

# Comparative Network Analysis Using KronFit

Gupta Sukrit, Puzis Rami and Kilimnik Konstantin

**Abstract** Comparative network analysis is an emerging line of research that provides insights into the structure and dynamics of networks by finding similarities and discrepancies in their topologies. Unfortunately, comparing networks directly is not feasible on large scales. Existing works resort to representing networks with vectors of features extracted from their topologies and employ various distance metrics to compare between these feature vectors. In this paper, instead of relying on feature vectors to represent the studied networks, we suggest fitting a network model (such as Kronecker Graph) to encode the network structure. We present the directed *fitting-distance* measure, where the distance from a network  $A$  to another network  $B$  is captured by the quality of  $B$ 's fit to the model derived from  $A$ . Evaluation on five classes of real networks shows that KronFit based distances perform surprisingly well.

**Keywords** Complex networks · Comparative analysis · Generative models · Distance metrics

## 1 Introduction

Comparative network analysis and network classification on the basis of structural similarity are at a nascent stage holding great potential. The topology of a network often encompasses important information on the functionality and dynamics of the system it represents. As case in point, structural similarity of road networks and fungal networks are the result of low cost and robustness being the main driving

---

G. Sukrit (✉) · P. Rami · K. Konstantin  
Department of Information Systems Engineering,  
Ben Gurion University of the Negev, 8410501 Beersheba, Israel  
e-mail: gupta@post.bgu.ac.il

P. Rami  
e-mail: puzis@bgu.ac.il

K. Konstantin  
e-mail: kilimnik@post.bgu.ac.il

forces in the network development [1]. Evaluation of network similarity is important in diverse research fields, particularly in computational biology, where it reveals previously unknown interactions and biological function of protein complexes.

So far network similarity does not have a concrete, rigorous definition. Therefore, researchers employ vectors of features extracted from the networks at various scales to compare networks. Existing features range from microscopic properties that describe interactions between individual nodes to macroscopic features that describe the network as a whole, e.g. average path length, degree distribution exponent, etc. Lately, mesoscopic features such as network motifs [2], graphlets [3], and backbones [4] have been utilized.

An important limitation of comparing networks based on feature vectors is that no single set of features can be claimed universal. Some network properties are good for comparing between protein interaction networks, other work well for social networks. In addition, feature extraction is computationally expensive in large networks [5]. This makes use of approaches that do not involve feature vectors a lucrative research problem. In this paper, we take a direction orthogonal to the conventional one and represent a network by its model rather than a set of features.

There are several statistical models that can capture the topology of a given network by fitting a small set of parameters [6–9]. The result of such fitting can be regarded as a compressed (imperfect) representation of the original network i.e. features of the network topology are recapitulated by a small number of metrics. *Similar networks should have similar models*. We explore this claim using series of distance metrics which are based on Kronecker Graphs model fitting algorithm (KronFit) [9]. We also take into account distance metrics derived from network features and compare them with KronFit based distance metrics in performing unsupervised clustering. We analyze the quality of clusters produced using each of these distance metrics, by evaluating against a number of cluster quality metrics. We show that *log-likelihood* (LL) of one network being generated by a Kronecker Graph model fitted to another network performs surprisingly well as a measure of similarity between networks.

The rest of this paper is structured as follows: We recapitulate the Kronecker Graphs generative model in Sect. 2. We proceed and develop series of network similarity estimators based on KronFit algorithm in Sect. 3, followed by description of baseline network distances in Sect. 4. In Sect. 5, we show that augmented network similarity estimators derived from KronFit perform surprisingly well based on the results of experiments performed with 5 classes of real networks. Sect. 6 discusses related works. Our conclusions on generative models as a tool in comparative analysis of networks are summarized in Sect. 7.

## 2 Background on KronFit

Let  $G = (V, E)$  denote a network where  $V = \{1, \dots, n\}$  is a set of  $n$  vertices and  $E \subseteq V^2$  is a set of  $m$  directed unweighted edges. We represent all undirected networks using the directed edges semantic  $((u, v) \in E \Leftrightarrow (v, u) \in E)$ . Although, we assume

unweighted networks, the proposed approach can easily be extended to weighted networks as well. Let  $A$  represent the  $n \times n$  adjacency matrix of  $G$  such that for any  $u, v \in V, A_{uv} = 1$  if  $(u, v) \in E$  and  $A_{uv} = 0$ , otherwise.

