**bGen LiveApp User Manual**

Curated by: Antonio Benedetti, Giulia Marchese

10th September 2023

[Antonio.x.benedetti@gsk.com](mailto:Antonio.x.benedetti@gsk.com)

[Giulia.x.benedetti@gsk.com](mailto:Giulia.x.benedetti@gsk.com)

**Index**

[Overview 2](#_Toc144994449)

[**1.** **Loading low-N Batches** 2](#_Toc144994450)

[1.1 Confirms Column Batch ID 2](#_Toc144994451)

[1.2 Select Batch Time 3](#_Toc144994452)

[1.3 Specify Resampling Interval 3](#_Toc144994453)

[**2.** **Settings of Parameters** 3](#_Toc144994454)

[2.1 Change advanced settings for J 3](#_Toc144994455)

[2.2 J automatic estimation 3](#_Toc144994456)

[2.3 Do you want to modify J? 4](#_Toc144994457)

[2.4 Stratified Resampling 4](#_Toc144994458)

[2.5 Estimate of the memory parameter (L) 4](#_Toc144994459)

[2.6 Change advanced settings for std\_th 4](#_Toc144994460)

[2.7 Change advanced settings for delta\_L 4](#_Toc144994461)

[2.8 L Estimation 5](#_Toc144994462)

[**3.** **Train the generator** 5](#_Toc144994463)

[**4.** **Generation of in-silico batches** 5](#_Toc144994464)

[4.1 Advanced settings for the generation of in-silico batches 6](#_Toc144994465)

[4.2 Select number of in-silico batches to be generated 6](#_Toc144994466)

[4.3 Select variable bounds 6](#_Toc144994467)

[4.4 Start generation 7](#_Toc144994468)

[**5.** **Plots** 7](#_Toc144994469)

[In this section end users will be able to interactively visualise the results. 7](#_Toc144994470)

[5.1 Variable Plots 7](#_Toc144994471)

[5.2 Advance analysis for F & Chi-squared 7](#_Toc144994472)

[**6.** **Save Excel** 8](#_Toc144994473)

[**References** 8](#_Toc144994474)

[**Appendix** 8](#_Toc144994475)

## Overview

bGen LiveApp is a MATLAB Live Script in which high-frequency variable trajectories from real batches (input) are used to generate in-silico low-frequency trajectories (output). The output from bGen LiveApp can be useful when building a process monitoring model is challenging due to the limited availability of historical batches data (“low-N” scenario).

bGen LiveApp has been developed using MATLAB®, version 9.14.0.2206163 (R2023a). MATLAB toolboxes required to run bGen LiveApp are reported in Table 1.

**Table 1.** *MATLAB toolboxes required to run bGen LiveApp*

|  |  |
| --- | --- |
| **Name** | **Version** |
| Statistics and Machine Learning Toolbox | 12.5 |

bGen LiveApp script is organised in sections. Control elements have been added to facilitate the interaction with the code and its application to different data templates and scenarios. For a better experience it is recommended to run each section individually following the sequential organization of the script. This will enable end users to interact more easily with the control elements embedded in the script.

The following sections of this user manual correspond numerically to the sections contained in bGen LiveApp script. Each of them contains information about the purpose of the Live Script section and instructions in case end users need to provide inputs or interact with the control elements or the code.

1. **Loading low-N Batches**

While running this section a load file window will open. End users need to select the Excel file containing the real batches data to import them. Accepted formats are .xlsx or any another format accepted by the MATLAB command readtable, such as .xls.

The requirements for the Excel file are the following:

* it must contain one column for the batch IDs, one column for the time instants, and at least one process variable column
* each process variable column is expected to contain data related only to that specific variable
* columns can be ordered in any way
* it must not contain empty column/s between other variables columns
* batch data must be variable-wise unfolded (Nomikos & MacGregor, 1994)
* batch IDs can have either numerical or textual format
* time and process variables strictly require a numerical format

### Confirms Column Batch ID

After loading the data end users need to select from the drop-down menu the column header corresponding to the batch ID column.

This section of the script also identifies the shortest batch duration. The current version of bGen LiveApp does not perform any type of alignment on batch data. If the input batches have different lengths, bGen LiveApp will truncate the data based on the shortest one.

### Select Batch Time

After loading the data end users need to select from the drop-down menu the column header corresponding to the time sample column and specify the unit of measurements from the subsequent dropdown menu.

This section of the script also initialises the value of J (number of resampled trajectories) that will be subsequently tuned (see section 2)

### Specify Resampling Interval

End users need to specify the resampling interval k in the edit field. k must be greater and multiple of the original sampling interval and such that the low-frequency trajectories contain at least two time-points.

