```
%with time dependent transition matrix
clear all;

global testdata;
global TN;
global dim;
iter=0;

data = xlsread('F:\EPOXYDOUBLEVOID6KVPHIQ5DEGFILE011.xlsx');
TN = 30;
dim =2 ;
kluster = 18;
testdata(1:TN,:) = data(1:TN,:);

weight(1,:) = data(1,:);
weight(2,:)= data(3,:);
weight(3,:) = data(5,:);
weight(4,:)= data(6,:);
weight(5,:) = data(8,:);
weight(6,:) = data(10,:);
weight(7,:)= data(11,:);
weight(8,:)= data(13,:);
weight(9,:) = data(15,:);
weight(10,:) = data(16,:);
weight(11,:) = data(18,:);
weight(12,:) = data(20,:);
weight(13,:) = data(21,:);
weight(14,:) = data(23,:);
weight(15,:) = data(25,:);
weight(16,:) = data(26,:);
weight(17,:) = data(28,:);
weight(18,:) = data(30,:);


weight_old = zeros(size(weight));       % create a zero matrix of same diment↙
ions of weight
difference = weight - weight_old;       % initialise difference.

iter_count = 1;                         % initialise iteration count.

while sum(sum(difference)) ~= 0 & iter_count ~= 500
    weight_old = weight;                % remember the weights of previous it↙
erations.

    for ii = 1:TN
        for jj = 1:kluster
```

```
            eq_dist(jj) = ((testdata(ii,:)-weight(jj,:)) * ((testdata(ii,:)-w
eight(jj,:))')) ;  % equiledian distance
        end
        [temp,near_class(ii)] = min(eq_dist);    % find the cluster which is
in minimum distance from the training exempler.
    end

    for ii = 1:kluster
        [a,b] = find(near_class == ii);
        temp_sum = 0;
        for jj = b
            temp_sum = temp_sum + testdata(jj,:);
        end
        if sum(a) == 0
            iter_count
            ii
        end
        weight(ii,:) = temp_sum / sum(a);
    end
    difference = abs(weight - weight_old);
    iter_count = iter_count+1;
    %----------------------------------min dist
end
for i=1:kluster
    for j=1:TN
        eq_dist(j)=sqrt((weight(i,:)-testdata(j,:))*((weight(i,:)-testdata(j,
:))'));
    end
    [temp,j1]=min(eq_dist);
    g(i,:)=testdata(j1,:);
end
%---------------------------------------
for i=1:kluster
    l=1;d=1;
    for j=1:144
        g1(i,l,d)=g(i,j);
        d=d+1;
        if d==3
            d=1;
            l=l+1;
        end
    end
end
%-------------------------------initial model
kluster=72;
o=18;
```

```
for k=1:o
    clear g
    g(:,:)=g1(k,:,:);
st=4;
for i=1:st
    c(i,:)=g(i,:);
end
cl=1;
while cl==1
    cl=1;
    for i=1:st
        for j=1:kluster
            dm(i,j)=sqrt((c(i,:)-g(j,:))*((c(i,:)-g(j,:))'));
        end
    end
    sm=zeros(st,kluster);
    [temp,temp1]=min(dm);
    for i=1:kluster
        sm(temp1(i),i)=1;
    end
    tsm=sum(sm,2);
    for i=1:st
        if tsm(i,1)>1
            temp2=0;
            for j=1:kluster
                if sm(i,j)==1
                    temp2=temp2+g(j,:);
                end
            end
            c(i,:)=temp2/tsm(i,1);
         end
    end
    for i=1:st
        for j=1:kluster
            dm(i,j)=sqrt((c(i,:)-g(j,:))*((c(i,:)-g(j,:))'));
        end
    end
    sm1=zeros(st,kluster);
    [temp,temp1]=min(dm);
    for i=1:kluster
        sm1(temp1(i),i)=1;
    end
    if sm==sm1
        cl=0;
    end
end
```

