ICT4Health Lab 4 - Clustering (K-Means)

Table of Contents

In this lab, hard and soft K-means clustering algorithm is implemented on dataset of arrythmia patients, same as Lab 3.

Data Preparation

```
close all;
clear all;
clc;

load('arrhythmia.mat','arrhythmia');

% finding and removing empty columns
s = sum(arrhythmia);
empty_col=find(s==0);
arrhythmia(:,empty_col) = [];

% setting value of last feature = 2 for all values >2
% in order to classify between either 'healthy' or 'arrhythmic'
 patients,
% ignoring different levels of arrhythmia
arrhythmiaAll=arrhythmia;
iii=find(arrhythmia(:,end)>2);
arrhythmia(iii,end)=2;

% Pre processing and normalizing data
y1 = arrhythmia(:,1:end-1); [N,F] = size(y1);
c = arrhythmia(:,end);
ymean = mean(y1); yvar = var(y1); o = ones(N,1);
y = (y1-o*ymean)./sqrt(o*yvar);

iii=find(c==1); jjj=find(c==2);
y1=y(iii,:);    y2=y(jjj,:);
x1=mean(y1);    x2=mean(y2);
xmeans = [x1;x2];
```

Hard K-means algorithm

```
K = 2; % no. of clusters
```

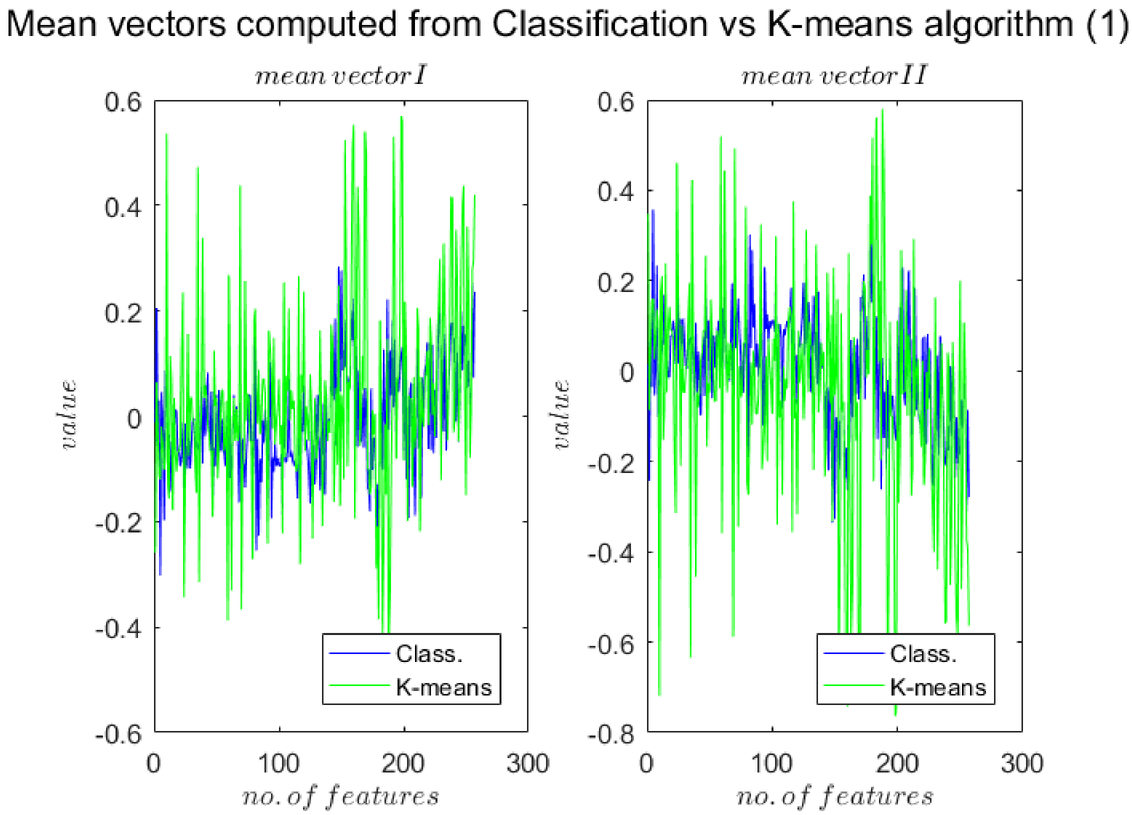
Initial decision vector(dec), using mean vectors from classification