Let  $M^{n \times n}$  be a matrix. Let  $M^{[k]}$  be the  $k$ th Kronecker product of the matrix, then its Kronecker square [10] is a matrix  $M^{[2]}$  of dimensions  $n^2 \times n^2$ . The items of  $M^{[2]}$  are

$$M_{ij}^{[2]} = M_{div(i,n),div(j,n)} \cdot M_{mod(i,n),mod(j,n)}$$

where  $div(i, n)$  is the integer quotient of  $i$  divided by  $n$  and  $mod(i, n)$  is the remainder. Leskovec et al. [9] have suggested using  $I^{2 \times 2}$  initiator matrices, where  $I_{ij} \in (0, 1)$ , raised to a Kronecker power of  $k$  as the basic building block in generation of large scale probabilistic adjacency matrices  $P = I^{[k]}$ . The dimension of all generated probabilistic adjacency matrices  $P$  is thus  $2^k \times 2^k$  where  $k$  is some integer. Every item  $P_{ij}$  is the probability of having an edge between the vertices  $i$  and  $j$  in the generated Kronecker networks. The log-likelihood of a given adjacency matrix  $A$  to be generated by drawing each edge  $(i, j)$  with the probability  $P_{ij}$  is:

$$LL = \sum_{i \in V} \sum_{j \in V} \log \left( P_{ij}^{(A_{ij})} (1 - P_{ij})^{(1-A_{ij})} \right) \tag{1}$$

The KronFit algorithm, suggested by the authors, finds the optimal initiator matrix such that  $LL$  is maximized. The authors observed that the  $2 \times 2$  initiator matrices are sufficient for a good match and choosing larger initiator matrices does not improve the results significantly.

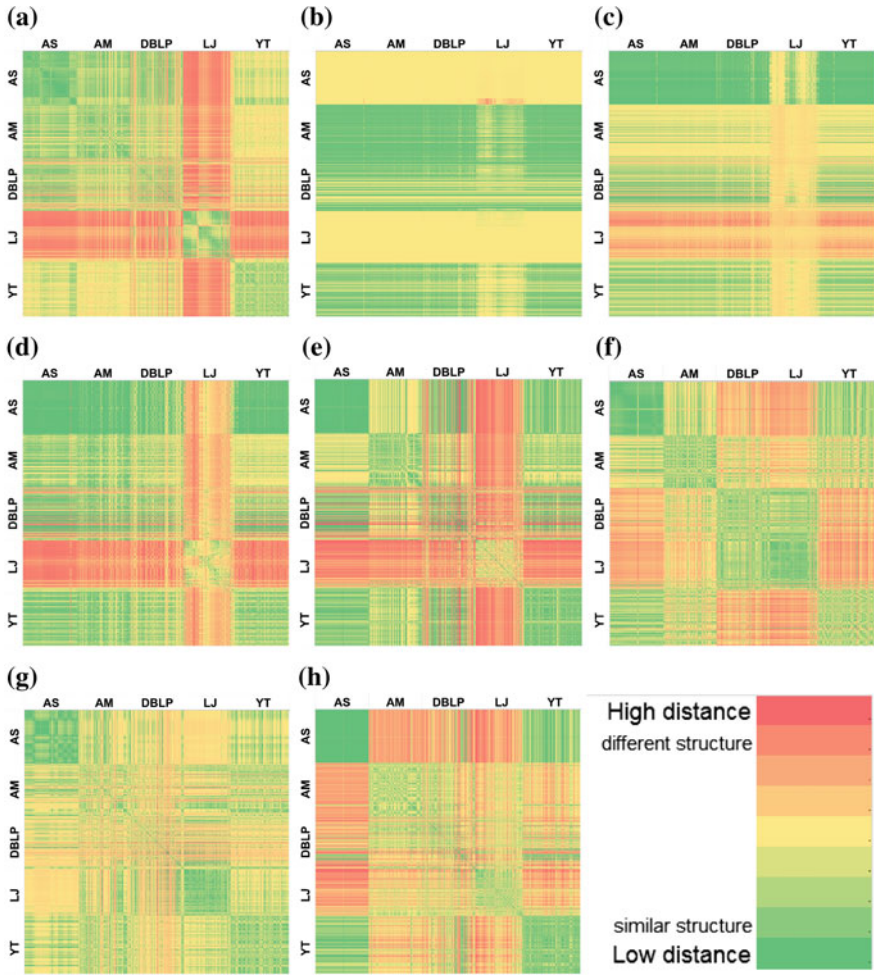
Leskovec et al. also suggest a method to reduce the complexity of  $LL$  calculation to  $O(m)$  by first computing the probability of a network with  $2^k$  vertices and no edges being generated from  $P$ . However, the number of vertices in real networks is not a power of two. Therefore, the network  $G$  is padded with  $2^k - n$  disconnected vertices. In the following discussions, we will assume a padded adjacency matrix  $\hat{A}$  whose dimensions are  $2^k \times 2^k$  where  $k = \lceil \log_2 n \rceil$ .  $\hat{A}$  is constructed from  $A$  by padding  $2^k - n$  rows and  $2^k - n$  columns with zeros.  $\hat{A}$  can be interpreted as an adjacency matrix of a network that is similar to  $G$  but has additional  $2^k - n$  isolated vertices, which we refer to as synthetic vertices in following discussions.

