Tutorial 10 – MathScript and System Identification  
(JEE344 Applied Control Engineering)

Learning Outcomes

- To use LabVIEW MathScript
- To apply least squares algorithm in estimation of system parameters
- To apply recursive least squares algorithm in estimation of system parameters

1. LabVIEW MathScript – Getting Started

A MathScript window that is opened via the Tools / MathScript window menu in LabVIEW is shown in the following figure:

![LabVIEW MathScript Window](image)

Figure 1 LabVIEW MathScript Window

The MathScript window consists of a number of (sub) windows:

- The Command Window where you type the commands to be executed (after you press the Enter key)
- The Output Window where executed commands and numerical results are shown.
- The Workspace window, which contains the Variables, Script, and History (sub)windows containing the following tabs:
Variables: Lists generated variables. The numerical value of these variables can be displayed. The value can also be plotted graphically by selecting Graphics First in the Variables dialog window.

Script: Opens a script editor. To open another Script editor: Select the menu (in the MathScript window) File / New Script Editor.

History: Shows a list of previous commands that you have executed.

The Workspace window contains a number of buttons:

- Open (script)
- Run runs the present code in the editor, even if the script has not been saved. Running includes compiling the code (i.e. generating executable code), before it is run (executed). Note that the script is not saved with the Run button.
- Save (script)
- New (script)

The Workspace window can be closed via the View menu in the MathScript window.

- In the Command window: Execute the following commands (by typing the text lines followed by the Enter button on the keyboard when at the end of each line):

  10+20 (Enter). Observe the result in the Output window.
  a=1;
  b=2;
  c=a+b

- Open the Variables window just to see the variables generated. Note that the scalars, as a, b, c etc., are displayed as 2 dimensional arrays (corresponding to a matrix in MATLAB).
- Open the History window to see the history of commands.

2. Least Squares Algorithm

2.1 Hands-on Exercise 1: LS algorithm and use of polyfit (see Example 4 in Chapter 4 Lecture Notes). The following data table was obtained from an experiment with a speed tachometer.

<table>
<thead>
<tr>
<th>k</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>input (rpm)</td>
<td>0</td>
<td>500</td>
<td>1000</td>
<td>1500</td>
<td>2000</td>
<td>2500</td>
<td>3000</td>
</tr>
<tr>
<td>output (V)</td>
<td>0</td>
<td>9.1</td>
<td>15.0</td>
<td>23.3</td>
<td>29.9</td>
<td>39.0</td>
<td>47.5</td>
</tr>
</tbody>
</table>

Estimate the parameters of the following model using Least Squares Algorithm:

\[ y = au + b \]

where \( y \) is output (V) and \( u \) is input (rpm).
2.2 Hands-on Exercise 2 Estimate parameters of the following model:

\[ y(k) = -a_1 y(k-1) - a_2 (k-2) + b_1 u(k-1) + b_2 u(k-2) \]

with data given in the following table (see Example 5, Chapter 4 Lecture Notes):

<table>
<thead>
<tr>
<th>k</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
</tr>
</thead>
<tbody>
<tr>
<td>y</td>
<td>0</td>
<td>-0.0889</td>
<td>0.0137</td>
<td>0.1564</td>
<td>0.4618</td>
<td>0.1771</td>
<td>0.3446</td>
<td>0.2171</td>
<td>-0.1558</td>
<td>0.0485...</td>
<td>-0.1879</td>
<td>-0.1123</td>
<td>0.0463</td>
<td>0.2003</td>
<td>0.5007</td>
<td>0.3846</td>
<td>-0.0172</td>
<td>0.1513...</td>
<td>-0.1162</td>
<td>0.1134</td>
</tr>
<tr>
<td>u</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>-1.0</td>
<td>1.0</td>
<td>-1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>-1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>-1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>-1.0</td>
<td>1.0</td>
<td>-1.0</td>
<td>1.0</td>
</tr>
</tbody>
</table>

2.3 Hands-on Exercise 3: Estimate manoeuvrability indices (see Example 6). Using the data for a 20-20 Z-test at:

http://academic.amc.edu.au/~hnguyen/JEE344ACE/DataFiles.zip

estimate T and K of the following model:

\[ T \dot{r} + r = K \delta \]

Results: \( T = 11.8964 \) (seconds) and \( K = 0.1145 \)
3. Recursive Least Squares Algorithm

3.1 Hands-on Exercise 4: Estimate parameters of a discrete-time model using the RLS (see Example 7)

3.2 Hands-on Exercise 5: Estimate manoeuvrability indices (also see Example 8) using RLS and the data for a 20-20 Z-test at:
   http://academic.amc.edu.au/~hnguyen/JEE344ACE/DataFiles.zip

4. Estimation of Bluefin’s Manoeuvrability Indices

4.1 Hands-on Exercise 6 (Data of Z-tests for Bluefin is provided)
Zig-zag test conducted for Bluefin was 10-10 Z-test. Only yaw rate and yaw angle were obtained with Crossbow. Rudder angle is not available. It is necessary to estimate rudder angle.

![Figure 3 10-10 Ztest for Bluefin](image)

5. Identification of Tank Level System

5.1 Hands-on Exercise 7: Estimate T and K for the tank in Control Lab using the simulation VI that was made in Tutorials 5 & 6 with LSA and RLSA. Modify the VI such that you can save data. Run the VI in 1) Man mode & 2) Auto mode (with P only control).

Conclusions
At this point the following LOs have been satisfied:
- To use LabVIEW MathScript
- To apply least squares algorithm in estimation of system parameters
- To apply recursive least squares algorithm in estimation of system parameters

References
http://techteach.no/labview/lv85/mathscript/index.htm