Eur. Phys. J. B (2017) 90: 157
DOI: 10.1140/epjb/e2017-70599-4

THE EUROPEAN PHYSICAL JOURNAL B

Regular Article

Revealing how network structure affects accuracy of link prediction

Jin-Xuan Yang and Xiao-Dong Zhang[a]

School of Mathematical Sciences, MOE-LSC and SHL-MAC, Shanghai Jiao Tong University, 800 Dongchuan Road, Shanghai 200240, P.R. China

Received 10 October 2016 / Received in final form 17 April 2017
Published online 28 August 2017 –

Abstract. Link prediction plays an important role in network reconstruction and network evolution. The network structure affects the accuracy of link prediction, which is an interesting problem. In this paper we use common neighbors and the Gini coefficient to reveal the relation between them, which can provide a good reference for the choice of a suitable link prediction algorithm according to the network structure. Moreover, the statistical analysis reveals correlation between the common neighbors index, Gini coefficient index and other indices to describe the network structure, such as Laplacian eigenvalues, clustering coefficient, degree heterogeneity, and assortativity of network. Furthermore, a new method to predict missing links is proposed. The experimental results show that the proposed algorithm yields better prediction accuracy and robustness to the network structure than existing currently used methods for a variety of real-world networks.

1 Introduction

Many real-world complex systems in nature and society can be described by complex networks, where a node in the network represents an individual or element, and a link is set when two nodes have interactions or reactions. To study network functions and network structures can help to understand the mechanism of complex systems [1]. Thus network analysis has attracted considerable attention from researchers of different scientific fields. As an important branch of network analysis, link prediction plays an important role in finding missing data and network reconstruction, predicting the evolution of networks [2–4] and understanding the network functions [5–9]. Protein interaction experiments show that there are huge amount of inaccurate data [5], and our knowledge about real interactions is very limited [6,10,11]. Of course, the applications of link prediction are not limited to the above mentioned items. It can also be applied to recommend system [12,13], and predict future conflict and individual preferences [14,15].

The main work of link prediction is to estimate the likelihood of the existence of a link between two nodes based on current links and interactions in a network [16]. The between pairs of nodes with high scores of likelihood are often regarded as evidence for existence of missing links. There are some link prediction methods based on the topological structure of a network, which take into account the number of common neighbors of two endpoints of a link, since two individuals which have more common friends are more likely to become friends [16]. Although these methods are simple and successful in dealing with specific networks, they are affected by network structure [17]. Moreover, it is reported that different methods may lead to networks that have different topological properties [10]. Thus to study how the network structure affects the accuracy of a link prediction method, or how to choose a suitable link prediction method, is an interesting problem according to some indices which can describe network structure. Currently, there is little literature on this subject.

The performances of many current link prediction methods are quite strongly dependent on the common neighbors and local paths [18]. The current indices to describe network structure, such as Laplacian eigenvalues [19], assortativity of network [20], degree heterogeneity [21] and clustering coefficient [1], can reveal small-world property [22] or clusters. But they pay less attention to strength of between pairs of nodes with common neighbors, and so they cannot provide a good reference for the choice of link prediction methods. On the other hand, a degree distribution function $p(k)$ can reveal the power-law degree distribution of scale-free networks [23], but link prediction is much more related to the size of degree of each node, that is, the fluctuation of degree sequence.

In this paper, we use common neighbors and the Gini coefficient to reveal how network structure affects the accuracy of link prediction, which can provide a good reference for the choice of a suitable algorithm of link prediction according to network structure. The statistical analysis shows high correlations between the common

[a] e-mail: xiaodong@sjtu.edu.cn

neighbors index, Gini coefficient index and other indices to describe network structure, such as Laplacian eigenvalues, clustering coefficient, degree heterogeneity and assortativity of network. Furthermore a new method to predict missing links is proposed. The experimental results show that the proposed algorithm yields better prediction accuracy and robustness to the network structure than existing currently used methods for a variety of real-world networks.

