**حل مسئله خوشه بندی با الگوریتم جاذبه گرانشی GSA در متلب**

**صورت مسئله:**

می خواهیم داده ها را در طول اجرای برنامه به شرح زیر ایجاد کنیم و سپس با استفاده از الگوریتم جاذبه گرانشی مراکز و تعداد خوشه داده ها را مشخص کنیم

تعداد نمونه در هر خوشه را 20 و در صد داده های پرت را 0.13 و 2داده ها را بعدی در نظر گرفتیم با پراکندگی و میانگین داده شده داده ها را با دستور زیر تولید می کنیم

اعداد توزیع نرمال با 2 سطر و 20 ستون ایجاد می شوند در پراکندگی ضرب شده و با میانگین جمع می شوند و نمونه ها را برای ما تولید می کنند.

C = sigma\*randn(dim,N);

for i = 1:N

C(:,i) = C(:,i) + mu;

end

پراکندگی

sigma1 = [.2 0 ; 0 0.7];

sigma2 = [1.0 0 ; 0 .1];

sigma3 = [.4 0 ; 0 .4];

sigma4 = [.2 0 ; 0 .2];

sigma5 = [.8 0 ; 0 .8];

میانگین(مرکز)

mu1=[-1 -1]+(rand(1,2)-0.5).\*2 .\* [1 1];

mu2=[6 -1]+(rand(1,2)-0.5)\*2 .\* [1 1];

mu3=[0 4]+(rand(1,2)-0.5)\*2 .\* [1 1];

mu4=[8 4]+(rand(1,2)-0.5)\*2 .\* [1 1];

mu5=[2 9]+(rand(1,2)-0.5)\*2 .\* [1 1];

mu6=[-5 -10]+(rand(1,2)-0.5)\*2 .\* [1 1];

**شرح کد:**

این سورس کد شامل 12 فایل می باشد که عبارتند از:

**MAIN\_GravClust.m:** تابع اصلی فراخوانی پیاده سازی الگوریتم جاذبه گرانشی برای حل مسیله خوشه بندی

%% MAIN\_GravClust.m

% Gravitational Clustering Algorithm

خروجی:

numClusters : تعداد خوشه های یافت شده

clusCen: مراکز خوشه ای یافت می شود

clc

close all;

clear all;

%% ورودی کاربر

epsilon\_r = 1; % فاصله ای که واحدهای توده متحرک در آن ترکیب می شوند

numMarker = 1; % تعداد واحدهای انبوه متحرک که در هر نمونه داده ایجاد می شوند

minNeighb = 7; % حداقل جرم واحد توده متحرک برای تعیین خوشه (m\_min)

pout = 0.13; % درصد داده های دور

NI = 20; % تعداد نمونه در هر خوشه

r\_x = 2 ; % شعاع اطراف نمونه ای که واحدهای توده متحرک تزریق می شوند

%% ایجاد داده ها

d=createData(NI, pout);

%% خوشه بندی را با جاذبه گرانشی انجام دهید

[numClusters, clusCen]=performGravitationalClustering(d,epsilon\_r,minNeighb,numMarker,r\_x);

**createData.m:**

مجموعه داده های استفاده شده نسخه خطی را ایجاد می کند

function s = createData(NI, pout)

sigma1 = [.2 0 ; 0 0.7];

sigma2 = [1.0 0 ; 0 .1];

sigma3 = [.4 0 ; 0 .4];

sigma4 = [.2 0 ; 0 .2];

sigma5 = [.8 0 ; 0 .8];

mu1=[-1 -1]+(rand(1,2)-0.5).\*2 .\* [1 1];

mu2=[6 -1]+(rand(1,2)-0.5)\*2 .\* [1 1];

mu3=[0 4]+(rand(1,2)-0.5)\*2 .\* [1 1];

mu4=[8 4]+(rand(1,2)-0.5)\*2 .\* [1 1];

mu5=[2 9]+(rand(1,2)-0.5)\*2 .\* [1 1];

mu6=[-5 -10]+(rand(1,2)-0.5)\*2 .\* [1 1];

dim=length(mu1);

s=[genClusterWithOutlier(mu2,sigma2,NI,pout,dim),...

