RESEARCH

An information-theoretic, all-scales approach to comparing networks

James P. Bagrow[1*] ⓘ and Erik M. Bollt[2]

*Correspondence:
james.bagrow@uvm.edu
[1]Department of Mathematics &
Statistics, Vermont Complex
Systems Center, University of
Vermont, Burlington, VT, USA
Full list of author information is
available at the end of the article

Abstract

As network research becomes more sophisticated, it is more common than ever for researchers to find themselves not studying a single network but needing to analyze sets of networks. An important task when working with sets of networks is network comparison, developing a similarity or distance measure between networks so that meaningful comparisons can be drawn. The best means to accomplish this task remains an open area of research. Here we introduce a new measure to compare networks, the Network Portrait Divergence, that is mathematically principled, incorporates the topological characteristics of networks at all structural scales, and is general-purpose and applicable to all types of networks. An important feature of our measure that enables many of its useful properties is that it is based on a graph invariant, the network portrait. We test our measure on both synthetic graphs and real world networks taken from protein interaction data, neuroscience, and computational social science applications. The Network Portrait Divergence reveals important characteristics of multilayer and temporal networks extracted from data.

Keywords: Network comparison, Graph similarity, Multilayer networks, Temporal networks, Weighted networks, Network portraits, GitHub, Arabidopsis, C. Elegans connectome

Introduction

Recent years have seen an explosion in the breadth and depth of network data across a variety of scientific domains (Bader et al. 2004; Lazer et al. 2009; Landhuis 2017). This scope of data challenges researchers, and new tools and techniques are necessary for evaluating and understanding networks. It is now increasingly common to deal with multiple networks at once, from brain networks taken from multiple subjects or across longitudinal studies (Whelan et al. 2012), to multilayer networks extracted from high-throughput experiments (De Domenico et al. 2015), to rapidly evolving social network data (Palla et al. 2007; Szell et al. 2010; Bagrow et al. 2011). A common task researchers working in these areas will face is comparing networks, quantifying the similarities and differences between networks in a meaningful manner. Applications for network comparison include comparing brain networks for different subjects, or the same subject before and after a treatment, studying the evolution of temporal networks (Holme and Saramäki 2012), classifying proteins and other sequences (Shervashidze et al. 2009; Yanardag and Vishwanathan 2015; Niepert et al. 2016), classifying online social networks (Yanardag and Vishwanathan 2015), or evaluating the accuracy of statistical or generative network

models (Hunter et al. 2008). Combined with a clustering algorithm, a network comparison measure can be used to aggregate networks in a meaningful way, for coarse-graining data and revealing redundancy in multilayer networks (De Domenico et al. 2015). Treating a network comparison measure as an objective function, optimization methods can be used to fit network models to data.

Approaches to network comparison can be roughly divided into two groups, those that consider or require two graphs defined on the same set of nodes, and those that do not. The former eliminates the need to discover a mapping between node sets, making comparison somewhat easier. Yet, two networks with identical topologies may have no nodes or edges in common simply because they are defined on different sets of nodes. While there are scenarios where assuming the same node sets is appropriate—for example, when comparing the different layers of a multilayer network one wants to capture explicitly the correspondences of nodes between layers (De Domenico et al. 2015)—here we wish to relax this assumption and allow for comparison without node correspondence, where no nodes are necessarily shared between the networks.

A common approach for comparison without assuming node correspondence is to build a comparison measure using a *graph invariant*. Graph invariants are properties of a graph that hold for all isomorphs of the graph. Using an invariant mitigates any concerns with the encoding or structural representation of the graphs, and the comparison measure is instead focused entirely on the topology of the network. Graph invariants may be probability distributions. Suppose P and Q represent two graph-invariant distributions corresponding to graphs G_1 and G_2, respectively. Then, a common approach to comparing G_1 and G_2 is by comparing P and Q. Information theory provides tools for comparing distributions, such as the Jensen-Shannon divergence:

$$D_{JS}(P, Q) = \frac{1}{2} KL(P \,||\, M) + \frac{1}{2} KL(Q \,||\, M) \tag{1}$$

where $KL(P \,||\, Q)$ is the Kullback-Leibler (KL) divergence (or relative entropy) between P and Q and $M = (P + Q)/2$ is the mixture distribution of P and Q. The Jensen-Shannon divergence has a number of nice properties, including that it is symmetric and normalized, making it a popular choice for applications such as ours (De Domenico and Biamonte 2016; Chen et al. 2018). In this work, we introduce a novel graph-invariant distribution that is general and free of assumptions and we can then use common information-theoretic divergences such as Jensen-Shannon to compare networks via these graph invariants.

The rest of this paper is organized as follows. In "Network portraits" section we describe network portraits (Bagrow et al. 2008), a graph invariant matrix representation of a network that is useful for visualization purposes but also capable of comparing pairs of networks. "An information-theoretic approach to network comparison" section introduces Network Portrait Divergences, a principled information-theoretic measure for comparing networks, building graph-invariant distributions using the information contained within portraits. Network Portrait Divergence has a number of desirable properties for a network comparison measure. In "Results" section we apply this measure to both synthetic networks (random graph ensembles) and real-world datasets (multilayer biological and temporal social networks), demonstrating its effectiveness on practical problems of interest. Lastly, we conclude in "Conclusion" section with a discussion of our results and future work.