2 Analytical results

2.1 Real-world network datasets

Consider an unweighted and undirected network $G = (V, E)$ with node set $V = \{v_1, v_2, \ldots, v_N\}$ and the observed link set E, where the size of E is m. The self-loops and multiple links are not allowed. The following test datasets conform to this definition. Naturally, more test samples are needed to obtain better accuracy for statistical analysis. The datasets of real-world networks used in this paper include (i) KA (Karate) – this is an interaction network of members of a university Karate club [24]; (ii) PI1 (Protein-structure-1) – this is a network of protein structure [25]; (iii) DP (Dolphins) – this is an animal relationship network with bottlenose dolphins [26]; (iv) SO (Social) – this is a social network of positive sentiment [25]; (v) MO (Movie) – this is a co-appearance network of characters in Hugo's novel Les Miserables [27]; (vi) PI2 (Protein-structure-2) – this is a protein-protein interactions network [25]; (vii) PK (Polbook) – this is a network of books about US politics (http://www.orgnet.com); (viii) WO (Word) – the data is a network of common adjective and noun adjacencies, as described by Newman [28]; (ix) FB (Football) – a network of American football games [29]; (x) JZ (Jazz) – this is a network related to jazz musicians [30]; (xi) NU (Nerve) – this data represents the neural network of C. elegans [22]. The original network is a directed and weighted network. We treat it as an undirected and unweighted one; (xii) US (USAir) – a network of US air transportation systems (http://vlado.fmf.uni-lj.si/pub/networks/data/default.html); (xiii) CEL (C. elegans) – this is a list of edges of the metabolic network of C. elegans [31]. The original self-looping network is treated as one without self-loops; (xiv) CIR (Circuit) – Electronic circuit ISCAS89 can be viewed as a network in which nodes are electronic components and connections are wires [25]; (xv) YEA (Yeast) – this is a network of gene interactions [32]; (xvi) EM (Email) – this is a network of e-mail interchanges between members of the University Rovira i Virgil [33]; (xvii) PB (Polblogs) – a directed network of hyperlinks between weblogs on US politics [34]. Here it is treated as an undirected one; (xviii) PW (Power) – this network represents the topology of the Western States Power Grid of the United States [22]. In Table 1 the parameters of test datasets are listed.

2.2 Correlation between indices

The adjacency matrix $A = (a_{ij})$ of a network G is an $N \times N$ matrix, where $a_{ij} = 1$ if the pair of nodes (i, j) is connected by a link in G, otherwise $a_{ij} = 0$. $k_i = \sum_j a_{ij}$ is the degree of node i in a network, and $D = \mathrm{diag}\{k_1, k_2, \ldots, k_N\}$ is the degree diagonal matrix. The Laplacian matrix L is given by $L = D - A$. We set $0 = \mu_1 \leq \mu_2 \leq \cdots \leq \mu_N$ as the eigenvalues of L [19]. In this paper a link $e = (i, j)$ is called as a link with common neighbors if node i, j have at least one common neighbor. $\Gamma(i)$ denotes the set of neighbors of node i. The first index used to describe the strength of links with common neighbors is c_n, which is defined as the fraction of these links between pairs of nodes with common neighbors in link set E,

$$c_n = \frac{\sum_{i<j} a_{ij}\delta_{ij}}{m}, \tag{1}$$

where $\delta_{ij} = 1$ if $\Gamma(i) \cap \Gamma(j) \neq \emptyset$, 0 otherwise. $0 \leq c_n \leq 1$, and 0 means that there is no triangle in a network, and 1 means that each link is contained in at least a triangle.

c_n also provides a new method for network decomposition. A network G is classified into two categories $G^c = \{G_1^c, G_2^c, \ldots, G_t^c\}$ and $G^n = \{G_1^n, G_2^n, \ldots, G_l^n\}$, where each link is contained in at least a triangle in each component G_i^c, and not any triangle in G_j^n. $G = G^c \cup G^n$. The corresponding decompositions of the adjacency matrix and Laplacian matrix are $A = \sum_{i=1}^{t} A_i^c + \sum_{j=1}^{l} A_j^n$ and $L = \sum_{i=1}^{t} L_i^c + \sum_{j=1}^{l} L_j^n$, where $a_{ij}^c = a_{ij}$ if $\Gamma(i) \cap \Gamma(j) \neq \emptyset$, 0 otherwise. $a_{ij}^n = a_{ij}$ if $\Gamma(i) \cap \Gamma(j) = \emptyset$, 0 otherwise $(i, j = 1, 2, \ldots, N)$.

The local clustering coefficient $c(i)$ of node i is defined as the probability that two distinct neighbors of i are connected. The clustering coefficient c of a network is the average of all nodes [1];

$$c(i) = \frac{2|E_i|}{k_i(k_i - 1)}, \tag{2}$$

where $|E_i|$ denotes the number of links that actually exist between these k_i nodes, and $c(i) = 0$ if $k_i = 0, 1$;

$$c = \frac{\sum_{i \in V} c(i)}{N}. \tag{3}$$

Although c_n and c both pay attention to the common neighbors, the index c_n is different from the clustering coefficient c. The clustering coefficient measures the tendency for nodes to form closely connected clusters. c_n pays more attention to the strength of links with common neighbors in a network. c_n and c have high positive correlation, where the Pearson correlation coefficient (CC) between c_n and c is 0.803 for real-world network datasets (see Fig. 1a, Tab. 2). In fact, this phenomenon can be explained according to the number of triangles. If two distinct neighbors j, k of node i are connected, that is a triangle $i \to j \to k \to i$, then $(i, j), (j, k), (k, i)$ are the links

Table 1. Test datasets of real-world networks. N – the number of nodes, m – the number of links, $\langle k\rangle$ – the average degree.