```
%-----------------------state seq
ss(k,:)=temp1(1,:);
 end
 display('the state sequences for all observations: ')
 ss
 %-----------------------------------------------------------------
cl1=1;
while cl1==1
    cl1=1;
%-------------------------------mu
clear temp temp1 temp3
for l=1:st
    temp=0;temp1=0;
    for i=1:o
        for j=1:kluster
            if ss(i,j)==l
                temp3(1,:)=g1(i,j,:);
                temp=temp+temp3;
                temp1=temp1+1;
            end
        end
    end
    mu(l,:)=temp/temp1;
end
%-------------------------------cv
clear temp temp3
for l=1:st
    temp=0;temp1=0;
    for i=1:o
        for j=1:kluster
            if ss(i,j)==l
                temp3(1,:)=g1(i,j,:);
                temp=temp+(((temp3(1,:)-mu(l,:))')*(temp3(1,:)-mu(l,:)));
                temp1=temp1+1;
            end
        end
    end
    cv(l,:,:)=temp/temp1;
end
%-----------------------------------tm(a)
clear a
clear temp temp1 temp2
temp=kluster-1;
for i=1:temp
    for j=1:st
        for k=1:st
```

```
            temp1=0;temp2=0;
            for l=1:o
                if ss(l,i)==j
                    temp2=temp2+1;
                    if ss(l,i+1)==k
                        temp1=temp1+1;
                    end
                end
            end
            if temp2==0
                a(i,j,k)=0;
            else
            a(i,j,k)=temp1/temp2;
            end
        end
    end
end
a1=a;
%---------------------------------py
for i=1:st
    temp=0;
    for j=1:o

            if ss(j,1)==i
                temp=temp+1;
            end

    end
    py(i)=(temp/o);
end
%---------------------------------b
clear temp temp1 temp2 temp3 b
for i=1:o
    for j=1:st
        temp(:,:)=cv(j,:,:);
        temp1=(det(temp))^0.5;
        for k=1:kluster
            temp2(1,:)=g1(i,k,:);
            temp3(:,:)=inv(temp);
            b(i,j,k)=((1/(((2*pi)^(dim/2))*temp1))*exp(-0.5*(temp2-mu(j,:))*(↙
temp3)*((temp2-mu(j,:))')));
            check=((1/(((2*pi)^(dim/2))*temp1))*exp(-0.5*(temp2-mu(j,:))*(tem↙
p3)*((temp2-mu(j,:))')));
        end
    end
end
```

```
%--------------------------------viterbi
clear temp temp1 temp2 temp3 a
for l=1:o
b1(:,:)=b(l,:,:);
for i=1:st
    del(1,i)=py(i)*b1(i,1);
    si(1,i)=0;
end
for i=2:kluster
    a(:,:)=a1(i-1,:,:);
    for j=1:st
        temp=0;
        for k=1:st
            temp(k)=del((i-1),k)*a(k,j);
        end
        [temp1,temp2]=max(temp);
        del(i,j)=temp1*b1(j,i);
        si(i,j)=temp2;
    end
end
t=kluster;clear temp;
temp=del(t,:);
[temp1,temp2]=max(temp);
psta=temp1;
ssta(t)=temp2;
for i=1:(kluster-1)
    t=t-1;
    ssta(t)=si((t+1),ssta(t+1));
end
sstar(l,:)=ssta(:);
end
display('new state sequence');
sstar
%------------------------
if ss==sstar
    cl1=0;
end
ss=sstar;
iter=iter+1;
end
%---------------------------------------------------------b2(b)
for i=1:o
    l=1;
    for j=1:kluster
        for k=1:st
            b2(i,l)=b(i,k,j);
            l=l+1;
        end
    end
end
%----------------------------
for i=1:o
    for j=1:kluster
        b3(i,j)=b(i,ss(i,j),j);
    end
end
%----------------------------
iter
```