Network portraits

Network portraits were introduced in (Bagrow et al. 2008) as a way to visualize and encode many structural properties of a given network. Specifically, the network portrait B is the array with (ℓ, k) elements

$$B_{\ell,k} \equiv \text{the number of nodes who have } k \text{ nodes at distance } \ell \qquad (2)$$

for $0 \leq \ell \leq d$ and $0 \leq k \leq N - 1$, where distance is taken as the shortest path length and d is the graph's diameter [1]. The elements of this array are computed using, e.g., Breadth-First Search. Crucially, no matter how a graph's nodes are ordered or labeled the portrait is identical. We draw several example networks and their corresponding portraits in Fig. 1.

This matrix encodes many structural features of the graph. The zeroth row stores the number of nodes N in the graph:

$$B_{0,k} = N\delta_{k,1}.$$

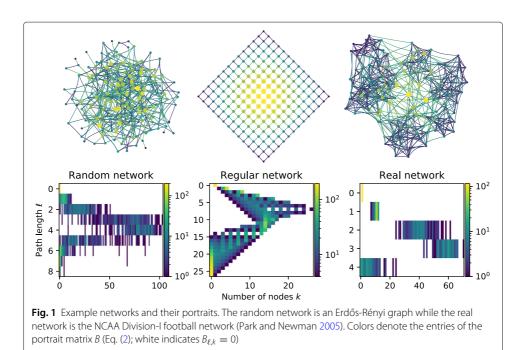The first row captures the degree distribution $P(k)$:

$$B_{1,k} = NP(k),$$

as neighbors are at distance $\ell = 1$. The second row captures the distribution of next-nearest neighbors, and so forth for higher rows. The number of edges M is $\sum_{k=0}^{N} kB_{1,k} = 2M$. The graph diameter d is

$$d = \max\{\ell \mid B_{\ell,k} > 0 \text{ for } k > 0\}.$$

The shortest path distribution is also captured: the number of shortest paths of length ℓ is $\frac{1}{2}\sum_{k=0}^{N} kB_{\ell,k}$. And the portraits of random graphs present very differently from highly ordered structures such as lattices (Fig. 1), demonstrating how dimensionality and regularity of the network is captured in the portrait (Bagrow et al. 2008).

One of the most important properties of portraits is that they are a graph invariant:

Fig. 1 Example networks and their portraits. The random network is an Erdős-Rényi graph while the real network is the NCAA Division-I football network (Park and Newman 2005). Colors denote the entries of the portrait matrix B (Eq. (2); white indicates $B_{\ell,k} = 0$)

Definition 1 *A **graph invariant** is a property of a graph that is invariant under graph isomorphism, i.e., it is a function f such that $f(G) = f(H)$ whenever G and H are isomorphic graphs.*

Theorem 1 *The network portrait (Eq. (2)) is a graph invariant.*

Proof Let $f : V_G \to V_H$ be a vertex bijection between two graphs $G = (V_G, E_G)$ and $H = (V_H, E_H)$ such that the number of edges between every pair of vertices (i, j) in G equals the number of edges between their images $(f(i), f(j))$ in H. Then G and H are isomorphic. Let $\ell_G(i, j)$ be the length of the shortest path between nodes i and j in G. For two isomorphic graphs G and H, $\ell_G(i, j) = \ell_H(f(i), f(j))$ for all i and j in G, since the shortest path tuples $(i, \ldots, j)$ in G and $(f(i), \ldots, f(j))$ in H are the same length. All elements in the matrix $B(G)$ are computed by aggregating the values of $\ell_G(i, j)$. Therefore, $B(G) = B(H)$.   □

Note that the converse is not necessarily true: that $f(G) = f(H)$ does not imply that G and H are isomorphic. As a counter-example, the non-isomorphic distance-regular dodecahedral and Desargues graphs have equal portraits (Bagrow et al. 2008).

Portraits of weighted networks The original work defining network portraits (Bagrow et al. 2008) did not consider weighted networks, where a scalar quantity w_{ij} is associated with each $(i, j) \in E$. An important consideration is that path lengths for weighted networks are generally computed by summing edge weights along a path, leading to path lengths $\ell \in \mathbb{R}$ (typically) instead of path lengths $\ell \in \mathbb{Z}$. To address this, in the Additional file 1 we generalize the portrait to weighted networks, specifically accounting for how real-valued path lengths must change the definition of the matrix B.

