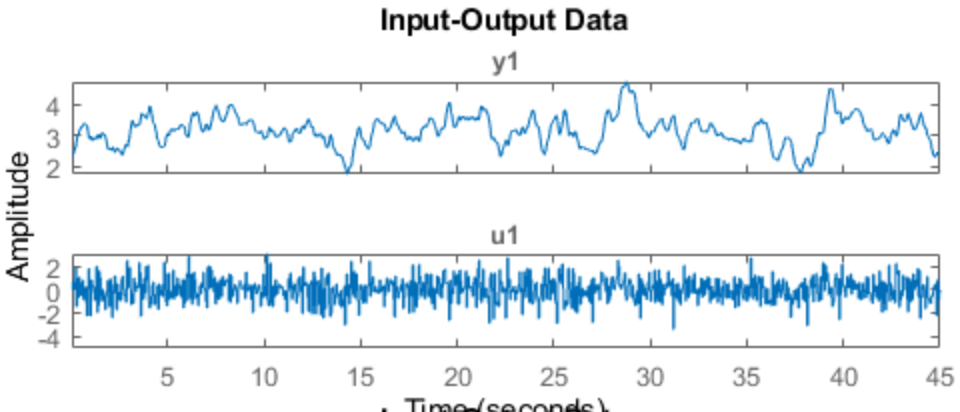
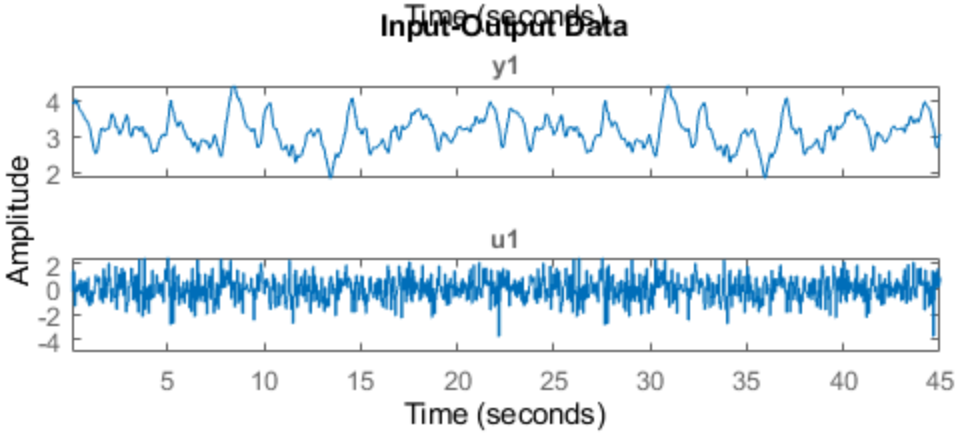
```
clear all, close all
load("iddata-18.mat");
subplot(211), plot(id)
subplot(212), plot(val)

index_plot=1; na=2; m=5;
%predictie

                phi= Neli(id.u,id.y,m,na,-1); marime_phi=size(phi);
                teta=phi\id.y;

                phi_val=Neli(val.u,val.y,m,na,-1);
                yhat=phi_val*teta;
                yhat_id=phi*teta;

        %eroare predictie identificare
        mse=id.y-yhat_id;
        MSE_PRE_ID=sum(mse.^2)/length(yhat_id);

        figure, plot(yhat_id), hold on, plot(id.y), title('MSE PRE
 identificare=', MSE_PRE_ID)

        %eroare predictie validare
        mse=val.y-yhat;
        MSE_PRE=sum(mse.^2)/length(yhat);

        figure, plot(yhat), hold on, plot(val.y), title('MSE PRE validare=',
 MSE_PRE)

   % simulare identificare
                for k=1:length(id.y)
                    if k==1
                        k_sim=Neli(id.u,0,m,na,k);
                    else
                        k_sim=Neli(id.u,Ysim_id,m,na,k);
                    end
                    Ysim_id(k)=k_sim*teta;
                end

            %eroare simulare
            mse=id.y-Ysim_id';
            MSE_SIM_ID=sum(mse.^2)/length(Ysim_id);

            figure, plot(Ysim_id), hold on, plot(id.y), title('MSE SIM
 identificare=', MSE_SIM_ID)

    % simulare validare
            Ysim=zeros(1,3);
                for k=1:length(val.y)
                    if k==1
                        k_sim=Neli(val.u,0,m,na,k);
                    else
```

