307/2525582
- Barrat, A., Barthelemy, M., Pastor-Satorras, R., & Vespignani, A. (2004). The architecture of complex weighted networks. *Proceedings of the National Academy of Sciences*, 101(11), 3747–3752. doi:10.1073/pnas.0400087101
- Barrat, A., Cattuto, C., Colizza, V., Gesualdo, F., Isella, L., Pandolfi, E., . . . van den Broeck, W. (2013). Empirical temporal networks of face-to-face human interactions. *The European Physical Journal Special Topics*, 222(6), 1295–1309. doi:10.1140/epjst/e2013-01927-7
- Belkin, M., & Niyogi, P. (2001). Laplacian eigenmaps and spectral techniques for embedding and clustering. In T. G. Dietterich, S. Becker, & Z. Ghahramani (Eds.), *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic* (pp. 585–591). Vancouver, Canada: MIT Press.
- Bhattacharya, K., Mukherjee, G., Saramaki, J., Kaski, K., & Manna, S. S. (2008). The international trade network: Weighted network analysis and modelling. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(02), P02002. doi:10.1088/1742-5468/2008/02/p02002
- Bilecen, B., Gamper, M., & Lubbers, M. J. (2018). The missing link: Social network analysis in migration and transnationalism. *Social Networks*, 53, 1–3. doi:10.1016/j.socnet.2017.07.001
- Bonchi, F., Galimberti, E., Gionis, A., Ordozgoiti, B., & Ruffo, G. (2019). Discovering polarized communities in signed networks. In W. Zhu, D. Tao, X. Cheng, P. Cui, E. Rundensteiner, D. Carmel, . . . J. X. Yu (Eds.), *Proceedings of the 28th ACM International Conference on Information and Knowledge Management* (pp. 961–970). New York, NY: ACM Press.

- Borgatti, S. P., Mehra, A., Brass, D. J., & Labianca, G. (2009). Network analysis in the social sciences. *Science*, 323(5916), 892–895. doi:10.1126/science.1165821
- Bregman, L. (1967). Proof of the convergence of Sheleikhovskii's method for a problem with transportation constraints. *USSR Computational Mathematics and Mathematical Physics*, 7(1), 191–204. doi:10.1016/0041-5553(67)90069-9
- Bruno, M., Sousa, S. F., Gürsoy, F., Serafino, M., Vianello, F. V., Vranić, A., & Boguñá, M. (2019). *Community detection in the hyperbolic space*. Retrieved from <https://arxiv.org/pdf/1906.09082>.
- Camp, E. F., Hobbs, J. P. A., De Brauwert, M., Dumbrell, A. J., & Smith, D. J. (2016). Cohabitation promotes high diversity of clownfishes in the Coral Triangle. *Proceedings of the Royal Society B: Biological Sciences*, 283(1827), 20160277. doi:10.1098/rspb.2016.0277
- Cao, J., Ding, C., & Shi, B. (2019). Motif-based functional backbone extraction of complex networks. *Physica A: Statistical Mechanics and its Applications*, 526, 121123. doi:10.1016/j.physa.2019.121123
- Cao, S., Lu, W., & Xu, Q. (2015). GraRep: Learning graph representations with global structural information. In J. Bailey, A. Moffat, C. C. Aggarwal, M. de Rijke, R. Kumar, V. Murdock, . . . J. X. Yu (Eds.), *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management* (pp. 891–900). New York, NY: ACM Press.
- Cao, S., Lu, W., & Xu, Q. (2016). Deep neural networks for learning graph representations. In D. Schuurmans & M. Wellman (Eds.), *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence* (pp. 1145–1152). Palo Alto, CA: AAAI Press.
- Cartwright, D., & Harary, F. (1956). Structural balance: A generalization of Heider's theory. *Psychological review*, 63(5), 277. doi:10.1037/h0046049
- Carvalho, R. C. D., & Charles-Edwards, E. (2020). The evolution of spatial networks of migration in Brazil between 1980 and 2010. *Population, Space and Place*, 26(7), e2332. doi:10.1002/psp.2332
- Caudillo-Cos, C., & Tapia-McClung, R. (2014). Patterns of internal migration of Mexican highly qualified population through network analysis. In B. Murgante, S. Misra, A. M. A. C. Rocha, C. Torre, J. G. Rocha, M. I. Falcão, . . . O. Gervasi (Eds.), *Computational science and its applications – ICCSA 2014* (pp. 169–184). Cham, Switzerland: Springer.
- Chakrabarti, A. S., & Sengupta, A. (2017). Productivity differences and inter-state migration in the U.S.: A multilateral gravity approach. *Economic Modelling*, 61, 156–168. doi:10.1016/j.econmod.2016.12.009