Comparing networks by comparing portraits

Given that a graph G admits a unique B-matrix makes these portraits a valuable tool for network comparison. Instead of directly comparing graphs G and G', we may compute their portraits B and B', respectively, and then compare these matrices. We review the comparison method in our previous work (Bagrow et al. 2008). First, compute for each portrait B the matrix C consisting of row-wise cumulative distributions of B:

$$C_{\ell,k} = \sum_{j=0}^{k} B_{\ell,j} \bigg/ \sum_{j=0}^{N} B_{\ell,j} . \tag{3}$$

The row-wise Kolmogorov-Smirnov test statistic K_ℓ between corresponding rows ℓ in C and C':

$$K_\ell = \max \left| C_{\ell,k} - C'_{\ell,k} \right|$$

allows a metric-like graph comparison. This statistic defines a two-sample hypothesis test for whether or not the corresponding rows of the portraits are drawn from the same underlying, unspecified distribution. If the two graphs have different diameters, the portrait for the smaller diameter graph can be expanded to the same size as the larger diameter graph by defining empty shells $\ell > d$ as $B_{\ell,k} = N\delta_{0,k}$. Lastly, aggregate the test statistics for all pairs of rows using a weighted average to define the similarity $\Delta(G, G')$

between G and G':

$$\Delta(G, G') \equiv \Delta(B, B') = \frac{\sum_\ell \alpha_\ell K_\ell}{\sum_\ell \alpha_\ell}, \tag{4}$$

where

$$\alpha_\ell = \sum_{k>0} B_{\ell,k} + \sum_{k>0} B'_{\ell,k} \tag{5}$$

is a weight chosen to increase the impact of the lower, more heavily occupied shells.

While we did develop a metric-like quantity for comparing graphs based on the KS-statistics (Eqs. (4) and (5)), we did not emphasize the idea. Instead, the main focus of the original portraits paper was on the use of the portrait for visualization. In particular, Eq. (4) is somewhat ad hoc. Here we now propose a stronger means of comparison using network portraits that is interpretable and grounded in information theory.

An information-theoretic approach to network comparison

Here we introduce a new way of comparing networks based on portraits. This measure is grounded in information theory, unlike the previous, ad hoc comparison measure, and has a number of other desirable attributes we discuss below.

The rows of B may be interpreted as probability distributions:

$$P(k \mid \ell) = \frac{1}{N} B_{\ell,k} \tag{6}$$

is the (empirical) probability that a randomly chosen node will have k nodes at distance ℓ. This invites an immediate comparison *per row* for two portraits:

$$KL(P(k \mid \ell) \,\|\, Q(k \mid \ell)) = \sum_k P(k \mid \ell) \log \frac{P(k \mid \ell)}{Q(k \mid \ell)}, \tag{7}$$

where $KL(p \,\|\, q)$ is the Kullback-Liebler (KL) divergence between two distributions p and q, and Q is defined as per Eq. (6) for the second portrait (i.e., $Q(k \mid \ell) = \frac{1}{N'} B'_{\ell,k}$). The KL-divergence admits an information-theoretic interpretation that describes how many extra bits are needed to encode values drawn from the distribution P if we used the distribution Q to develop the encoding instead of P.

However, while this seems like an appropriate starting point for defining a network comparison, Eq. (7) has some drawbacks:

1   $KL(P(k) \,\|\, Q(k))$ is undefined if there exists a value of k such that $P(k) > 0$ and $Q(k) = 0$. Given that rows of the portraits are computed from individual networks, which may have small numbers of nodes, this is likely to happen often in practical use.

2   The KL-divergence is not symmetric and does not define a distance.

3   Defining a divergence for each pair of rows of the two matrices gives $\max(d, d') + 1$ separate divergences, where d and d' are the diameters of G and G', respectively. To define a scalar comparison value (a similarity or distance measure) requires an appropriate aggregation of these values, just like the original approach proposed in (Bagrow et al. 2008); we return to this point below.

The first two drawbacks can be addressed by moving away from the KL-divergence and instead using, e.g., the Jensen-Shannon divergence or Hellinger distance. However, the

last concern, aggregating over $\max(d, d') + 1$ difference quantities, remains for those measures as well.

Given these concerns, we propose the following, utilizing the shortest path distribution encoded by the network portraits. Consider choosing two nodes uniformly at random with replacement. The probability that they are connected is

$$\frac{\sum_c n_c^2}{N^2}, \tag{8}$$

where n_c is the number of nodes within connected component c, the sum $\sum_c n_c^2$ runs over the number of connected components, and the n_c satisfy $\sum_c n_c = N$. Likewise, the probability the two nodes are at a distance ℓ from one another is

$$\frac{\#\text{ paths of length }\ell}{\#\text{ paths}} = \frac{1}{\left(\sum_c n_c^2\right)} \sum_{k=0}^{N} k B_{\ell,k}. \tag{9}$$

Lastly, the probability that one of the two nodes has $k - 1$ other nodes at distance ℓ is given by

$$\frac{k B_{\ell,k}}{\sum_{k'} k' B_{\ell,k'}}. \tag{10}$$

We propose to combine these probabilities into a single distribution that encompasses the distances between nodes weighted by the "masses" or prevalences of other nodes at those distance, giving us the probability for choosing a pair of nodes at distance ℓ and for one of the two randomly chosen nodes to have k nodes at that distance ℓ:

$$P(k, \ell) = \frac{\sum_c n_c^2}{N^2} \frac{\sum_{k'} k' B_{\ell,k'}}{\sum_c n_c^2} \frac{k B_{\ell,k}}{\sum_{k'} k' B_{\ell,k'}} = \frac{k B_{\ell,k}}{N^2} \tag{11}$$

and likewise for $Q(k, \ell)$ using B' instead of B. However, this distribution is not normalized

$$\sum_k \sum_\ell k B_{\ell,k} = \sum_c n_c^2 \neq N^2 \tag{12}$$

unless the graph G is connected. It will be advantageous for this distribution to be normalized in all instances, therefore, we condition this distribution on the two randomly chosen nodes being connected:

$$P(k, \ell) = \frac{\sum_{k'} k' B_{\ell,k'}}{\sum_c n_c^2} \frac{k B_{\ell,k}}{\sum_{k'} k' B_{\ell,k'}} = \frac{k B_{\ell,k}}{\sum_c n_c^2}. \tag{13}$$