## **Settings of Parameters**

In this section advanced settings for J (number of resampled trajectories) and L (memory parameter for each process variable) can be modified before proceeding to their tuning and estimation.

### Change advanced settings for J

If end users want to change the default advanced settings for the tuning of J, the checkbox control of this section needs to be ticked and the end users need to input the new values for max\_J and p\_J in the corresponding edit field control. End users can also restore the default values by right clicking the edit field control and selecting “Restore Default Value”.

max\_J is the maximum possible value for J/N, where N is the number of available real batches. It must be integer and greater or equal than 2. Note that this upper bound does not apply when J is manually set by the user (see section 2.3). The default value is 2000.

p\_J is a lower bound for the probability that all the high-frequency time points in a low-frequency sampling interval are sampled during the stratified resampling operation, and this must happen for each low-frequency sampling interval. This parameter must be greater than 0 and smaller than 1. The closer it is to 1, the bigger J is. The default value is 0.999.

### J automatic estimation

If end users want to automatically estimate J (preferred and default option), the checkbox control of this section needs to be ticked.

If end users want to input a specific value for J, the checkbox control of this section needs to be unticked and end user can proceed to the next section.

### Do you want to modify J?

If end users want to change the value of J obtained from the automatic estimation process or input a more suitable value of J based on experience, the checkbox control of this section needs to be ticked and the users need to specify the new value in the edit field control.

### Stratified Resampling

After defining J, this script section will proceed with the stratified resampling process.

### Estimate of the memory parameter (L)

In this section the advanced settings delta\_L and std\_th for the estimation of L are initialised to their default values.

delta\_L enables to tune the memory. This value must be between 0 and 1. For a specific variable, it is suggested to reduce it to improve the time correlation of the in-silico trajectories. The reduction might bring some benefits in terms of coverage too. It is suggested to increase it if the in-silico trajectories are too noisy. The default value is 0.99 for all the variables.

std\_th is a threshold for scaling. If the standard deviation of a column is smaller than this value, then the column is considered as a constant vector. The default value is 1e-6.

### Change advanced settings for std\_th

If end users want to change the default advanced setting std\_th, the checkbox control of this section needs to be ticked and the end user needs to input the new value in the edit field control. End users can also restore the default value by right clicking the edit field control and selecting “Restore Default Value”.

### Change advanced settings for delta\_L

There are two options to modify the advanced setting delta\_L:

1. if end users want to change the default advanced setting delta\_L for ALL the variables, the checkbox control change\_delta\_L\_ALL needs to be ticked and the end user needs to input the new value in the corresponding edit field control. This new setting will be applied to ALL the variables.
2. to change the default advanced setting delta\_L for individual variables and assign different values to each of them, the end users need to select the variable of interest from the drop-down menu, tick the checkbox control change\_delta\_L and input the new delta\_L value in the corresponding edit field control for the chosen variable.

Before proceeding to the estimation of L, end users can visualise the advanced settings properties of delta\_L for each variable in the table embedded at the end of this section of the script.

### L Estimation

L is automatically estimated in this section. No input is required from end users.

## **Train the generator**

Run this section to train the in-silico generator. No input is required from end users.

## **Generation of in-silico batches**

Advanced settings for the in-silico batch generation process are initialised at the beginning of this section.

R\_per\_iter is the number of generated in-silico trajectories for each iteration. End users can consider increasing it to save time if the plan is to get a large number of in-silico trajectories. It must be integer and positive. Default value is 10.

num\_z is the number of samples from the real interval used to compute the reconstruction errors between the theoretical F and Chi-squared distributions and the fitted ones (it must be greater or equal than 2). Increasing it may improve the precision in the fitting error evaluation but will also increase the computational cost. Default value is 1e4.

num\_rect\_max is the maximum number of bins available to fit F and Chi-squared distributions. Consider increasing it only if the relative warning appears. It must be integer and positive. Default value is 100.

std\_th is a threshold for scaling. If the standard deviation of a column is smaller than this value, then the column is considered as a constant vector. It may lead to numerical problems if too small or large. Default value is 1e-6.

exp\_var\_th is the minimum percentage variance that needs to be explained by the calibrated PCA model (used during fitting). Default value is 50.

conf\_nipals allows to tune the required precision for the convergence of NIPALS algorithm during PCA building. Reducing it may improve the precision of the PCA model and but will increase the computational cost. Default value is 1e-3.

iter\_nipals is the maximum number of NIPALS algorithm iterations. End users can consider increasing it only if fitting results do not seem satisfactory. Default value is 1e4.

delta\_F\_th is used only if the number of in-silico batches to be generated is determined though an automatic procedure (see 4.2 for details). It is the error threshold for the F distribution fitting with the T^2 values obtained from a training matrix built both with real and in-silico batches. Its reduction is likely to lead to an increase of the number of generated in-silico trajectories. It must be greater than zero. Default value is 0.002.

delta\_chi2\_th is used only if the number of in-silico batches to be generated is determined though an automatic procedure (see 4.2 for details). It is the error threshold for the Chi-squared distribution fitting with the Q values obtained from a training matrix built both with real and in-silico batches. Its reduction is likely to lead to an increase of the number of generated in-silico trajectories. It must be greater than zero. Default value is 0.002.