### 3 KronFit Network Distances

**Model distance (MD).** An initiator matrix  $I^{2 \times 2} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$  contains four values that lie in the range of  $[0, 1]$  and are considered equally important for fitting a correct model. Leskovec et al. [9] proposed to compare networks by comparing their initiator matrices. We use *euclidean distance (ED)* to perform such kind of comparison. An *ED* heatmap on 500 networks of five different types<sup>1</sup> is presented in Fig. 1a.

---

<sup>1</sup>Details on evaluated data set are presented in Sect. 5.1.



**Fig. 1** Network similarity metrics. **a** MD. **b** FD. **c** SFD. **d** NFD. **e** Density. **f** Clust.Coeff. **g** Diameter. **h** Degree distr.

**Fitting Distance (FD).** Given a network  $G$ , an initiator matrix  $I$  found by the KronFit algorithm maximizes the log-likelihood of  $G$  being generated using  $I$ . The likelihood of similar networks being generated by the same model is maximized as well. Structurally dissimilar networks should, therefore, have lower likelihood of being generated by the model. In the rest of this paper we denote the initiator matrix fit to a network  $G$  as  $I_G$ . We will refer to the network  $G$  as *prototype*. We denote by  $LL_{GS}$  the log-likelihood of a *subject network*  $S$  being generated from  $I_G$ .

These  $LL_{GS}$  values can be used as inverse distance measure between the networks.  $LL_{GS}$  values are negative numbers ranging from  $-2.6E + 08$  to  $-376$  in current study. The value  $LL_{GS} = -2.6E + 08$  was obtained for a prototype being a community

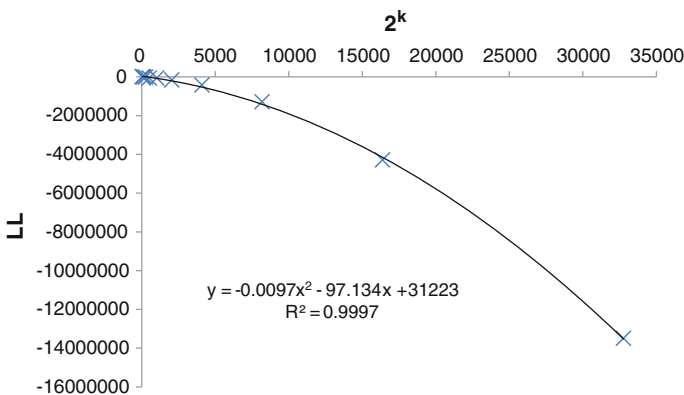
in the Live Journal social network tested against one of the Autonomous Systems topologies. In the rest of this paper, we denote the negation of  $LL_{GS}$  as the *fit-distance* (*FD*) measure between networks.

$LL$  computation time is linear at the number of edges. Thus the time complexity of computing the FD matrix is  $O(k^2\hat{m})$  where  $k$  is the number of networks in the data set and  $\hat{m}$  is the average number of edges. In contrast, MD is easier to compute because for each pair of networks it requires calculating the euclidean distance between the respective initiator matrices, an operation which can be regarded as constant time.

See Fig. 1b for a heatmap of *FD*. The columns in this figure correspond to prototype networks and rows correspond to subject networks. We can clearly notice the large horizontal strips in this heatmap. In contrast to *MD* which is a symmetric measure,  $LL_{GS}$  can differ significantly from  $LL_{SG}$ . There are several factors that affect this asymmetry. For example, KronFit does not work equally well for all kinds of networks. We notice that types of prototype networks with high average *FD* (the yellow horizontal stripes in Fig. 1b) are better differentiated from other networks using this measure. Network size is another significant factor that affects the  $LL$  calculation. The more nodes a network  $S$  has the lower is its  $LL_{GS}$  value w.r.t. any prototype  $G$ .