This now defines a single (joint) distribution P (Q) for all rows of B (B') which can then be used to define a single KL-divergence between two portraits:

$$KL(P(k, \ell) \,||\, Q(k, \ell)) = \sum_{\ell=0}^{\max(d,d')} \sum_{k=0}^{N} P(k, \ell) \log \frac{P(k, \ell)}{Q(k, \ell)} \tag{14}$$

where the log is base 2.

Definition 2 *The **Network Portrait Divergence** $D_{JS}(G, G')$ between two graphs G and G' is the Jensen-Shannon divergence as follows,*

$$D_{JS}(G, G') \equiv \frac{1}{2} KL(P \,||\, M) + \frac{1}{2} KL(Q \,||\, M) \tag{15}$$

where $M = \frac{1}{2}(P + Q)$ is the mixture distribution of P and Q. Here P and Q are defined by Eq. (11), and the $KL(\cdot \,||\, \cdot)$ is given by Eq. (14).

The Network Portrait Divergence $0 \leq D_{JS} \leq 1$ provides a single value to quantify the dissimilarity of the two networks by means of their distance distributions, with smaller D_{JS} for more similar networks and larger D_{JS} for less similar networks. Unlike the KL divergence, D_{JS} is symmetric, $D_{JS}(G, G') = D_{JS}(G', G)$ and $\sqrt{D_{JS}}$ is a metric (Endres and Schindelin 2003).

The Network Portrait Divergence has a number of desirable properties[2]. It is grounded in information theory, which provides principled interpretation of the divergence measure. It compares networks based entirely on the structure of their respective topologies: the measure is independent of how the nodes in the network are indexed and, further, does not assume the networks are defined on the same set of nodes. Network Portrait Divergence is relatively computationally efficient; unlike graph edit distance measures, for example, because Network Portrait Divergence is based on a graph invariant and expensive optimizations such as "node matching" are not needed. Both undirected and directed networks are treated naturally, and disconnected networks can be handled without any special problems. Using the generalization of network portraits to weighted networks (see Additional file 1), the Network Portrait Divergence can also be used to compare weighted networks. Lastly, all scales of structure within the two networks contribute simultaneously to the Network Portrait Divergence via the joint neighbor-shortest path length distribution (Eq. (11)), from local structure to motifs to the large scale connectivity patterns of the two networks.
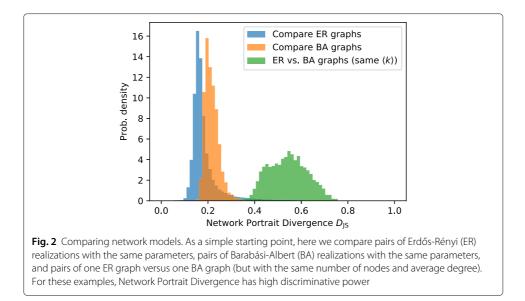
Results

Now we explore the use of the Network Portrait Divergence (Definition 2) to compare networks across a variety of applications. We study both synthetic example graphs, to benchmark the behavior of the comparison measure under idealized conditions. Then real world network examples are presented to better capture the types of comparison tasks researchers may encounter.

Synthetic networks

To understand the performance of the Network Portrait Divergence, we begin here by examining how it relates different realizations of the following synthetic graphs:

1 Erdős-Rényi (ER) graphs $G(N, p)$ (Erdös and Rényi 1959), the random graph on N nodes where each possible edge exists independently with constant probability p;

2 Barabási-Albert (BA) graphs $G(N, m)$ (Barabási and Albert 1999), where N nodes are added sequentially to a seed graph and each new node attaches to m existing nodes according to preferential attachment.

Figure 2 shows the distributions of D_{JS} for different realizations of ER and BA graphs with the same parameters, as well as the cross-comparison distribution using D_{JS} to compare one realization of an ER graph with one realization of a BA graph where both graphs have the same average degree $\langle k \rangle$ but other parameters may vary. Overall we note that realizations drawn from the same ensemble have relatively small Network Portrait Divergence, with ER falling roughly in the range $0 < D_{JS} < 0.5$ and BA between $0.1 < D_{JS} < 0.4$ (both smaller than the $\max(D_{JS}) = 1$). In contrast, D_{JS} is far higher when comparing one ER to one BA graph, with $D_{JS} > 0.6$ in our simulations. The Network Portrait Divergence captures the smaller differences between different realizations of a single

