Supplementary Information for

How to Simulate Outliers with the Desired Properties

Alba González-Cebrián, Francisco Arteaga, Abel Folch-Fortuny and Alberto Ferrer

Alba González-Cebrián.
E-mail: algonceb@upv.es
Francisco Arteaga
E-mail: francisco.arteaga@ucv.es
Abel Folch-Fortuny
E-mail: Abel.Folch-Fortuny@dsm.com
Alberto Ferrer
E-mail: aferrer@eio.upv.es

This PDF file includes:

This document provides the Matlab code to reproduce the examples from the Results section in the article. All code lines are also in the file exeMatlab.m, available in the GitHub repository https://github.com/albagc/SCOUTer.git. We also encourage the user to check the documentation for further information about the functions used in this work.

Contents

1. Intallation from GitHub

The first step to do is to download the functions from https://github.com/albagc/SCOUTer.git. After unzipping it, the folder must be set as the Matlab working directory.

2. Principal Component Analysis Model

A. PCA-MB and Exploration. The file `exeMatlab.m` is the script that reproduces the examples from the Results section, but also includes examples about how to fit and explore the PCA model. The following results have been obtained using Matlab version R2020a 9.8.0.1323502. The first lines load the data (in the file `exeMatlab.m`), build a PCA model and obtain the score and distance plots (Figure S1).

```
load exampleX
pcamodel_ref = pcamb_classic(X, 2, 0.05, 'cent');
pcaout = pcame(X, pcamodel_ref);
figure('Name', 'PCA Model for reference data set')
dscplot(X, pcamodel_ref, 'click', 'off');
```

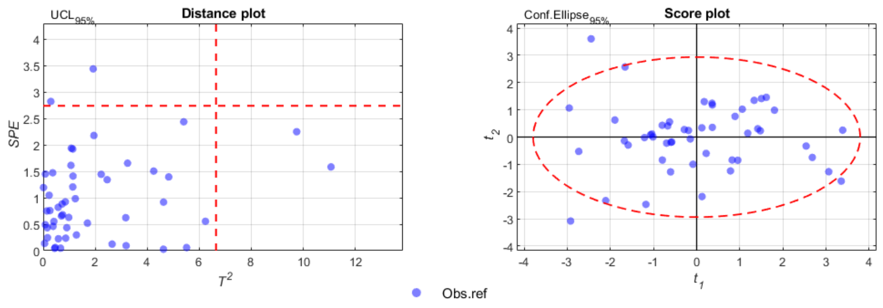


Fig. S1. Distance plot (left) and score plot (right) of the reference data set.

It is important to notice that the input argument `pcamodel_ref`, in functions `pcame` and `dscplot`, is a struct with fields that contain relevant information about the PCA model. Thus, **users can use already built PCA models by storing the parameters as fields of a struct**. For further information about the format of the struct and the required fields, please check the documentation file.

B. Interacting with plots. Regardining the third input argument of the `dscplot` function, it switches on (or off) the plot interactivity option. This is set as an input argument of the `distplot` and `scoreplot` functions as well.

B.1. Click on observations. By setting the interaction parapeter to `'on'`, the output figure enables the user to click on observations from the distance and score plots.

```
figure('Name', 'PCA Model plot'),
dscplot(X, pcamodel_ref, 'click', 'on');
```

Initially, a text box appears indicating the user to select an observation from the plots above. By selecting one observation from the left graph, the user can display further information about its SPE, Hotelling-T^2 and its contributions (Figure S2).