- Charron, N. (2013). Impartiality, friendship-networks and voting behavior: Evidence from voting patterns in the Eurovision song contest. *Social Networks*, 35(3), 484–497. doi:10.1016/j.socnet.2013.05.005
- Charyyev, B., & Gunes, M. H. (2019). Complex network of United States migration. *Computational Social Networks*, 6(1), 1. doi:10.1186/s40649-019-0061-6
- Chen, C., Tao, Y., & Lin, H. (2019). *Dynamic network embeddings for network evolution analysis*. Retrieved from <https://arxiv.org/pdf/1906.09860>.
- Chen, Y. W., Ni, L., & Ospina-Forero, L. (2021). Visualising internal migration flows across local authorities in England and Wales. *Environment and Planning A: Economy and Space*, 53(4), 616–618. doi:10.1177/0308518X20968568
- Chinazzi, M., Fagiolo, G., Reyes, J. A., & Schiavo, S. (2013). Post-mortem examination of the international financial network. *Journal of Economic Dynamics and Control*, 37(8), 1692–1713. doi:10.1016/j.jedc.2013.01.010
- Coscia, M., & Neffke, F. M. (2017). Network backboning with noisy data. In C. Baru, B. Thuraisingham, Y. Papakonstantinou, & Y. Diao (Eds.), *Proceedings of the 2017 IEEE 33rd International Conference on Data Engineering (ICDE)* (pp. 425–436). New York, NY: IEEE Press.
- D'Angelo, S., Murphy, T. B., & Alfò, M. (2019). Latent space modelling of multidimensional networks with application to the exchange of votes in Eurovision song contest. *The Annals of Applied Statistics*, 13(2), 900–930. doi:10.1214/18-AOAS1221
- Dağdelen, G., & Karakılıç, İ. Z. (2011). *Disaggregating citizenship: Transnational partition of the "economic" from the "socio-cultural" in the case of a Turkish community in London*. Paper presented at the International RC21 Conference, Amsterdam.
- Danchev, V., & Porter, M. A. (2020). *Migration networks: Applications of network analysis to macroscale migration patterns*. Retrieved from <https://arxiv.org/pdf/2002.10992>.
- Davis, J. A. (1967). Clustering and structural balance in graphs. *Human Relations*, 20(2), 181–187. doi:10.1177/001872676702000206
- Derr, T., Ma, Y., & Tang, J. (2018). Signed graph convolutional networks. In F. Zhu, J. Yu, D. Tao, & B. Thuraisingham (Eds.), *Proceedings of the 2018 IEEE International Conference on Data Mining (ICDM)* (pp. 929–934). New York, NY: IEEE Press.
- Dianati, N. (2016). Unwinding the hairball graph: Pruning algorithms for weighted complex networks. *Physical Review E*, 93, 012304. doi:10.1103/PhysRevE.93.012304

- Dickinson, J. Y. (1966). Aspects of Italian immigration to Philadelphia. *The Pennsylvania Magazine of History and Biography*, 90(4), 445–465.
- Domagalski, R., Neal, Z. P., & Sagan, B. (2021). Backbone: An R package for extracting the backbone of bipartite projections. *PLOS ONE*, 16(1), 1–20. doi:10.1371/journal.pone.0244363
- Doreian, P., & Mrvar, A. (2009). Partitioning signed social networks. *Social Networks*, 31(1), 1–11. doi:10.1016/j.socnet.2008.08.001
- Doreian, P., & Mrvar, A. (2015). Structural balance and signed international relations. *Journal of Social Structure*, 16(1529-1227), 1–49. doi:10.21307/joss-2019-012
- Drbohlav, D., & Džúrová, D. (2007). "Where are they going?": Immigrant inclusion in the Czech Republic (a case study on Ukrainians, Vietnamese, and Armenians in Prague). *International Migration*, 45(2), 69–95. doi:10.1111/j.1468-2435.2007.00404.x
- Dunne, J. A., Williams, R. J., & Martinez, N. D. (2002). Food-web structure and network theory: The role of connectance and size. *Proceedings of the National Academy of Sciences*, 99(20), 12917–12922. doi:10.1073/pnas.192407699
- Esmailian, P., & Jalili, M. (2015). Community detection in signed networks: The role of negative ties in different scales. *Scientific Reports*, 5(1), 14339. doi:10.1038/srep14339
- Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In E. Simoudis, J. Han, & U. Fayyad (Eds.), *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining* (pp. 226–231). Palo Alto, CA: AAAI Press.
- Euler, L. (1736). Solutio problematis ad geometriam situs pertinentis. *Commentarii Academiae Scientiarum Imperialis Petropolitanae*, 8, 128–140.
- European Broadcasting Union. (2003). *Final of Riga 2003*. Retrieved from <https://eurovision.tv/event/riga-2003>.
- Fang, C., Kohram, M., Meng, X., & Ralescu, A. (2011). *Graph embedding framework for link prediction and vertex behavior modeling in temporal social networks*. Retrieved from <https://www.researchgate.net/publication/264885614>.
- Fenn, D., Suleman, O., Efstathiou, J., & Johnson, N. F. (2006). How does Europe make its mind up? connections, cliques, and compatibility between countries in the Eurovision song contest. *Physica A: Statistical Mechanics and its Applications*, 360(2), 576–598. doi:10.1016/j.physa.2005.06.051
- Fernandes, T. T., Dáttilo, W., Silva, R. R., Luna, P., Braz, A. B., & Morini, M. S. C. (2020). Cohabitation and niche overlap in the occupation of twigs by

