AMRITA
VISHWA VIDYAPEETHAM
DEEMED TO BE UNIVERSITY UNDER SECTION 3 OF UGC ACT, 1956

Unit 1 - Sem 4 - 22MAT230

Mathematics for Computing 4

Dr Sunil Kumar S and Prof K P Soman

School of Artificial Intelligence

Amrita Vishwa Vidyapeetham

If you find any mistakes or have any comments to share,

I would be grateful to receive them at s_sunilkumar@cb.amrita.edu

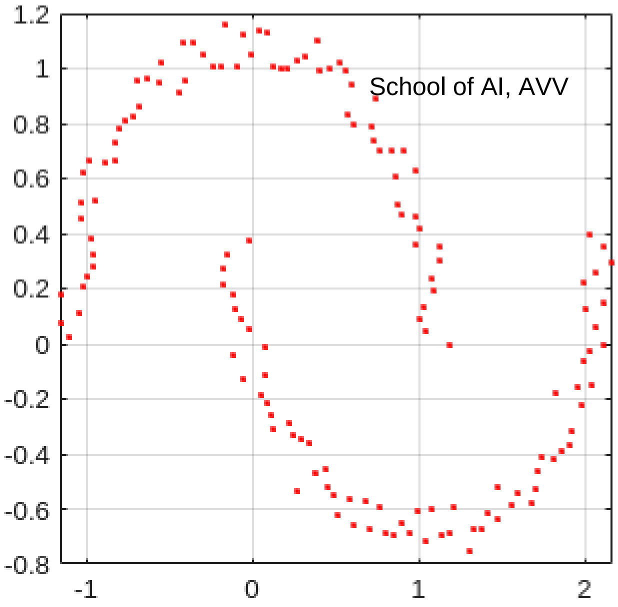
https://github.com/mfcpjt/MFC4_22MAT230

```
clearvars
clear all
ready = true;
PUBLISH = ready;
```

```
function X = moon2dataset()
    N = 144;
    r = 1 + 0.2*rand(1,N);
    theta = linspace(0,2*pi,N);
    X = [r.*cos(theta);r.*sin(theta)]';
    X(X(:,2)<0,:) = X(X(:,2)<0,:) + [1,0.4];
end
```

```
% rng(100)

X = moon2dataset();
plot(X(:,1),X(:,2),'r.')
axis square
grid on
```



```
N = size(X,1);
```

Compute the distance matrix

d is the distance matrix

```
d = dist(X');
```

d2 is the distance squared matrix.

```
d2 = d.*d;
```

sort the distance matrix

```
sd = sort(d,2);
```

self tuning σ

Personalized σ for each data point.

Take the distance to the n^{th} data point as the σ

Here n is taken as 4.

Distance to the third nearest point is taken

```
s = sd(:,4);
lsm = s*s';
```

Build the similarity matrix

```
S = exp(-d2 ./ lsm);
```

Compute the Adjacency, Degree and symmetric normalized Laplacian matrices

```
W = S - diag(diag(S));
D = diag(sum(W));
L_u = D - W;
L_n = eye(N) - pinv(D)*W;
L_ns = eye(N) - D^(-0.5)*W*D^(-0.5);
meanW = mean(W,"all")
```

```
meanW = 
0.0153
```

weak links are removed.

(weaklings dont survive ! pun not intended)

```
W(W < meanW)=0;
```

```
[eVec, eVal] = eig(L_ns);
```

Check the number of zero eigen values of L_u

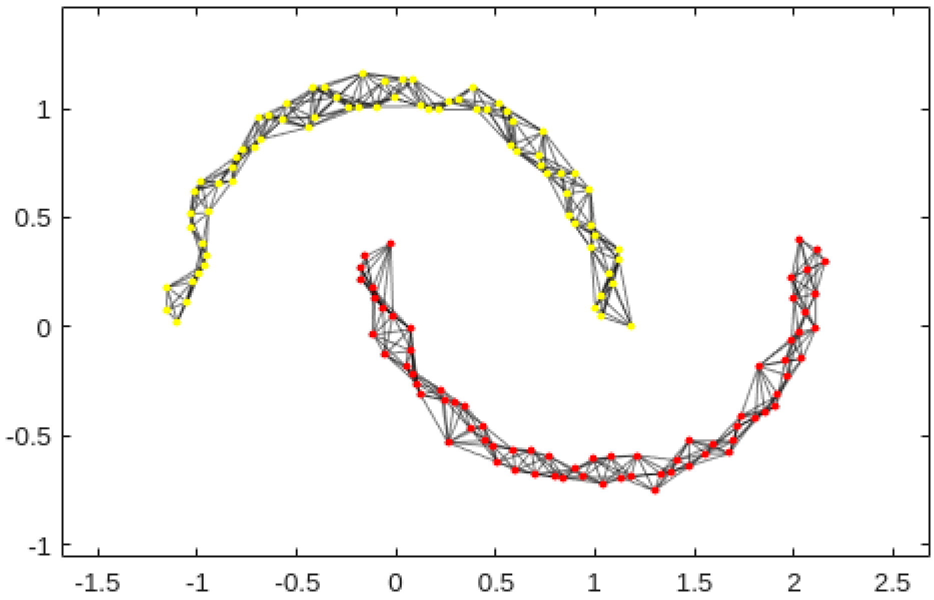
```
[eVec1, eVal1] = eig(L_u);
diag(eVal1)
```

```
ans = 144×1
   -0.0000
    0.0000
    0.0070
    0.0082
    0.0272
    0.0306
    0.0641
    0.0687
    0.1096
    0.1204
    0.1840
    0.1893
    0.2495
    0.2749
    0.3266
      ⋮
```

```
[s_eVal, idx] = sort(diag(eVal),"ascend");
s_eVec = eVec(:,idx);
```