Network	N	m	$\langle k\rangle$	Network	N	m	$\langle k\rangle$	Network	N	m	$\langle k\rangle$
KA	34	78	4.588	PK	105	441	8.400	CEL	453	2025	8.940
PI1	53	123	4.642	WO	112	425	7.589	CIR	512	819	3.199
DP	62	159	5.129	FB	115	613	10.661	YEA	688	1078	3.134
SO	67	142	4.239	JZ	198	2742	27.697	EM	1133	5451	9.622
MO	77	254	6.597	NU	297	2148	14.465	PB	1490	16 715	22.436
PI2	95	213	4.484	US	332	2126	12.807	PW	4941	6594	2.669

with common neighbors. Thus if there are more triangles in the network with high clustering coefficient, then the probability of links possessing common neighbors is also high (high c_n). Similarly, if there is high c_n in the network, then there are more links with common neighbors, i.e., there are more triangles. So c_n is high in small-world networks, but some biological networks and technological networks indicate low c_n, such as CIR, YEA and PW networks in Figure 1a. Thus to some extent c_n also reveals the small-world property of real-world networks. The most remarkable feature is that the accuracies of most link prediction methods based on the common neighbors have higher correlation with c_n than c (see Fig. 2 in Methods and experiments section).

The second index to describe the properties of the network structure is the Gini coefficient based on the Lorenz curve. The Gini coefficient was proposed by Hirschman, and was used to measure inequality among the values of a frequency distribution (such as income distribution). An alternate expression of the Gini coefficient can be written as [35]:

$$g = \frac{2\sum_{i=1}^{N} i x_i}{N\sum_{i=1}^{N} x_i} - \frac{N+1}{N}, \tag{4}$$

where for a distribution on the values x_i, $i = 1$ to N, indexed in non-decreasing order ($x_i \le x_{i+1}$). The Gini coefficient can also be apply to measure the inequality of degree sequence and Laplacian eigenvalues sequence of a network. Let $k_1 \le k_2 \le \cdots \le k_N$ and $\mu_1 \le \mu_2 \le \cdots \le \mu_N$. Since $\sum_{i=1}^{N} k_i = \sum_{i=1}^{N} \mu_i = 2m$, the Gini coefficient g_d of degree sequence and g_μ of Laplacian eigenvalues sequence are defined as:

$$g_d = \frac{\sum_{i=1}^{N} i k_i - (N+1)m}{Nm}, \tag{5}$$

$$g_\mu = \frac{\sum_{i=1}^{N} i \mu_i - (N+1)m}{Nm}, \tag{6}$$

which reveal the uneven extent of a sequence distribution. The Gini coefficient ranges from 0 to 1, where 0 means perfect equality and 1 complete inequality.

g_d and g_μ have high positive correlation (see Fig. 1e, Tab. 2, $CC = 0.965$), where only the top-15 real-world networks are tested since with increasing of N it becomes more time-consuming to compute all eigenvalues of matrix L. Below, we prove the property that g_μ is no less than g_d, which shows that the uneven degree sequence means the uneven Laplacian eigenvalues sequence.

Property: $g_\mu \ge g_d$ (For an undirected and unweighted network G, the uneven degree sequence means the uneven Laplacian eigenvalues sequence).

Proof: It is sufficient to prove that:

$$\sum_{i=1}^{N} i\mu_i \ge \sum_{i=1}^{N} i k_i, \tag{7}$$

according to the definition of g_μ, g_d. Since:

$$\sum_{i=1}^{N} i\mu_i = \sum_{j=1}^{N}\sum_{i=j}^{N} \mu_i. \tag{8}$$

By the following Schur inequality [36],

$$\sum_{i=t}^{N} \mu_i \ge \sum_{i=t}^{N} k_i, \quad t = 1, 2, \ldots, N. \tag{9}$$

We complete the proof.

We found that there is correlation between g_d, g_μ and current parameters to describe network properties. Table 2 gives the correlated extent of various parameters calculated from the Pearson correlation coefficient. Barabási and Albert found that in many real-world networks there are a small amount of nodes with large degree, and most nodes have few links, that is, the power-law degree distribution $p(k) \propto k^{-\beta}$. The holds for the internet network, protein interaction network, metabolic network and so on. These are called scale-free networks [37]. These nodes with large degree are called rich nodes. Rich nodes are interconnected with high probability to form a so-called rich-club [38]. Besides the degree distribution function $p(k)$, the degree heterogeneity $H = \frac{\langle k^2\rangle}{\langle k\rangle^2}$ [21] is used to describe the scale-free network. For a scale network or rich-club network, there is a high H and uneven degree distribution. Thus the g_d is high for such networks. There must be high positive correlation between g_d and H. The testing in real-world network datasets in Figure 1b supports our finding, and in Table 2 the Pearson correlation coefficient between g_d and H is 0.812.