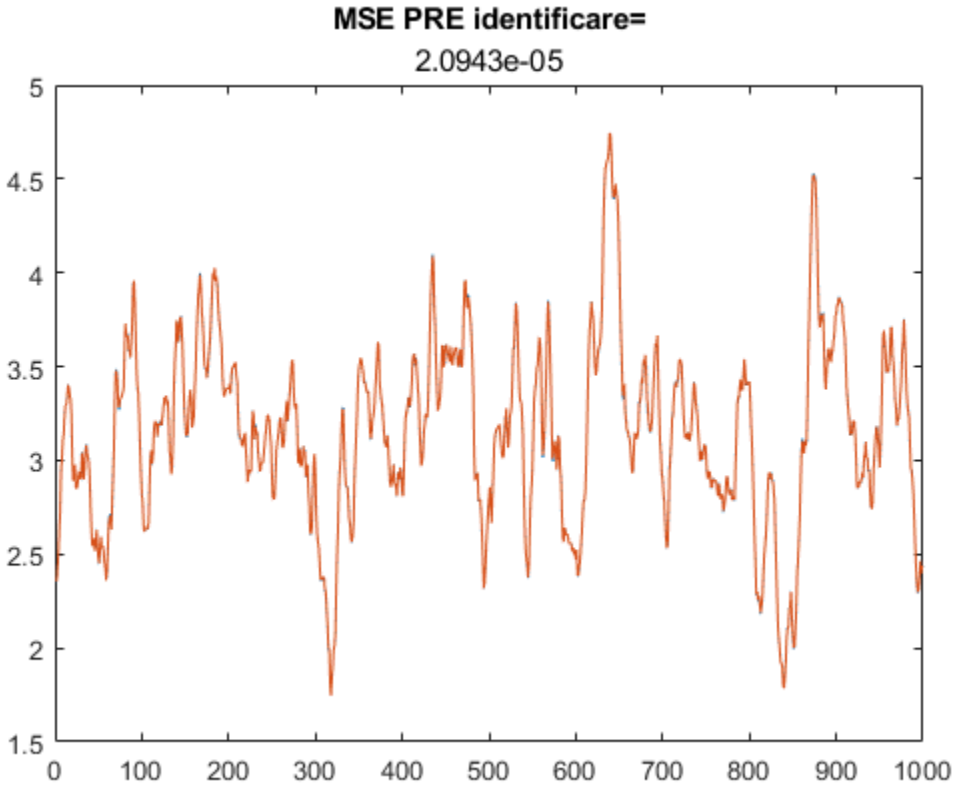
```
                        k_sim=Neli(val.u,Ysim,m,na,k);
                    end
                    Ysim(k)=k_sim*teta;
                end

            %eroare simulare
            mse=val.y-Ysim';
            MSE_SIM=sum(mse.^2)/length(Ysim);

            figure, plot(Ysim), hold on, plot(val.y), title('MSE SIM
 validare=', MSE_SIM)

function Fi= Neli(u,y,m,na, deep)

    if deep<=0
        for n=1:na
            for i=1:length(y)
                if i<=n
                  coloane_y(i,n)=0; coloane_u(i,n)=0;
                else
                  coloane_y(i,n)=y(i-n); coloane_u(i,n)=u(i-n);
                end
            end
        end
        Matri=ones(length(y),1);     %initializre Matri
    else
        for n=1:na
            if deep<=n
              coloane_y(1,n)=0; coloane_u(1,n)=0;
            else
              coloane_y(1,n)=y(deep-n); coloane_u(1,n)=u(deep-n);
            end
    end
        Matri=1;     %initializre Matri
    end



    for pow=1:m
        Matri=[Matri,coloane_y.^pow];   Matri=[Matri,coloane_u.^pow];       %
 aduagare u si y simpli
    end

    Simple=[coloane_u,coloane_y]; Marime_s=size(Simple);
    placebo=Simple;
    prime=1;    Marime_p=size(Simple);

    while Marime_p(1,2)>1
            for secon=prime:Marime_p(1,2)-1
                if prime==1 & secon==1
                    Combinati=Simple(:,prime).*Simple(:,secon+1);
                else
                    Combinati=[Combinati,Simple(:,prime).*Simple(:,secon+1)];
                end
```

