User Guide for LF SS Regression for www.r3eda.com
15 June 2016 – R. Russell Rhinehart

User Guide – "Generic LF Static Model Regression.xlsm"

This VBA macro uses leapfrogging (LF) to best fit a model to data. LF seeks model coefficient values to minimize the root-mean-square deviation (square root of the sum of squared deviations divided by number of data) between data and model. Convergence is based on the rms-value of a random selected subset coming to steady-state with iteration progress, and the optimizer starts N times (from randomized DV-values) and reports the best of N.

The data is from an application that arose from a model-based control desire to calculate the valve stem position required to produce a desired air flow rate. The model is derived from first principles pressure losses in the pipe and valve, with the valve characteristic (modified equal percentage) modeled as a power-law relation.

$$u = a\left(\frac{F}{\sqrt{2783.2 - kF^2}}\right)^b$$

Where u is the controller output (nominally the valve position) and F is the flow rate. The three model coefficients are a, b, and k. The constants in the model, 2783.2 and 2 represent expectations. The value for coefficient b is expected to be between 1/3 and 1, and the values of a and k must be positive. Seems simple, but it is nonlinear, contains multiple local optima, and a large value of k leads to an execution error!

Data are from a 1" pipe system, pilot-scale, fully computer instrumented and controlled. Experimentally, we chose the u value (controller output), waited to steady-state, and measured the flow rate. In use however, the equation is the inverse of the normal process model. It needs to provide the correct u-value to obtain a desired flow rate.

Where to find things

- The worksheet tab "Main" is for user inputs and to observe results.
- The worksheet tab "DataIN" is for user to input experimental data.
- The worksheet tab "DataRESULT" summarizes the N optimization trials, sorted by OF value.
- Subroutine "Response_Model" in the VBA module contains the model, and also recognizes if the argument of the square root function could be negative.
- Subroutine "ConstraintTest" in the VBA module tests to see if DV values violate constraints.
- Subroutine "Assign" in the VBA module assigns LF player DV values to the model coefficients.

- Declarations in the VBA Code are where you define variable names.

If you wish to use this program for your individual application you'll need to work with those 6 items.

Disclaimer

The user accepts all responsibility for the use of this program or the use of results from this program.

User Guide

Main Worksheet

You can enter your own model coefficient values in Column 8 and then click on the "Display model" button. The data are the red dots on the lower left graph and the modeled values are the black astericks connected by the black connect-the-dots curve. The middle graph is a parity plot of modeled w.r.t. experimental y-data, and permits a view of the distribution of residuals. If you enter variable values that violate a constraint "FAIL" will appear in Cells(8, 6). The rms value will appear in Cells(12, 6).

To run the optimizer, first enter a hypothesized range of coefficient values in the green cells of Columns 9 and 10. (The optimizer is not constrained by that range and will seek an external optimum.) Then choose an option in the green Column 15 cells, then press the "Start Regression Optimizer" button. You can also change optimizer criteria in Column 4 cells.

Optimizer results are placed in the cells associated with the yellow background. Column 6 contains optimizer progress identifiers. Once converged, DV*, OF* and associated data (trial number, number of function evaluations, number of iterations) are listed in Columns 1 to 14.

If you place a desired confidence ($0<c<1$) basis in Cells(3, 15) and a desired best fraction ($0<f<1$) in Cells(4, 15), Cells(5, 15) will reveal the number of trials needed to be c confident that at least one of the N trials will have landed in an rms value that represents one of the best f possible.

I like to choose a "Y" in Cells(8, 15) to observe the optimizer progress. The upper right graph reveals the random subset rms value w.r.t. iteration number (blue dots) and the red filtered value. When the red filtered value relaxes to a noisy steady state convergence is declared. A threshold for the R-statistic in steady-state identification of 0.85 is a good value. This is a scale independent, application independent left-hand critical value of a ratio of variances. The green line in the upper right graph is the number of the lead LF player (associated with the secondary axis). As long as it keeps jumping between players, all is well. Noteworthy, there is an outlier I the data. As a result, when it is sampled in the random set, it makes a blue dot that is definitely not consistent with the bulk of the blue dots. Such a pattern provides insight to the data.

After N trials, the graph on the lower right indicates the CDF (cumulative distribution function) of OF* values. This can provide insight on the solution. Steps in the CDF of OF* graph reveal local optima that trapped the optimizer, and horizontal axis values indicate the OF range of local traps. Desirably, the first rise will be steep and long, indicating that many trials found the same OF* value. However, any rise is not perfectly vertical. A numerical optimizer stops in the close proximity of the optimum, not exactly on it. So, if there is only one optimum, found by all trials, then they will not all have the exactly the same OF value, and the CDF of OF curve will not be a vertical line. But in this case, it will have a very small range on the horizontal axis.

DataIN Worksheet

You can enter your data here. You can use the "DeleteOldData" button to delete old data or you can use any method. Place your modeled output data in Column 5, and associated input(s) in Columns 6 or after. Do not skip data. Sort the data by input value (Column 6) for viewing convenience in the graphs.

The VBA program places the modeled output in Column 4. Do not place your data there.

Data point (70.75, 52) is the rogue data point. You might want to delete it, then move the data below up a row, and rerun the optimizer.

Sub Response_model() in the VBA Module

You can enter VBA code for your model here.

Sub ConstraintTest() in the VBA Module

You can enter VBA code to look at constraints here.

Sub Assign() in the VBA Module

There are three decision variables DVs in this application, one for each model coefficient. The LP players have a player number and a dimension index. The array name is "PlayerPosition". To determine the OF-value of the player, the rms-value of model to data, you need to assign player position values, DV values, to the model coefficients.

Declarations in the VBA Module

I choose to use the "Option Explicit" declaration in the header. This means that all variables need to be explicitly defined. Accordingly, if you decide to use a model with different variable or coefficient names you need to declare them here.