To modify the values of the abovementioned parameters, proceed to the next section.

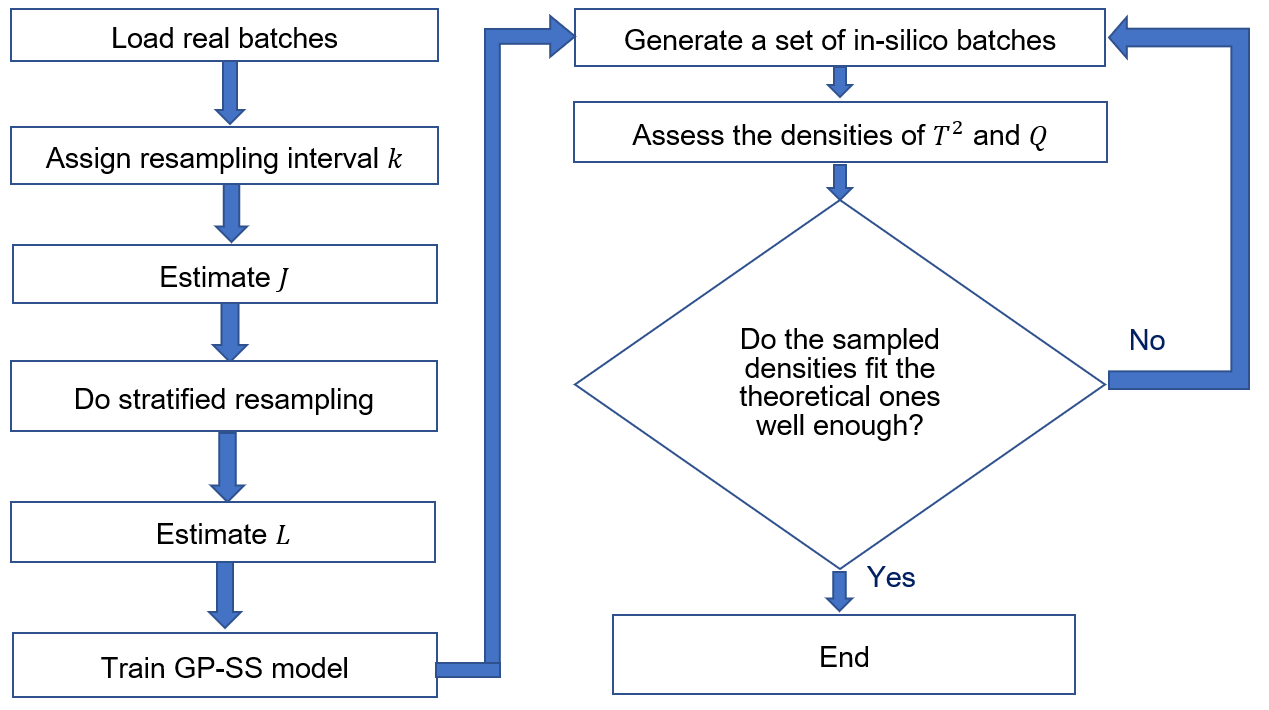
### Advanced settings for the generation of in-silico batches

If changes to the default parameters settings are required, end users can enter the desired values in the corresponding edit field control. End users can also restore the default values by right clicking the edit field control and selecting “Restore Default Value”.

### Select number of in-silico batches to be generated

End users need to select one of the following options from the drop-down menu to proceed with the in-silico batch generation process:

* “assigned” option (default) will generate as many in-silico batches as specified by the user in the edit field control app.R\_fixed. After selecting this option end users can modify the value in the edit field control if desired. The default value is 20 in-silico batches.
* “automatic” option will generate as many in-silico batches as necessary to obtain reconstructed F and Chi-squared distributions, from the PCA Hotelling T^2 and Residual Q respectively, that follow the theoretical ones to a reasonably close extent. For better contextualization of the algorithm behind this automatic batch number evaluation please refer to Figure 1 and to *Gasparini et al. 2023.*



**Figure 1.** *Simplified flow chart illustrating the stopping criterion for automatic in-silico batch number estimation*

### Select variable bounds

In this section end users can set upper and lower bounds for the in-silico generated variables by following these steps:

1. Users need to select the variable of interest from the drop-down menu.
2. To set the upper bound tick the checkbox and proceed to add the value of interest in the corresponding edit field. Untick the checkbox to remove any previously set bound.
3. To set the lower bound tick the checkbox and proceed to add the value of interest in the corresponding edit field. Untick the checkbox to remove any previously set bound.

