

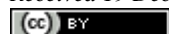
Article

Screening node attributes that significantly influence node centrality in the network

WenJun Zhang

School of Life Sciences, Sun Yat-sen University, Guangzhou 510275, China; International Academy of Ecology and Environmental Sciences, Hong Kong
E-mail: zhwj@mail.sysu.edu.cn, wjzhang@iaees.org

Received 19 December 2015; Accepted 16 January 2016; Published online 1 September 2016



Abstract

In present study, I proposed a method to screen node attributes that significantly influencing node centrality in the network. One of four node centralities, degree centrality, closeness centrality, betweenness centrality, and circuit centrality, can be used as the dependent variable and attribute-by-node data are used as the data of independent variables. Stepwise linear regression method was applied to screen statistically significant node attributes from candidate attributes. Matlab codes of the algorithm are provided also.

Keywords network; node attributes; node centrality; attribute screening; stepwise linear regression.

Selforganizology
ISSN 2410-0080
URL: <http://www.iaees.org/publications/journals/selforganizology/online-version.asp>
RSS: <http://www.iaees.org/publications/journals/selforganizology/rss.xml>
E-mail: selforganizology@iaees.org
Editor-in-Chief: WenJun Zhang
Publisher: International Academy of Ecology and Environmental Sciences

1 Introduction

In nature, many networks are created by connections between nodes that have some correlations, similarities or complementarities in node attributes (Zhang, 2011, 2012a, 2012b, 2015a, 2015c). They include, for example, networks of protein-protein interactions, ecological networks, some social networks, etc. Node attributes determine the evolution and topological structure of such networks. However, in most cases, we do not know which attributes are significant or crucial. Utilization of insignificant attributes will increase the noise of network information. Thus it is necessary to screen significant node attributes from a lot of candidate attributes. In present study, I try to propose a method to screen node attributes that significantly influence node centrality in the network.

2 Algorithm

Suppose there is a weighted network X with m nodes (Zhang, 2012a), its weighted adjacency matrix is $d=(d_{ij})_{m \times m}$. $d_{ij}=w_{ij}$, if two nodes v_i and v_j are adjacent, and $d_{ij}=0$, if v_i and v_j are not adjacent, where w_{ij} is the weight of the link from node v_i to node v_j , $i, j=1, 2, \dots, m$. For unweighted network, $w_{ij}=1$, if two nodes v_i and v_j are adjacent, $i, j=1, 2, \dots, m$. Adjacency matrix d is a symmetric matrix, i.e., $d=d'$. Known n candidate attributes (i.e., traits) for m nodes (e.g., for m proteins, the attributes can be pH, isoelectric point, molecular

weight, or for m species, they can be n alleles, etc.). The raw data matrix is $x=(x_{ij})_{n \times m}$.

First, calculate the following centrality indices of nodes, which represent one of the topological properties of a network (Shams and Khansari, 2014; Zhang, 2012a, 2012b, 2012c; Khansari et al., 2016).

(1) Degree centrality. It is a local centrality based on neighbourhood. It reflects the node influence on its neighbourhoods. A node degree centrality is the sum of the weights of the links attached to that node. It represents the whole involvement of a node in the network (Opsahl et al., 2010; Khansari et al., 2016)

$$y_i = \sum_{j=1}^n w_{ij}$$

where w_{ij} is the weight of link from node i to j .

(2) Closeness centrality. It is a distance-based measure. For a distance-based measure, different values (cost, connection strength, etc.) can be assigned to different links in weighted networks. It is a global centrality which represents the independence of a node in the network (Freeman, 1978). Closeness centrality is defined as reciprocal of the sum of the node's geodesic distances to all other nodes (the distances of the shortest paths) in the network

$$y_i = 1 / \sum_{j=1}^n d_{ij}$$

where, d_{ij} is the weighted geodesic path between node i to j . Dijkstra algorithm (Dijkstra, 1959; Zhang, 2012a) is used to calculate the shortest path and corresponding distance between two nodes.