**Scaled Fitting Distance (SFD).** The log likelihood metric for a network crudely measures how well the synthetic networks generated from the initiator matrix will match the original network. The larger the networks we strive to generate, the more variations can be there and thus, the likelihood of generating a particular subject network of the same size drops. Figure 2 shows that  $LL$  scales as the number of elements in the padded adjacency matrix  $\hat{A}^{2^k \times 2^k}$ .

Next, we adjust the  $-LL_{GS}$  values by the factor of  $1/2^{2k}$ . The resulting distance measure is presented in Fig. 1c. It is clear that the adjusted  $LL$  measure is more informative but the asymmetry is still there. The fitting distance obtained from SLL is referred to as Scaled Fitting Distance (SFD), in the rest of the paper.



**Fig. 2** Log-likelihood (averaged over all prototype networks and subject networks having the same  $k = \lceil \log_2 n \rceil$ ) as a function of the dimension of the padded adjacency matrix ( $2^k$ )

**Normalized Fitting Distance (NFD).** Equation 1 inadvertently incorporates an extra factor that affects the magnitude of log likelihood. While calculating log likelihood, they also consider the probability of edges formed by vertices that weren't part of the initial network. The largest deviations are found in networks whose size is just above a power of two ( $n = 2^k + i$  where  $i$  is some small integer). In such networks, the number of synthetically added vertices (zero padding discussed in Sect. 2) grows to the scale of original network size and notably affects the log likelihood value. Kim and Leskovec [11] propose a corrected log-likelihood  $LL_{N_z}$  that disregards padded nodes. In this paper we focus on KronFit and the original  $LL$  rather than  $LL_{N_z}$  and KronEM proposed in [11].

While comparing a subject network to the existing prototypes one would like to find the closest prototypes. In this case, maintaining consistency among subject networks is not required. We define a Normalized Fitting Distance (*NFD*) such that the most similar network (self) is at distance zero and the farthest network is at distance unity.

$$NLL_{GS} = \frac{LL_{GS} - \max_{G'} LL_{G'S}}{\min_{G'} LL_{G'S} - \max_{G'} LL_{G'S}}$$

Since all values are negative,  $NLL_{GS}$  is a positive real number in range  $[0, 1]$ . The distance metric obtained from NLL is termed as Normalized Fit-Distance (NFD).

## 4 Baseline Network Distances

We compare the KronFit-based network distances to four common baseline metrics: Density, Clustering Coefficient, Diameter, and Degree distribution [12].

**Density** of a network is the fraction of existing edges out of possible edges in the network. For undirected networks  $Density = \frac{2m}{n(n-1)}$  where  $m$  is the number of edges and  $n$  is the number of nodes. The *Density* network distance is simply the absolute difference between the densities of two networks.

**Clustering Coefficient (CC)** of a node is the fraction of existing edges between node's neighbors out of all possible edges between them. CC of a node can be regarded as the density of its ego network. Average CC is a common macroscopic network feature. Similar to *Density*, the *CC* network distance is the absolute difference between the networks' average CCs.

**Diameter** is the distance between the two farthest nodes in the network. We consider the absolute difference between network diameters as the *Diameter* network distance.

**Degree Distribution** is one of the the most common microscopic properties that are used to describe networks. In this paper, we consider an euclidean distance between normalized degree distributions [3] as the eighth network distance measure.

**Table 1** Details of the evaluated networks (all values are averaged over 100 networks of each type)

| Network type | n     | m      | Avg. degree | Avg. CC | Diameter | $LLGG$   |
|--------------|-------|--------|-------------|---------|----------|----------|
| AS           | 6,060 | 23,891 | 7.78        | 0.362   | 10       | -188,666 |
| Amazon       | 96    | 326    | 7.12        | 0.560   | 8.7      | -1,422   |
| DBLP         | 662   | 2,224  | 10.1        | 0.832   | 10       | -13,406  |
| LiveJournal  | 270   | 14,320 | 116.2       | 0.857   | 3.5      | -38,279  |
| Youtube      | 288   | 862    | 5.64        | 0.366   | 6.7      | -5,098   |

## 5 Evaluation

### 5.1 Data and Procedure

We conducted experiments on 500 networks containing five different types of networks. 100 snapshots of the Autonomous Systems relationships between the years 1997 and 2000 were obtained from [13]. The rest of the networks were obtained from the SNAP network collection [14] where we selected 100 largest network files from each one of the following data sets: Amazon (AM), DBLP, LiveJournal (LJ), and YouTube (YT). The evaluated data is briefly summarized in Table 1.