- arthropods in the leaf litter of Brazilian Atlantic Forest. *Insectes Sociaux*, 67(2), 239–247. doi:10.1007/s00040-020-00753-w
- Foti, N. J., Hughes, J. M., & Rockmore, D. N. (2011). Nonparametric sparsification of complex multiscale networks. *PLOS ONE*, 6(2), e16431. doi:10.1371/journal.pone.0016431
- Gangal, V., Narwekar, A., Ravindran, B., & Narayanam, R. (2016). Trust and distrust across coalitions: Shapley value based centrality measures for signed networks. In D. Schuurmans & M. Wellman (Eds.), *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence* (pp. 4210–4211). Palo Alto, CA: AAAI Press.
- Garcia, D., & Tanase, D. (2013). Measuring cultural dynamics through the Eurovision song contest. *Advances in Complex Systems*, 16(08), 1350037. doi:10.1142/S0219525913500379
- Gemmetto, V., Cardillo, A., & Garlaschelli, D. (2017). *Irreducible network backbones: Unbiased graph filtering via maximum entropy*. Retrieved from <https://arxiv.org/pdf/1706.00230>.
- Goldade, T., Charyyev, B., & Gunes, M. H. (2018). Network analysis of migration patterns in the United States. In C. Cherifi, H. Cherifi, M. Karsai, & M. Musolesi (Eds.), *Complex networks & their applications VI* (pp. 770–783). Cham, Switzerland: Springer.
- Gonçalves, B., Perra, N., & Vespignani, A. (2011). Modeling users' activity on Twitter networks: Validation of Dunbar's number. *PLOS ONE*, 6(8), e22656. doi:10.1371/journal.pone.0022656
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. Cambridge, MA: MIT Press.
- Gower, J. C. (1975). Generalized procrustes analysis. *Psychometrika*, 40(1), 33–51. doi:10.1007/bf02291478
- Goyal, P., Kamra, N., He, X., & Liu, Y. (2018). *DynGEM: Deep embedding method for dynamic graphs*. Retrieved from <https://arxiv.org/pdf/1805.11273>.
- Grady, D., Thiemann, C., & Brockmann, D. (2012). Robust classification of salient links in complex networks. *Nature Communications*, 3(1), 864. doi:10.1038/ncomms1847
- Granovetter, M. S. (1973). The strength of weak ties. *American journal of sociology*, 78(6), 1360–1380.
- Greenwood, M. J. (1997). Internal migration in developed countries. In M. R. Rosenzweig & O. Stark (Eds.), *Handbook of population and family economics* (pp. 647–720). Amsterdam, Netherlands: Elsevier.

- Grover, A., & Leskovec, J. (2016). Node2vec: Scalable feature learning for networks. In B. Krishnapuram, M. Shah, A. Smola, C. Aggarwal, D. Shen, & R. Rastogi (Eds.), *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 855–864). New York, NY: ACM Press.
- Gupta, M., & Mishra, R. (2020). Network projection-based edge classification framework for signed networks. *Decision Support Systems*, *135*, 113321. doi:10.1016/j.dss.2020.113321
- Gürsoy, F., & Badur, B. (in press). Extracting the signed backbone of intrinsically dense weighted networks. *Journal of Complex Networks*.
- Gürsoy, F., Haddad, M., & Bothorel, C. (2021). *Alignment and stability of embeddings: Measurement and inference improvement*. Retrieved from <https://arxiv.org/pdf/2101.07251>.
- Guveli, A., Ganzeboom, H. B. G., Platt, L., Nauck, B., Baykara-Krumme, H., Eroğlu, Ş., ... Spierings, N. (2016). *Intergenerational consequences of migration: Socio-economic, family and cultural patterns of stability and change in Turkey and Europe*. Basingstoke, United Kingdom: Palgrave Macmillan.
- Haddad, M., Bothorel, C., Lenca, P., & Bedart, D. (2020). TemporalNode2vec: Temporal node embedding in temporal networks. In H. Cherifi, S. Gaito, J. F. Mendes, E. Moro, & L. M. Rocha (Eds.), *Complex networks and their applications VIII* (pp. 891–902). Cham, Switzerland: Springer.
- Harary, F., Lim, M.-H., & Wunsch, D. C. (2002). Signed graphs for portfolio analysis in risk management. *IMA Journal of Management Mathematics*, *13*(3), 201–210. doi:10.1093/imaman/13.3.201
- Heider, F. (1946). Attitudes and cognitive organization. *The Journal of Psychology*, *21*(1), 107–112. doi:10.1080/00223980.1946.9917275
- Hewapathirana, I. U., Lee, D., Moltchanova, E., & McLeod, J. (2020). Change detection in noisy dynamic networks: A spectral embedding approach. *Social Network Analysis and Mining*, *10*(1), 14. doi:10.1007/s13278-020-0625-3
- Hoche, S., Nürnberger, A., & Flach, P. (2007). Network analysis in natural sciences and engineering. *AI Communications*, *20*, 229–230. Retrieved from <https://content.iospress.com/journals/ai-communications/>.
- Hosseini-Pozveh, M., Zamanifar, K., & Naghsh-Nilchi, A. R. (2019). Assessing information diffusion models for influence maximization in signed social networks. *Expert Systems with Applications*, *119*, 476–490. doi:10.1016/j.eswa.2018.07.064
- Hou, C., Zhang, H., He, S., & Tang, K. (2020). GloDyNE: Global topology preserving dynamic network embedding. *IEEE Transactions on Knowledge and Data Engineering*. Advance online publication. doi:10.1109/TKDE.2020.3046511