genClusterWithOutlier(mu3,sigma3,NI,pout,dim),...

genClusterWithOutlier(mu4,sigma4,NI,pout,dim),...

genClusterWithOutlier(mu5,sigma5,NI,pout,dim)];

% خوشه مربعی

s\_v=[];

for i=-2:0.5:0

for j=-2:0.5:0

tmp=[i;j];

s\_v=[s\_v tmp];

end

end

s=[s s\_v];

s=s(:,randperm(length(s(1,:))));

% بعدا خوشه ظاهر شد

NI=15;

s2=[genClusterWithOutlier(mu2,sigma2,NI,pout,2),...

genClusterWithOutlier(mu3,sigma3,NI,pout,2),...

genClusterWithOutlier(mu4,sigma4,NI,pout,2),...

genClusterWithOutlier(mu5,sigma5,NI,pout,2),...

genClusterWithOutlier(mu6,sigma5,20,pout,2)];

s\_v=[];

for i=-2:0.5:0

for j=-2:0.5:0

tmp=[i;j];

s\_v=[s\_v tmp];

end

end

s2=[s2 s\_v];

s2=s2(:,randperm(length(s2(1,:))));

s=[s s2];

end

**genClusterWithOutlier.m:** خوشه را با داده های پرت تولید کنید

function XA = genClusterWithOutlier(mu,sigma,N,percentage,dim)

X=injectParticles(mu.',sigma,N,dim); % یک خوشه با میانگین و انحراف معیار داده شده ایجاد کنید

if percentage == 0

XA=X;

XA(dim+1,:)=zeros(1,length(XA(1,:)));

else

A=rand(1,N);

B=A>=(1-percentage);

tmp=X(:,B==1);

if ~isempty(tmp)

XA(:,B==1)=tmp+injectParticles(zeros(dim,1),3\*eye(dim,dim),length(tmp(1,:)),dim);

XA(:,B==0)=X(:,B==0);

else

XA=X ;

end

end

end

**performGravitationalClustering.m:**

خوشه بندی گرانشی را انجام دهید

function [numClusters\_fin,clusCen\_fin]=performGravitationalClustering(dataSamples,epsilon\_r,minNeighb,numMarker,r\_x)

flagPlot=1; % برای گزینه کشیدن ، پرچم را روی 1 تنظیم کنید

numSamples = length(dataSamples(1,:));

dim = length(dataSamples(:,1));

for timeindx=1:numSamples

samples= dataSamples(:,1:timeindx);

% برای هر نمونه داده جدید حرکت واحد جرم ایجاد کنید

if (exist('p')==0)

numPart=0;

else

numPart = length(p);

end

x = genCluster(samples(:,end),r\_x,numMarker,dim);

Sx = size(x);

for ii = 1:Sx(2)

p(numPart+ii)=createParticle(1,x(:,ii));

end

%% خوشه بندی داده ها را برای مرحله زمان فعلی انجام دهید

p=performClustering(p, samples);

%% جابه جایی واحدهای جرم را ترکیب می کند.

p= combineParticles(p,epsilon\_r);

%% کشیدن

if flagPlot==1

numCenters=0;

clusCen=[];

clf

hold on;

if dim==2 %کشیدن دو بعدی

plot(samples(1,1:end),samples(2,1:end),'b\*','Markersize',8);

for i=1:length(p)

if p(i).mass <= minNeighb

plot(p(i).pos(1),p(i).pos(2),'ro','Markersize',8,'LineWidth',p(i).mass);

end

if p(i).mass > minNeighb

numCenters=numCenters+1;

if norm([-1 -1]'-p(i).pos)<=2

plot(p(i).pos(1),p(i).pos(2),'cx','Markersize', 15,'LineWidth',5);