First we have computed the degree distribution, diameter, average clustering coefficient, and density for each network. We normalized the degrees of vertices to fit the range (0, 1]. Then for every pair of consecutive deciles of the normalized degrees ( $d_i, d_{i+1}$ ), we calculated the fraction of vertices whose normalized degrees are between  $d_i$  and  $d_{i+1}$ . Along with diameter, average clustering coefficient, and density this results in 13 features that describe each network.

We proceed by calculating the initiator matrices for all 500 networks in the data set. We set the initiator matrix,  $I^{[1]} = \begin{bmatrix} 0.9 & 0.5 \\ 0.5 & 1 \end{bmatrix}$  (these are the default values used in Leskovec et al. [9] and these represent the general trend in fitted values for most networks) and configure the KronFit algorithm with the following parameters: 50 iterations for gradient descent; learning rate equal to  $10^5$ ; minimal and maximal gradient steps equal to 0.005 and 0.05 respectively; 100,000 samples per gradient estimation; and 10,000 warm-up samples. The initiator matrices add four features to the description of each network. Based on this data set we compute the distances between each pair of network using the eight metrics defined in Sects. 3 and 4, namely: Model Distance ( $MD$ ), Fitting Distance ( $FD$ ), Scaled Fitting Distance ( $SFD$ ), Normalized Fitting Distance ( $NFD$ ),  $Density$ , Clustering Coefficient ( $CC$ ),  $Diameter$ , and Degree Distribution ( $Deg$ ). In the next subsections we analyze the quality of the distance metrics.

**Table 2** Comparison of quality of clusters obtained using different distance metrics using purity, prediction strength (PS), adjusted Rand index (Rand), and Fowlkes-Mallows index (FM)

| Distance Metric | Purity       | PS           | Rand         | FM           |
|-----------------|--------------|--------------|--------------|--------------|
| MD              | 0.608        | 0.340        | 0.398        | 0.555        |
| Density         | 0.502        | 0.268        | 0.192        | 0.426        |
| Degree          | 0.592        | 0.360        | 0.348        | 0.486        |
| CC              | 0.634        | 0.432        | 0.381        | 0.518        |
| Diameter        | 0.516        | 0.288        | 0.215        | 0.412        |
| <i>FD</i>       | <b>0.765</b> | <b>0.503</b> | <b>0.637</b> | <b>0.724</b> |
| <i>SFD</i>      | 0.435        | 0.258        | 0.162        | 0.437        |
| <i>NFD</i>      | 0.393        | 0.237        | 0.104        | 0.518        |

## 5.2 Cluster Analysis

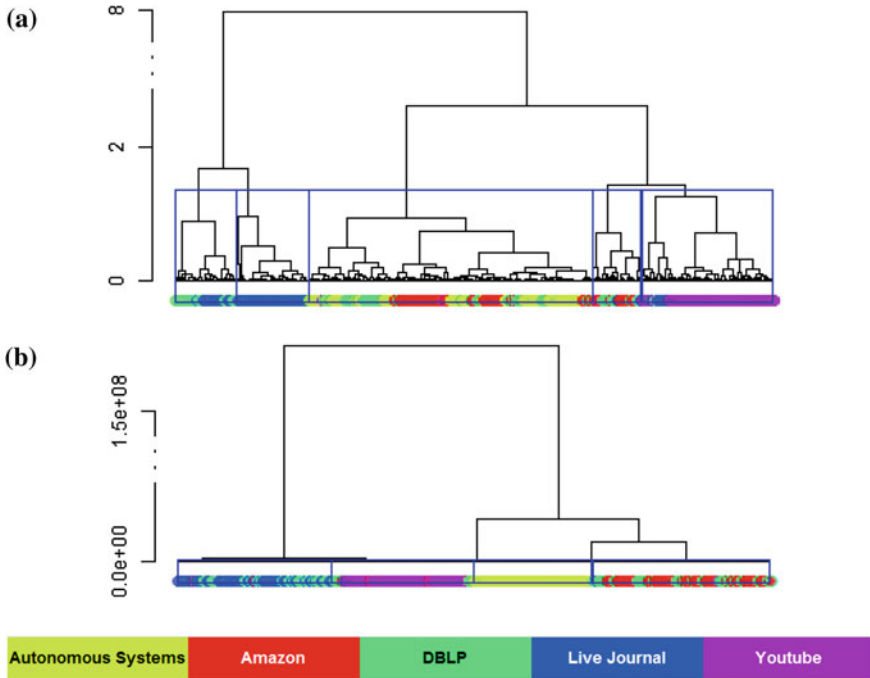
The standard way of evaluating the quality of a distance metric is through application of unsupervised clustering or supervised classification algorithms which require distances between entities to be evaluated. Examples of such algorithms are k-means [15] or Ward’s algorithm [16] for unsupervised or k-nearest neighbors [17] for supervised methods. Here we focus on the clustering algorithms.