- Hu, B., Wang, H., & Yu, Z. (2019). Drug side-effect prediction via random walk on the signed heterogeneous drug network. *Molecules*, *24*(20). doi:10.3390/molecules24203668
- Hu, Y., & Zhu, D. (2009). Empirical analysis of the worldwide maritime transportation network. *Physica A: Statistical Mechanics and its Applications*, *388*(10), 2061–2071. doi:10.1016/j.physa.2008.12.016
- Iacono, G., Ramezani, F., Soranzo, N., & Altafini, C. (2010). Determining the distance to monotonicity of a biological network: A graph-theoretical approach. *IET Systems Biology*, *4*(3), 223–235. doi:10.1049/iet-syb.2009.0040
- Isella, L., Stehlé, J., Barrat, A., Cattuto, C., Pinton, J., & van den Broeck, W. (2011). What's in a crowd? analysis of face-to-face behavioral networks. *Journal of Theoretical Biology*, *271*(1), 166–180. doi:10.1016/j.jtbi.2010.11.033
- Jiang, B., Zhang, Z.-L., & Towsley, D. (2015). Reciprocity in social networks with capacity constraints. In L. Cao, C. Zhang, T. Joachims, G. Webb, D. D. Margineantu, & G. Williams (Eds.), *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 457–466). New York, NY: ACM Press.
- Ju, W., Chen, L., Li, B., Liu, W., Sheng, J., & Wang, Y. (2020). A new algorithm for positive influence maximization in signed networks. *Information Sciences*, *512*, 1571–1591. doi:10.1016/j.ins.2019.10.061
- Kikkawa, J. (1968). Ecological association of bird species and habitats in eastern Australia; similarity analysis. *Journal of Animal Ecology*, *37*(1), 143–165. doi:10.2307/2716
- Kim, J., Park, H., Lee, J.-E., & Kang, U. (2018). SIDE: Representation learning in signed directed networks. In P.-A. Champin, F. Gandon, L. Médini, M. Lalmas, & P. G. Ipeirotis (Eds.), *Proceedings of the 2018 World Wide Web Conference* (pp. 509–518). Geneva, Switzerland: International World Wide Web Conferences Steering Committee.
- Kipf, T. N., & Welling, M. (2016). *Semi-supervised classification with graph convolutional networks*. Retrieved from <https://arxiv.org/pdf/1609.02907>.
- Klugman, J. (2009). *Human development report 2009. overcoming barriers: Human mobility and development*. Basingstoke, United Kingdom: Palgrave Macmillan.
- Kobayashi, T., Takaguchi, T., & Barrat, A. (2019). The structured backbone of temporal social ties. *Nature Communications*, *10*(1), 220. doi:10.1038/s41467-018-08160-3
- Kojaku, S., & Masuda, N. (2019). Constructing networks by filtering correlation matrices: A null model approach. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *475*(2231), 20190578. doi:10.1098/rspa.2019.0578

- Kok, P. (1997). The definition of migration and its application: Making sense of recent South African census and survey data. *Southern African Journal of Demography*, 7(1), 19–30.
- Kritzer, J. P., DeLucia, M.-B., Greene, E., Shumway, C., Topolski, M. F., Thomas-Blate, J., . . . Smith, K. (2016). The importance of benthic habitats for coastal fisheries. *BioScience*, 66(4), 274–284. doi:10.1093/biosci/biw014
- Kulisiewicz, M., Kazienko, P., Szymanski, B. K., & Michalski, R. (2018). Entropy measures of human communication dynamics. *Scientific Reports*, 8(1), 15697. doi:10.1038/s41598-018-32571-3
- Kulkarni, V., Al-Rfou, R., Perozzi, B., & Skiena, S. (2015). Statistically significant detection of linguistic change. In A. Gangemi, S. Leonardi, & A. Panconesi (Eds.), *Proceedings of the 24th International Conference on World Wide Web* (pp. 625–635). Geneva, Switzerland: International World Wide Web Conferences Steering Committee.
- Kullback, S., & Leibler, R. A. (1951). On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1), 79–86. doi:10.1214/aoms/1177729694
- Kunegis, J., Preusse, J., & Schwagereit, F. (2013). What is the added value of negative links in online social networks? In D. Schwabe, V. Almeida, H. Glaser, R. Baeza-Yates, & S. Moon (Eds.), *Proceedings of the 22nd International Conference on World Wide Web* (pp. 727–736). New York, NY: ACM Press.
- Lee, E. S. (1966). A theory of migration. *Demography*, 3(1), 47–57. doi:10.2307/2060063
- Leskovec, J., Huttenlocher, D., & Kleinberg, J. (2010). Signed networks in social media. In E. Mynatt, G. Fitzpatrick, S. Hudson, K. Edwards, & T. Rodden (Eds.), *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 1361–1370). New York, NY: ACM Press.
- Levorato, M., Figueiredo, R., Frota, Y., & Drummond, L. (2017). Evaluating balancing on social networks through the efficient solution of correlation clustering problems. *EURO Journal on Computational Optimization*, 5(4), 467–498. doi:10.1007/s13675-017-0082-6
- Li, R.-H., Dai, Q., Qin, L., Wang, G., Xiao, X., Yu, J. X., & Qiao, S. (2021). Signed clique search in signed networks: Concepts and algorithms. *IEEE Transactions on Knowledge and Data Engineering*, 33(2), 710–727. doi:10.1109/TKDE.2019.2904569
- Liang, S., Zhang, X., Ren, Z., & Kanoulas, E. (2018). Dynamic embeddings for user profiling in Twitter. In Y. Guo & F. Farooq (Eds.), *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 1764–1773). New York, NY: ACM.