elseif norm([6 -1]'-p(i).pos)<=2

plot(p(i).pos(1),p(i).pos(2),'gx','Markersize', 15,'LineWidth',5);

elseif norm([0 4]'-p(i).pos)<=2

plot(p(i).pos(1),p(i).pos(2),'kx','Markersize', 15,'LineWidth',5);

elseif norm([8 4]'-p(i).pos)<=2

plot(p(i).pos(1),p(i).pos(2),'yx','Markersize', 15,'LineWidth',5);

elseif norm([2 9]'-p(i).pos)<=2

plot(p(i).pos(1),p(i).pos(2),'mx','Markersize',15,'LineWidth',5);

elseif norm([-5 -10]'-p(i).pos)<=2

plot(p(i).pos(1),p(i).pos(2),'rx','Markersize', 15,'LineWidth',5);

end

clusCen=[clusCen p(i).pos];

end

end

end

if timeindx >minNeighb \*numMarker

disp(['Number of clusters: ' num2str(numCenters)]);

title(['Number of found clusters: ' num2str(numCenters)])

end

xlim([-15 20])

ylim([-15 20])

xlabel('Feature 1')

ylabel('Feature 2')

set(gca,'fontsize',16)

pause(0.1)

else

for i=1:length(p)

if p(i).mass > minNeighb

numCenters=numCenters+1;

clusCen=[clusCen p(i).pos];

end

end

end

numClusters\_fin(1,timeindx)=numCenters;

clusCen\_fin{1,timeindx}=clusCen;

end

**genCluster.m:**تولید خوشه

function C = genCluster(mu,sigma,N,dim)

C = sigma\*randn(dim,N);

for i = 1:N

C(:,i) = C(:,i) + mu;

end

end

**performClustering.m:**انجام خوشه بندی

ورودی:

p : مجموعه فعلی واحدهای جرم قابل جابه جایی

samples : مجموعه ای از نمونه های داده که باید بصورت خوشه ای درآیند

خروجی:

p :مجموعه به روز شده واحدهای جرم قابل جابه جایی

function p= performClustering(p, samples)

t = 1; % مرحله زمان را تنظیم کنید

% محاسبه نیروی وارد بر هر واحد جرم متحرک

for i=1:length(p)

F = zeros(size(samples));

for l = 1:length(samples(1,:))

F(:,l) = calcF(samples(:,l),1,p(i).pos,p(i).mass);

end

p(i).force=sum(F.');

% پارامتر میرایی را تعیین کنید

k = .1;

k\_0=0.3;

k\_d=5;

p(i).r\_koeff= 3; % f\_damp

F\_d = -(p(i).velo).'.\*p(i).r\_koeff.';

% محدودیت برای محدود کردن حداکثر نیرو (f\_damp محاسبه شده بر اساس آخرین مرحله) برای نیروهای فعلی

if(norm(F\_d)>.8.\*norm(p(i).velo))

F\_d = -.8.\*p(i).velo.';

end

if(norm(p(i).force) >.3)

p(i).force = p(i).force./norm(p(i).force) \* .3;

end

d = (p(i).force+F\_d).';

p(i).dF = d; % اختصاص نیرو

p(i).velo = p(i).velo+(p(i).dF).\*t; % تعیین سرعت

p(i).pos=p(i).pos+p(i).velo\*t; % اختصاص موقعیت جدید

end

end

**:combineParticles.m** ذرات را ترکیب کنید

ورودی:

p : مجموعه فعلی از واحدهای جرم متحرک

epsilon\_r : شعاعی که در آن دو واحد جرم متحرک ترکیب می شوند

خروجی:

p : مجموعه به روز شده واحدهای جرم متحرک

function p= combineParticles(p,epsilon\_r)

distMat=zeros(length(p));

particleDeletion = [];

particleSurvivor=[];

for j=1:length(p)

for i = 1:length(p)

distMat(j,i)=norm(p(j).pos-p(i).pos);

end

end

meanDist = mean(distMat);

distMat(distMat==0) = inf;

minVal = min(min(distMat));

while(minVal < epsilon\_r)

idx = find(distMat == minVal); % موقعیت عناصر در ماتریس فاصله

idx=mod(idx,length(p));

idx(find(idx==0))=length(p);

