

ДОДАТОК А

Тексти програм

```

%Battle of Sexes on BA-Network Procedure
clear all
close all
n = 200; m = 3;
[G,degTheor] = genNetG(n,m,'blueup');%'balance');
G.Nodes.Degree = degree(G);
[rhoDeg, rhoCat, jm] = assortG(G);
plotG(G, 'circle', 'Before');
plotDegG(G, degTheor, 'Before');
W = 5; V = 5; payoff1 = [W,-1;-1,V]; payoff2 = [V,-1;-1,W];
nIter = n*m*3;
rho(1) = rhoCat;
for k = 1:nIter
    rezEdge = oneRoundG(G, payoff1, payoff2);
    if min(rezEdge(:))==W
        break;
    end
    i = randsample(2*numedges(G),1,true,W-rezEdge(:));
    exEdge = 1 + mod(i-1,numedges(G));
    exNode = G.Edges.EndNodes(i);
    G = rmedge(G, exEdge);
    G = prefLinkG(G, exNode);
    [i, rho(k+1)] = assortG(G);
end
rezEdge = oneRoundG(G, payoff1, payoff2);
[rhoDeg2, rhoCat2, jm2] = assortG(G);
rho(end+1) = rhoCat2;

plotG(G, 'circle', 'After');
G.Nodes.Degree = degree(G);
plotDegG(G, degTheor, 'After');

[~,order] = sort(degree(G),'descend');
[H,idx] = reordernodes(G,order);
%plotG(H, 'circle', 'SortByDegree');
[~,order] = sort(H.Nodes.Class);
[K,idx] = reordernodes(H,order);
plotG(K, 'circle', 'SortByClass');
figure; imagesc(adjacency(K)); colormap(summer);
figure; plot(rho); grid on;
%{
rez = zeros(1, numel(nodeClass));
for i = 1:edges
    rez(edgeList(:,i)) = rez(edgeList(:,i)) + rezEdge(:,i)';
end
%}

```

```

function [G, degTheor] = genNetG(n,m,classType)
A = genBA(n, m);
G = graph(A);
idx = find(rand(numedges(G),1) < 0.5);
E = G.Edges.EndNodes;
H = E;
H(idx,1) = E(idx,2);
H(idx,2) = E(idx,1);
G = graph(H(:,1),H(:,2));

[~,order] = sort(degree(G), 'descend');
G = reordernodes(G,order);
if nargin==2
    classType = 'random';
end
if strcmp(classType, 'balance')
    n1 = 0; n2 = 0; c = zeros(n,1); d = degree(G);
    for node = 1:n
        if n1 <= n2
            c(node) = 1; n1 = n1 + d(node);
        else
            c(node) = 2; n2 = n2 + d(node);
        end
    end
    G.Nodes.Class = c;
elseif strcmp(classType, 'blueup')
    n1 = 0; n2 = numedges(G); c = zeros(n,1) + 2; d =
degree(G);
    for node = 1:n
        if n1 + d(node) > n2
            break;
        end
        c(node) = 1; n1 = n1 + d(node);
    end
    G.Nodes.Class = c;
else
    G.Nodes.Class = 1 + (rand(n,1) < 0.5);
end
G.Nodes.Labels = [1:n]';
colors = zeros(n,3); colors(G.Nodes.Class==2,1) = 1;
colors(G.Nodes.Class==1,3) = 1;
G.Nodes.Colors = colors;
%G.Edges.Weight = [];

m1 = m + 1;
k = m : n-1;
N = zeros(n-1,n); N(m,m1) = m1; N(m1:n-1,m1) = 0;
for t = m1:n-1
    N(m,t+1) = N(m,t) - N(m,t)*m/2/(t-m1/2) + 1;
    for s = m1:n-1
        N(s,t+1)=N(s,t)+(N(s-1,t)*(s-1)/2-N(s,t)*s/2)/(t-m1/2);
    end
end
end

```

```

sL = sum(N.*((1:n-1)'*ones(1,n)));
Pr = N(m:end,end)'/n;
degTheor = pmf2rangeDist(k, Pr, n);

```

```

function rezEdge = oneRoundG(G, payoff1, payoff2)
strategy1 = G.Nodes.Class(G.Edges.EndNodes(:,1));
strategy2 = G.Nodes.Class(G.Edges.EndNodes(:,2));
rezEdge1 = payoff1(strategy1+2*strategy2-2);
rezEdge2 = payoff2(strategy1+2*strategy2-2);
rezEdge = [rezEdge1, rezEdge2];

```

```

function [rhoDeg, rhoCat, m] = assortG(G)
d = degree(G);
s1 = sum(d);
s2 = sum(d.^2);
s3 = sum(d.^3);
n3 = d'*adjacency(G)*d;
rhoDeg = (n3-s2^2/s1)/(s3-s2^2/s1);
h1 = subgraph(G, G.Nodes.Class==1);
m(1,1) = 2 * numedges(h1);
h2 = subgraph(G, G.Nodes.Class==2);
m(2,2) = 2 * numedges(h2);
m(1,2) = (2 * numedges(G) - m(1,1) - m(2,2)) / 2;
m(2,1) = m(1,2);
rhoCat = (m(1,1)*m(2,2)-m(1,2)*m(2,1)) / (m(1,1)+m(1,2)) /
(m(2,1)+m(2,2));

```

```

function G = prefLinkG(G, excludedNode)
n = 2 * numedges(G);
notk = [neighbors(G,excludedNode); excludedNode];
i = randi(n);
while ismember(G.Edges.EndNodes(i),notk)
    i = randi(n);
end
G = addedge(G,excludedNode, G.Edges.EndNodes(i));

```

```

function plotG(G, layoutForm, titleText)
if nargin < 3
    titleText='';
    if nargin < 2
        layoutForm = 'circle';
    end
end
md = max(G.degree);
if md > 24
    ms = 2 + round(G.degree/md*24);
else
    ms = 2 + G.degree;
end

```

```

if numnodes(G) > 50
    lab = '';
    ec = [0.5 0.5 0.5];
else
    lab = G.Nodes.Labels;
    ec = [0.2 0.2 0.2];
end
figure; plot(G, 'NodeColor',G.Nodes.Colors,...
    'MarkerSize',ms,'layout',layoutForm,'NodeLabel',lab,
    'EdgeColor',ec);
title(titleText);

function plotDegG(G, degTheor, titleText)
TabG = sortrows(G.Nodes, 'Degree', 'descend'); x =
[1:height(TabG)]';
figure; loglog(x, TabG.Degree, '-k'); hold on;
loglog(x(TabG.Class==1), TabG.Degree(TabG.Class==1), 'o', 'Marker
FaceColor', 'b', 'MarkerEdgeColor', 'r', 'MarkerSize', 6);
loglog(x(TabG.Class==2), TabG.Degree(TabG.Class==2), 'o', 'Marker
FaceColor', 'r', 'MarkerEdgeColor', 'b', 'MarkerSize', 6);
if nargin > 1
    loglog(x, degTheor, '-.', 'Color', [0 0.5 0.2], 'LineWidth', 1);
end
hold off; grid on; title(titleText);

```