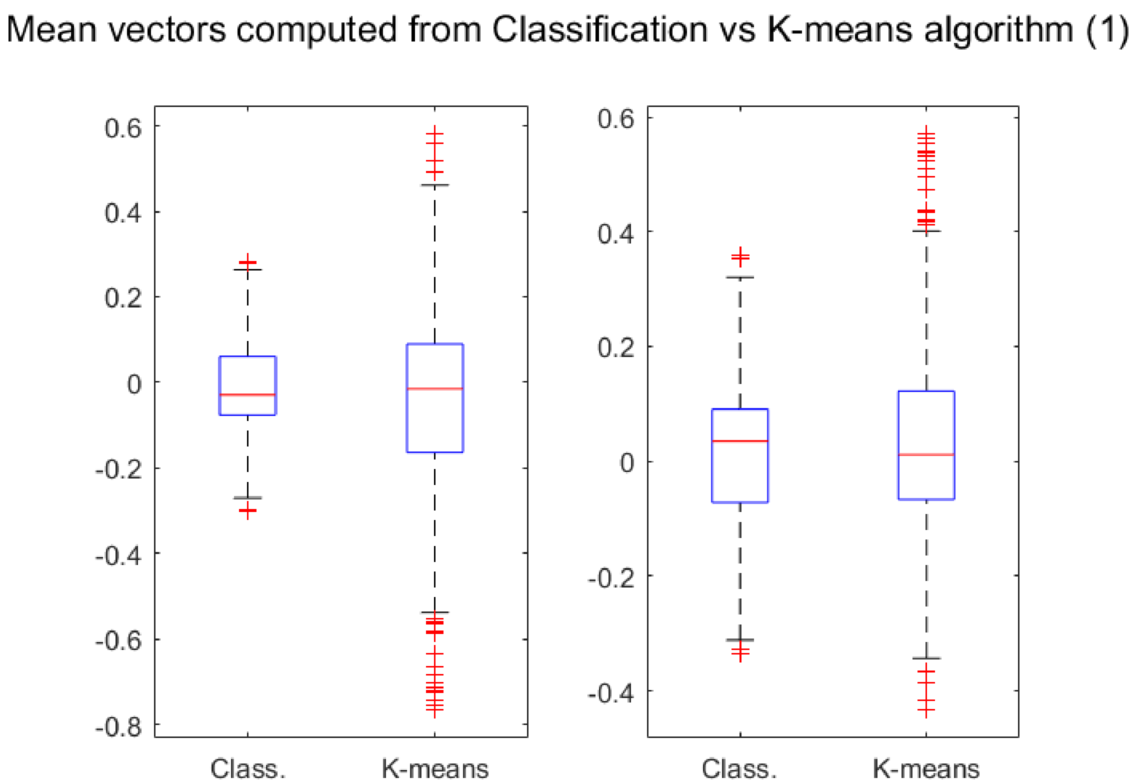
```
[~,dec] = max(bsxfun(@minus,y*xmeans',dot(xmeans,xmeans,2)'/2),[],2);
```

```
prev_dec = 0;
while any(dec ~= prev_dec)
    alloc = sparse(dec,1:N,1,K,N,N); % Allocating each
 observation(patient) to cluster
    means = (spdiags(1./sum(alloc,2),0,K,K)*alloc)*y; %New means for
 each cluster
    prev_dec = dec;
    [~,dec] = max(bsxfun(@minus,y*means',dot(means,means,2)'/2),
[],2); %Finding minimum distance
end
xs=1:F;

figure
subplot(1,2,1)
plot(xs,xmeans(1,:),'b',xs,means(1,:),'g')
title('$mean\,vector I$','Interpreter','latex')
xlabel('$no.\,of\,features$','Interpreter','latex')
ylabel('$value$','Interpreter','latex')
legend('Class.','K-means','Location','southeast')
subplot(1,2,2)
plot(xs,xmeans(2,:),'b',xs,means(2,:),'g')
title('$mean\,vector II$','Interpreter','latex')
xlabel('$no.\,of\,features$','Interpreter','latex')
ylabel('$value$','Interpreter','latex')
legend('Class.','K-means','Location','southeast')
suptitle('Mean vectors computed from Classification vs K-means
 algorithm (1)')
% variance of the new mean vectors computed by k-means algo is almost
 two
% to three times larger than mean vectors from classification
cmp_vec1 = [xmeans(1,:);means(2,:)];
cmp_vec2 = [xmeans(2,:);means(1,:)];
figure
subplot(1,2,1)
boxplot(cmp_vec1','Labels',{'Class.','K-means'})
subplot(1,2,2)
boxplot(cmp_vec2','Labels',{'Class.','K-means'})
suptitle('Mean vectors computed from Classification vs K-means
 algorithm (1)')


cmp=(dec==c);
result = sum(cmp)/N; % 0.5752
% result stores the ratio of k-means decisions matched with doctors
```