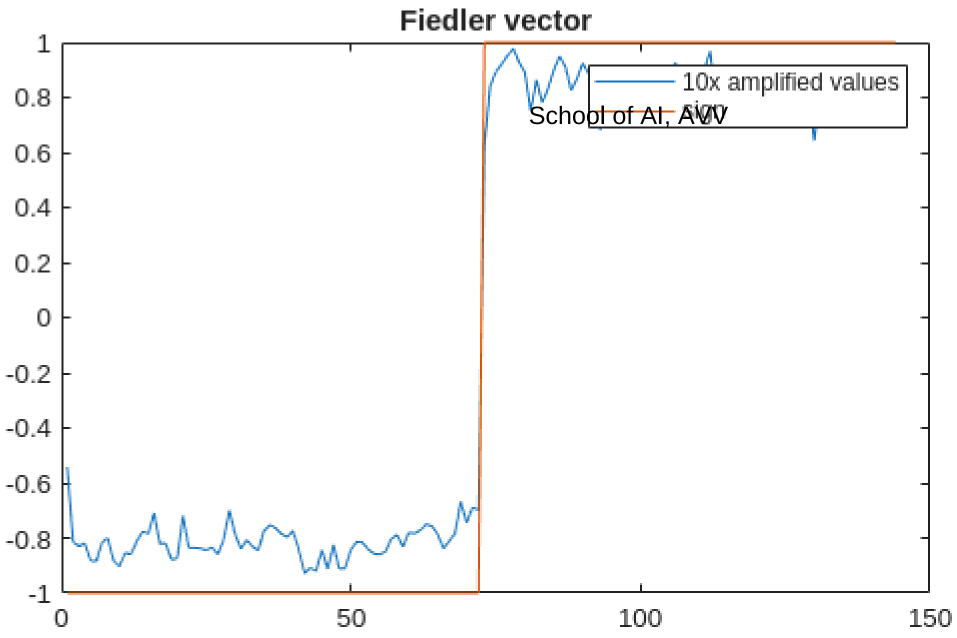
```
fv = s_eVec(:,2);
```

```
pG =
plot(graph(W),"XData",X(:,1),"YData",X(:,2),"EdgeColor","black","NodeLabel",
[]);
highlight(pG, find(fv>0), "NodeColor","red")
highlight(pG,find(fv<0),"NodeColor","yellow")
```



Sign of Fiedler vector used for classification

```
plot(10*fv);hold on
plot(sign(fv)); hold off
title("Fiedler vector")
legend("10x amplified values","sign")
```



Split the graph into 2 sub graphs based on the sign of the fiedler vector

All datapoints corresponding to the **positive** components of fv belong to one cluster

```
W1 = W(fv>0,fv>0);
min(W1,[],"all")
```

```
ans = 
0
```

```
meanW1 = mean(W1,"all")
```

```
meanW1 = 
0.0291
```

```
max(W1,[],"all")
```

```
ans = 
0.8545
```

```


W1(W1 < 2*meanW1) = 0;
```

All datapoints corresponding to the **negative** components of fv belong to another cluster

```
W2 = W(fv<0,fv<0);
min(W2,[],"all")
```

```
ans = 
0
```

```
meanW2 = mean(W2,"all")
```

```
meanW2 =
0.0317
```

