APPENDIX C

SCORE CODE OF SINGULAR SPECTRUM ANALYSIS

APPENDIX C

Score Code of Singular Spectrum Analysis

```
% -----------------------------------------------------------------
x1=load('D:\ไฟล์ input ของแต่ละสถานี\KGT25.txt')
for L=12:31
fName1 = 'm'
fName2 = 'p'
fName1 = [fName1 int2str(L) fName2]
fName3 = 'KGT25.txt'
for q=1:L
   fName  = [fName1 int2str(q) fName3]
   I=[1:q]
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% -----------------------------------------------------------------
%    Author: Francisco Javier Alonso Sanchez    e-mail:fjas@unex.es
%    Departament of Electronics and Electromecanical Engineering
%    Industrial Engineering School
%    University of Extremadura
%    Badajoz
%    Spain
% -----------------------------------------------------------------
%
% SSA generates a trayectory matrix X from the original series x1
% by sliding a window of length L. The trayectory matrix is aproximated
% using Singular Value Decomposition. The last step reconstructs
% the series from the aproximated trayectory matrix. The SSA applications
% include smoothing, filtering, and trend extraction.
% The algorithm used is described in detail in: Golyandina, N., Nekrutkin,
% V., Zhigljavsky, A., 2001. Analisys of Time Series Structure - SSA and
% Related Techniques. Chapman & Hall/CR.
% x1 Original time series (column vector form)
% L  Window length
% y  Reconstructed time series
% r  Residual time series r=x1-y
% vr Relative value of the norm of the approximated trajectory matrix with respect
%     to the original trajectory matrix
% The program output is the Singular Spectrum of x1 (must be a column vector),
% using a window length L. You must choose the components be used to reconstruct
%the series in the form [i1,i2:ik,...,iL], based on the Singular Spectrum appearance.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% Step1 : Build trayectory matrix
            N=length(x1);
            if L>N/2;L=N-L;end
            K=N-L+1;
            X=zeros(L,K);
            for i=1:K
```

```
        X(1:L,i)=x1(i:L+i-1);
        end
% Step 2: SVD
    S=X*X';
    [U,autoval]=eig(S)
    [d,i]=sort(-diag(autoval))
    d=-d;
    U=U(:,i)
    sev=sum(d)
    %plot((d./sev)*100),hold on,plot((d./sev)*100,'rx');
    %title('Singular Spectrum');xlabel('Eigenvalue Number');ylabel('Eigenvalue (%
    Norm of trajectory matrix retained)')
    V=(X')*U;
    rc=U*V';
    % Step 3: Grouping
    %I=input('Choose the agrupation of components to reconstruct the series
    %in the form I=[i1,i2:ik,...,iL]  ')
    Vt=V';
     rca=U(:,I)*Vt(I,:);
    % Step 4: Reconstruction
    y=zeros(N,1);
    Lp=min(L,K);
    Kp=max(L,K);
    for k=0:Lp-2
    for m=1:k+1;
    y(k+1)=y(k+1)+(1/(k+1))*rca(m,k-m+2);
    end
     end
    for k=Lp-1:Kp-1
    for m=1:Lp;
    y(k+1)=y(k+1)+(1/(Lp))*rca(m,k-m+2);
    end
    end
    for k=Kp:N
    for m=k-Kp+2:N-Kp+1;
    y(k+1)=y(k+1)+(1/(N-k))*rca(m,k-m+2);
    end
    end
    %figure;subplot(2,1,1);hold on;xlabel('Data poit');ylabel('Original and
    reconstructed series')
    %plot(x1);grid on;plot(y,'r')
    r=x1-y;
    %subplot(2,1,2);plot(r,'g');xlabel('Data poit');ylabel('Residual series');grid on
    %vr=(sum(d(I))/sev)*100;
    dlmwrite(fName,y)
    end
    end
```