In order to measure the quality of a clustering algorithm with respect to a given distance metric, a variety of measures can be used. Average inter-cluster distance ( $ICD_O$ ), average intra-cluster distance ( $ICD_I$ ), the Dunn index [18], Calinski and Harabasz index [19, 20] are only a few examples of such measures. Some algorithms directly optimize one of the cluster quality measures. For example, k-means minimizes the sum of square distances between elements and centers of their clusters. *FD* is clearly superior according to all these metrics as depicted in Table 2.

Given a dissimilarity matrix the objective of a clustering algorithm is partitioning the elements into a set of clusters such that every element is similar to other elements within its cluster and not similar to elements of other clusters. Next, we evaluated the distance metrics with all hierarchical clustering algorithms available in standard distribution of the R programming language. The best overall results were obtained with the Ward’s clustering algorithm which takes a square of the input dissimilarities (distances). Thus, in the rest of this subsection we depict the results using this algorithm.

Figure 3 compares the cluster hierarchies based on the Model Distance (MD) and the proposed Fit Distance (FD). The FD hierarchy is strict with a clear partition into four clusters while the MD hierarchy is more detailed and results clusters of uneven size when cut at any level. We cut all hierarchies to produce five clusters due to the five network types in the data set. By color-coding the leaf nodes based on the types of the respective networks we can visualize the purity of the produced clusters. The networks in the MD hierarchy are mixed up while in the FD hierarchy





**Fig. 3** Ward hierarchical clustering based on **a** the Model Distance and **b** the Fit Distance

the clusters are much more pure with YouTube cluster being the easiest to identify in both hierarchies.

In order to quantitatively evaluate the accuracy of the clustering, we calculate cluster purity, prediction strength [21], adjusted Rand index [22], and Fowlkes-Mallows index [23]. Purity evaluates the extent to which clusters are homogeneous. Purity of a clustering  $C_1, \dots, C_5$  of the 500 networks is

$$Purity = \sum_j \max_i \frac{n_{ij}}{500}$$

where  $n_{ij}$  is the number of networks of type  $i$  in a cluster  $j$ . Other measures also evaluate the produced clustering versus a gold standard, which is derived from the types of the networks as listed in Table 1.

From Table 2, we can clearly see that FD is superior to MD and all other distance metrics. Purity of 0.76 means that only 24% of the networks were included as a minority group in their cluster. Note that SFD and NFD are both extremely inefficient performing worse than the baseline in most cases. Based on these results we can conclude that FD is better suited for unsupervised clustering than other metrics and that normalizing the log-likelihood to produce a good fit distance should be performed with a great care.

**Table 3** Average distances between networks and the inter-cluster/intra-cluster distance ratio for then gold standard and the random reference

| Distance Metric | Gold      |         |               | Random  |         |               |
|-----------------|-----------|---------|---------------|---------|---------|---------------|
|                 | $ICD_O$   | $ICD_I$ | $ICD_O/ICD_I$ | $ICD_O$ | $ICD_I$ | $ICD_O/ICD_I$ |
| MD              | 0.374     | 0.169   | 2.210         | 0.334   | 0.334   | 1.000         |
| Density         | 0.339     | 0.154   | 2.200         | 0.303   | 0.303   | 1.000         |
| Degree          | 0.822     | 0.408   | 2.010         | 0.740   | 0.741   | 0.999         |
| CC              | 0.324     | 0.119   | 2.710         | 0.283   | 0.283   | 1.000         |
| Diameter        | 5.594     | 3.737   | 1.500         | 5.230   | 5.230   | 1.000         |
| $FD$            | 1,165,784 | 53,503  | <b>21.800</b> | 943,813 | 950,231 | 0.993         |
| $SFD$           | 0.207     | 0.186   | 2.320         | 0.328   | 0.325   | 1.009         |
| $NFD$           | 0.181     | 0.061   | 2.930         | 0.842   | 0.842   | 1.000         |

### 5.3 Separation of Network Types

In previous subsection we showed that fit-distance is a good distance (a.k.a. dissimilarity) measure for unsupervised analysis of network collections. Unfortunately, many clustering algorithms are sensitive to the distributions and scale of the distance values. We, therefore, strive to evaluate the quality of the distance metrics directly, without proxies such as classification or clustering algorithms.