% حداقل فاصله پتانسیلی دو ذره را بررسی کنید

% همه دیگران

minMeanDist = min(meanDist(idx));

% ذره را با حداقل فاصله از دیگران حفظ کنید -> احتمال همگرایی در آینده بیشتر است

minMeanDistIdx = find(meanDist == minMeanDist);

if length(idx)==1

deleteIdx = idx(idx~=minMeanDistIdx);

winnerIdx = idx(idx==minMeanDistIdx);

else

deleteIdx = idx(1);

winnerIdx = idx(2);

end

particleDeletion = [particleDeletion deleteIdx];

particleSurvivor = [particleSurvivor winnerIdx];

distMat(deleteIdx,:) = inf;

distMat(:,deleteIdx) = inf;

minVal = min(min(distMat));

end

% واحدهای جرمی قابل جابه جایی را به روز کنید / حذف کنید

for i=1:length(particleDeletion)

p(particleSurvivor(i)).mass = p(particleDeletion(i)).mass +p(particleSurvivor(i)).mass;

p(particleSurvivor(i)).subP=[p(particleSurvivor(i)).subP p(particleDeletion(i))];

end

p=p(setdiff([1:length(p)],particleDeletion));

end

**gravitationalClustering.m:**الگوریتم خوشه بندی با جاذبه گرانشی

ورودی:

dataSamples : مجموعه داده کاملی که باید خوشه بندی شود

epsilon\_r : شعاعی که در آن دو واحد جرم متحرک ترکیب می شوند

m\_min :آستانه واحد جرم متحرک برای تعریف خوشه

num\_marker : تعداد واحدهای جرم متحرک که به ازای هر نمونه داده تزریق می شوند

r\_x : محدوده ای را برای شعاع تعیین می کند که در آن واحد جرم متحرک تزریق می شود در اطراف یک نمونه داده

خروجی:

numClusters\_fin : تعداد خوشه های شناسایی شده

clusCen\_fin :م وقعیت های مرکز خوشه ای مربوطه

function [numClusters\_fin,clusCen\_fin]=GravitationalClustering(dataSamples,epsilon\_r,m\_min,numMarker,r\_x)

flagPlot=1; % برای گزینه کشیدن ، پرچم را روی 1 تنظیم کنید

numSamples = length(dataSamples(1,:));

dim = length(dataSamples(:,1));

for timeindx=1:numSamples

samples= dataSamples(:,1:timeindx);

% برای هر نمونه داده جدید حرکت واحد جرم ایجاد کنید

if (exist('p')==0)

numPart=0;

else

numPart = length(p);

end

x = injectParticles(samples(:,end),r\_x,numMarker,dim);

Sx = size(x);

for ii = 1:Sx(2)

p(numPart+ii)=createParticle(1,x(:,ii));

end

%% خوشه بندی داده ها را برای مرحله زمان فعلی انجام دهید

p=performClustering(p, samples);

%% جابه جایی واحدهای جرم را ترکیب می کند.

p= combineParticles(p,epsilon\_r);

%% کشیدن

if flagPlot==1

numCenters=0;

clusCen=[];

clf

hold on;

if dim==2 %کشیدن دوبعدی

plot(samples(1,1:end),samples(2,1:end),'b\*','Markersize',8);

for i=1:length(p)

if p(i).mass <= m\_min

plot(p(i).pos(1),p(i).pos(2),'ro','Markersize',8,'LineWidth',p(i).mass);

end

%به روز رسانی تعداد خوشه ها

if p(i).mass > m\_min

numCenters=numCenters+1;