(3) Betweenness centrality. It is a distance-based measure. It represents the node's ability to control the data flow in the network (Freeman, 1978). This measure is the proportion of number of geodesic paths that pass through the given node to total number of geodesic paths between any pair of nodes in the network (Khansari et al., 2016)

$$y_i = n_i / s$$

where n_i is the number of weighted geodesic paths which pass through node i , and s is the total number of weighted geodesic paths between any pair of nodes in the network.

(4) Circuit centrality. Here I proposed circuit centrality to characterize the contribution of a node in different possible subgraphs. It is a distance-based measure and based on the concept of subgraph centrality (Khansari et al., 2016). This measure is the proportion of number of fundamental circuits (i.e., cycles, or closed paths, or loops, see Paton (1969) and Zhang (2012a)) that pass through the given node to total number of fundamental circuits in the network

$$y_i = n_i / s$$

where n_i is the number of fundamental circuits which pass through node i , and s is the total number of fundamental circuits in the network.

In present study, I use the stepwise linear regression to screen node attributes. The stepwise linear regression is a multi-variable regression that can screen statistically significant variables into the linear regression equation (Zhang and Fang, 1982). The full multi-variable linear regression equation is (Qi et al., 2016)

$$y=b_0+ b_1 x_1+ b_2 x_2+...+ b_n x_n$$

where x_i is the i -th attribute. Let

$$l_{ij} = l_{ji} = \frac{\sum_{k=1}^m x_{jk} x_{ik} - [\sum_{k=1}^m x_{jk} \sum_{k=1}^m x_{ik}] / m}{\sum_{k=1}^m x_{ik} y_k - [\sum_{k=1}^m x_{ik} \sum_{k=1}^m y_k] / m}$$

$$i, j = 1, 2, \dots, n$$

Correlation coefficients between-attribute and between attribute and centrality index y are

$$r_{ij} = l_{ij} / (l_{ii} l_{jj})^{0.5}$$

$$r_{iy} = l_{iy} / (l_{ii} l_{yy})^{0.5}$$

Solve the equation

$$r_{i1} b'_1 + r_{i2} b'_2 + \dots + r_{in} b'_n = r_{iy}$$

$$i = 1, 2, \dots, n$$

The variance contribution of each attribute is

$$v_i = r_{iy}^2 / r_{ii}^2$$

Let $v_k = \max v_i$, and calculate $F = (m-l-1)v_k/q$, where l is the number of attributes screened into the equation, q is the square of residuals. For first screening, $q = v_k$. If $F \geq F_a$, screen the attribute x_k into the equation ($F_a = 0.1$, etc.), or else remove x_k . The correlation matrix are changed as the following

$$\begin{aligned} r_{ij} &= r_{ij} - r_{ik} r_{kj} / r_{kk} & i, j \neq k \\ r_{kj} &= r_{kj} / r_{kk} & j \neq k, i = k \\ r_{jk} &= -r_{jk} / r_{kk} & j \neq k, i = k \\ r_{kk} &= 1 / r_{kk} & i = k, j = k \end{aligned}$$

where k is the k -th screened or removed attribute. Calculate $v_k(l+1) = \max v_i(l+1)$, and $F = (m-l-2) v_k(l+1) / (q(l) - v_k(l+1))$. If $F(l+1) \geq F_a$, screen the attribute x_k into the equation, and change the correlation matrix. Let $v_k = \max v_i$, where x_k is the attribute already in the equation, $F_k = (m-l-1) v_k(l) / q(l)$, where q is the r_{yy} in the inverse matrix of correlation matrix. If $F_k \leq F_a$, remove the attribute x_k from the equation, otherwise screen into the attribute. Repeat the procedure above, until no attribute can be screened into or remove from the equation.

By doing so, the linear regression equation is obtained as the following

$$y = \tilde{b}_0 + \tilde{b}_1 x_1 + \dots + \tilde{b}_j x_j + \dots + \tilde{b}_k x_k$$

and the attributes remained in the equation are qualified node attributes.