- Liang, X., Hsu, L. M., Lu, H., Sumiyoshi, A., He, Y., & Yang, Y. (2017). The rich-club organization in rat functional brain network to balance between communication cost and efficiency. *Cerebral Cortex*, 28(3), 924–935. doi:10.1093/cercor/bhw416
- Liebig, J., & Rao, A. (2016). Fast extraction of the backbone of projected bipartite networks to aid community detection. *EPL (Europhysics Letters)*, 113(2), 28003. doi:10.1209/0295-5075/113/28003
- Liu, X. [Xi], Andris, C., & Desmarais, B. A. (2019). Migration and political polarization in the U.S.: An analysis of the county-level migration network. *PLOS ONE*, 14(11), 1–16. doi:10.1371/journal.pone.0225405
- Liu, X. [Xueyan], Song, W., Musial, K., Zhao, X., Zuo, W., & Yang, B. (2020). Semi-supervised stochastic blockmodel for structure analysis of signed networks. *Knowledge-Based Systems*, 195, 105714. doi:10.1016/j.knosys.2020.105714
- Ma, Y., Zhu, X., & Yu, Q. (2019). Clusters detection based leading eigenvector in signed networks. *Physica A: Statistical Mechanics and its Applications*, 523, 1263–1275. doi:10.1016/j.physa.2019.04.061
- Mahdavi, S., Khoshraftar, S., & An, A. (2018). dynnode2vec: Scalable dynamic network embedding. In D. Kossmann, B. Liu, N. Abe, H. Liu, & C. Pu (Eds.), *Proceedings of the 2018 IEEE International Conference on Big Data* (pp. 3762–3765). New York, NY: IEEE Press.
- Maier, G., & Vyborny, M. (2005). *Internal migration between US States - A social network analysis*. Retrieved from <https://ideas.repec.org/p/wiw/wus009/1084.html>.
- Mantzaris, A. V., Rein, S. R., & Hopkins, A. D. (2018a). Examining collusion and voting biases between countries during the Eurovision song contest since 1957. *Journal of Artificial Societies and Social Simulation*, 21(1), 1. doi:10.18564/jasss.3580
- Mantzaris, A. V., Rein, S. R., & Hopkins, A. D. (2018b). Preference and neglect amongst countries in the Eurovision song contest. *Journal of Computational Social Science*, 1(2), 377–390. doi:10.1007/s42001-018-0020-2
- Marcaccioli, R., & Livan, G. (2019). A Pólya urn approach to information filtering in complex networks. *Nature Communications*, 10(1). doi:10.1038/s41467-019-08667-3
- Massey, D. S., Arango, J., Hugo, G., Kouaouci, A., Pellegrino, A., & Taylor, J. E. (1993). Theories of international migration: A review and appraisal. *Population and Development Review*, 19(3), 431–466. doi:10.2307/2938462

- Mastrandrea, R., Fournet, J., & Barrat, A. (2015). Contact patterns in a high school: A comparison between data collected using wearable sensors, contact diaries and friendship surveys. *PLOS ONE*, *10*(9), 1–26. doi:10.1371/journal.pone.0136497
- Mazzarisi, P., & Lillo, F. (2017). Methods for reconstructing interbank networks from limited information: A comparison. In F. Abergel, H. Aoyama, B. K. Chakrabarti, A. Chakraborti, N. Deo, D. Raina, & I. Vodenska (Eds.), *Econophysics and sociophysics: Recent progress and future directions* (pp. 201–215). Cham, Switzerland: Springer.
- McAuliffe, M., & Khadria, B. (Eds.). (2019). *World migration report 2020*. Geneva, Switzerland: International Organization for Migration.
- Menczer, F., Fortunato, S., & Davis, C. A. (2020). *A first course in network science*. Cambridge, United Kingdom: Cambridge University Press.
- Mercado, P., Bosch, J., & Stoll, M. (2020). Node classification for signed social networks using diffuse interface methods. In U. Brefeld, E. Fromont, A. Hotho, A. Knobbe, M. Maathuis, & C. Robardet (Eds.), *Machine learning and knowledge discovery in databases* (pp. 524–540). Cham, Switzerland: Springer.
- Moreno, J. L. (1934). *Who shall survive? A new approach to the problem of human interrelations*. Washington, DC: Nervous and Mental Disease Publishing.
- Neal, Z. (2014). The backbone of bipartite projections: Inferring relationships from co-authorship, co-sponsorship, co-attendance and other co-behaviors. *Social Networks*, *39*, 84–97. doi:10.1016/j.socnet.2014.06.001
- Newman, M. (2003). The structure and function of complex networks. *SIAM Review*, *45*(2), 167–256. doi:10.1137/S003614450342480
- Newman, M. (2018). *Networks*. Oxford, United Kingdom: Oxford University Press.
- Organisation for Economic Co-operation and Development. (2021). *OECD statistics*. Retrieved from <https://stats.oecd.org>.
- Ou, M., Cui, P., Pei, J., Zhang, Z., & Zhu, W. (2016). Asymmetric transitivity preserving graph embedding. In B. Krishnapuram, M. Shah, A. Smola, C. Aggarwal, D. Shen, & R. Rastogi (Eds.), *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 1105–1114). New York, NY: ACM Press.
- Palmucci, A., Liao, H., Napoletano, A., & Zaccaria, A. (2020). Where is your field going? a machine learning approach to study the relative motion of the domains of physics. *PLOS ONE*, *15*(6), 1–16. doi:10.1371/journal.pone.0233997
- Passino, F. S., Bertiger, A. S., Neil, J. C., & Heard, N. A. (2019). *Link prediction in dynamic networks using random dot product graphs*. Retrieved from <https://arxiv.org/pdf/1912.10419>.