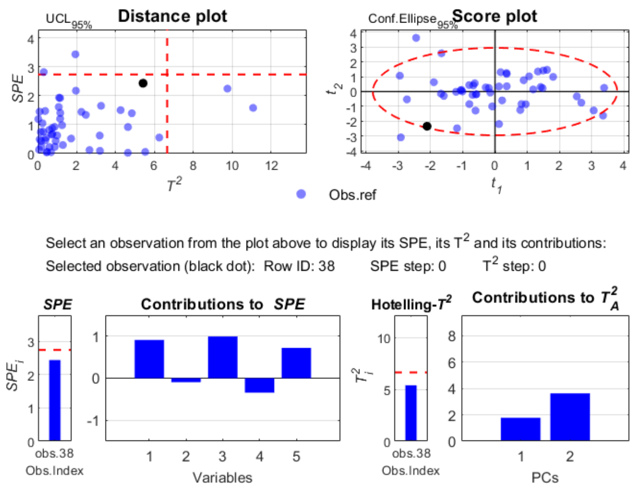
B.2. Brush observations. Alternatively, one can use the brush in order to select observations from the plot. This possibility is only available with the interactivity switched **off**, and it requires to run one more command to export the index of the selected data to the variable workspace.

```
figure('Name', 'PCA Model plot'),
dscplot(X, pcamodel_ref, 'click', 'off')
```

After running these lines, the Figure S1 is displayed and the user can brush the data. The selected points will appear as black dots in both subplots (Figure S3). Once the observations have been selected and the Figure is still open, running the following line, a variable is created storing the index of the brushed observations:

```
idloc = get(gcf, 'UserData');
```

Afterwards, this vector can be used to select the observations that will be used as an input for the `scout` function.



Select an observation from the plot above to display its SPE, its T^2 and its contributions:
Selected observation (black dot): Row ID: 38 SPE step: 0 T^2 step: 0

Fig. S2. Interactive figure with the distance plot (left) and the score plot (right) of the reference data set.

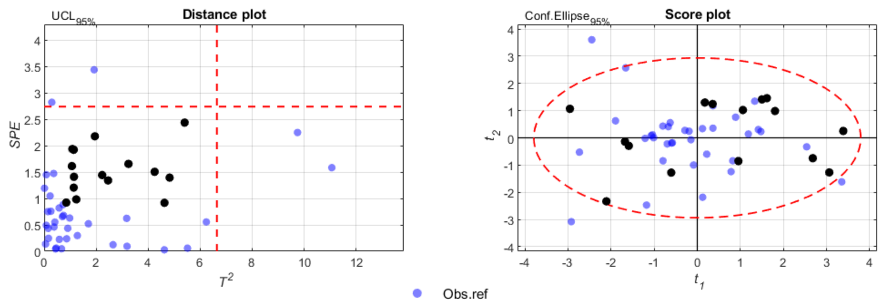
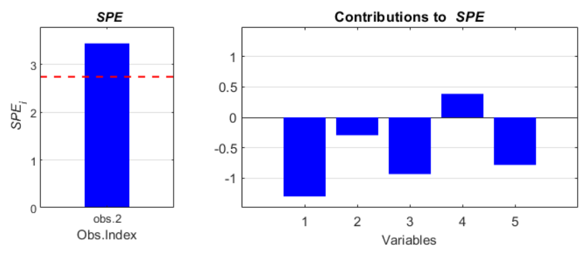


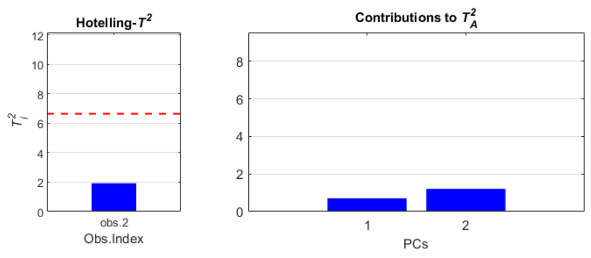
Fig. S3. Brushing observations from distance plot (left) and the score plot (right) of the reference data set.

C. Contribution plots. If the user wants a plot only with the information about the SPE or the T_A^2, the functions **speinfo** and **ht2info** can be used. In Figure S4 the SPE (Figure S4a) and the T_A^2 (Figure S4b) information about the observation with maximal SPE is provided.

```
[~, imaxspe] = max(pcaout.SPE);
figure('Name', 'SPE information'),
speinfo(pcaout.SPE, pcaout.E, pcamodel_ref.limspe, imaxspe);
figure('Name', 'H-T2 information'),
ht2info(pcaout.T2, pcaout.T2cont, pcamodel_ref.limt2, imaxspe);
```



(a) Information about the SPE.