End users need to repeat the abovementioned steps for any variable of interest. At the end of this script section end users can visualise a table with the bounding properties for each variable:

* If LB\_set is 0 the variable will not have a lower bound. If it is 1 the variable will have a lower bound.
* LB\_value is the variable lower bound value, and it will be applied only if LB\_set is 1.
* If UB\_set is 0 the variable will not have an upper bound. If it is 1 the variable will have an upper bound.
* UB\_value is the variable upper bound value, and it will be applied only if LB\_set is 1.

By default, no bounds are applied to variables.

### Start generation

Run this section to generate the in-silico batches. No input from end users is required.

## **Plots**

## In this section end users will be able to interactively visualise the results.

### Variable Plots

In this section end users will be able to interactively visualise the variables trends in time.

From the drop-down menu end users can select one of the following visualization types:

Real, to visualise the variable time trends for the real batches only

Real and in-silico, to visualise the variable time trends for real and in-silico batches

In-silico, to visualise the variable time trends for in-silico batches only

Coverage, to visualise the variable coverage in time. This indicator assesses in a univariate way the coverage of the Normal Operating Conditions (NOCs), for a specific variable and time instant. It ranges between 0 (bad coverage) and 1 (perfect coverage).

After selecting the visualization type, end users can proceed with the selection of the variable of interest from the following drop-down menu.

Any time one of these two drop-down menu items is changed, the plot embedded in this script section will be updated.

### Advance analysis for F & Chi-squared

In this section end users will be able to interactively visualise the theoretical and approximated F and Chi-squared distribution plots by selecting the option of interest in the drop-down menu.

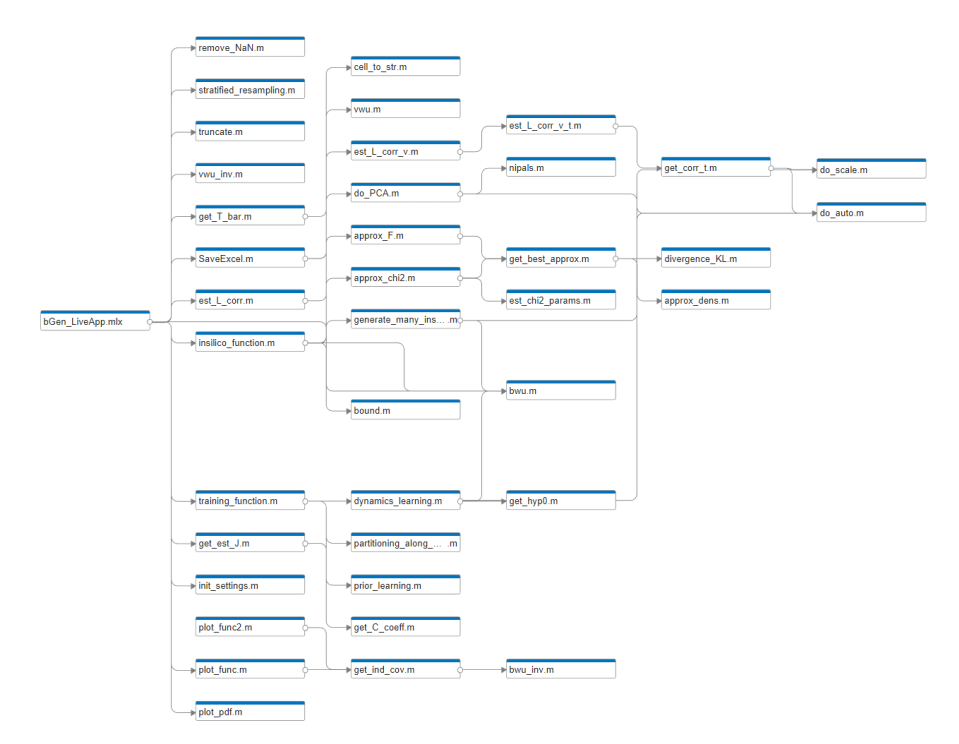
## **Save Excel**

To save the results as Excel file tick the checkbox in this section and select the in-silico batch ID format (textual or numerical options are available).

## **References**

Nomikos, P., & MacGregor, J. F. (1994). Monitoring batch processes using multiway principal component analysis. *AIChE Journal, 40*(8), 1361-1375.

## **Appendix**



**Figure 2.** *bGen LiveApp functions dependency plot*