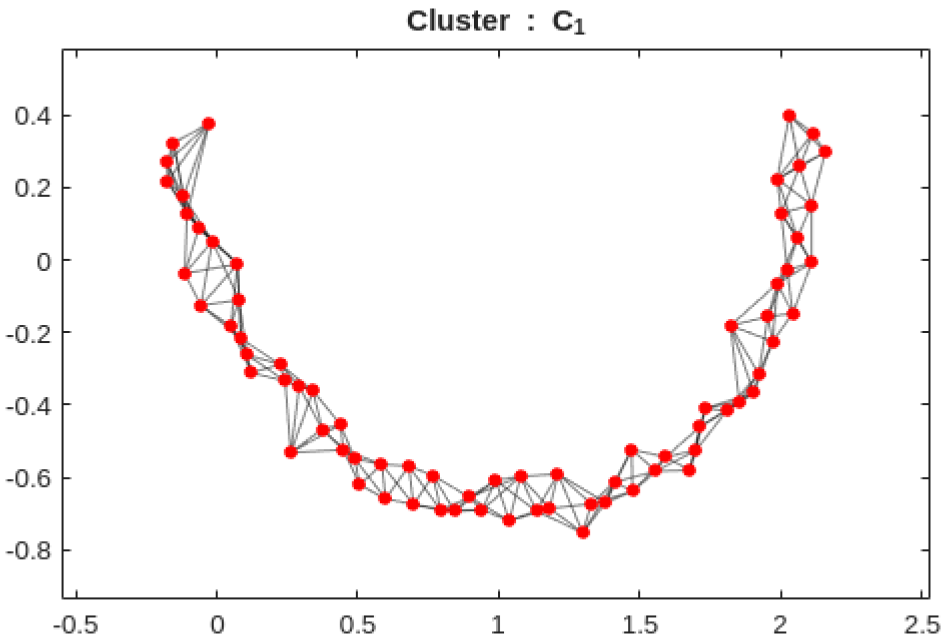
```
max(W2,[],"all")
```

```
ans =
0.9061
```

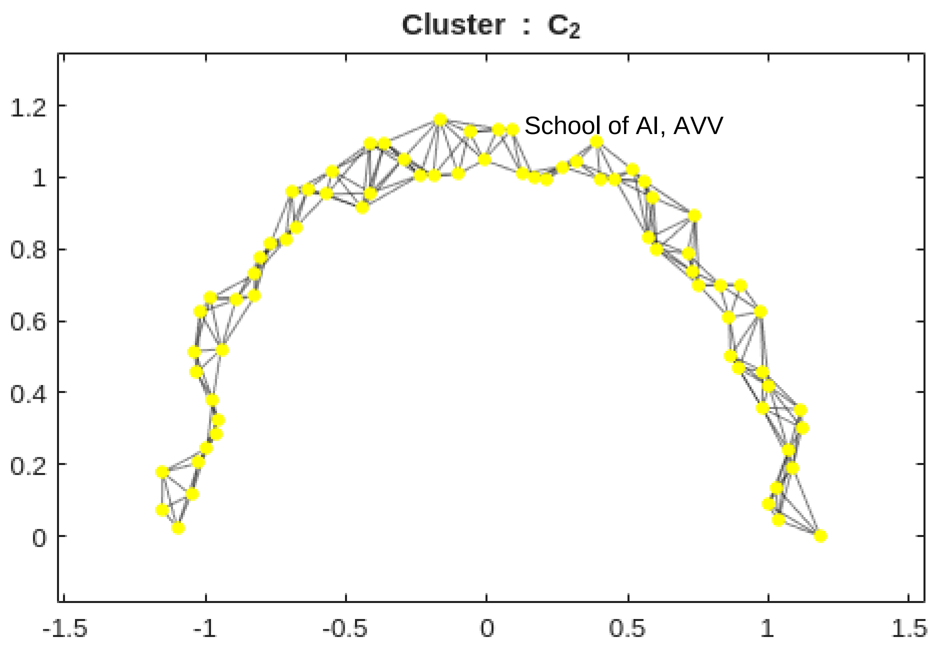
```


W2(W2 < 2*meanW2) = 0;
```

```
plot(graph(W1),"NodeColor","red","XData",X(fv>0,1),"YData",X(fv>0,2),"EdgeColor","black","NodeLabel",[])
title("Cluster : C_1")
```



```
plot(graph(W2),"NodeColor","yellow","XData",X(fv<0,1),"YData",X(fv<0,2),"EdgeColor","black","NodeLabel",[])
title("Cluster : C_2")
```



```
if(PUBLISH == ready)
    path = '/media/user/DATA4LINUX/new1/Repos/Mine/MFC4_22MAT230/';
    mlxfile = matlab.desktop.editor.getActive().Filename;
    [~, name, ext] = fileparts(mlxfile);
    outfile = [path, name, ext,'.pdf']
    export(matlab.desktop.editor.getActive().Filename, outfile);
    if ispc
        winopen(outfile);
    elseif ismac
        system(['open ' char(outfile)]);
    else
        system("env -u LD_LIBRARY_PATH xdg-open '" + outfile + "' &");
    end
end
```