The assortativity of network r is also called assortative mixing, which denotes the tendency of network nodes to joint other nodes preferentially with opposite or similar

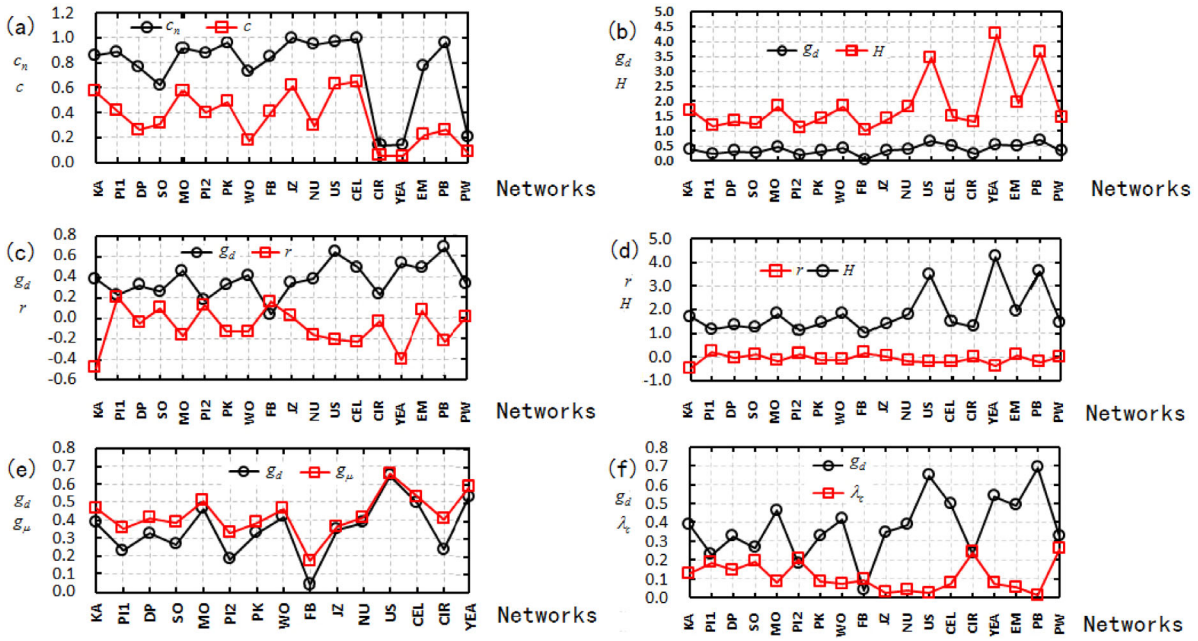

Fig. 1. The results of various parameters in real-world networks. c – clustering coefficient, r – assortativity coefficient, $H = \frac{\langle k^2 \rangle}{\langle k \rangle^2}$ – degree heterogeneity. $\lambda_c = \frac{\langle k \rangle}{\langle k^2 \rangle}$ – spreading threshold. (a) The correlation between c_n and c. (b) The correlation between g_d and H. (c) The correlation between g_d and r. (d) The correlation between r and H. (e) The correlation between g_d and g_μ. (f) The correlation between g_d and λ_c.

Table 2. The Pearson correlation coefficient (CC) between parameters.

	c_n	c	g_d	H	g_d	r	r	H	g_d	g_μ	g_d	λ_c
CC	0.803		0.812		–0.661		–0.622		0.965		–0.623	

properties. If the nodes with large degree tend to be connected with the nodes with large degree, the network has positive correlation; if the nodes with large degree and small degree nodes are connected, the network has negative correlation. The expression of r is proposed by Newman [20] as:

$$r = \frac{S_e S_1 - S_2^2}{S_3 S_1 - S_2^2}, \tag{10}$$

where $S_1 = \sum_i k_i = 2m, S_2 = \sum_i k_i^2, S_3 = \sum_i k_i^3, S_e = 2\sum_{i<j} a_{ij} k_i k_j$.

The statistical analysis on the above datasets shows that H and r have negative correlation, and between g_d and r there is negative correlation to some extent (see Figs. 1c and 1d, Tab. 2). Since for degree heterogeneity networks, there are few nodes with large degree connected to small degree nodes, the H is high, but r is negative. The above analysis shows g_d and H have high positive correlation, so g_d and r have negative correlation. For example, for the KA, US, YEA and PB networks, there is a node with large size of degree connecting a lot of small size nodes. Figure 1 also shows that they have uneven degree sequences, high dis-assortativity and degree heterogeneity. The US network and PB network have obvious rich-club phenomenon [17,39]. Here the two networks have high g_d ($g_d = 0.647$ for US, $g_d = 0.690$ for PB). Thus g_d can reveal the rich-club phenomenon to some extent.

Interestingly, g_d has negative correlation with the epidemic spreading threshold $\lambda_c = \frac{\langle k \rangle}{\langle k^2 \rangle}$ [40] (correlation coefficient $CC = -0.623$). Since for scale-free networks, g_d is high, but these networks have very low epidemic spreading threshold λ_c. In Figure 1f the US network and PB network have very high g_d and low λ_c, which implies that the two networks are more likely to spread diseases since the node with large size of degree is susceptible to infection and transmission of viruses. The another key application of g_d is it better reveals the accuracy of PA link prediction algorithm (see Fig. 2 in Methods and experiments section).

3 Methods and experiments

In this section a new link prediction method is proposed, which has better prediction accuracy and better robustness to network structure than other methods. The above two indices can be better applied to reveal how network structure affects the accuracy of link prediction methods.