Fig. 2 Comparing network models. As a simple starting point, here we compare pairs of Erdős-Rényi (ER) realizations with the same parameters, pairs of Barabási-Albert (BA) realizations with the same parameters, and pairs of one ER graph versus one BA graph (but with the same number of nodes and average degree). For these examples, Network Portrait Divergence has high discriminative power

networks ensemble with correspondingly smaller values of D_{JS} while the larger differences between networks from different ensembles are captured with larger values of D_{JS}.
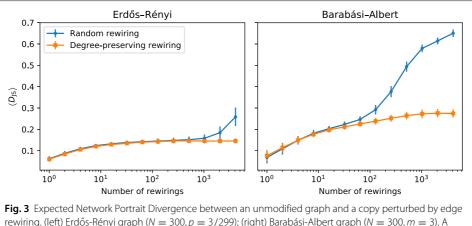
Measuring network perturbations with Network Portrait Divergence

Next, we ask how well the Network Portrait Divergence measures the effects of network perturbations. We performed two kinds of rewiring perturbations to the links of a given graph G: (i) *random rewiring*, where each perturbation consists of deleting an existing link chosen uniformly at random and inserting a link between two nodes chosen uniformly at random; and (ii) *degree-preserving rewirings* (Li et al. 2018), where each perturbation consists of removing a randomly chosen pair of links (i, j) and (u, v) and inserting links (i, u) and (j, v). The links (i, j) and (u, v) are chosen such that $(i, u) \notin E$ and $(j, v) \notin E$, ensuring that the degrees of the nodes are constant under the rewiring.

We expect that random rewirings will lead to a stronger change in the network than the degree-preserving rewiring. To test this, we generate an ER or BA graph G, apply a fixed number n of rewirings to a copy of G, and use the Network Portrait Divergence to compare the networks before and after rewirings. Figure 3 shows how D_{JS} changes on average as a function of the number of rewirings, for both types of rewirings and both ER and BA graphs. The Network Portrait Divergence increases with n, as expected. Interestingly, below $n \approx 100$ rewirings, the different types of rewirings are indistinguishable, but for $n > 100$ we see that random rewirings lead to a larger divergence from the original graph than degree-preserving rewirings. This is especially evident for BA graphs, where the scale-free degree distribution is more heavily impacted by the random rewiring than for ER graphs. The overall D_{JS} is also higher in value for BA graphs than ER graphs. This is plausible because the ER graph is already maximally random, whereas many correlated structures exist in a random realization of the BA graph model that can be destroyed by perturbations (Albert and Barabási 2002).

Comparing real networks

We now apply the Network Portrait Divergence to real world networks, to evaluate its performance when used for several common network comparison tasks. Specifically, we

Fig. 3 Expected Network Portrait Divergence between an unmodified graph and a copy perturbed by edge rewiring. (left) Erdős-Rényi graph ($N = 300, p = 3/299$); (right) Barabási-Albert graph ($N = 300, m = 3$). A random rewiring is the deletion of an edge chosen uniformly at random followed by the insertion of a new edge between two nodes chosen uniformly at random. Degree-preserving rewiring chooses a pair of edges (u, v) and (x, y) and rewires them across nodes to (u, x) and (v, y) such that the degrees of the chosen nodes remain unchanged (Li et al. 2018). Errorbars denote ± 1 s.d

study two real-world multiplex networks, using D_{JS} to compare across the layers of these networks. We also apply D_{JS} to a temporal network, measuring how the network changes over time. This last network has associated edge weights, and we consider it as both an unweighted and a weighted network.

The datasets for the three real-world networks we study are as follows:
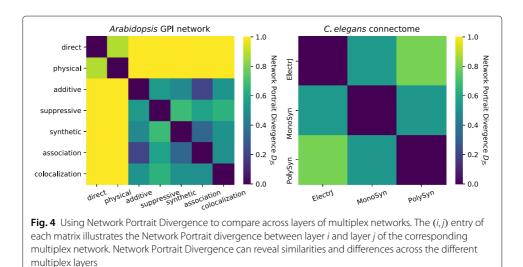
***Arabidopsis* GPI network** The Genetic and Protein Interaction (GPI) network of *Arabidopsis Thaliana* taken from BioGRID 3.2.108 (Stark et al. 2006; De Domenico et al. 2015). This network consists of 6,980 nodes representing proteins and 18,654 links spread across seven multiplex layers. These layers represent different interaction modes from direct interaction of protein and physical associations of proteins within complexes to suppressive and synthetic genetic interactions. Full details of the interaction layers are described in (De Domenico et al. 2015).

***C. elegans* connectome** The network taken from the nervous system of the nematode *C. elegans*. This multiplex network consists of 279 nodes representing non-pharyngeal neurons and 5,863 links representing electrical junctions and chemical synapses. Links are spread across three layers: an electric junction layer (17.6% of links), a monadic chemical synapse layer (28.0% of links), and a polyadic chemical synapse layer (54.4% of links) (Chen et al. 2006; De Domenico et al. 2015). The first layer represents electrical coupling via gap junctions between neurons, while links in the other layers represent neurons coupled by neurotransmitters. *C. elegans* has many advantages as a model organism in general (Brenner 1974; White et al. 1986), and its neuronal wiring diagram is completely mapped experimentally (White et al. 1986; Hall et al. 2008), making its connectome an ideal test network dataset.