The following are Matlab codes (nodeIndicesScreen.m) of the algorithm

```
%Reference: Zhang WJ. 2016. Screening node attributes that significantly influence node centrality in the network.
Selforganizology, 3(3): 75-86
raw=input('Input the file name of node-by-attribute data (e.g., raw.txt, raw.xls, etc. The matrix is z=(zij)n*m, where n is the
number of candidate attributes, m is total number of nodes): ','s');
adj=input('Input the file name of adjacency matrix of unweighted or weighted network (e.g., adj.txt, adj.xls, etc. Adjacency
matrix is d=(dij)m*m, where m is the number of nodes in the network. dij=1 for unweighted network and dij=wij for weighted
```

```

network (wij is the weight for the link vi to vj), if vi and vj are adjacent, and dij=0, if vi and vj are not adjacent; i, j=1,2,..., m):
','s');
fs=input('Input the F threshold value for screening attributes (e.g., 0.1, 0.05): ');
choice=input('Input the type of topological property of nodes (1: Degree centrality; 2: Closeness centrality; 3: Betweenness
centrality): ');
raw=load(raw); nw=size(raw,1); m=size(raw,2); n=nw+1;
x=zeros(n,m);
adj=load(adj);
newdata=zeros(nw,m); xb=zeros(1,n); sg=zeros(1,n); ds=zeros(1,n); degr=zeros(1,m);
switch choice
    case 1
        degr=sum(adj);
    case 2
        [ss,pat,distances,paths]=Dijkstra(adj);
        for i=1:m
            degr(i)=1/sum(distances(i,:));
        end
    case 3
        [ss,pat,distances,paths]=Dijkstra(adj);
        for i=1:m
            degr(i)=pat(i)/ss;
        end
end
a=zeros(n);
x=[raw;degr];
iss="";
for i=1:n
    c=0;
    for j=1:m
        c=c+x(i,j);
    end
    xb(i)=c/m;
    c=0;
    for j=1:m
        c=c+(x(i,j)-xb(i))^2;
    end
    sg(i)=sqrt(c);
end
h=sg(n);
for i=1:n-1
    for j=i+1:n
        c=0;
        for k=1:m
            c=c+(x(i,k)-xb(i))*(x(j,k)-xb(j));
        end
end

```

```

a(i,j)=c/(sg(i)*sg(j)); a(j,i)=a(i,j);
end; end
for i=1:n
xb(i)=i; sg(i)=0; a(i,i)=1;
end
l=0; s=0;
while (n>=1)
if (l==n-1) break; end
ma=0;
for i=1:n
ds(i)=xb(i);
end
for i=1:n-1
if (ds(i)==0) continue; end
if (a(i,i)<1e-05) continue; end
v1=a(i,n)*a(n,i)/a(i,i);
if (v1>ma) ma=v1; k=i; end
end
f1=ma*(m-1-2)/(a(n,n)-ma);
if (f1<=fs) break; end
xb(k)=0; sg(k)=k;
l=l+1;
for i=1:n
for j=1:n
if ((i~=k) & (j~=k)) a(i,j)=a(i,j)-a(i,k)*a(k,j)/a(k,k); end
end; end
for j=1:n
if (j~=k) a(k,j)=a(k,j)/a(k,k); a(j,k)=-a(j,k)/a(k,k); end
end
a(k,k)=1/a(k,k);
r=sqrt(1-a(n,n));
yn=h*sqrt(a(n,n)/(m-1-1));
if (s==0) s=1; continue; end
lab=0;
while (n>=1)
ma=-1e+18;
for i=1:n
ds(i)=sg(i);
end
for i=1:n-1
if (ds(i)==0) continue; end
if (a(i,i)<1e-05) continue; end
v1=a(i,n)*a(n,i)/a(i,i);
if (v1>ma) ma=v1;k=i; end
end