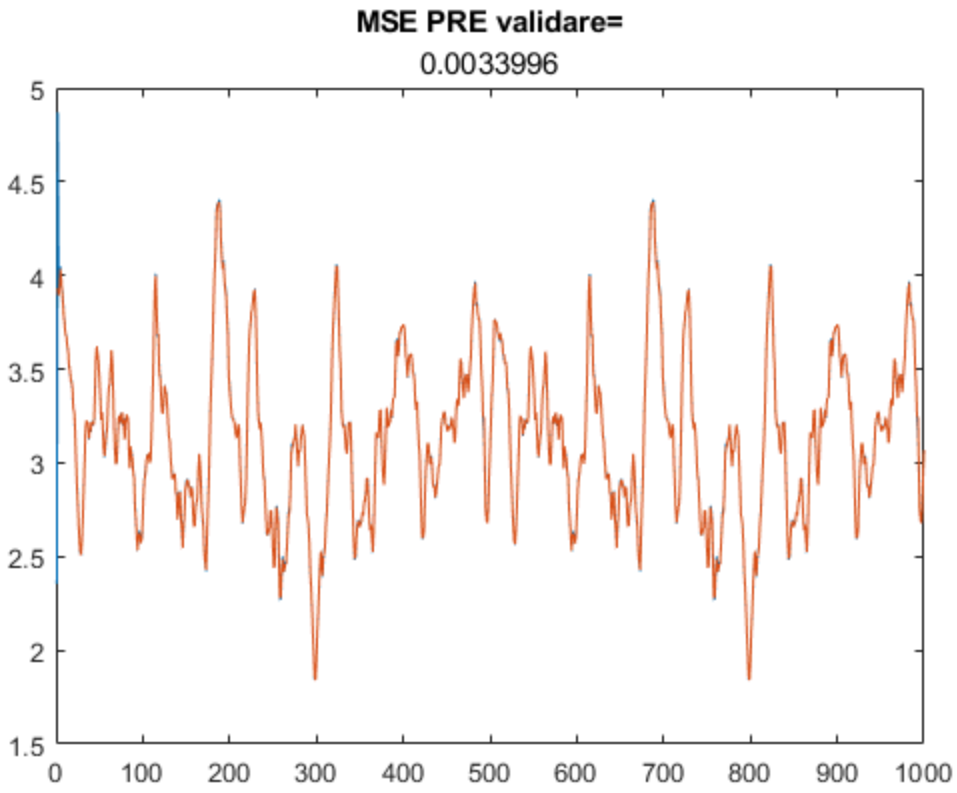
```
            end
            placebo(:,1)=[]; Marime_p=size(placebo); prime=prime+1;
    end

    Marime_c=size(Combinati);

    %combinatile neliniare pana la ordinul 3
        for i=1:Marime_c(1,2)
            for j=1:Marime_s(1,2)
                if i==1 & j==1
                    Combinati2=Combinati(:,1).*Simple(:,1);
                else
                    Combinati2=[Combinati2, Combinati(:,i).*Simple(:,j)];
                end
            end
        end
        FinalCombinati=Combinati2;

      %combinatile neliniare de la ordinul 4 in sus
      Marime_c2=size(Combinati2);
      for ordin=4:m
            for i=1:Marime_c2(1,2)
                for j=1:Marime_s(1,2)
                    if i==1 & j==1
                        Combinati2=Combinati(:,1).*Simple(:,1);
                    else
                        Combinati2=[Combinati2, Combinati2(:,i).*Simple(:,j)];
                    end
                end
            end
            FinalCombinati=[FinalCombinati, Combinati2];
      end


                    if m==1
                        Fi=Matri;
                    elseif m==2
                        Fi=[Matri,Combinati];
                    else
                        Fi=[Matri,Combinati,FinalCombinati];
                    end
end
```