Mean vectors computed from Classification vs K-means algorithm (1)
mean vector I
mean vector II
value
no. of features
Class.
K-means


Mean vectors computed from Classification vs K-means algorithm (1)
Class.
K-means

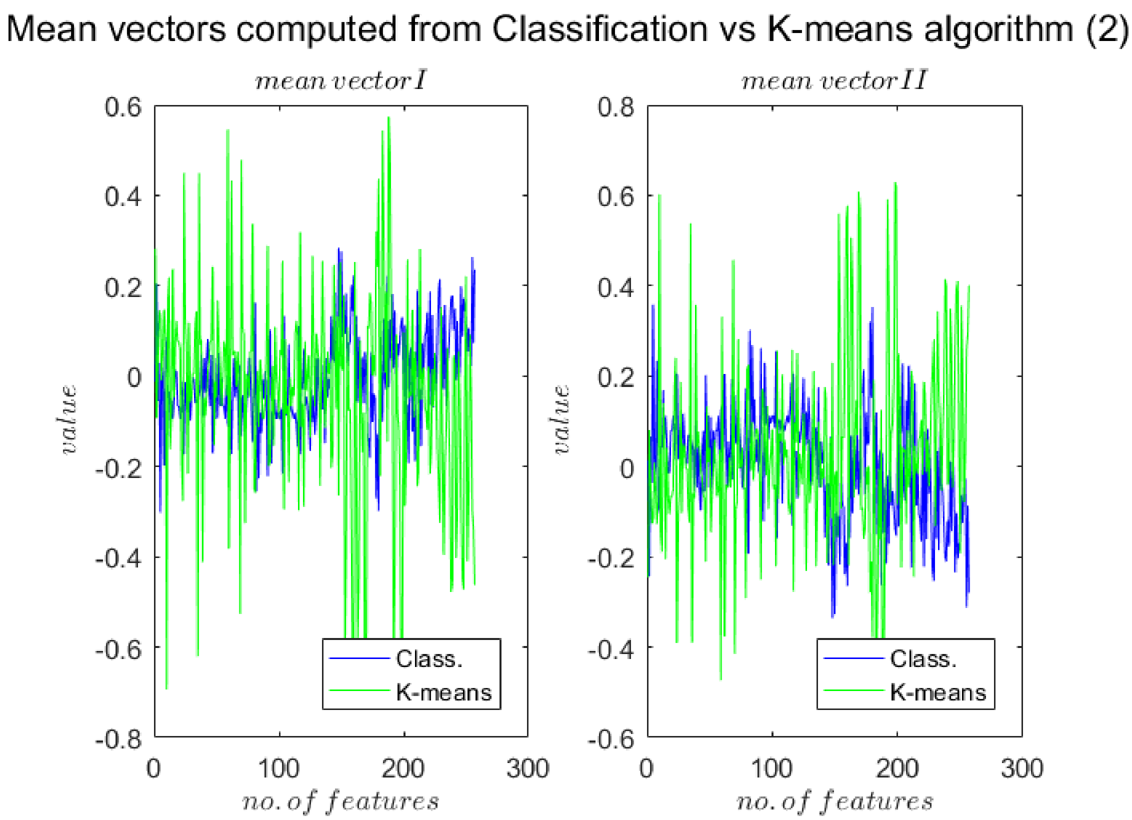
Starting with random init vectors

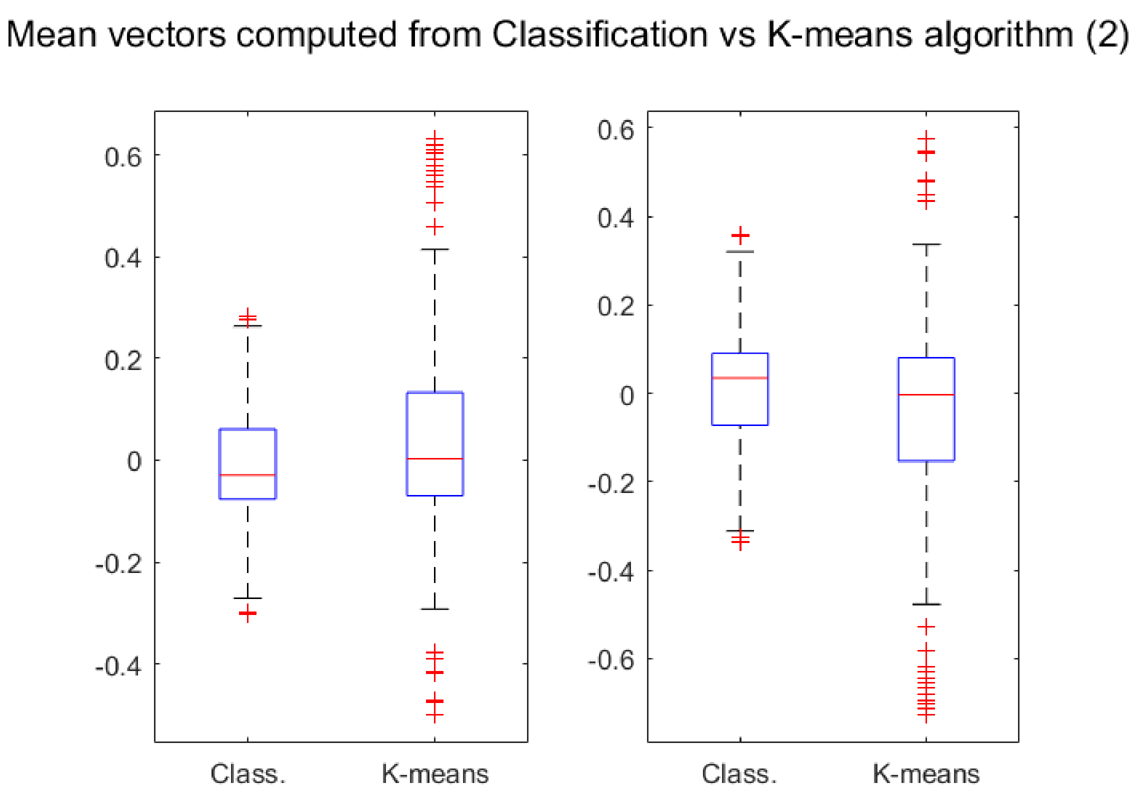
```
dec2 = ceil(rand(N,1)*K);
prev_dec = 0;
while any(dec2 ~= prev_dec)
    alloc = sparse(dec2,1:N,1,K,N,N); % Allocating each
 observation(patient) to cluster
    means = (spdiags(1./sum(alloc,2),0,K,K)*alloc)*y; %New means for
 each cluster
    prev_dec = dec2;
    [~,dec2] = max(bsxfun(@minus,y*means',dot(means,means,2)'/2),
[],2); %Finding minimum distance
end

figure
subplot(1,2,1)
plot(xs,xmeans(1,:),'b',xs,means(1,:),'g')
title('$mean\,vector I$','Interpreter','latex')
xlabel('$no.\,of\,features$','Interpreter','latex')
ylabel('$value$','Interpreter','latex')
legend('Class.','K-means','Location','southeast')
subplot(1,2,2)
plot(xs,xmeans(2,:),'b',xs,means(2,:),'g')
title('$mean\,vector II$','Interpreter','latex')
xlabel('$no.\,of\,features$','Interpreter','latex')
ylabel('$value$','Interpreter','latex')
legend('Class.','K-means','Location','southeast')
suptitle('Mean vectors computed from Classification vs K-means
 algorithm (2)')

cmp_vec1 = [xmeans(1,:);means(2,:)];
cmp_vec2 = [xmeans(2,:);means(1,:)];
figure
subplot(1,2,1)
boxplot(cmp_vec1','Labels',{'Class.','K-means'})
subplot(1,2,2)
boxplot(cmp_vec2','Labels',{'Class.','K-means'})
suptitle('Mean vectors computed from Classification vs K-means
 algorithm (2)')

cmp2=(dec2==c);
result2 = sum(cmp2)/N; % 0.4425
% result stores the ratio of k-means decisions matched with doctors
```