if norm([-1 -1]'-p(i).pos)<=2

plot(p(i).pos(1),p(i).pos(2),'cx','Markersize', 15,'LineWidth',5);

elseif norm([6 -1]'-p(i).pos)<=2

plot(p(i).pos(1),p(i).pos(2),'gx','Markersize', 15,'LineWidth',5);

elseif norm([0 4]'-p(i).pos)<=2

plot(p(i).pos(1),p(i).pos(2),'kx','Markersize', 15,'LineWidth',5);

elseif norm([8 4]'-p(i).pos)<=2

plot(p(i).pos(1),p(i).pos(2),'yx','Markersize', 15,'LineWidth',5);

elseif norm([2 9]'-p(i).pos)<=2

plot(p(i).pos(1),p(i).pos(2),'mx','Markersize',15,'LineWidth',5);

elseif norm([-5 -10]'-p(i).pos)<=2

plot(p(i).pos(1),p(i).pos(2),'rx','Markersize', 15,'LineWidth',5);

end

clusCen=[clusCen p(i).pos];

end

end

end

if timeindx >m\_min \*numMarker

disp(['Number of clusters: ' num2str(numCenters)]);

title(['Number of found clusters: ' num2str(numCenters)])

end

xlim([-15 20])

ylim([-15 20])

xlabel('Feature 1')

ylabel('Feature 2')

set(gca,'fontsize',16)

pause(0.1)

%% تعیین تعداد خوشه ها

else

for i=1:length(p)

if p(i).mass > m\_min

numCenters=numCenters+1;

clusCen=[clusCen p(i).pos];

end

end

end

numClusters\_fin(1,timeindx)=numCenters;

clusCen\_fin{1,timeindx}=clusCen;

end

**calcF.m:** محاسبه نیروی وارد بر هر جرم متحرک

function F=calcF(x,x\_mass,x2,x2\_mass)

q=log10(norm(x-x2)+1)+2;

r\_diff = x-x2;

r\_diff\_n = norm(r\_diff);

F = x\_mass\*x2\_mass\* (r\_diff)./r\_diff\_n.^q;

end

**injectParticles.m:**

یک خوشه با میانگین و انحراف معیار داده شده ایجاد کنید

function C = injectParticles(mu,sigma,N,dim)

C = sigma\*randn(dim,N);

for i = 1:N

C(:,i) = C(:,i) + mu;

end

end

**RUN\_Demo.m:** نسخه ی نمایشی را اجرا کنید

خروجی:

numClusters تعداد خوشه های یافت شده

clusCen مراکز خوشه ها یافت می شود

close all;

clear all;

%% ورودی کاربر

epsilon\_r = 1; % فاصله ای که واحدهای توده متحرک در آن ترکیب می شوند

numMarker = 1; % تعداد واحدهای جرم متحرک که در هر نمونه داده ایجاد می شوند

minNeighb = 7; % حداقل جرم واحد جرم متحرک برای تعیین خوشه (m\_min)

pout = 0.13; % درصد داده های پرت

NI = 20; % تعداد نمونه در هر خوشه

r\_x = 2 ; % شعاع اطراف نمونه ای که واحدهای جرم متحرک تزریق می شوند

%% ایجاد داده ها

d=createData(NI, pout);

%% خوشه بندی را با الگوریتم جاذبه گرانشی را انجام دهید

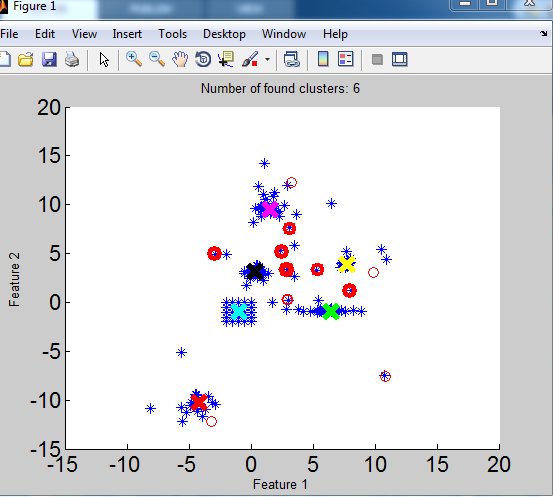
[numClusters, clusCen]=GravitationalClustering(d,epsilon\_r,minNeighb,numMarker,r\_x);

کافی است یکی از دو فایل **RUN\_Demo.m** یا **MAIN\_GravClust.m** را اجرا کنید تا نتیجه را ببینید.