```

```

f1=-ma*(m-1-1)/a(n,n);
if (f1>fs) lab=1; break; end
sg(k)=0; xb(k)=k; l=l-1;
for i=1:n
for j=1:n
if ((i~=k) & (j~=k)) a(i,j)=a(i,j)-a(i,k)*a(k,j)/a(k,k); end
end; end
for j=1:n
if (j~=k) a(k,j)=a(k,j)/a(k,k); a(j,k)=-a(j,k)/a(k,k); end
end
a(k,k)=1/a(k,k);
r=sqrt(1-a(n,n)); yn=h*sqrt(a(n,n)/(m-1-1));
end;
if (lab==1) continue; end
end
for i=1:n-1
a(i,1)=sg(i);
end
for i=1:n
c=0;
for j=1:m
c=c+x(i,j);
end
xb(i)=c/m;
c=0;
for j=1:m
c=c+(x(i,j)-xb(i))^2;
end
sg(i)=sqrt(c);
end
h=sg(n);
c=0;
for i=1:n-1
if (a(i,1)==0) continue; end
ds(i)=a(i,n)*sg(n)/sg(i); a(i,2)=ds(i); c=c+ds(i)*xb(i);
end
s=xb(n)-c;
iss=strcat(iss,'Screened attributes: \n');
nm=0;
for i=1:n-1
if (a(i,1)==0) continue; end
if (ds(i)~=0) iss=strcat(iss,'Attribute-',num2str(i)); end
if ((ds(i+1)~=0) & (i<n-1)) iss=strcat(iss,','); end
if (ds(i)~=0)
nm=nm+1;

```

```

for j=1:m
newdata(nm,j)=x(i,j);
end; end
end
fprintf('\nNew attribute-by-node data\n')
disp([newdata(1:nm,:)])
iss=strcat(iss,'\nStepwise regression equation:\n');
iss=strcat(iss,'y=',num2str(s));
for i=1:n-1
if (a(i,1)==0) continue; end
if (ds(i)>0) e1=num2str(ds(i)); end
if (ds(i)<0) e1=num2str(abs(ds(i))); end
if (ds(i)>0) iss=strcat(iss,'+',e1,'Attribute',num2str(i)); end
if (ds(i)<0) iss=strcat(iss,'-',e1,'Attribute',num2str(i)); end
end
iss=strcat(iss,'\nCorrelation coefficient R=',num2str(r),' ', 'F value=',num2str(fs),'\n');
fprintf(iss)

```

The functions, Dijkstra.m, foundCircuit.m, which are used to calculate the shortest path and corresponding distance between two nodes, and to calculate the fundamental circuits in the network respectively (Zhang, 2012a), are as the following

```

function [ss,pat,distances,paths]=Dijkstra(d)
% d: weighted adjacency matrix; ss: total number of paths; pat: number of paths passing through each node; distances: matrix of
distances between different nodes; paths: string of paths and distances between any of two nodes
v=size(d,1);
p=zeros(1,v); w=zeros(1,v); a=zeros(1,v); b=zeros(1,v);
pat=zeros(1,v);
distances=zeros(v);
for i=1:v
for j=1:v
if ((d(i,j)==0) & (i~=j)) d(i,j)=inf; end
end; end
paths="";
su=0;
for j=1:v-1
for k=j+1:v
for i=1:v
p(i)=0; w(i)=0;
a(i)=inf;
end
a(j)=0; w(j)=1; n=j; h=0;
while (v>0)
ma=inf;
for i=1:v

```

```

if (w(i)==1) continue; end
iv=d(n,i)+a(n);
if (iv<a(i)) a(i)=iv; b(i)=n; end
if (a(i)>ma) continue; end
ma=a(i); h=i;
end
w(h)=1;
if (h==k) break; end
n=h;
end
sd=a(k); p(1)=k; c=k;
for i=2:v
if (c==j) break; end
p(i)=b(c); c=b(c);
end
paths=strcat(paths,'Shortest path from ',num2str(j),' to ',num2str(k),'\n');
for i=v:-1:1
if ((p(i)~=0) & (sd~=inf))
if (i>1) paths=strcat(paths,num2str(p(i)),'->'); end
if (i<=1) paths=strcat(paths,num2str(p(i)),'\n'); end
end; end
for i=v:-1:1
for h=1:v
if (p(i)==h) pat(h)=pat(h)+1; break; end
end; end
if (sd~=inf) paths=strcat(paths,'Distance=',num2str(sd),'\n'); distances(j,k)=sd; distances(k,j)=sd; end
if (sd==inf) paths=strcat(paths,'No path','\n'); su=su+1; end
end; end
ss=v*(v-1)/2-su;

