Math 285 Homework4 Fall2015

Jingmei Lu 11/27/2015

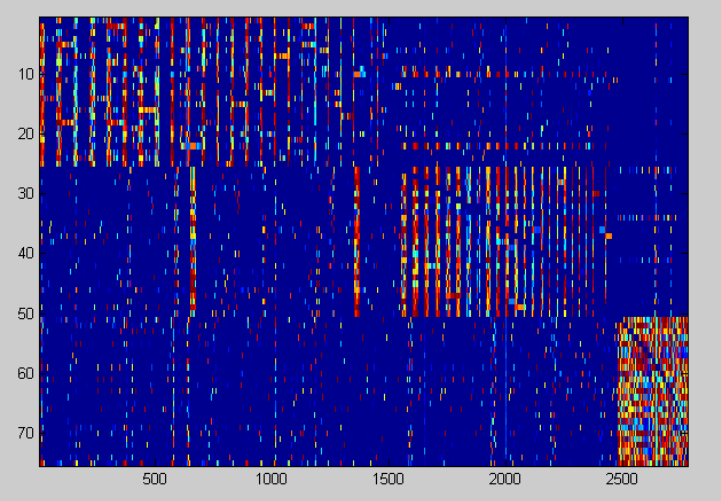
Problem 1

(a)

code

```
%% problem (a) compute Als
n = 75;
sigma = 0.01;
for i = 1:n
    index = 0;
    for j = 1:n-1
        for k = j+1:n
            index = index + 1;
            Y = [X(i,:); X(j,:); X(k,:)];
            center = mean(Y,1); % average of the rows
            Y_tilde = Y - repmat(center, 3, 1); % move center to origin point
            S = svd(Y_tilde,'econ');
            els = S(2);
            if i == j || i == k
                A_ls(i,index) = 0;
            else
                A_ls(i,index) = exp(-(els^2)/(2*(sigma^2)));
            end
        end
    end
end

imagesc(A_ls)
```

Result

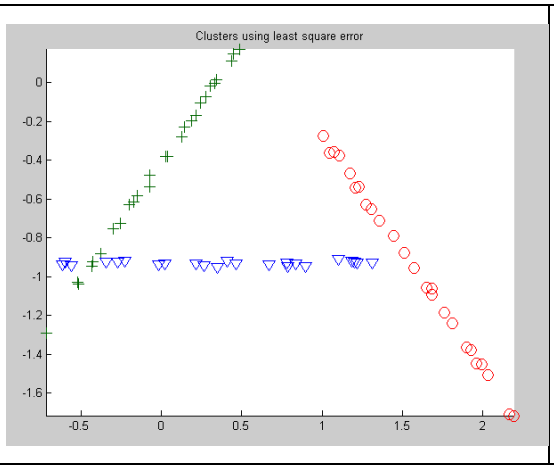


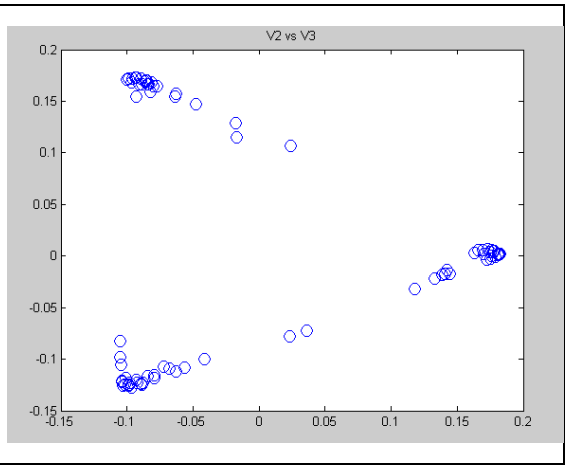
(b)