In the following analysis we use the gold standard, data set partitioned by the network types, for evaluation of the distance metric quality using standard cluster quality metrics ( $ICD_I$ ,  $ICD_O$ , Dunn Index, etc.). We use random partition as a reference point. An appropriate cluster quality metric should produce the same value for the random partition regardless the scale of the distance metric used. This requirement is especially important in current study because the log-likelihood values differ by several orders of magnitude from other distance metrics as presented in Table 3. Therefore we employ the intra-cluster/inter-cluster distance ratio ( $ICD_O/ICD_I$ ), as it the most stable according to the random reference (see rightmost column in Table 3). This measure can be regarded as a normalized separation index since it is agnostic to the scale of the evaluated distance metric.

High  $ICD_O/ICD_I$  value indicates good separation between the clusters. In our case,  $ICD_O/ICD_I$  is the upper bound on the cluster quality that can be produced by any clustering algorithm. For example, if the degree distribution distance is accurate, then for this particular set of 500 networks the best partition can be at most twice as good as the random partition.

For this purpose, inter-cluster distance and intra-cluster distance of clusters formed by different distance metrics were calculated. A low value for  $ICD_I$  implies that the networks in the same cluster were very similar to each other, which is desirable. Similarly, a higher  $ICD_O$  implies that different clusters were far away from each other and easily distinguishable. Note that different scales are introduced with clustering

using different distance metrics and we divided  $ICD_O$  and  $ICD_I$ , to normalize the scales. From Table 3, it can be easily seen that FD performs better than all the other distance metrics by a large factor. Comparing with the model distance, FD produces clusters that are considerably more distinguishable.

No clustering algorithm can perform better than the gold standard. Thus, cluster quality measure applied to the gold standard with respect to a distance metric results in the highest quality that can be achieved.

## 6 Related Work

Pržulj [3] uses 73 constraints in the form of graphlets to compare PPI networks with their synthetically generated counterparts. The approach is especially appropriate for partially known networks where global topology characteristics are biased but some parts of the networks are well studied and contain reliable local information. This approach also requires munificent computing resources.

In [24], the focus is on comparing mesoscopic properties of networks. Networks are decomposed into communities of different sizes, starting with a single node and a maximum of  $n$  nodes. They compare networks based on different parameters calculated for different community sizes. In [25], networks were compared based on a measure called  $n$ -tangle density, which is basically the edge density in sub-graphs of  $n$  nodes from the network. The  $n$ -tangle density is calculated for different values of  $n$  and these densities are compared to evaluate network similarity. This method cannot find node to node correspondence between similar networks, therefore, making it difficult to pinpoint the source of anomaly (if any) in a network.

Aliakbary et al. [26] used several network features like average shortest path, degree distribution, average density, average clustering, etc. to compare networks for unsupervised machine learning task. They also compare the effectiveness of their approach to Euclidean distance metric based on KronFit initiator matrix and demonstrate its inferiority. It may be noted here that there are many different initiator matrices that can encode the original network with the same likelihood. For example, two initiator matrices  $\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$  and  $\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}$  have non zero euclidean distance of  $\sqrt{2}$ , while both encode isomorphic graphs yielded by Kronecker product of any degree. As demonstrated in Sect. 5, the correct way of utilizing the full power of KronFit is by calculating the likelihood of a network being generated from an initiator matrix and not by comparing the initiator matrices directly.

## 7 Discussion and Future Work

Network comparison is an emerging research area with wide applications in social and biological networks analysis. In this paper, we propose a fit-distance ( $FD$ ) distance metric between a subject network and the model derived from a prototype

network. We demonstrated  $FD$  with the Kronecker Graphs model and showed that it is superior to direct comparison between the models using euclidean distance and to the baseline network distance measures.

One of the interesting features of  $FD$  is that prototype networks that can easily be distinguished from other types of networks using this measure, receive high  $FD$  values in general (see the yellow stripes in Fig. 1b). Although, the differences between subject networks are not comprehensible to the eye in this sub-figure, they are quite significant as can be seen from the normalized values in Fig. 1d. We attribute the success of  $FD$  in clustering and its extremely high  $ICD_o/ICD_l$  value to this natural weighting of “easy” and “hard” prototype networks.

The primary objective of this paper was to draw the attention of scholars to the network comparison opportunities opened by some generative and descriptive network models. In the nearest future, more accurate fitting distance measures should be developed based on new network models such as MAGFit [7] or KronEM [8]. With the advance of model-based network comparison methods, we expect to see machine learning models that classify networks directly, without the need for feature extraction.

