Dynamic system identification using sequential search and FLS

Gbr / 18.05.2019

https://uk.mathworks.com//fuzzy/nonlinear-system-identification.html

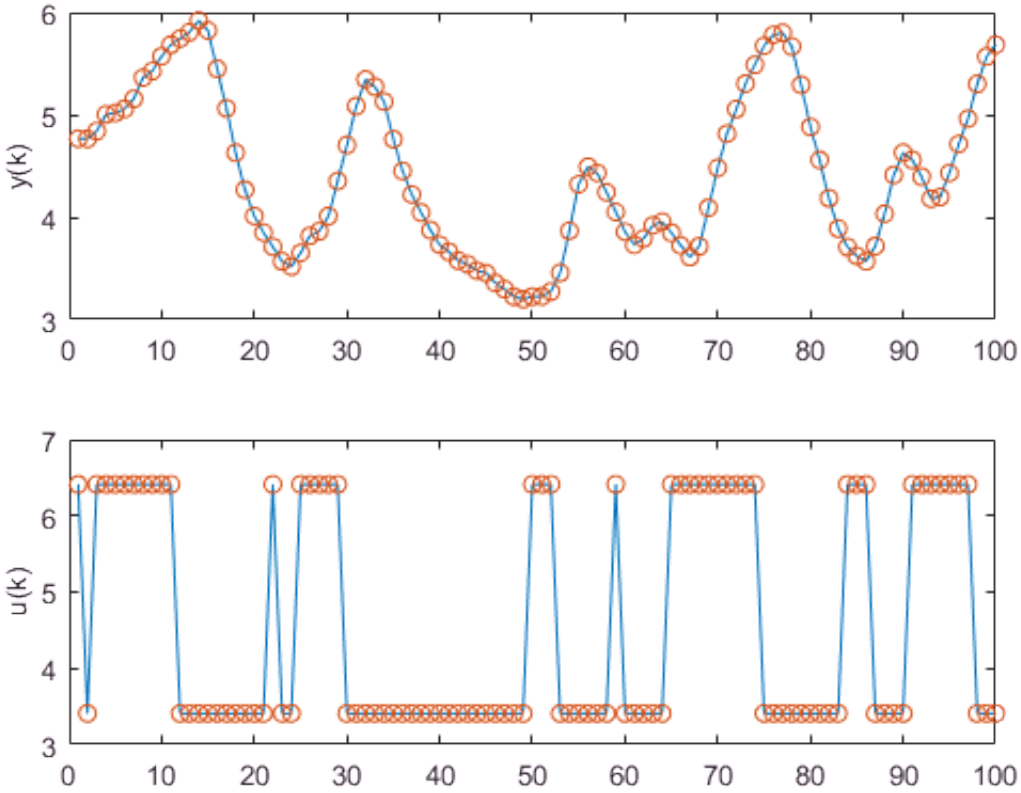
```
%% Load data
load drydemodata  % u2 - inputs, y2 outputs; 1000 items in total
data_n = length(y2);
output = y2;
```

Prepare the data for potential inputs for the fuzzy model:

4 previous outputs; 6 previous inputs

$$\underbrace{y(k-1), y(k-2), y(k-3), y(k-4)}_{\text{iesiri anterioare}}, \underbrace{u(k-1), u(k-2), u(k-3), u(k-4), u(k-5), u(k-6)}_{\text{intrari anterioare}}$$

```
% input
input = [[0; y2(1:data_n-1)] ...
        [0; 0; y2(1:data_n-2)] ...
        [0; 0; 0; y2(1:data_n-3)] ...
        [0; 0; 0; 0; y2(1:data_n-4)] ...
        [0; u2(1:data_n-1)] ...
        [0; 0; u2(1:data_n-2)] ...
        [0; 0; 0; u2(1:data_n-3)] ...
        [0; 0; 0; 0; u2(1:data_n-4)] ...
        [0; 0; 0; 0; 0; u2(1:data_n-5)] ...
        [0; 0; 0; 0; 0; 0; u2(1:data_n-6)]];
data = [input output];
data(1:6, :) = [];  % remove the first 6 rows of data
input_name = char('y(k-1)','y(k-2)','y(k-3)','y(k-4)','u(k-1)','u(k-2)','u(k-3)','u(k-4)','u(k
index = 1:100;
subplot(2,1,1)
plot(index,y2(index),'-',index,y2(index),'o')
ylabel('y(k)','fontsize',10)
subplot(2,1,2)
plot(index,u2(index),'-',index,u2(index),'o')
ylabel('u(k)','fontsize',10)
```



A heuristic approach to input selection is called sequential forward search, in which each input is selected sequentially to optimize the total squared error. This can be done by the function seqsrch.