(b) Information about the T_A^2.

Fig. S4. Additional information about the observation with maximum SPE of the reference data set.

All the information about the mathematical expressions and meanings of contributions for the SPE and the T^2 can be found in (1).

3. Simulate Controlled Outliers

To generate outliers, the function `scout` is used. In this section, the settings to reproduce the Results section from the paper are illustrated.

A. One-step simulation case. In Figure S5, the example from *Case I* is recreated, generating a set with one step and $T_y^2 = 40$ for all the observations.

```
T2target = 40*ones(size(X, 1), 1);
outonestep = scout(X, pcamodel_ref, 'simple', 't2y', T2target);
Xall = [X; outonestep.X];
obstag = [zeros(size(X, 1), 1); outonestep.tag];
figure('Name', 'PCA Model plot inter off'),
dscplot(Xall, pcamodel_ref, 'click', 'off', 'obstag', obstag, ...
     'steps_spe', outonestep.step_spe, 'steps_t2', outonestep.step_t2);
```

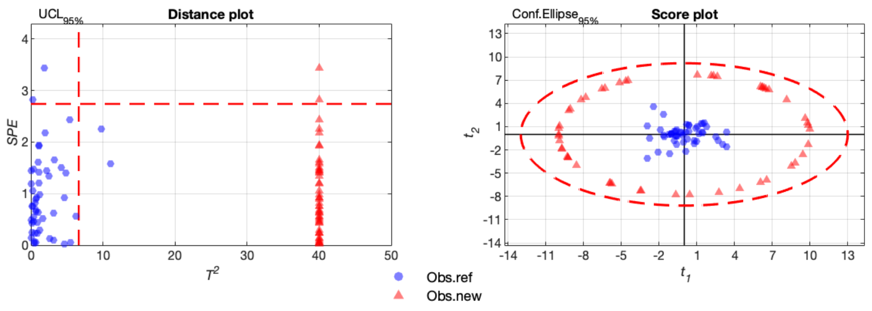
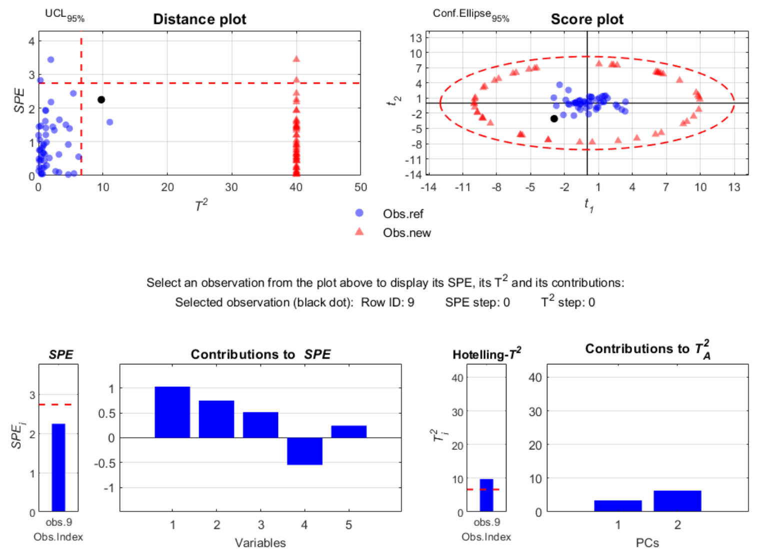


Fig. S5. Figure with the distance plot (left) and the score plot (right) of the reference data set.