A standard metric AUC (areas under the receiver operating characteristic curve) [41] is often used to evaluate the accuracy of a link prediction method. In order to evaluate the performance of an algorithm, the observed link set E is randomly divided into two disjoint subsets: the probe

Table 3. The link prediction methods used for comparison are computed as follows, where $\Gamma(i)$ denotes the set of neighbors of node i. $|\Gamma(i) \cap \Gamma(j)|$ is the number of common neighbors of node i, j, and k_i is the degree of i. A is adjacency matrix, and $\epsilon = 0.01$. For details please refer to [16].

CN	$s_{ij} = \lvert\Gamma(i) \cap \Gamma(j)\rvert$
Salton (Sal)	$s_{ij} = \frac{\lvert\Gamma(i)\cap\Gamma(j)\rvert}{\sqrt{k_i \times k_j}}$
Jaccard	$s_{ij} = \frac{\lvert\Gamma(i)\cap\Gamma(j)\rvert}{\lvert\Gamma(i)\cup\Gamma(j)\rvert}$
Sϕrensen (Sen)	$s_{ij} = \frac{2\lvert\Gamma(i)\cap\Gamma(j)\rvert}{k_i + k_j}$
Hub Promoted Index (HPI)	$s_{ij} = \frac{\lvert\Gamma(i)\cap\Gamma(j)\rvert}{\min\{k_i, k_j\}}$
Hub Depressed Index (HDI)	$s_{ij} = \frac{\lvert\Gamma(i)\cap\Gamma(j)\rvert}{\max\{k_i, k_j\}}$
Leicht-Holme-Newman (LHN)	$s_{ij} = \frac{\lvert\Gamma(i)\cap\Gamma(j)\rvert}{k_i \times k_j}$
Adamic-Adar (AA)	$s_{ij} = \sum_{z\in\Gamma(i)\cap\Gamma(j)} \frac{1}{\log k_z}$
Resource Allocation (RA)	$s_{ij} = \sum_{z\in\Gamma(i)\cap\Gamma(j)} \frac{1}{k_z}$
Local Path (LP)	$s^{LP} = A^2 + \epsilon A^3$
Preferential Attachment (PA)	$s_{ij} = k_i \times k_j$

set E^P (this accounts for 10% of links generally) and the training set E^T (this accounts for 90% of links). E^P is used for testing and is regarded as unknown information. E^T is viewed as known information. A link prediction method provides an ordered list of scores of all links in $U - E^T$ (scores represent the likelihood of missing links), where U is a universal set for $N(N-1)/2$ links. At each time, we will select randomly a link in $U - E$ and a link in probe set E^P to compare their scores. After comparison of n times, there are n' times the links in E^P having higher scores and n'' times they have same scores. The AUC is defined as

$$AUC = \frac{n' + 0.5n''}{n}. \tag{11}$$

A good prediction method should have high AUC, that is, the links in the probe set have higher scores than nonexisting links.

A link prediction method gives the scores between pairs of nodes of missing links. The higher scores represent higher likelihood of missing links. There are many link prediction methods based on the common neighbors metric [16], which are listed in Table 3.

A common feature of these methods is that they pay more attention to the common neighbors of two nodes, since two individuals which have more common friends are more likely to become friends. The Preferential Attachment method is originated from the evolution model where a new link chooses nodes with probability proportional to their degree [23]. The Local Path method takes advantage of the information of the next nearest neighbors.

In a network if two nodes i and j have degree k_i and k_j, respectively, they choose their neighbors at random. The total expected number of common neighbors between two nodes is $k_i k_j/N$. It is thought to be more likely to possess a missing link where between the two nodes is a high difference of the actual number of common neighbors and the expected number choosing randomly, that is

$$P_{ij}^{(1)} = CN_{ij} - k_i k_j/N, \tag{12}$$

where $CN_{ij} = |\Gamma(i) \cap \Gamma(j)|$ is the actual number of common neighbors of node i, j. This probability shows the similarity between two nodes to some extent [42]. Furthermore, the connections of the neighbors of node i, j are considered. Let $x \in \Gamma(i)$, $y \in \Gamma(j)$ $(x \neq y)$. Similarly, the expected number of links between x and y if links are placed at random is $k_x k_y/2m$, and the actual number of links falling in between x and y is a_{xy} (the element of the adjacency matrix). For a real-world network it is more possible that a significant fraction of links fall in between pairs of nodes of $\Gamma(i)$ and $\Gamma(j)$ than at random if there is a missing link (i, j), that is so-called short loops [28],

$$P_{ij}^{(2)} = \sum_{\substack{x\in\Gamma(i),\\ y\in\Gamma(j),\\ x\neq y}} (a_{xy} - k_x k_y/2m). \tag{13}$$