```

```
function [num,n,circuits]=foundCircuit(d)
```

```
% d: weighted adjacency matrix; num: total number of fundamental circuits; n: number of fundamental circuits containing each
node; circuits: string of all circuits
```

```

v=size(d,1);
l=zeros(1,v); vp=zeros(1,v); ts=zeros(1,v); circuit=zeros(1,v*(v-1)/2);
n=zeros(1,v);
num=0;
for i=1:v
l(i)=-1;
end
circuits="";
t=1;
while (v>0)
its=1;
ts(1)=t; l(t)=0;

```



```

while (v>0)
if (its==0) break; end
r=ts(its); lm=l(r)+1;
for w=1:v
if (d(r,w)<=0) continue; end
if ((d(r,w)>0) & ((l(w)+1)==0))
ts(its)=w;
its=its+1;
vp(w)=r; l(w)=lm;
d(r,w)=0; d(w,r)=0;
continue; end
num=num+1; a=vp(w);
m=1;
circuit(1)=r; j=r;
while (v>0)
j=vp(j);
m=m+1;
circuit(m)=j;
if (j==a) break; end
end
m=m+1;
circuit(m)=w;
circuits=strcat(circuits,'Number of fundamental circuit: ',num2str(num),'\n');
circuits=strcat(circuits,'Fundamental circuit: ');
for j=1:m
circuits=strcat(circuits,num2str(circuit(j)),'->');
end
circuits=strcat(circuits,num2str(circuit(1)),'\n');
for i=1:v
for j=1:m
if (circuit(j)==i) n(i)=n(i)+1; break; end
end; end
d(r,w)=0; d(w,r)=0;
end
its=its-1;
end
la=0;
for t=t:v
if (l(t)==-1) la=1; break; end
end
if (la==1) continue; end
break;
end

```

3 Application Example

Data of 17 common HLA-DQB1 alleles (candidate attributes) for the world's 12 human races and populations (nodes) (17×12; Table 1) are from Zhang and Qi (2014) and Zhang (2015c, supplementary material; here the transposition of HLA_DQB1.txt). In addition, an adjacency matrix (12×12; HLA_DQB1_adj.txt; Zhang, 2015c, supplementary material), derived from linear correlation analysis, is given.

Table 1 The world's 12 human races and populations (nodes) and 17 common HLA-DQB1 alleles (candidate attributes)

Allele ID	Alleles	1	2	3	4	5	6	7	8	9	10	11	12
1	DQB1*0201	102	325	238	186	87	135	143	140	92	106	46	60
2	DQB1*0301	71	140	214	169	230	225	159	191	146	245	199	130
3	DQB1*0302	163	114	71	59	82	49	77	67	8	53	38	40
4	DQB1*0303	153	44	71	119	184	163	170	174	62	202	86	170
5	DQB1*0401	10	44	12	51	0	58	50	45	15	64	15	0
6	DQB1*0402	41	18	71	25	0	0	11	0	0	0	0	0
7	DQB1*0501	123	53	48	25	20	37	77	45	15	43	38	80
8	DQB1*0502	51	0	48	68	71	106	28	11	92	43	331	380
9	DQB1*05031	71	44	48	76	36	56	44	6	0	74	46	90
11	DQB1*05032	10	0	0	0	0	0	0	0	31	0	0	0
12	DQB1*0504	0	0	24	25	0	0	0	0	46	0	0	0
13	DQB1*0601	10	44	36	102	158	88	99	129	146	85	136	30
14	DQB1*0602	153	105	36	42	36	47	121	124	331	64	23	0
15	DQB1*0603	10	35	60	25	5	0	17	34	8	0	0	0
16	DQB1*0604	10	18	24	25	5	12	0	28	0	21	0	0
17	DQB1*06051	20	18	0	0	20	0	6	0	0	0	0	10
18	DQB1*null	0	0	0	0	0	25	0	6	7	0	42	10

Table 2 Screened node attributes when different node centralities (dependent variables) and statistic significance levels are used. '+' denotes positive influence and '-' denotes negative influence.