By setting the `'click'` input argument to `'on'`, the distance and score plot are rendered enabling the interaction, as shown in Figure S6.

```
figure('Name', 'PCA Model plot inter on'),
dscplot(Xall, pcamodel_ref, 'click', 'on', 'obstag', obstag, ...
     'steps_spe', outonestep.step_spe, 'steps_t2', outonestep.step_t2);
```



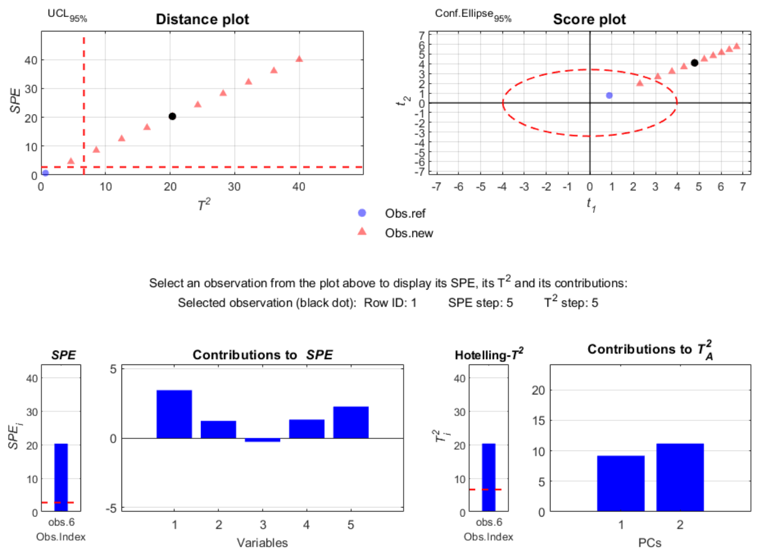
Select an observation from the plot above to display its SPE, its T^2 and its contributions:
Selected observation (black dot): Row ID: 9 SPE step: 0 T^2 step: 0

Fig. S6. Interactive figure with the distance plot (left) and the score plot (right) of the reference data set.

Note that in the interactive plots, the row number of the selected observation according to the reference data set, as well as the steps at which it is obtained, are displayed. In order to do this, this information must be provided in the `'obstag'`, `'steps_spe'` and `'steps_t2'` input arguments of the `dscplot` function. These arguments hold for the row number of the shifted observations in the reference matrix, for the steps performed for the SPE and for the T_A^2, respectively. This information is easily retrieved, since most arguments are fields of the output struct (`outonestep` in the lines above) obtained by the function `scout`.

B. Step-wise simulation case. In the following example, the result from *Case II* is obtained in Matlab. Since the selected observation is chosen randomly, a seed is set in order to ensure the same results when the code is executed. In this case, the third input argument of `scout` is set to `'steps'`. In Figure S7 it can be seen the sequence of ten linearly-spaced steps that are performed from the reference observation $\mathbf{x}$ until reaching the SPE_y and T_y^2 target values.

```
rng(0207) % Ensures the same results
indsel = randperm(size(X, 1), 1);
x = X(indsel, :);
outsteps = scout(x, pcamodel_ref, 'steps', 't2y', 40, 'spey', 40, 'nsteps', 10);
Xall = [x; outsteps.X];
obstag = [0; outsteps.tag];
figure('Name', 'PCA Model plot'),
dscplot(Xall, pcamodel_ref, 'click', 'on', 'obstag', obstag, ...
     'steps_spe', outsteps.step_spe, 'steps_t2', outsteps.step_t2);
```



Select an observation from the plot above to display its SPE, its T^2 and its contributions:
Selected observation (black dot): Row ID: 1 SPE step: 5 T^2 step: 5

Fig. S7. Interactive plots after generating an outlier with 10 linearly-spaced steps between the reference and target values.