Mean vectors computed from Classification vs K-means algorithm (2)
$mean\ vector\ I$
$mean\ vector\ II$
$value$
$no.\ of\ features$
Class.
K-means
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
0.8
100
200
300


Mean vectors computed from Classification vs K-means algorithm (2)
Class.
K-means
0.6
0.4
0.2
0
-0.2
-0.4
-0.6

Clustering with k=4 (number of clusters = 4)

```
K=4;
rng(2);
dec4_1 = ceil(rand(N,1)*K);
rng(3);
dec4_2 = ceil(rand(N,1)*K);
prev_dec1 = 0;
prev_dec2 = 0;

while any(dec4_1 ~= prev_dec1)
    alloc = sparse(dec4_1,1:N,1,K,N,N); % Allocating each
 observation(patient) to cluster
    means = (spdiags(1./sum(alloc,2),0,K,K)*alloc)*y; %New means for
 each cluster
    prev_dec1 = dec4_1;
    [~,dec4_1] = max(bsxfun(@minus,y*means',dot(means,means,2)'/2),
[],2); %Finding minimum distance
end

while any(dec4_2 ~= prev_dec2)
    alloc = sparse(dec4_2,1:N,1,K,N,N); % Allocating each
 observation(patient) to cluster
    means = (spdiags(1./sum(alloc,2),0,K,K)*alloc)*y; %New means for
 each cluster
    prev_dec2 = dec4_2;
    [~,dec4_2] = max(bsxfun(@minus,y*means',dot(means,means,2)'/2),
[],2); %Finding minimum distance
end
ii{1} = find(dec4_1==1); jj{1} = find(dec4_2==1);
ii{2} = find(dec4_1==2); jj{2} = find(dec4_2==2);
ii{3} = find(dec4_1==3); jj{3} = find(dec4_2==3);
ii{4} = find(dec4_1==4); jj{4} = find(dec4_2==4);

matcmp = zeros(4,4);
for i=1:4
    for j=1:4
        matcmp(i,j) = length(intersect(ii{i},jj{j}));
    end
end
% using intersect(A,B) to find common elements in each cluster and
 then
% listing them in matrix form we obtain following matrix
%      j1   j2  j3  j4
%    _ _ _ _ _ _ _ _ _
% i1 | 75    4  58   0
% i2 | 60   29   4   2
% i3 |  1  142   4   5
% i4 |  6    1   1  60
% we can observe that some clusters are almost similar and some are
% somewhat similar
```

Soft K-means clustering

```
R = y'*y/N;
[U,D] = eig(R);

d=diag(D);d1=d/sum(d);d1c=cumsum(d1);
rem_eig=1e-3; nrem=(d1c<rem_eig);
UL=U; UL(:,nrem)=[];
z=y*UL; z=z./(o*sqrt(var(z)));
[N,F] = size(z);

% Clustering with 2 classes
K = 2;
rng(4);
dec_m = ceil(rand(N,1)*K);
pis = ones(K,1)*(1/K);
varK = ones(K,1);
prev_decm = 0;
while any(dec_m ~= prev_decm)
    alloc = sparse(dec_m,1:N,1,K,N,N); % Allocating each
 observation(patient) to cluster
    means = (spdiags(1./sum(alloc,2),0,K,K)*alloc)*z; %New means for
 each cluster
    prev_decm = dec_m;

    rhoz=z*means';
    en1=diag(z*z'); en2=diag(means*means');
    [Uy,Vy] = meshgrid(en2,en1);
    distz=Uy+Vy-2*rhoz;
    %[dist,decz]=min(distz,[],2);
    %dist = abs(bsxfun(@minus,z*means',dot(means,means,2)'/2));
    mat1 = spdiags(1./(2*varK),0,K,K);
    mat2 = distz*mat1;

    mat11 = spdiags(varK,0,K,K);

    varKm = log(pis./((2*pi*varK).^(F/2)));
    new_dist = bsxfun(@minus,mat2,varKm');
    [~,dec_m] = min(new_dist,[],2);

    % pis update
    Nk = sum(alloc,2);
    pis = Nk/N;

    % varK update
    for i=1:K
        list = find(alloc(i,:) == 1);
        zn = z(list,:);
        diff_s = sum(diag((zn - means(i,:))*(zn + means(i,:))'));
        varK(i) = diff_s/((Nk(i)-1)*F);
    end
end
resultm = sum((dec_m==c))/N; % 0.6659
```

Published with MATLAB® R2016b