1. Static System Identifications

* Problem:

1. Solution

1. Bias of LSE

1. is independent and , where
2. Accuracy (Variance) of LSE ( the standard deviation =

* The unknown of the covariance of

The notation :

* : the estimator . : the mean
* ,

1. Dynamic System Identification
   1. Linear Dynamic Systems

Assume the shift operator introduce then

where

%% Kim’ comment

The term is called as a colored noise, where as is called as a white noise

%%%%%%%

* + 1. Transfer Function Models

1. **FIR(finite impulse response)**

with

1. **IIR(infinite impulse response)**

-**ARX(AutoRegressive eXogenous)** :

* Autoregressor: , eXogenous:
* Matlab command

:

+ deadtime(delay time):

- **ARMAX(AutoRegressive Moving Average eXogenous)**

1. General whole class of transfer function

with

++ :

* + 1. Equation Error Identification
* FIR
* The regressor
* The unknown
* The Observed data corresponding the input data

Then

* **Example**

1. Output

++at

++ at t=3

++at

🡪

🡪with these LSE is

🡪the LSE

* **ARX Regressor**

See the text book page 118 for in (6.16)

* Example.

1. Measured data

++ at (Let e(t) delete for simplicity)

++ at t=3

++ at t=4

* LSE model
* ARMAX model in (6.6)
* Problem in ARMAX

-How to find

1) Example

++ at t=1,2,3,4

Find LSE as ARX

++ calculate the residual error

++ Now we have measured data and

* With the regressor matrix, calculate a new estimator as
* Matlab example:

1. FIR

%%%% a sample program

%y(t) = u(t-1) + 0.5u(t-2) + e(t)

A = [1];

B = [0 1 0.5];

sys0 = idpoly(A,B) % original system

% check the sys0 with the

% generate input and noise

u = iddata([ ],idinput(300,'rbs'));

e = iddata([ ],randn(300,1));

% generate output

y = sim(sys0,[u e]);

% for regressor matix,

z = [y,u];

figure(1)

idplot(z); grid on

% LSE

sys = arx(z,[0 2 1]) % estimate system

% compare the real output with the output of the LSE system

ye =sim(sys,[u e]);

figure(2)

plot(y,'b');grid on; hold on

plot(ye,'r');

title('FIR sytem : real output(blue) with LSE output(red)')

hold off

%%%comparison

Unknown system x(t) : y(t)= u(t-1) + 0.5u(t-2)

Number of the sampling : 300

Input: Random binary Sequence

Measurement noise: Gaussian ~ N(0,1)

Measured data : sim(sys0,[u,e])

%%LSE results

1. ARX

%y(t) = 1.5y(t-1) - 0.7y(t-2) + u(t-1) + 0.5u(t-2)

A = [1 -1.5 0.7];

B = [0 1 0.5];

sys0 = idpoly(A,B);

u= iddata([ ],idinput(300,'rbs'));

e = iddata([ ],randn(300,1));

% input = sin wave

% t = 0:0.1:29.9;

% input = sin(2\*pi\*t);

% u = iddata([ ],input',0.1);

% e = iddata([ ],randn(300,1),0.1);

% generate output

y = sim(sys0,[u e]);

% for regressor matix,

z = [y,u];

figure(3)

idplot(z); grid on

% LSE

sys = arx(z,[2 2 1])

%%%comparison

Unknown system x(t): y(t) = 1.5y(t-1) -0.7y(t-2) + u(t-1) + 0.5u(t-2)

Number of the sampling: 300

Input: Random binary Sequence

Measurement noise: Gaussian ~ N(0,1)

Measured data: sim(sys0,[u,e])

%%LSE results

1. ARMAX

% define a discrete system

%y(t) = 1.5y(t-1) - 0.7y(t-2) + u(t-1) + 0.5u(t-2) + e(t) + 0.5e(t-1)

A = [1 -1.5 0.7];

B = [0 1 0.5];

C = [1 0.5]

sys0 = idpoly(A,B,C)

u= iddata([ ],idinput(300,'rbs'));

e = iddata([ ],randn(300,1));

% generate output

y = sim(sys0,[u e]);

% for regressor matix,

z = [y,u];

figure(5)

idplot(z); grid on

% LSE

sys = armax(z,[2 2 1 1])

%%%comparison

Unknown system x(t): y(t) = 1.5y(t-1) -0.7y(t-2) + u(t-1) + 0.5u(t-2)+e(t)+0.5e(t-1)

Number of the sampling: 300

Input: Random binary Sequence

Measurement noise: Gaussian ~ N(0,1)

Measured data: sim(sys0,[u,e])

%%LSE results

* Given the unknown systems, using appropriate matlab commands, find the best estimator.
* Condition :

- The input is RBS, the number of sampling = 300,

- The output measurement error ~ N(0,1)

* FIR model
* ARX model
* ARMAX model