EE211 MATLAB Assign1

Hanlin Cai

FZU 832002117

MU 20122161

**Statement：**

I chose dataset Gr3 and Gr13 to complete the system identification, and all the Code & Pic were created by myself.

**Procedure(1)**

图表, 条形图

描述已自动生成图表, 散点图

描述已自动生成

Fig1-2

图表

描述已自动生成图表, 散点图

描述已自动生成

Fig3-4

**Procedure(2)**

**Code1-1 :**

Use struc & arxstruc (ARX) function to fit the model then verify the fitting accuracy.

%% Assign1-1 2022/05/03 Hanlin Cai

% 使用Matlba内置函数进行拟合，并判断拟合精度

% Use struc & arxstruc (ARX) function to fit the model

% Then verify the fitting accuracy

clear;clc;close all;

%% 1、导入建模训练集

% Import the modeling training set

data1 = importdata('2data/Gr3.txt');

data2 = data1.data;

data3 = data2(1:1000,:);

data4 = data2(2001:3000,:);

t1 = data3(:,1); %time\_1

u1 = data3(:,2); %input\_1

y1 = data3(:,3); %output\_1

t2 = data4(:,1); %time\_2

u2 = data4(:,2); %input\_2

y2 = data4(:,3); %output\_2

% plot(t2,y2,'o')

plot(t1,y1,'-')

xlabel('t')

ylabel('y')

title('Original corresponding curve');

Ts = 1000;

ze = iddata(y1,u1,Ts);

zv = iddata(y2,u2,Ts);

NN = struc(1:3, 1:3, 0:2);

V = arxstruc(ze,zv,NN);

%% 2、使用卷积神经网络，计算模型表现

% Convolutional neural networks are used to compute model representations.

% [u1, t1] = bodyfat\_dataset;

% net = feedforwardnet(10);

% net.performParam.regularization = 0.01;

% net.performFcn

%

% net = train(net, u1, t1);

% y = net(u1);

% perf = perform(net, t2, y2)

%% 3、增加噪音

% Increase the noise

% 1、均匀分布噪声 Uniformly distributed noise

u3 = u1 + 0.1\*(2\*rand(size(u1))-1).\*u1;

% 2、正态分布噪声 Normally distributed noise

u4 = u2 + 0.1\*randn(size(u2)).\*(u2);

%% 4、验证模型精准度

% Verify model accuracy

order = selstruc(V,0);

M1 = arx(ze,order)

% 增加噪音进入信号部分

Z1 = iddata(y1,u1,0.05);

Z2 = iddata(y1,u3,0.05);

Z3 = iddata(y1,u4,0.05);

% Z4 = iddata(y2,u1,0.05);

% M2 = arx(Z1 , [5,6,0])

% compare(M2, Z1);

% hold on;

%% 5、绘制Loss Function

% ident;

% u1 y1

% u2 y2

% u3 y1 noise

For dataset Gr3 (1:1000):

图形用户界面

描述已自动生成

图形用户界面

描述已自动生成图形用户界面

描述已自动生成

Fig5-7

For dataset Gr3 (1001:3000):

图形用户界面

中度可信度描述已自动生成

图形用户界面

描述已自动生成

图形用户界面, 应用程序

描述已自动生成

Fig8-10

**Procedure(3)**

**Code1-2 :**

The intermediate parameters were calculated using the least square method.

%% Assign1-2 2022/05/03 Hanlin Cai

% The least square method was used for fitting

clear;clc;close all;

%% 1、导入建模训练集

% Import the modeling training set

data1 = importdata('2data/Gr3.txt');

data2 = data1.data;

data3 = data2(1:1000,:);

data4 = data2(2001:3000,:);

t1 = data3(:,1); %time\_1

u1 = data3(:,2); %input\_1

y1 = data3(:,3); %output\_1

t2 = data4(:,1); %time\_2

u2 = data4(:,2); %input\_2

y2 = data4(:,3); %output\_2

data4 = importdata('2data/Gr13.txt');

data5 = data4.data;

data6 = data5(1:1000,:);

t3 = data6(:,1); %time\_1

u3 = data6(:,2); %input\_1

y3 = data6(:,3); %output\_1

% % plot(t1,y1,'o')

%% 使用最小二乘法计算中间参数

% The intermediate parameters were calculated using the least square method

% y1 = O\*(u1)

% 定义O为中间变量 Define O as the intermediate variable

u11 = (u1)';

u22 = (u2)';

u33 = (u3)';

% O = ( (u1)'\*u1 )^(-1) \* (u1)' \* y1;