Open source developer collaboration network This network represents the software developers working on open source projects hosted by IBM on GitHub (https://github.com/ibm). This network is temporal, evolving over the years 2013-2017, allowing us to compare its development across time. Aggregating all activity, this network consists of 679 nodes and 3,628 links. Each node represents a developer

who has contributed to the source code of the project, as extracted from the git metadata logs (Bird et al. 2009; Kalliamvakou et al. 2014; Klug and Bagrow 2016). Links occur between developers who have edited at least one source code file in common, a simple measure of collaboration. To study this network as a *weighted network*, we associate with each link (i, j) an integer weight w_{ij} equal to the number of source files edited in common by developers i and j.

For these data, the Network Portrait Divergence reveals several interesting facets of the multilayer structure of the Arabidopsis network (Fig. 4). First, both the direct interaction and physical association layers are the most different, both from each other and from the other layers, with the synthetic interactions layer being most different from the direct and physical layers. The remaining five layers show interesting relationships, with suppressive and synthetic genetic interactions being the most distinct pair ($D_{\mathrm{JS}} = 0.553$) among these five layers. The additive and suppressive layers were also distinct ($D_{\mathrm{JS}} = 0.365$). De Domenico et al. observed a similar distinction between the suppressive and synthetic layers (De Domenico et al. 2015).

The multilayer *C. elegans* network, consisting of only three layers, is easier to understand than *Arabidopsis*. Here we find that the electrical junction layer is more closely related to the monadic synapse layer than it is to the polyadic synapse layer, while the polyadic layer is more closely related to the monadic synapse layer than to the electrical junction layer. The *C. elegans* data aggregated all polyadic synapses together into one layer accounting for over half of the total links in the network, but it would be especially interesting to determine what patterns for dyadic, triadic, etc. synapses can be revealed with the Network Portrait Divergence.

The third real-world network we investigate is a temporal network (Fig. 5). This network encodes the collaboration activities between software developers who have contributed to open source projects owned by IBM on GitHub.com. Here each node represents a developer and a links exist between two developers when they have both edited at least one file in common among the source code hosted on GitHub. This network is growing over time as more projects are open-sourced by IBM, and more developers join and contribute to those projects. We draw the IBM developer network for each year from 2013 to 2017 in
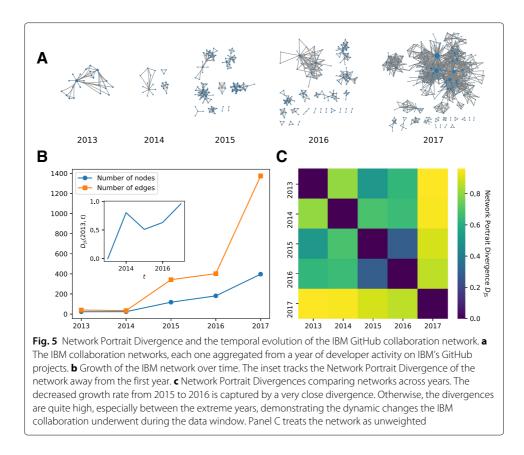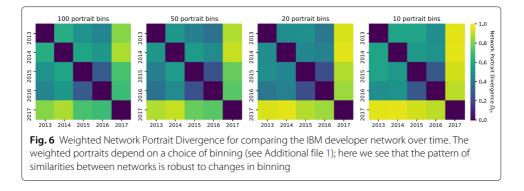

Fig. 4 Using Network Portrait Divergence to compare across layers of multiplex networks. The (i, j) entry of each matrix illustrates the Network Portrait divergence between layer i and layer j of the corresponding multiplex network. Network Portrait Divergence can reveal similarities and differences across the different multiplex layers

Fig. 5 Network Portrait Divergence and the temporal evolution of the IBM GitHub collaboration network. **a** The IBM collaboration networks, each one aggregated from a year of developer activity on IBM's GitHub projects. **b** Growth of the IBM network over time. The inset tracks the Network Portrait Divergence of the network away from the first year. **c** Network Portrait Divergences comparing networks across years. The decreased growth rate from 2015 to 2016 is captured by a very close divergence. Otherwise, the divergences are quite high, especially between the extreme years, demonstrating the dynamic changes the IBM collaboration underwent during the data window. Panel C treats the network as unweighted

Fig. 5a, while Fig. 5b shows the change in size of these networks over time. Lastly, Fig. 5c demonstrates how D_{JS} captures patterns in the temporal evolution of the network, in particular revealing the structural similarity between 2015 and 2016, a period that showed a distinct slowdown in network growth compared with prior and subsequent years.

Figure 6 shows the Network Portrait Divergence comparing different years of the IBM developer network, as per Fig. 5 but now accounting for edge weights. Shortest path lengths were found using Dijkstra's algorithm based on reciprocal edge weights (see Additional file 1 for details). The weighted portraits require binning the path length distributions (see Additional file 1). Here we show four such binnings, based on quantiles of the weighted path length distributions, from $b = 100$ bins (each bin accounts for 1% of shortest paths) to $b = 10$ (each bin accounts for 10% of shortest paths). Note that each D_{JS} is defined with its own portrait binning, as the quantiles of $\mathcal{L} = \mathcal{L}(G) \cup \mathcal{L}(G')$ may vary across different pairs of networks, where $\mathcal{L}(G) = \{\ell_{ij} \mid i,j \in V \wedge \ell_{ij} < \infty\}$ is the set of all unique shortest path lengths in G. Overall, the relationships between the different time periods of the network do not depend strongly on the choice of binning, and we capture patterns across time similar to, though not identical to, the patterns found analyzing the unweighted networks (shown in Fig. 5).