A new link prediction method is proposed, which sets scores between pairs of nodes as

$$s_{ij} = P_{ij}^{(1)} + P_{ij}^{(2)}. \tag{14}$$

Our method further reduces the probability of a "Degenerate State" [18] – the probability that each pair of nodes is assigned same score. This phenomenon is obvious for the CN method, since there are more pairs of nodes with the same number of common neighbors. Thus our method gives better prediction accuracy. The comparison of prediction accuracy under the AUC metric in 10 real-world networks is given in Table 4, where the AA and RA method give close results due to their similar form. The PA method performs better for scale-free networks (high g_d), since for these networks the probability that a new link connects i and j is proportional to $k_i \times k_j$, So for the KA network and YEA network, the PA method gives better results (they have high g_d, $g_d = 0.385$ for KA, $g_d = 0.537$ for YEA). The accuracy of the LP method is higher than the CN method, since when $\epsilon = 0$, the LP degenerates to the CN method. The remarkable feature of our method is its high prediction accuracy in the CIR network, YEA network and PW network, because our method pays attention to connections of neighbors of node i, j besides vertex similarity, that is P_{ij}^2. Conversely, most link prediction methods give their worst results in these networks, since they depend on more the common neighbors, but these three networks possess low c_n (see Fig. 1a).

Overall, most methods give relatively good results in those networks with more the common neighbors (high c_n) except the PA method. For example, the CN method is completely dependent on the common neighbors metric. The Sal, Jaccard, Sen, HPI, HDI, and LHN methods are different normalization methods based on the common neighbors. The PA method gives relatively good results in those networks with high g_d. The CN method is simple,

Table 4. The comparison of accuracy of various link prediction methods under the AUC metric with 10% probe set in 10 real-world networks. The results are the average of 20 realizations for each network, and the probe set E^P is randomly removed every time. The highest value for each network is labeled in boldface.

AUC	KA	PI1	DP	SO	PI2	PK	FB	CIR	YEA	PW
PA	0.680	0.484	0.650	0.541	0.398	0.675	0.259	0.392	0.626	0.442
Jaccard	0.633	0.848	0.803	0.746	0.881	0.875	0.859	0.549	0.542	0.586
LHN	0.624	0.848	0.782	0.745	0.882	0.845	0.861	0.549	0.542	0.586
HDI	0.620	0.846	0.804	0.744	0.880	0.863	0.858	0.549	0.542	0.586
HPI	0.734	0.843	0.785	0.748	0.879	0.891	0.857	0.549	0.543	0.586
Sen	0.633	0.848	0.803	0.746	0.881	0.875	0.859	0.549	0.542	0.586
Sal	0.659	0.847	0.797	0.747	0.880	0.882	0.859	0.549	0.542	0.586
CN	0.682	0.832	0.767	0.746	0.873	0.887	0.848	0.549	0.544	0.586
RA	0.758	0.840	0.807	0.748	0.876	0.901	0.848	0.549	0.546	0.586
AA	0.744	0.840	**0.808**	0.748	0.876	0.900	0.848	0.549	0.545	0.586
LP	0.717	0.867	0.800	0.748	0.889	**0.908**	0.862	0.603	0.753	0.631
Our	**0.781**	**0.888**	0.802	**0.770**	**0.927**	0.894	**0.889**	**0.696**	**0.880**	**0.796**

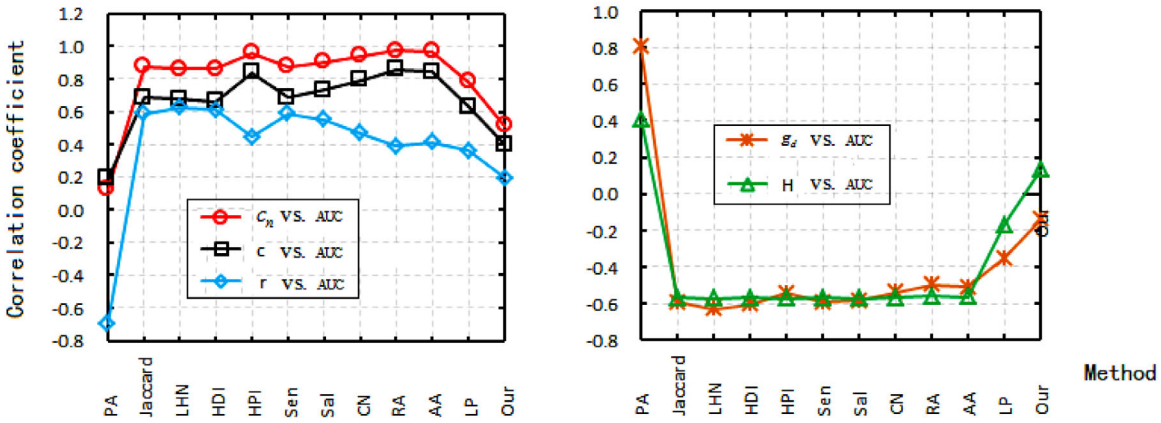

Fig. 2. Pearson correlation coefficients of prediction accuracy AUC versus other parameters (c_n versus AUC, c versus AUC, r versus AUC, g_d versus AUC, H versus AUC) with respect to various methods in 10 networks.

but it leads to degenerate state. The RA and AA methods assign more weights for low degree common neighbors and reduce a degenerate state. Thus they obtain better accuracy in many networks. The RA method, AA method, LP method and our method give better accuracy than others. The LP method and our method can predict these links that generate short loops of length more than 3. Thus for most networks they further improve the prediction accuracy at the cost of being a little more time-consuming.