O1 = ( (u11)\*u1 )^(-1) \* (u11) \* y1; % O1 = 0.0789

O2 = ( (u22)\*u2 )^(-1) \* (u22) \* y2; % O2 = 0.0668

O3 = ( (u33)\*u3 )^(-1) \* (u33) \* y3; % O3 = 0.4267

**Procedure(4)**

For dataset Gr3 (1:1000):

图形用户界面

描述已自动生成

Fig11

For dataset Gr3 (1001:3000):

图形用户界面, 应用程序

描述已自动生成

Fig12

**Procedure(5)**

**ARX Function :**

For dataset Gr3 (1:1000):

Discrete-time ARX model: A(z)y(t) = B(z)u(t) + e(t)

A(z) = 1 - 0.08718 z^-1 + 0.08923 z^-2 + 0.0005606 z^-3

B(z) = 0.07952 - 0.3896 z^-1 - 0.4509 z^-2

For dataset Gr3 (1001:3000):

Discrete-time ARX model: A(z)y(t) = B(z)u(t) + e(t)

A(z) = 1 - 0.08825 z^-1 + 0.09134 z^-2 + 0.001028 z^-3

B(z) = 0.07846 - 0.39 z^-1 - 0.4492 z^-2

**Least Square Method :**

For dataset Gr3 (1:1000):

O1 = ( (u11)\*u1 )^(-1) \* (u11) \* y1; % O1 = 0.0789

For dataset Gr3 (1001:3000):

O2 = ( (u22)\*u2 )^(-1) \* (u22) \* y2; % O2 = 0.0668

**Procedure(6)**

%% 3、增加噪音

% Increase the noise

% 1、均匀分布噪声 Uniformly distributed noise

u3 = u1 + 0.1\*(2\*rand(size(u1))-1).\*u1;

% 2、正态分布噪声 Normally distributed noise

u4 = u2 + 0.1\*randn(size(u2)).\*(u2);

For dataset Gr3 (1:1000):

**图形用户界面

描述已自动生成**

Fig13

For dataset Gr3 (1001:3000):

**图形用户界面

描述已自动生成**

Fig14

**Procedure(7)**

For dataset Gr3 (1:1000): FFT

图形用户界面, 图表, 折线图

描述已自动生成

Fig15

For dataset Gr3 (1001:3000): FFT

图形用户界面, 图表, 折线图

描述已自动生成

Fig16

**Code1-3 :**

Using Fast Fourier Transfor to Plot the spectrum of the data

%% Assign1-3 2022/05/03 Hanlin Cai

% FFT快速傅立叶变换，处理函数部分

% Fast Fourier Transform

clear;clc;close all;

%% 1、导入建模训练集

% Import the modeling training set

data1 = importdata('2data/Gr3.txt');

data2 = data1.data;

data3 = data2(1:1000,:);

data4 = data2(1001:2000,:);

t1 = data3(:,1); %time\_1

u1 = data3(:,2); %input\_1

y1 = data3(:,3); %output\_1

t2 = data4(:,1); %time\_2

u2 = data4(:,2); %input\_2

y2 = data4(:,3); %output\_2

Ts = 1000;

ze = iddata(y1,u1,Ts);

zv = iddata(y2,u2,Ts);

NN = struc(1:3, 1:3, 0:2);

V = arxstruc(ze,zv,NN);

%% 验证模型精准度，同A1-1

% order = selstruc(V,0);

% M = iv4(ze,order);

%% 增加噪音

% 1、均匀分布噪声

u2 = u1 + 0.1\*(2\*rand(size(u1))-1).\*u1;

% 2、正态分布噪声

u3 = u1 + 0.1\*randn(size(u1)).\*(u1);

%% FFT函数

% 1、对信号进行fft变换，获取信号频率分布

L=length(u1);

T=(u1(end)-u1(1))/L; % 采样周期等于采样长度除去采样点数

Fs1=1/T; % 采样频率

f1=Fs1\*(0:(L/2))/L;

L=length(u2);

T=(u2(end)-u2(1))/L; % 采样周期等于采样长度除去采样点数

Fs2=1/T; % 采样频率

f2=Fs2\*(0:(L/2))/L;

% 2、傅里叶变换

S1=y1(1:L);

Y1 = fft(S1);

P2 = abs(Y1/L);

P1 = P2(1:L/2+1);

P1(2:end-1) = 2\*P1(2:end-1);

S2=y2(1:L);

Y2 = fft(S2);

P3 = abs(Y2/L);

P4 = P3(1:L/2+1);

P4(2:end-1) = 2\*P4(2:end-1);

% 3、plot

figure;

plot(f1,P1)

% plot(f2,P4)

% title('FFT')

xlabel('f (Hz)')

ylabel('y')

Hanlin Cai

FZU 832002117

MU 20122161

2022/05/03