Code

```
%% problem (b) Apply multiway Ncut to W_ls
W_ls = A_ls * transpose(A_ls);

% find k smallest eigenvectors
De_ls = diag(sum(W_ls,2));
I_ls = eye(size(De_ls,1));
L_ls = I_ls - De_ls\W_ls;

[V_ls,D_ls] = eig(L_ls);
[B_ls, ind_ls] = sort(diag(D_ls),'ascend');
[B_ls, V_ls(ind_ls)];

labels_ls = kmeans(V_ls(:,2:3), 3, 'Replicates', 10);
figure; gcplot(X, labels_ls); title('Clusters using least square error')

error_percentage_ls =
computing_percentage_of_misclassified_points(labels_ls,trueLabels)

[~, ~, intravars_ls] = kmeans(V_ls(:,2:3), 3, 'Replicates', 10);
scatter_ls = sum(intravars_ls)
```

Result





Error rate

error_percentage_ls =	0.0267

Total Scatter

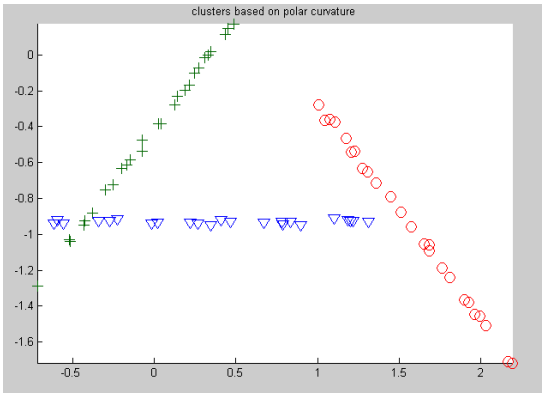
scatter_ls =	0.0783

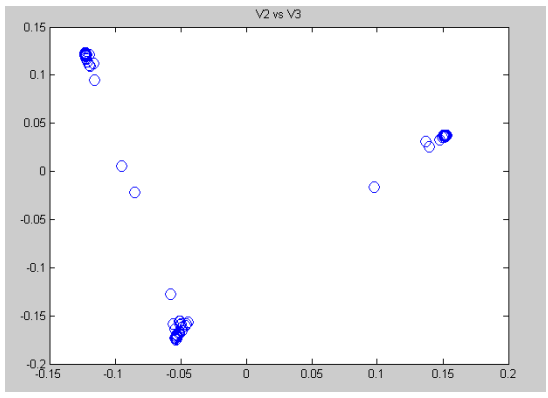
(c)

code

```
%% problem (c)
W_pc = A_pc * transpose(A_pc);

% find k smallest eigenvectors
De_pc = diag(sum(W_pc,2));
I_pc = eye(size(De_pc,1));
L_pc = I_pc - De_pc\W_pc;

[V_pc,D_pc] = eig(L_pc);
[B_pc, ind_pc] = sort(diag(D_pc),'ascend');
[B_pc, V_pc(ind_pc)];
figure; plot(V_pc(:,2), V_pc(:,3), 'o', 'MarkerSize', 10); title('V2 vs
V3')

labels_pc = kmeans(V_pc(:,2:3), 3, 'Replicates', 10);
figure; gcplot(X, labels_pc); title('clusters based on polar curvature')
error_percentage_pc =
computing_percentage_of_misclassified_points(labels_pc,trueLabels)
[~, ~, intravars_pc] = kmeans(V_pc(:,2:3), 3, 'Replicates', 10);
scatter_pc = sum(intravars_pc)
```

Result





Error rate

```
error_percentage_pc =       0.0267
```

Total Scatter

```
scatter_pc =       0.0409
```

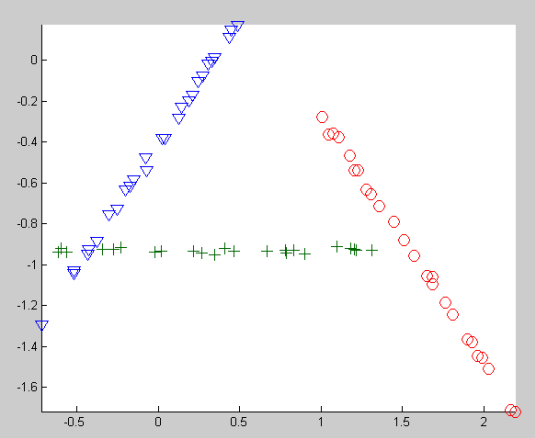
From V2 vs V3 plot of (b) and (c), we can find that the points in (c) are more concentrated. And the total scatter of Apc is smaller than the total scatter of A_{LS}, so Apc is better.