Furthermore, we compare the correlation between prediction accuracy AUC and c_n, c, r, g_d, H (see Fig. 2). Because most of these methods are based on common neighbor metric, c_n and c both pay attention to the common neighbors. The experimental results further confirm that the AUC of the Jaccard, LHN, HDI, HPI, Sen, Sal, CN, RA and AA methods have positive correlation with c_n, c. The index c_n can better describe the correlations than c in Figure 2, since it pays more attention to the strength of links with the common neighbors. Thus c_n provides a good reference for choice of link prediction methods according to network structure, i.e. it is not suitable to use these methods based on common neighbor metric to predict missing links for these networks with low c_n. For example, in Table 4 the AUC of most algorithms for the CIR, YEA and PW networks are low ($c_n = 0.137$ for the CIR network, $c_n = 0.143$ for the YEA network, $c_n = 0.208$ for the PW network in Fig. 1a).

In Figure 2, the AUC of the PA method has positive correlations with g_d and H, and negative correlation with r, but g_d shows higher correlations with the prediction accuracy of the PA method. Thus it is suitable to use the PA method for some networks with high g_d (KA network and YEA network) and rich-club networks (US network and PB network). The PA method achieves link prediction with minimal time consumption since only the information of node degree is needed.

Figure 2 indicates that the c_n index and g_d index can better reveal the correlation between the prediction accuracy and network structure than other indices. Thus they can provide better reference for choice of link prediction methods. It is not difficult to find that the LP method and our method have low correlations with these parameters from Figure 2, and better prediction accuracy from Table 4. Thus our method has better robustness to network structure. Especially, the RA method, AA method, LP method and our method give better results in these networks with high c_n, such as the PI1 network, PI2 network and PK network.

4 Conclusion and discussion

How network structure affects the accuracy of link prediction methods is an interesting problem, but in the past few years there are little literature on this subject. In this paper, we use the common neighbors index c_n and the Gini coefficient index g_d to reveal the relation between them. Furthermore a new method to predict missing links is proposed. Although c_n and c both pay attention to the common neighbors, the experimental results show that the index c_n has higher correlation with prediction accuracy AUC than the clustering coefficient c, since c_n pays more attention to the strength of links with common neighbors. Therefore c_n can provide a good reference for choice of link prediction methods based on common neighbor metric (i.e., it is not suitable for these networks with low c_n to use these methods). To some extent c_n can also reveal the small-world property of real-world networks.

Indices g_d and g_μ have high positive correlation. They can reveal the rich-club phenomenon. The experimental results show that there is a high positive correlation between g_d and the prediction accuracy of the PA method. Thus for some rich-club networks, such as the US network and PB network, the PA method can be used to achieve link prediction with minimal time consumption since only the information of node degree is needed. Meanwhile, g_d has strong positive correlation with H, and negative correlation with r, which can be explained reasonably by network structure. Furthermore we proved that g_μ is no less than g_d, which indicates that for an undirected and unweighted network, the uneven degree sequence means the uneven Laplacian eigenvalues sequence. An interesting situation is that g_d has negative correlation with the epidemic spreading threshold λ_c, which can be explained by the characteristic of a scale-free network. For example, there are some networks with high g_d, such as the US network and PB network, where a node with large size of degree is susceptible to infection and transmits virus.

For most networks the LP method and our method further improve the prediction accuracy at little additional time cost, and reduce the degenerate state. Our method shows better robustness to network structure and better prediction accuracy than the LP method. The complexity of two methods is the same, $O(l^3N)$, where l is the time complexity to traverse the neighborhood of a node [18].

Beyond that, a new method of network decomposition is proposed according to c_n. The relation between g_d and g_μ provides some conveniences to study the fluctuation of eigenvalue sequence, since it is easy to obtain degree sequence, but computing all eigenvalues is a large time cost for large-scale networks, or even impossible. It is reported that the community structure in a network has correlation with the fluctuation of eigenvalues of the Laplacian matrix [43]. Thus c_n and g_d shed some light on further research for network properties and network structure.

The authors would like to thank the anonymous referees for their valuable comments and suggestions to improve the final version of the paper. We acknowledge Newman for the datasets. This work is supported by the National Natural Science Foundation of China (Nos. 11531001 and 11271256), the Joint NSFC-ISF Research Program (jointly funded by the National Natural Science Foundation of China and the Israel Science Foundation (No. 11561141001)), Innovation Program of Shanghai Municipal Education Commission (No. 14ZZ016) and Specialized Research Fund for the Doctoral Program of Higher Education (No. 20130073110075).