## References

1. Bebber, D.P., Hynes, J., Darrah, P.R., Boddy, L., Fricker, M.D.: Biological solutions to transport network design. *Proceedings of the Royal Society of London B: Biological Sciences* **274**(1623), 2307–2315 (2007)
2. Milo, R., et al.: Network motifs: simple building blocks of complex networks. *Science* **298**, 824827 (2002)
3. Pržulj, Natasa: Biological network comparison using graphlet degree distribution. *Bioinformatics* **23**(2), e177–e183 (2007)
4. Serrano, M.Á., Boguná, M., Vespignani, A.: Extracting the multiscale backbone of complex weighted networks. *Proc. Nat. Acad. Sci.* **106**(16), 6483–6488 (2009)
5. Baskerville, Kim; Paczuski, Maya: Subgraph ensembles and motif discovery using an alternative heuristic for graph isomorphism. *Phys. Rev. E* **74**(5), 051903 (2006)
6. Airoldi, E.M., Blei, D.M., Fienberg, S.E., Xing, E.P.: Mixed membership stochastic blockmodels. In: *Advances in Neural Information Processing Systems*, pp. 33–40 (2009)
7. Myunghwan, K., Leskovec, J.: Multiplicative attribute graph model of real-world networks. *Internet Math.* **8**(1–2), 113–160 (2012)
8. Davis, M., Liu, W., Miller, P., Hunter, R.F., Kee, F.: AGWAN: A Generative Model for Labelled, Weighted Graphs. In: *New Frontiers in Mining Complex Patterns*, pp. 181–200. Springer International Publishing (2014)
9. Leskovec, J., Chakrabarti, D., Kleinberg, J., Faloutsos, C., Ghahramani, Z.: Kronecker graphs: an approach to modeling networks. *J. Mach. Learn. Res.* **11**, 985–1042 (2010)
10. Neudecker, H.: A note on Kronecker matrix products and matrix equation systems. *SIAM J. Appl. Math.* **17**(3), 603–606 (1969)
11. Kim, M., Leskovec, J.: The network completion problem: inferring missing nodes and edges in networks. In: *SDM*, pp. 47–58 (2011)
12. Newman, M.: *Networks: An Introduction*. Oxford University Press, Oxford (2009)
13. U. of Oregon Route Views Project. Online data and reports: <http://www.routeviews.org>. The CAIDA UCSD, AS Relationships Dataset (years 1997–2000). <http://www.caida.org/data/active/as-relationships/>

14. Leskovec, J., Krevl, A.: Stanford Large Network Dataset Collection, June 2014. <http://snap.stanford.edu/data>
15. MacQueen, J.: Some methods of classification and analysis of multivariate observations. In: LeCam, L.M., Neyman, J., (eds.), *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability*, p. 281. University of California Press, Berkeley, CA (1967)
16. Ward, Jr., J.H.: Hierarchical grouping to optimize an objective function. *J. Am. Stat. Assoc.* **58**(301), 236–244 (1963)
17. Cover, T.M., Hart, P.E.: Nearest neighbor pattern classification. *IEEE Trans. Inf. Theory* **13**(1), 21–27 (1967)
18. Halkidi, M., Batistakis, Y., Vazirgiannis, M.: On clustering validation techniques. *J. Intell. Inf. Syst.* **17**, 107–145 (2001)
19. Calinski, T., Harabasz, J.: A Dendrite method for cluster analysis. *Commun. Stat.* **3**, 1–27 (1974)
20. Hennig, C., Liao, T.: How to find an appropriate clustering for mixed-type variables with application to socio-economic stratification. *J. Roy. Stat. Soc. Ser. C. Appl. Stat.* **62**, 309–369 (2013)
21. Tibshirani, R., Walter, G.: Cluster validation by prediction strength. *J. Comput. Graph. Stat.* **14**(3), 511–528 (2005)
22. Gordon, A.D.: *Classification*, 2nd edn. Chapman & Hall/CRC, Boca Raton, FL (1999)
23. Fowlkes, E.B., Mallows, C.L.: A method for comparing two hierarchical clusterings. *J. Am. Stat. Assoc.* **78**, 553–569 (1983)
24. Onnela, J.-P., et al.: Taxonomies of networks from community structure. *Phys. Rev. E* **86**(3), 036104 (2012)
25. Gallos, L.K., Fefferman, N.H.: Revealing effective classifiers through network comparison. *EPL (Europhys. Lett.)* **108**(3), 38001 (2014)
26. Aliakbary, S., Motallebi, S., Rashidian, S., Habibi, J., Movaghar, A.: Distance metric learning for complex networks: towards size-independent comparison of network structures. *Chaos: An Interdisciplinary. J. Nonlinear Sci.* **25**(2), 023111 (2015)