(d)

code

```
[sampleLabels,averageL2Error] = scc(X, 1, 3);
figure; gcplot(X, sampleLabels)
error_percentage_down =
computing_percentage_of_misclassified_points(sampleLabels,trueLabels)
```

Result



Error rate

```
error_percentage_down =       0.0267
```

(e)

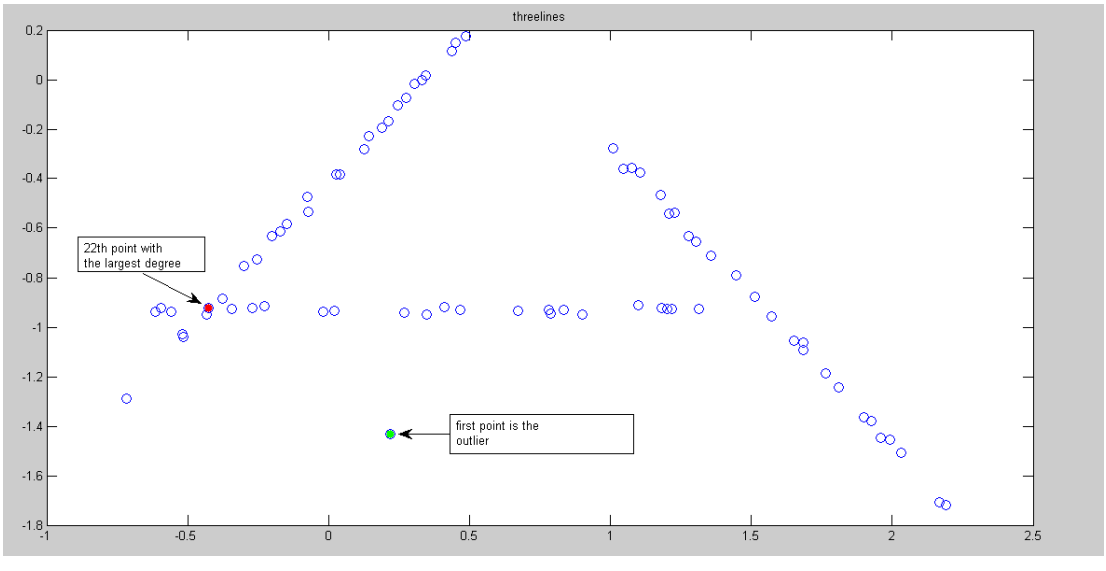
code

```
%% problem (e)
X(1,:) = X(1,:) - [0,0.5]
n = 75;
sigma = 0.01;
for i = 1:n
    index = 0;
    for j = 1:n-1
        for k = j+1:n
            index = index + 1;
            Y = [X(i,:); X(j,:); X(k,:)];
            center = mean(Y,1); % average of the rows
            Y_tilde = Y - repmat(center, 3, 1); % move center to origin point
            [U,S,V] = svd(Y_tilde,'econ');
            els = S(2,2);
            if i == j || i == k
                A(i,index) = 0;
            else
                A(i,index) = exp(-(els^2)/(2*(sigma^2)));
            end
```

```
        end
    end
end

W = A * transpose(A)
D = sum(W,2)
[D_sort, index] = sort(D)
index(1)
index(75)

figure; plot(X(:,1), X(:,2), 'o', 'MarkerSize', 8); title('threelines')
hold on
scatter(X(index(1),1), X(index(1),2), 'green', 'filled')
scatter(X(index(75),1), X(index(75),2), 'red', 'filled')
hold off
```