- Peixoto, T. P. (2020). *The Netzschleuder network catalogue and repository*. Retrieved from <https://networks.skewed.de/>.
- Perozzi, B., Al-Rfou, R., & Skiena, S. (2014). DeepWalk: Online learning of social representations. In S. Macskassy, C. Perlich, J. Leskovec, W. Wang, & R. Ghani (Eds.), *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 701–710). New York, NY: ACM Press.
- Pitoski, D., Lampoltshammer, T. J., & Parycek, P. (2021). Network analysis of internal migration in Croatia. *Computational Social Networks*, 8(1), 10. doi:10.1186/s40649-021-00093-0
- Radicchi, F., Ramasco, J. J., & Fortunato, S. (2011). Information filtering in complex weighted networks. *Physical Review E*, 83(4), 1–9. doi:10.1103/PhysRevE.83.046101
- Ravenstein, E. G. (1885). The laws of migration. *Journal of the Statistical Society of London*, 48(2), 167–235. doi:10.2307/2979181
- Ravenstein, E. G. (1889). The laws of migration. *Journal of the Royal Statistical Society*, 52(2), 241–305. doi:10.2307/2979333
- Reniers, G. (1999). On the history and selectivity of Turkish and Moroccan migration to Belgium. *International Migration*, 37(4), 679–713. doi:10.1111/1468-2435.00090
- Riccaboni, M., Rossi, A., & Schiavo, S. (2013). Global networks of trade and bits. *Journal of Economic Interaction and Coordination*, 8(1), 33–56. doi:10.1007/s11403-012-0101-x
- Robbins, H., & Monro, S. (1951). A stochastic approximation method. *The Annals of Mathematical Statistics*, 22(3), 400–407. doi:10.1214/aoms/1177729586
- Roberts, J. A., Perry, A., Roberts, G., Mitchell, P. B., & Breakspear, M. (2017). Consistency-based thresholding of the human connectome. *NeuroImage*, 145, 118–129. doi:10.1016/j.neuroimage.2016.09.053
- Rogers, R. (1983). Return migration in comparative perspective. *International Migration Review*, 17(1_suppl), 277–299. doi:10.1177/019791838301701S40
- Rokach, L., & Maimon, O. (2005). Clustering methods. In O. Maimon & L. Rokach (Eds.), *Data mining and knowledge discovery handbook* (pp. 321–352). Boston, MA: Springer.
- Roweis, S. T., & Saul, L. K. (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500), 2323–2326. doi:10.1126/science.290.5500.2323

- Saiz, H., Gómez-Gardeñes, J., Nuche, P., Girón, A., Pueyo, Y., & Alados, C. L. (2017). Evidence of structural balance in spatial ecological networks. *Ecography*, *40*(6), 733–741. doi:10.1111/ecog.02561
- Schiavo, S., Reyes, J., & Fagiolo, G. (2010). International trade and financial integration: A weighted network analysis. *Quantitative Finance*, *10*(4), 389–399. doi:10.1080/14697680902882420
- Schich, M., & Meirelles, I. (2017). Arts, humanities and complex networks reach critical threshold. *Leonardo*, *50*(5), 440–440. doi:10.1162/LEON_e_01484
- Schoch, D. (2020). *signnet: An R package to analyze signed networks*. Retrieved from <https://github.com/schochastics/signnet/>.
- Schönemann, P. H. (1966). A generalized solution of the orthogonal procrustes problem. *Psychometrika*, *31*(1), 1–10. doi:10.1007/BF02289451
- Serrano, M. Á., Boguñá, M., & Vespignani, A. (2009a). Extracting the multiscale backbone of complex weighted networks. *Proceedings of the National Academy of Sciences*, *106*(16), 6483–6488. doi:10.1073/pnas.0808904106
- Serrano, M. Á., Boguñá, M., & Vespignani, A. (2009b). Reply to slater: Extracting the backbone of multiscale networks. *Proceedings of the National Academy of Sciences*, *106*(26), E67. doi:10.1073/pnas.0905243106
- Sgrignoli, P., Metulini, R., Schiavo, S., & Riccaboni, M. (2015). The relation between global migration and trade networks. *Physica A: Statistical Mechanics and its Applications*, *417*, 245–260. doi:10.1016/j.physa.2014.09.037
- Singer, U., Guy, I., & Radinsky, K. (2019). Node embedding over temporal graphs. In C. Sierra (Ed.), *Proceedings of the 28th International Joint Conference on Artificial Intelligence* (pp. 4605–4612). Palo Alto, CA: AAAI Press.
- Slater, P. B. (1974). College student mobility in the United States in fall 1968: Graph-theoretic structuring of transaction flows. In E. D. Goldfield (Ed.), *Proceedings of the American Statistical Association Social Statistics Section* (pp. 440–446). Alexandria, VA: American Statistical Association.
- Slater, P. B. (1975a). A hierarchical regionalization of RSFSR administrative units using 1966–69 migration data. *Soviet Geography*, *16*(7), 453–465. doi:10.1080/00385417.1975.10640088
- Slater, P. B. (1975b). The identification of Turkish regions using 1965 lifetime interprovincial migration data. In E. D. Goldfield (Ed.), *Proceedings of the American Statistical Association Social Statistics Section* (pp. 207–213). Alexandria, VA: American Statistical Association.
- Slater, P. B. (1976a). A hierarchical regionalization of Japanese prefectures using 1972 interprefectural migration flows. *Regional Studies*, *10*(1), 123–132. doi:10.1080/09595237600185121