Conclusion

In this paper we have introduced a measure, the Network Portrait Divergence, for comparing networks, and validated its performance on both synthetic and real-world network data. Network Portrait Divergence provides an information-theoretic interpretation that

Fig. 6 Weighted Network Portrait Divergence for comparing the IBM developer network over time. The weighted portraits depend on a choice of binning (see Additional file 1); here we see that the pattern of similarities between networks is robust to changes in binning

naturally encompasses all scales of structure within networks. It does not require the networks to be connected, nor does it make any assumptions as to how the two networks being compared are related, or indexed, or even that their node sets are equal. Further, Network Portrait Divergence can naturally handle both undirected and directed, unweighted networks, and we have introduced a generalization for weighted networks. The Network Portrait Divergence is based on a graph invariant, the network portrait. Comparison measures based on graph invariants are desirable as they will only be affected by the topology of the networks being studied, and not other externalities such as the format or order in which the networks are recorded or analyzed. The computational complexity of the Network Portrait Divergence compares favorably to many other graph comparison measures, particularly spectral measures, but it remains a computation that is quadratic in the number of nodes of the graph. To scale to very large networks will likely require further efficiency gains, probably from approximation strategies to efficiently infer the shortest path distributions (Potamias et al. 2009).

Our approach bears superficial similarities with other methods. Graph distances and shortest path length distributions are components common to many network comparison methods, including our own, although the Network Portrait Divergences utilizes a unique composition of all the shortest path length distributions for the networks being compared. At the same time, other methods, including ours, use the Jensen-Shannon divergence to build comparison measures. For example, the recent work of Chen et al. (2018) uses the Shannon entropy and Jensen-Shannon Divergence of a probability distribution computed by normalizing e^A, the exponential of the adjacency matrix also known as the communicability matrix. This is an interesting approach, as are other approaches that examine powers of the adjacency matrix, but it suffers from a drawback: when comparing networks of different sizes, the underlying probability distributions must be modified in an ad hoc manner (Chen et al. 2018). The Network Portrait Divergence, in contrast, does not need such modification.

The Network Portrait Divergence, and other methods, is based upon the Jensen-Shannon divergence between graph-invariant probability distributions, but many other information-theoretic tools exist for comparing distributions, including f-divergences such as the Hellinger distance or total variation distance, Bhattacharyya distance, and more. Using different measures for comparison may yield different interpretations and insights, and thus it is fruitful to better understand their use across different network comparison problems.

Network Portrait divergence lends itself well to developing statistical procedures when combined with suitable graph null models. For example, one could quantify the randomness of a structural property of a network by comparing the real network to a random model that controls for that property. Further, to estimate the strength or significance of an observed divergence between graphs G_1 and G_2, one could generate a large number of random graph null proxies for either G_1 or G_2 (or both) and compare the divergences found between those nulls with the divergence between the original graphs. These comparisons could be performed using a Z-test or other statistical procedure, as appropriate. Exploring this and other avenues for Network Portrait Divergence-based statistical procedures is a fruitful direction for future work.

In general, because of the different potential interpretations and insights that researchers performing network comparison can focus on, the network comparison problem lacks quantitative benchmarks. These benchmarks are useful for comparing different approaches systematically. However, the comparison problem is not as narrowly defined as, for example, graph partitioning, and thus effective methods may highlight very different facets of comparison. While specific benchmarks can be introduced for specific facets, due to a lack of standardized, systematic benchmarks, most researchers introducing new comparison measures focus on a few tasks of interest, as we do here. Bringing clarity to the literature by defining and validating an appropriate benchmarking suite would be a valuable contribution to the network comparison problem, but we consider this to be beyond the scope of our current work. Instead, we have focused our method on highlighting several areas, particularly within real world applications but also using some intuitive synthetic scenarios, where the method is effective.

As network datasets increase in scope, network comparison becomes an increasingly common and important component of network data analysis. Measures such as the Network Portrait Divergence have the potential to help researchers better understand and explore the growing sets of networks within their possession.

Endnotes

[1] Note that a distance $\ell = 0$ is admissible, with two nodes i and j at distance 0 when $i = j$. This means that the matrix B so defined has a zeroth row. It also has a zeroth column, as there may be nodes that have zero nodes at some distance ℓ. This occurs for nodes with eccentricity less than the graph diameter.

[2] Code implementing Network Portrait Divergence is available at https://github.com/bagrow/network-portrait-divergence/.

Additional file

Additional file 1: Generalization of network portraits and portrait divergences to weighted networks. (PDF 501 kb)

Author details
[1]Department of Mathematics & Statistics, Vermont Complex Systems Center, University of Vermont, Burlington, VT, USA.
[2]Department of Mathematics, Clarkson University, Potsdam, NY, USA.