Result

```
>> index(1)
ans =        1
>> index(75)
ans =       22
```



Since index is the index of the sorted degree, D(1) is the least degree and D(75) is the largest degree, so index(1) is the index of point with the least degree, which is outlier. index(75) is the index of point with the largest degree, which is the 22th point. I also label these two points in the plot.

problem 2

(a)

equation for line1: $a_1x + b_1y = 0$

equation for line2: $a_2x + b_2y = 0$

equation for line3: $a_3x + b_3y = 0$

So overall the points in the union of the three lines must all satisfy the following equation:

$$(a_1x + b_1y)(a_2x + b_2y)(a_3x + b_3y) = 0$$

We can expand above equation to the form:

$$c_{11}x^3+c_{12}x^2y+c_{21}xy^2+c_{22}y^3=0$$

We should fit above polynomial to the entire data set.

(b)

equation for plane1: $a_1x + b_1y + c_1z = 0$

equation for plane2: $a_2x + b_2y + c_2z= 0$

So the polynomial we should fit to the entire data set is:

$$d_{11}x^2+d_{22}y^2+d_{33}z^2+d_{12}xy+d_{13}xz+d_{23}yz = 0$$

(c)

equation for subspace1:

$$a_1x_1+ b_1x_2+ c_1x_3+ d_1x_4+e_1x_5+f_1x_6+g_1x_7+h_1x_8+j_1x_9+k_1x_{10} = 0$$

equation for subspace2:

$$a_2x_1+ b_2x_2+ c_2x_3+ d_2x_4+e_2x_5+f_2x_6+g_2x_7+h_2x_8+j_2x_9+k_2x_{10} = 0$$

So the polynomial we should fit to the entire data set is:

$$\begin{aligned}
&k_{11}x_1^2+k_{22}x_2^2+k_{33}x_3^2+k_{44}x_4^2+k_{55}x_5^2+k_{66}x_6^2+k_{77}x_7^2+k_{88}x_8^2+k_{99}x_9^2+k_{00}x_{10}^2\\
&+k_{12}x_1x_2+k_{13}x_1x_3+k_{14}x_1x_4+k_{15}x_1x_5+k_{16}x_1x_6+k_{17}x_1x_7+k_{18}x_1x_8\\
&+k_{19}x_1x_9+k_{10}x_1x_{10}+k_{23}x_2x_3+k_{24}x_2x_4+k_{25}x_2x_5+k_{26}x_2x_6+k_{27}x_2x_7+k_{28}x_2x_8\\
&+k_{29}x_2x_9+k_{20}x_2x_{10}+k_{34}x_3x_4+k_{35}x_3x_5+k_{36}x_3x_6+k_{37}x_3x_7+k_{38}x_3x_8+k_{39}x_3x_9\\
&+k_{30}x_3x_{10}+k_{45}x_4x_5+k_{46}x_4x_6+k_{47}x_4x_7+k_{48}x_4x_8+k_{49}x_4x_9+k_{40}x_4x_{10}+k_{56}x_5x_6+k_{57}x_5x_7\\
&+k_{58}x_5x_8+k_{59}x_5x_9+k_{50}x_5x_{10}+k_{67}x_6x_7+k_{68}x_6x_8+k_{69}x_6x_9+k_{60}x_6x_{10}+k_{78}x_7x_8\\
&+k_{79}x_7x_9+k_{70}x_7x_{10}+k_{89}x_8x_9+k_{80}x_8x_{10}+k_{90}x_9x_{10}=0
\end{aligned}$$

(d)

equation for line1: $a_1x + b_1y + c_1 = 0$

equation for line1: $a_2x + b_2y + c_2 = 0$

So the polynomial we should fit to the entire data set is:

$$d_{11}x^2 + d_{22}y^2 + d_{12}xy + d_1x + d_2y + d_3 = 0$$

(e)

To define a line in R^{10}, we need 9 equations, each equation has 10 unknowns. Then if we want to use GPCA to cluster two lines in R^{10}, we need to union 18 equations together. So the problem is very difficult.