- Slater, P. B. (1976b). A multiterminal network-flow analysis of an unadjusted Spanish interprovincial migration table. *Environment and Planning A: Economy and Space*, 8(8), 875–878. doi:10.1068/a080875
- Slater, P. B. (2008). *Hubs and clusters in the evolving United States internal migration network*. Retrieved from <https://arxiv.org/pdf/0809.2768>.
- Slater, P. B. (2009). A two-stage algorithm for extracting the multiscale backbone of complex weighted networks. *Proceedings of the National Academy of Sciences*, 106(26), E66. doi:10.1073/pnas.0904725106
- Smith, R. J. (1979). The ethnic Japanese in Brazil. *Journal of Japanese Studies*, 5(1), 53–70. doi:10.2307/132086
- Snijders, T. A., van de Bunt, G. G., & Steglich, C. E. (2010). Introduction to stochastic actor-based models for network dynamics. *Social Networks*, 32(1), 44–60. doi:10.1016/j.socnet.2009.02.004
- SocioPatterns. (2021). *SocioPatterns*. Retrieved from <http://www.sociopatterns.org/>.
- Stehlé, J., Voirin, N., Barrat, A., Cattuto, C., Isella, L., Pinton, J.-F., . . . Vanhems, P. (2011). High-resolution measurements of face-to-face contact patterns in a primary school. *PLOS ONE*, 6(8), 1–13. doi:10.1371/journal.pone.0023176
- Svete, A., & Hostnik, J. (2020). *It is not just about the melody: How Europe votes for its favorite songs*. Retrieved from <https://arxiv.org/pdf/2002.06609>.
- Szell, M., Lambiotte, R., & Thurner, S. (2010). Multirelational organization of large-scale social networks in an online world. *Proceedings of the National Academy of Sciences*, 107(31), 13636–13641. doi:10.1073/pnas.1004008107
- Tang, C., Dong, X., Lian, Y., & Tang, D. (2020). Do Chinese hospital services constitute an oligopoly? evidence of the rich-club phenomenon in a patient referral network. *Future Generation Computer Systems*, 105, 492–501. doi:10.1016/j.future.2019.12.001
- Tang, J. [Jian], Qu, M., Wang, M., Zhang, M., Yan, J., & Mei, Q. (2015). LINE: Large-scale information network embedding. In A. Gangemi, S. Leonardi, & A. Panconesi (Eds.), *Proceedings of the 24th International Conference on World Wide Web* (pp. 1067–1077). Geneva, Switzerland: International World Wide Web Conferences Steering Committee.
- Tang, J. [Jie], Zhang, J., Yao, L., Li, J., Zhang, L., & Su, Z. (2008). ArnetMiner: Extraction and mining of academic social networks. In Y. Li, B. Liu, & S. Sarawagi (Eds.), *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 990–998). New York, NY: ACM Press.
- Tang, J. [Jiliang], Aggarwal, C., & Liu, H. (2016a). Node classification in signed social networks. In S. C. Venkatasubramanian & W. Meira (Eds.), *Proceedings*

of the 2016 SIAM International Conference on Data Mining (SDM) (pp. 54–62). Philadelphia, PA: Society for Industrial and Applied Mathematics.