To set a non-linear spacing between steps (Figure S8), the input parameters `'gspe'` and `'gt2'` of the `scout.m` function should have values different from one.

```
outsteps_2 = scout(x, pcamodel_ref, 'steps', 't2y', 40, 'spey', 40, 'nsteps', 10,...
    'gspe',1.5,'gt2',0.5);
Xall = [x; outsteps_2.X];
obstag = [0; outsteps_2.tag];
figure, dscplot(Xall, pcamodel_ref, 'click', 'off', 'obstag', obstag, ...
     'steps_spe', outsteps_2.step_spe, 'steps_t2', outsteps_2.step_t2);
```

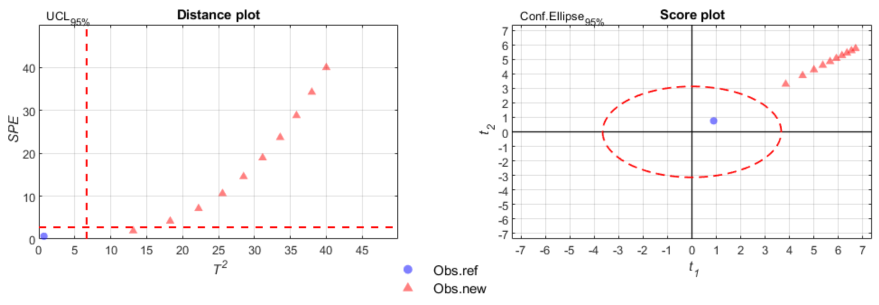
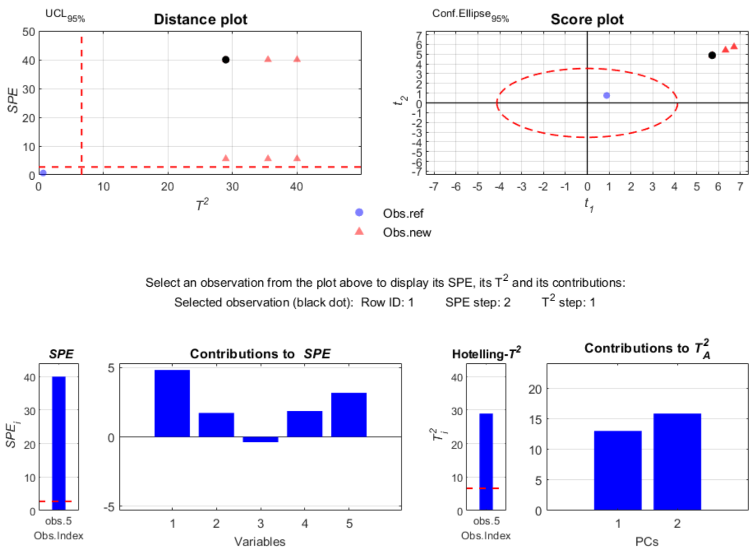


Fig. S8. Plots after generating an outlier with 10 non-linearly-spaced steps between the reference and target values.

C. Grid-wise simulation case. Finally, the grid-wise simulation (*Case III* from the Results section)is illustrated as well. In this case, the same observation from the previous example will be shifted with different number of steps for the SPE and T_A^2 and also with different spacing between these steps for each parameter (Figure S9).

```
outgrid = scout(x, pcamodel_ref, 'grid', 't2y', 40, 'spey', 40, ...
    'nstepsspe', 2, 'nstepst2', 3, 'gspe', 3, 'gt2', 0.3);
Xall = [x; outgrid.X];
obstag = [0; outgrid.tag];
figure('Name', 'PCA Model plot'),
dscplot(Xall, pcamodel_ref, 'click', 'on', 'obstag', obstag, ...
     'steps_spe', outgrid.step_spe, 'steps_t2', outgrid.step_t2);
```



Select an observation from the plot above to display its SPE, its T^2 and its contributions:
Selected observation (black dot): Row ID: 1 SPE step: 2 T^2 step: 1

Fig. S9. Interactive plots after generating an outlying grid of observations.

The third input argument of the `scout.m` function is set to `'grid'` in this case.
In order to have detailed information about the functions implemented in Matlab, see the Matlab Documentation file or type `help <function name>` in the Matlab Command Prompt.

References

1. Ferrer A (2007). "Multivariate Statistical Process Control Based on Principal Component Analysis (MSPC-PCA): Some Reflections and a Case Study in an Autobody Assembly Process." Quality Engineering, 19(4), 311–325. ISSN 0898-2112. doi:10.1080/08982110701621304.