Attributes	Node centrality		Closeness centrality		Betweenness centrality		Circuit centrality	
	F=0.1	F=0.05	F=0.1	F=0.05	F=0.1	F=0.05	F=0.1	F=0.05
1 DQB1*0201	+	+			+	+		
2 DQB1*0301			-	-	-	-	-	-
3 DQB1*0302	-	-						
4 DQB1*0303	+	+	+	+	+	+	+	+
5 DQB1*0401							+	+
6 DQB1*0402			-	-			+	+
7 DQB1*0501							-	-
8 DQB1*0502	-	-	-	-				
9 DQB1*05031			-	-				
10 DQB1*05032	-	-	+	+	-	-	-	-
11 DQB1*0504	+	+			+	+	+	+
12 DQB1*0601	+	+	+	+	+	+		
13 DQB1*0602			-	-	+	+	+	+
14 DQB1*0603	+	+			+	+	+	+
15 DQB1*0604	-	-	-	-	-	-	-	-
16 DQB1*06051			-	-				
17 DQB1*null	+	+			+	+		
<i>R</i>	0.9999	0.9999	0.9999	0.9999	1	1	0.9998	0.9998

In Table 1, nodes 1 to 12 represent Tibetan, Uighur, Kazak, Xingjiang Han, Taiwanese, Hong Kong, Northern Han, Shanghai Han, Hunan Han, Manchu, Buyi, and Dai (Geng et al., 1995; Chang and Hawkins, 1997; Mizuki et al., 1997, 1998, and others).

Using different node centralities and statistic significances, screening results of node attributes are listed in Table 2. As an example, the stepwise linear regression equation for node centrality and $F=0.1$ is

$$y = -0.29311 + 0.00899 \times \text{Attribute1} - 0.00963 \times \text{Attribute3} + 0.03587 \times \text{Attribute4} - 0.01650 \times \text{Attribute8} - 0.25115 \times \text{Attribute10} + 0.10231 \times \text{Attribute11} + 0.01261 \times \text{Attribute12} + 0.05268 \times \text{Attribute14} - 0.08376 \times \text{Attribute15} + 0.09326 \times \text{Attribute17}$$

$$R = 0.9999$$

4 Discussion

In present study, a single topological property, i.e., a type of node centrality is used as the dependent variable of stepwise regression, of which degree centrality is particularly important. Degree centrality is directly correlated to network structure, link prediction, and network evolution (Zhang, 2012a, 2015a, 2015b; Zhang and Li, 2015a, 2015b). To find node attributes that significantly determine the comprehensive topological structure of the network, multiple properties, e.g., node centralities, connectivity (Zhang, 2012a), connectance (2011, 2012a), aggregation index (Zhang and Zhan, 2011), etc., should be jointly used. In this case, the stepwise regression of multiple properties vs. multiple node attributes (Zhang and Fang, 1982) is suggested for use.

Acknowledgment

I am thankful to the support of Discovery and Crucial Node Analysis of Important Biological and Social Networks (2015.6-2020.6), from Yangling Institute of Modern Agricultural Standardization, High-Quality Textbook *Network Biology* Project for Engineering of Teaching Quality and Teaching Reform of Undergraduate Universities of Guangdong Province (2015.6-2018.6), from Department of Education of Guangdong Province, and Project on Undergraduate Teaching Reform (2015.7-2017.7), from Sun Yat-sen University, China.