**نتایج:**

**پس از اجرای فایل MAIN\_GravClust.m ، 6 خوشه پیدا می شود و مراکز هر خوشه با رنگی متفاوت از مرکز خوشه دیگر نمایش داده می شود.**

**در ادامه نتیجه را مشاهده کنید...**

****

**Number of clusters: 0**

**Number of clusters: 0**

**Number of clusters: 0**

**Number of clusters: 0**

**Number of clusters: 0**

**Number of clusters: 0**

**Number of clusters: 0**

**Number of clusters: 0**

**Number of clusters: 0**

**Number of clusters: 0**

**Number of clusters: 0**

**Number of clusters: 0**

**Number of clusters: 0**

**Number of clusters: 0**

**Number of clusters: 0**

**Number of clusters: 0**

**Number of clusters: 0**

**Number of clusters: 0**

**Number of clusters: 0**

**Number of clusters: 0**

**Number of clusters: 0**

**Number of clusters: 0**

**Number of clusters: 1**

**Number of clusters: 2**

**Number of clusters: 2**

**Number of clusters: 2**

**Number of clusters: 2**

**Number of clusters: 2**

**Number of clusters: 2**

**Number of clusters: 2**

**Number of clusters: 2**

**Number of clusters: 2**

**Number of clusters: 2**

**Number of clusters: 2**

**Number of clusters: 2**

**Number of clusters: 2**

**Number of clusters: 2**

**Number of clusters: 2**

**Number of clusters: 2**

**Number of clusters: 2**

**Number of clusters: 2**

**Number of clusters: 2**

**Number of clusters: 2**

**Number of clusters: 2**

**Number of clusters: 3**

**Number of clusters: 3**

**Number of clusters: 3**

**Number of clusters: 3**

**Number of clusters: 3**

**Number of clusters: 3**

**Number of clusters: 3**

**Number of clusters: 3**

**Number of clusters: 3**

**Number of clusters: 3**

**Number of clusters: 3**

**Number of clusters: 3**

**Number of clusters: 3**

**Number of clusters: 3**

**Number of clusters: 3**

**Number of clusters: 3**

**Number of clusters: 3**

**Number of clusters: 4**

**Number of clusters: 4**

**Number of clusters: 4**

**Number of clusters: 4**

**Number of clusters: 4**

**Number of clusters: 4**

**Number of clusters: 4**

**Number of clusters: 4**

**Number of clusters: 4**

**Number of clusters: 4**

**Number of clusters: 4**

**Number of clusters: 4**

**Number of clusters: 4**

**Number of clusters: 4**

**Number of clusters: 4**

**Number of clusters: 4**

**Number of clusters: 4**

**Number of clusters: 4**

**Number of clusters: 4**

**Number of clusters: 4**

**Number of clusters: 4**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 5**

**Number of clusters: 6**

**Number of clusters: 6**

**Number of clusters: 6**

**Number of clusters: 6**

**Number of clusters: 6**

**Number of clusters: 6**

**Number of clusters: 6**

**Number of clusters: 6**

**Number of clusters: 6**

**Number of clusters: 6**

**Number of clusters: 6**

**Number of clusters: 6**

**Number of clusters: 6**

**Number of clusters: 6**

**Number of clusters: 6**

**Number of clusters: 6**

**Number of clusters: 6**

**Number of clusters: 6**

**Number of clusters: 6**

**Number of clusters: 6**

**Number of clusters: 6**

**>>**