References

Albert R, Barabási A.-L (2002) Statistical mechanics of complex networks. Rev Mod Phys 74(1):47

Bader JS, Chaudhuri A, Rothberg JM, Chant J (2004) Gaining confidence in high-throughput protein interaction networks. Nat Biotechnol 22(1):78

Bagrow JP, Bollt EM, Skufca JD, Ben-Avraham D (2008) Portraits of complex networks. EPL (Europhys Lett) 81(6):68004

Bagrow JP, Wang D, Barabasi A-L (2011) Collective response of human populations to large-scale emergencies. PLoS ONE 6(3):17680

Barabási A-L, Albert R (1999) Emergence of scaling in random networks. Science 286(5439):509–512

Bird C, Rigby PC, Barr ET, Hamilton DJ, German DM, Devanbu P (2009) The promises and perils of mining git. In: Mining Software Repositories, 2009. MSR'09. 6th IEEE International Working Conference On. IEEE, New York. pp 1–10

Brenner S (1974) The genetics of caenorhabditis elegans. Genetics 77(1):71–94

Chen BL, Hall DH, Chklovskii DB (2006) Wiring optimization can relate neuronal structure and function. Proc Natl Acad Sci USA 103(12):4723–4728

Chen D, Shi D-D, Qin M, Xu S-M, Pan G-J (2018) Complex network comparison based on communicability sequence entropy. Phys Rev E 98(1):012319

De Domenico M, Biamonte J (2016) Spectral entropies as information-theoretic tools for complex network comparison. Phys Rev X 6:041062

De Domenico M, Nicosia V, Arenas A, Latora V (2015) Structural reducibility of multilayer networks. Nat Commun 6:6864

De Domenico M, Porter MA, Arenas A (2015) MuxViz: a tool for multilayer analysis and visualization of networks. J Complex Netw 3(2):159–176

Endres DM, Schindelin JE (2003) A new metric for probability distributions. IEEE Trans Inf Theory 49(7):1858–1860

Erdös P, Rényi A (1959) On random graphs, I. Publ Math Debr 6:290–297

Hall DH, Altun ZF, et al (2008) C. Elegans Atlas. Cold Spring Harbor Laboratory Press, New York

Holme P, Saramäki J (2012) Temporal networks. Phys Rep 519(3):97–125

Hunter DR, Goodreau SM, Handcock MS (2008) Goodness of fit of social network models. J Am Stat Assoc 103(481):248–258

Kalliamvakou E, Gousios G, Blincoe K, Singer L, German DM, Damian D (2014) The promises and perils of mining github. In: Proceedings of the 11th Working Conference on Mining Software Repositories. ACM, New York. pp 92–101

Klug M, Bagrow JP (2016) Understanding the group dynamics and success of teams. R Soc Open Sci 3(4):160007

Landhuis E (2017) Neuroscience: Big brain, big data. Nature 541:559–561

Lazer D, Pentland AS, Adamic L, Aral S, Barabasi AL, Brewer D, Christakis N, Contractor N, Fowler J, Gutmann M, et al (2009) Life in the network: the coming age of computational social science. Sci (NY) 323(5915):721

Li Z, Mucha PJ, Taylor D (2018) Network-ensemble comparisons with stochastic rewiring and *von Neumann* entropy. SIAM J Appl Math 78(2):897-920. https://doi.org/10.1137/17M1124218

Park J, Newman ME (2005) A network-based ranking system for us college football. J Stat Mech Theory Exp 2005(10):10014

Potamias M, Bonchi F, Castillo C, Gionis A (2009) Fast shortest path distance estimation in large networks. In: Proceedings of the 18th ACM Conference on Information and Knowledge Management. ACM, New York. pp 867–876

Niepert M, Ahmed M, Kutzkov K (2016) Learning convolutional neural networks for graphs. In: Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48, ICML'16. JMLR.org, New York. pp 2014–2023. http://dl.acm.org/citation.cfm?id=3045390.3045603

Palla G, Barabási A-L, Vicsek T (2007) Quantifying social group evolution. Nature 446(7136):664

Shervashidze N, Vishwanathan S, Petri T, Mehlhorn K, Borgwardt K (2009) Efficient graphlet kernels for large graph comparison. In: Artificial Intelligence and Statistics. PMLR, Florida. pp 488–495

Stark C, Breitkreutz B-J, Reguly T, Boucher L, Breitkreutz A, Tyers M (2006) BioGRID: a general repository for interaction datasets. Nucleic Acids Res 34(suppl_1):535–539

Szell M, Lambiotte R, Thurner S (2010) Multirelational organization of large-scale social networks in an online world. Proc
    Natl Acad Sci USA 107(31):13636–13641
Whelan R, Conrod PJ, Poline J-B, Lourdusamy A, Banaschewski T, Barker GJ, Bellgrove MA, Büchel C, Byrne M, Cummins
    TD, et al (2012) Adolescent impulsivity phenotypes characterized by distinct brain networks. Nat Neurosci 15(6):920
White JG, Southgate E, Thomson JN, Brenner S (1986) The structure of the nervous system of the nematode
    caenorhabditis elegans. Philos Trans R Soc Lond B Biol Sci 314(1165):1–340
Yanardag P, Vishwanathan S (2015) Deep graph kernels. In: Proceedings of the 21th ACM SIGKDD International
    Conference on Knowledge Discovery and Data Mining. ACM, New York. pp 1365–1374

Publisher's Note