References

- Barabasi AL, Albert R. 1999. Emergence of scaling in random networks. *Science*, 286(5439): 509
- Chang YW, Hawkins BR. 1997. HLA Class I and Class II frequencies of a Hong Kong Chinese population based on bone marrow donor registry data. *Human Immunology*, 56: 125-135
- Dijkstra EW. 1959. A note on two problems in connexion with graphs. *Numerischemathematik*, 1(1): 269-271
- Freeman LC. 1978. Centrality in social networks conceptual clarification. *Social Networks*, 1: 215-239
- Geng L, Imanishi T, Tokunaga K, et al. 1995. Determination of HLA class II alleles by genotyping in a Manchu population in the northern part of China and its relationship with Han and Japanese populations. *Tissue Antigens*, 46: 111-116
- Khansari M, Kaveh A, Heshmati Z, et al. 2016. Centrality measures for immunization of weighted networks. *Network Biology*, 6(1): 12-27
- Mizuki N, Ohno S, Ando H, et al. 1998. Major histocompatibility complex class II alleles in an Uyur population in the Silk Route of Northwest China. *Tissue Antigens*, 51:287-292

- Mizuki N, Ohno S, Sato T, et al. 1997. Major histocompatibility complex class II alleles in Kazak and Han populations in the Silk Route of Northwest China. *Tissue Antigens*, 50: 527-534
- Newman ME. 2001. Scientific collaboration networks. II. Shortest paths, weighted networks, and centrality. *Physical Review E*, 64(1): 016132
- Opsahl T, Agneessens F, Skvoretz J. 2010. Node centrality in weighted networks: Generalizing degree and shortest paths. *Social Networks*, 32(3): 245-251
- Paton K. 1969. An algorithm for finding a fundamental set of cycles of a graph. *Communications of the ACM*, 12(9): 514-518
- Qi YH, Liu GH, Zhang WJ. 2016. A Matlab program for stepwise regression. *Network Pharmacology*, 1(1): 36-40
- Shams B, Khansari M. 2014. Using network properties to evaluate targeted immunization algorithms. *Network Biology*, 4(3): 74-94
- Zhang WJ. 2011. Constructing ecological interaction networks by correlation analysis: hints from community sampling. *Network Biology*, 1(2): 81-98
- Zhang WJ. 2012a. *Computational Ecology: Graphs, Networks and Agent-based Modeling*. World Scientific, Singapore
- Zhang WJ. 2012b. How to construct the statistic network? An association network of herbaceous plants constructed from field sampling. *Network Biology*, 2(2): 57-68
- Zhang WJ. 2012c. Several mathematical methods for identifying crucial nodes in networks. *Network Biology*, 2(4): 121-126
- Zhang WJ, 2015a. A generalized network evolution model and self-organization theory on community assembly. *Selforganizology*, 2(3): 55-64
- Zhang WJ. 2015b. Calculation and statistic test of partial correlation of general correlation measures. *Selforganizology*, 2(4): 65-77
- Zhang WJ. 2015c. Prediction of missing connections in the network: A node-similarity based algorithm. *Selforganizology*, 2(4): 91-101
- Zhang WJ, Li X. 2015a. General correlation and partial correlation analysis in finding interactions: with Spearman rank correlation and proportion correlation as correlation measures. *Network Biology*, 5(4): 163-168
- Zhang WJ, Li X. 2015b. Linear correlation analysis in finding interactions: Half of predicted interactions are undeterministic and one-third of candidate direct interactions are missed. *Selforganizology*, 2(3): 39-45
- Zhang WJ, Qi YH. 2014. Pattern classification of HLA-DRB1 alleles, human races and populations: Application of self-organizing competitive neural network. *Selforganizology*, 1(3-4): 138-142
- Zhang WJ, Zhan CY. 2011. An algorithm for calculation of degree distribution and detection of network type: with application in food webs. *Network Biology*, 1(3-4): 159-170
- Zhang YT, Fang KT. 1982. *Introduction to Multivariate Statistics*. Science Press, Beijing, China