- Tang, J. [Jiliang], Aggarwal, C., & Liu, H. (2016b). Recommendations in signed social networks. In J. Bourdeau, J. A. Hendler, R. N. Nkambou, I. Horrocks, & B. Y. Zhao (Eds.), *Proceedings of the 25th International Conference on World Wide Web* (pp. 31–40). Geneva, Switzerland: International World Wide Web Conferences Steering Committee.
- Tang, J. [Jiliang], Chang, Y., Aggarwal, C., & Liu, H. (2016). A survey of signed network mining in social media. *ACM Computing Surveys*, 49(3). doi:10.1145/2956185
- Tang, J. [Jiliang], Hu, X., & Liu, H. (2014). Is distrust the negation of trust? The value of distrust in social media. In L. Ferres, G. Rossi, V. Almeida, & E. Herder (Eds.), *Proceedings of the 25th ACM Conference on Hypertext and Social Media* (pp. 148–157). New York, NY: ACM Press.
- Torres, L., Blevins, A. S., Bassett, D. S., & Eliassi-Rad, T. (2020). *The why, how, and when of representations for complex systems*. Retrieved from <https://arxiv.org/pdf/2006.02870>.
- Tumminello, M., Aste, T., Di Matteo, T., & Mantegna, R. N. (2005). A tool for filtering information in complex systems. *Proceedings of the National Academy of Sciences*, 102(30), 10421–10426. doi:10.1073/pnas.0500298102
- Tumminello, M., Miccichè, S., Lillo, F., Piilo, J., & Mantegna, R. N. (2011). Statistically validated networks in bipartite complex systems. *PLOS ONE*, 6(3), e17994. doi:10.1371/journal.pone.0017994
- Turkish Statistical Institute. (2021). *Address Based Population Registration System results*. Retrieved from <https://biruni.tuik.gov.tr/>.
- United States Census Bureau. (2019). *State-to-state migration flows*. Retrieved from <https://census.gov/data/tables/time-series/demo/geographic-mobility/state-to-state-migration.html>.
- van de Bunt, G. G., Van Duijn, M. A., & Snijders, T. A. (1999). Friendship networks through time: An actor-oriented dynamic statistical network model. *Computational & Mathematical Organization Theory*, 5(2), 167–192. doi:10.1023/A:1009683123448
- van der Leij, M., & Goyal, S. (2011). Strong ties in a small world. *Review of Network Economics*, 10(2). doi:10.2202/1446-9022.1278
- Vanhems, P., Barrat, A., Cattuto, C., Pinton, J.-F., Khanafer, N., Régis, C., . . . Voirin, N. (2013). Estimating potential infection transmission routes in hospital wards using wearable proximity sensors. *PLOS ONE*, 8(9), 1–9. doi:10.1371/journal.pone.0073970

- Wan, C., Fang, Y., Wang, C., Lv, Y., Tian, Z., & Wang, Y. (2019). Signrank: A novel random walking based ranking algorithm in signed networks. *Wireless Communications and Mobile Computing, 2019*, 4813717. doi:10.1155/2019/4813717
- Wang, D., Cui, P., & Zhu, W. (2016). Structural deep network embedding. In B. Krishnapuram, M. Shah, A. Smola, C. Aggarwal, D. Shen, & R. Rastogi (Eds.), *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 1225–1234). New York, NY: ACM Press.
- Wei, Y., Song, W., Xiu, C., & Zhao, Z. (2018). The rich-club phenomenon of China's population flow network during the country's spring festival. *Applied Geography, 96*, 77–85. doi:10.1016/j.apgeog.2018.05.009
- Xiao, H., Ordozgoiti, B., & Gionis, A. (2020). Searching for polarization in signed graphs: A local spectral approach. In Y. Huang, I. King, T.-Y. Liu, & M. van Steen (Eds.), *Proceedings of The Web Conference 2020* (pp. 362–372). New York, NY: ACM Press.
- Xu, J., Tao, Y., Yan, Y., & Lin, H. (2018). Exploring evolution of dynamic networks via diachronic node embeddings. *IEEE Transactions on Visualization and Computer Graphics, 26*(7), 2387–2402. doi:10.1109/TVCG.2018.2887230
- Yahoo! (2021). *The Yahoo Webscope Program*. Retrieved from <https://webscope.sandbox.yahoo.com/>.
- Yakar, M., & Eteman, F. S. (2017). Türkiye'de iller arası göçlerin nodexl ile sosyal ağ analizi. *Göç Dergisi, 4*(1), 82–109. doi:10.33182/gd.v4i1.573
- Ou-Yang, L., Dai, D., & Zhang, X. (2015). Detecting protein complexes from signed protein-protein interaction networks. *IEEE/ACM Transactions on Computational Biology and Bioinformatics, 12*(6), 1333–1344. doi:10.1109/TCBB.2015.2401014
- Zhan, X. X., Hanjalic, A., & Wang, H. (2019). Information diffusion backbones in temporal networks. *Scientific Reports, 9*(1), 6798. doi:10.1038/s41598-019-43029-5
- Zhang, P., Song, X., Xue, L., & Gu, K. (2019). A new recommender algorithm on signed networks. *Physica A: Statistical Mechanics and its Applications, 520*, 317–321. doi:10.1016/j.physa.2019.01.054
- Zhang, X., Zhang, Z., Zhao, H., Wang, Q., & Zhu, J. (2014). Extracting the globally and locally adaptive backbone of complex networks. *PLOS ONE, 9*(6). doi:10.1371/journal.pone.0100428
- Zhou, L., Yang, Y., Ren, X., Wu, F., & Zhuang, Y. (2018). Dynamic network embedding by modeling triadic closure process. In S. A. McIlraith &

K. Q. Weinberger (Eds.), *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence* (pp. 571–578). Palo Alto, CA: AAAI Press.

Zhou, S., & Mondragon, R. J. (2004). The rich-club phenomenon in the internet topology. *IEEE Communications Letters*, 8(3), 180–182.
doi:10.1109/LCOMM.2004.823426

Zhou, Y., Liu, W., Pei, Y., Wang, L., Zha, D., & Fu, T. (2019). Dynamic network embedding by semantic evolution. In C. Jayne, Z. Somogyvari, P. Angelov, & M. Roveri (Eds.), *Proceedings of the 2019 International Joint Conference on Neural Networks (IJCNN)* (pp. 1–8). New York, NY: IEEE Press.