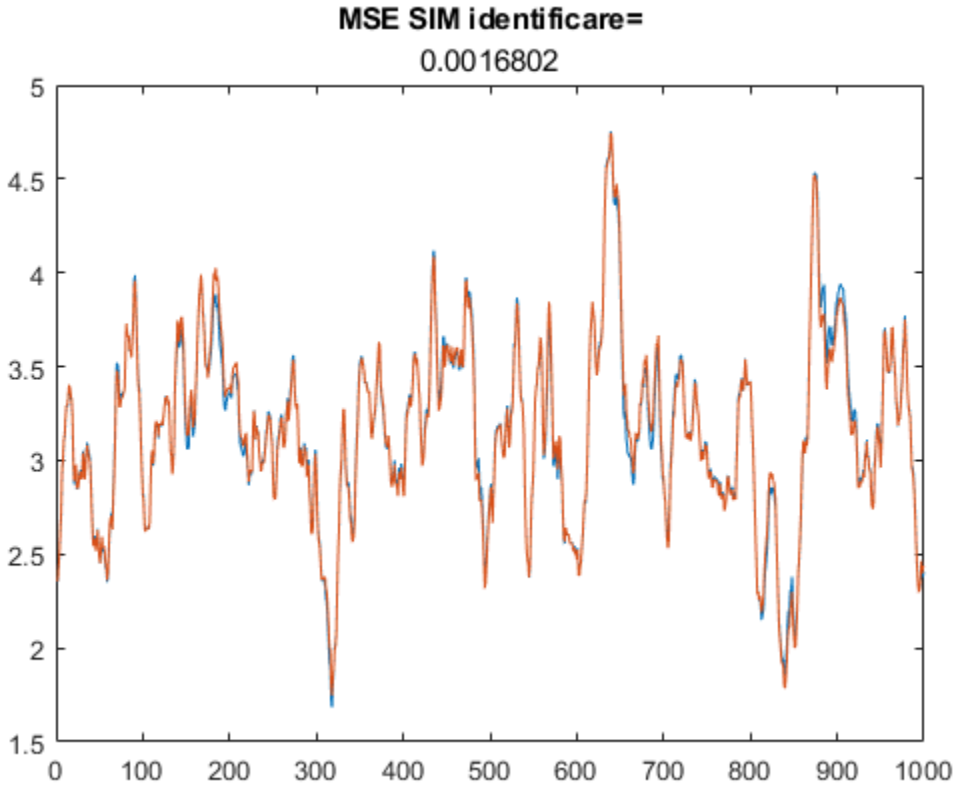
```
Warning: Updating objects saved with previous MATLAB version...
Resave your MAT files to improve loading speed.
Warning: Rank deficient, rank = 68, tol =  3.537366e-09.
```

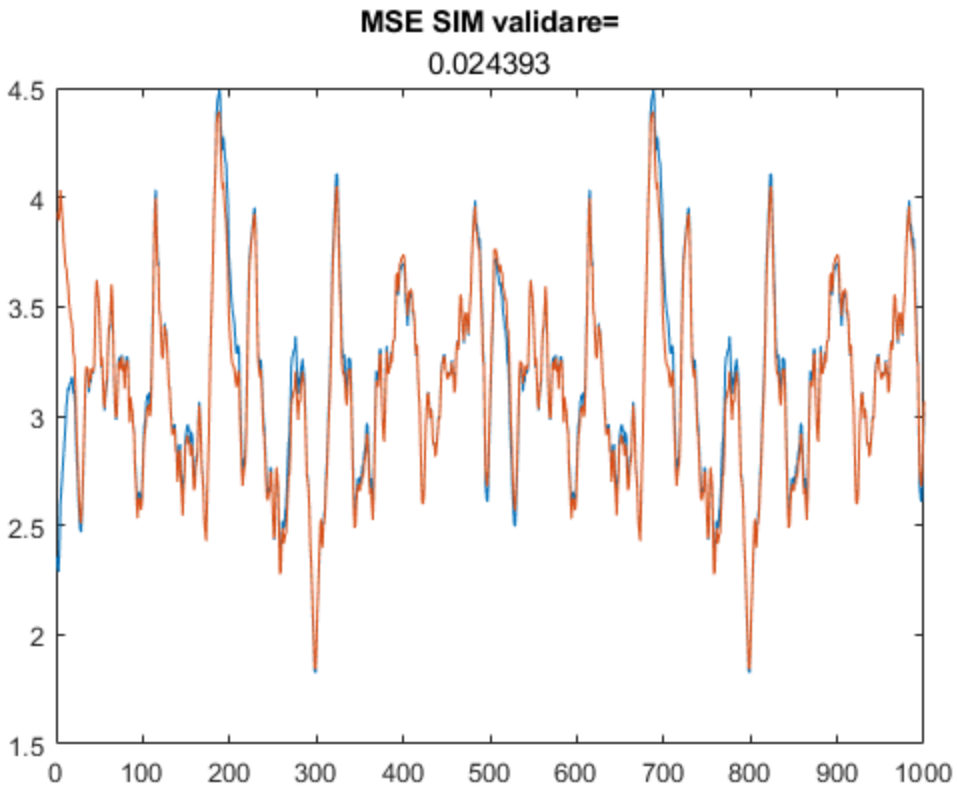

Input-Output Data
y1
u1
Amplitude
Time (seconds)


Input-Output Data
y1
u1
Amplitude
Time (seconds)


MSE PRE identificare=
2.0943e-05


MSE PRE validare=
0.0033996
5
4.5
4
3.5
3
2.5
2
1.5
0
100
200
300
400
500
600
700
800
900
1000


MSE SIM identificare=
0.0016802
5
4.5
4
3.5
3
2.5
2
1.5
0
100
200
300
400
500
600
700
800
900
1000



Published with MATLAB® R2021b