Author contribution statement

This manuscript was completed by J.X. Yang and X.D. Zhang. X.D. Zhang provided many good ideas and methods to achieve this work. J.X. Yang deduced the mathematical equation and model. Afterward J.X. Yang finished the experiment and data processing, and wrote the manuscript. X.D. Zhang revised the manuscript. All authors reviewed the manuscript and approved the final version of the manuscript.

References

1. R. Albert, A.L. Barabási, Rev. Mod. Phys. **74**, (2002) 47
2. Q.M. Zhang, L. Lü, W.Q. Wang et al., PLoS ONE **8**, e55437 (2013)
3. W.Q. Wang, Q.M. Zhang, T. Zhou, EPL **98**, 28004 (2012)
4. Q.M. Zhang, X.K. Xu, Y.X. Zhu, T. Zhou, Sci. Rep. **5**, 10350 (2015)
5. Von C. Mering, R. Krause, B. Snel et al., Nature **417**, 399 (2002)
6. L.A.H. Amaral, Proc. Natl. Acad. Sci. USA **105**, 6795 (2008)
7. A.L. Barabási et al., Physica A **311**, 590 (2002)
8. S.N. Dorogovtsev, J.F. Mendes, Adv. Phys. **51**, 1079 (2002)
9. B. Bringmann, M. Berlingerio, F. Bonchi, A. Gionis, IEEE Intell. Syst. **25**, 26 (2010)
10. H. Yu, P. Braun, M.A. Yildirim et al., Science **322**, 104 (2008)
11. M.P.H. Stumpf, T. Thorne et al., Proc. Natl. Acad. Sci. USA **105**, 6959 (2008)
12. L. Lü, M. Medo, C.H. Yeung et al., Phys. Rep. **519**, 1 (2012)
13. A. Fiasconaro, M. Tumminello, V. Nicosia et al., Phys. Rev. E **92**, 012811 (2015)
14. N. Rovira-Asenjo, T. Gumí, M. Sales-Pardo, R. Guimerà, Sci. Rep. **3**, 1999 (2013)
15. R. Guimerà, A. Llorente, E. Moro, M. Sales-Pardo, PLoS ONE **7**, e44620 (2012)
16. L. Lü, T. Zhou, Physica A **390**, 1150 (2011)
17. T. Zhou, L. Lü, Y.C. Zhang, Eur. Phys. J. B **71**, 623 (2009)
18. L. Lü, C.H. Jin, T. Zhou, Phys. Rev. E **80**, 046122 (2009)
19. R. Merris, Linear Algebra Appl. **197**, 143 (1994)
20. M.E.J. Newman, Phys. Rev. E **67**, 026126 (2003)
21. V. Sood, S. Redner, Phys. Rev. Lett. **94**, 178701 (2005)
22. D.J. Watts, S.H. Strogatz, Nature **393**, 440 (1998)
23. A.L. Barabási, R. Albert, Science **286**, 509 (1999)
24. W.W. Zachary, J. Anthropol. Res. **33**, 452 (1977)
25. R. Milo, S. Itzkovitz, N. Kashtan et al., Science **303**, 1538 (2004)

26. D. Lusseau et al., Behav. Ecol. Sociobiol. **54**, 396 (2003)
27. D.E. Knuth, *The Stanford GraphBase: A Platform for Combinatorial Computing* (Addison-Wesley, Reading, MA, 1993)
28. M.E.J. Newman, Phys. Rev. E **74**, 036104 (2006)
29. M. Girvan, M.E.J. Newman, Proc. Natl. Acad. Sci. USA **99**, 7821 (2002)
30. P. Gleiser, L. Danon, Adv. Complex Syst. **6**, 565 (2003)
31. J. Duch, A. Arenas, Phys. Rev. E **72**, 027104 (2005)
32. R. Milo, S. Shen-Orr, S. Itzkovitz et al., Science **298**, 824 (2002)
33. R. Guimera, L. Danon, A. Diaz-Guilera et al., Phys. Rev. E **68**, 065103 (2003)
34. L.A. Adamic, N. Glance, in *3rd Int. Workshop on Link Discov.*, ACM, 2005, pp. 36–43
35. T. Ogwang, Oxford Bull. Econ. Stat. **62**, 123 (2000)
36. A.E. Brouwer, W.H. Haemers, *Spectra of Graphs* (New York, Springer, 2011)
37. A.L. Barabási, R. Albert, Science **286**, 509 (1999)
38. S. Zhou, R.J. Mondragón, IEEE Commun. Lett. **8**, 180 (2004)
39. V. Colizza, A. Flammini, M.A. Serrano, A. Vespignani, Nat. Phys. **2**, 110 (2006)
40. M. Boguná, R. Pastor-Satorras, Phys. Rev. E **64**, 047104 (2002)
41. J.A. Hanley, B.J. McNeil, Radiology **143**, 29 (1982)
42. E.A. Leicht, P. Holme, M.E.J. Newman, Phys. Rev. E **73**, 026120 (2006)
43. B. He, L. Gu, X.D. Zhang, J. Stat. Mech. **2012**, P02012 (2012)