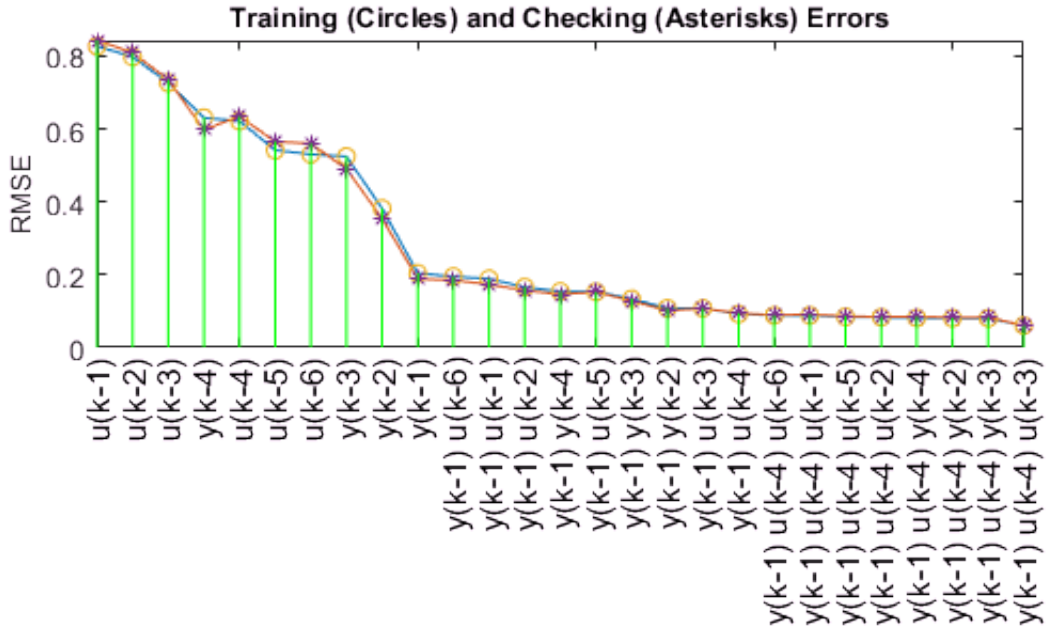
```
trn_data_n = 300;
trn_data = data(1:trn_data_n,:);
chk_data = data(trn_data_n+1:trn_data_n+300,:);
[~,elapsed_time] = seqsrch(3,trn_data,chk_data,input_name); % #ok<*ASGLU>
```

```
Selecting input 1 ...
ANFIS model 1: y(k-1) --> trn=0.2043, chk=0.1888
ANFIS model 2: y(k-2) --> trn=0.3819, chk=0.3541
ANFIS model 3: y(k-3) --> trn=0.5245, chk=0.4903
ANFIS model 4: y(k-4) --> trn=0.6308, chk=0.5977
ANFIS model 5: u(k-1) --> trn=0.8271, chk=0.8434
ANFIS model 6: u(k-2) --> trn=0.7976, chk=0.8087
ANFIS model 7: u(k-3) --> trn=0.7266, chk=0.7349
ANFIS model 8: u(k-4) --> trn=0.6215, chk=0.6346
ANFIS model 9: u(k-5) --> trn=0.5419, chk=0.5650
ANFIS model 10: u(k-6) --> trn=0.5304, chk=0.5601
Currently selected inputs: y(k-1)

Selecting input 2 ...
ANFIS model 11: y(k-1) y(k-2) --> trn=0.1085, chk=0.1024
ANFIS model 12: y(k-1) y(k-3) --> trn=0.1339, chk=0.1283
ANFIS model 13: y(k-1) y(k-4) --> trn=0.1542, chk=0.1461
```

```
ANFIS model 14: y(k-1) u(k-1) --> trn=0.1892, chk=0.1734
ANFIS model 15: y(k-1) u(k-2) --> trn=0.1663, chk=0.1574
ANFIS model 16: y(k-1) u(k-3) --> trn=0.1082, chk=0.1077
ANFIS model 17: y(k-1) u(k-4) --> trn=0.0925, chk=0.0948
ANFIS model 18: y(k-1) u(k-5) --> trn=0.1533, chk=0.1531
ANFIS model 19: y(k-1) u(k-6) --> trn=0.1952, chk=0.1853
Currently selected inputs: y(k-1) u(k-4)

Selecting input 3 ...
ANFIS model 20: y(k-1) u(k-4) y(k-2) --> trn=0.0808, chk=0.0822
ANFIS model 21: y(k-1) u(k-4) y(k-3) --> trn=0.0806, chk=0.0836
ANFIS model 22: y(k-1) u(k-4) y(k-4) --> trn=0.0817, chk=0.0855
ANFIS model 23: y(k-1) u(k-4) u(k-1) --> trn=0.0886, chk=0.0912
ANFIS model 24: y(k-1) u(k-4) u(k-2) --> trn=0.0835, chk=0.0843
ANFIS model 25: y(k-1) u(k-4) u(k-3) --> trn=0.0609, chk=0.0604
ANFIS model 26: y(k-1) u(k-4) u(k-5) --> trn=0.0848, chk=0.0867
ANFIS model 27: y(k-1) u(k-4) u(k-6) --> trn=0.0890, chk=0.0894
Currently selected inputs: y(k-1) u(k-3) u(k-4)
```




```
fprintf('\nElapsed time = %f\n',elapsed_time);
```

```
Elapsed time = 0.505000
```

```
winH1 = gcf;
```