How to Simulate Outliers with the Desired Properties

Alba González-Cebrián[a], Francisco Arteaga[b], Abel Folch-Fortuny[c], Alberto Ferrer[a]

[a] *Multivariate Statistics Engineering Research Group, Universitat Politècnica de València, 46022 València, Spain*
[b] *Universidad Católica de Valéncia, 46001 València, Spain*
[c] *DSM Biotechnology Center, 2613 AX Delft, The Netherlands*

Abstract

Deviating multivariate observations are used typically to test the performance of outlier detection methods. Yet, the generation of outlying cases itself usually appears as a secondary methodological step in methods comparison. In the literature, outliers are defined using certain distribution parameters which differ from those of the clean or reference data. However, these parameters change among authors, leading to a lack of a standard and measurable definition of the characteristics simulated outliers. This makes the comparison between methods hard and its results dependent on the procedure followed to simulate the data. In order to set a standard procedure, a framework to simulate outliers is defined here. Since it is based on certain specifications for both the Squared Prediction Error (SPE) and Hotelling's T^2 statistics from a Principal Component Analysis (PCA) model, tuning them becomes a simple and efficient task. This procedure has been implemented in a set of Matlab functions.

Keywords: PCA, Outliers, Squared Prediction Error, Hotelling's T^2, Simulation, Matlab

1. Introduction

Principal component analysis (PCA) models are specially useful in the context of highly correlated data sets, given its dimensionality and noise reduction power. This is accomplished by obtaining the A latent variables or principal components (PCs) that are linear combinations of the K original variables (usually with $A << K$). These A components explain most of the

Email address: algonceb@upv.es (Alba González-Cebrián)

variance of the K original variables. Beyond the use of PCA as a model itself, it is also widely used for Exploratory Data Analysis (EDA), given the effectiveness offered by the compression of a high-dimensional space to a lower dimension representation that retains most of its variability. One of the reasons why EDA is a good practice before any further use of a data set, is that during this prior steps one can deal with events such as missing data or the potential existence of rare events, also named outliers [1, 2, 3]. When PCA is used in an EDA framework, a model is built, which is known as the PCA Model Building (PCA-MB) stage. In its basic definition, PCA uses least squares parameters which can be very distorted by the influence of outliers. In order to deal with this issue, several approaches that avoid this negative effect have been proposed in literature, assembled in what are known as robust PCA methods. There are plenty of strategies to conduct PCA in a robust way. However, beyond the particularities of each proposal, what basically defines these algorithms is their ability to neglect the influence of potential outliers during the PCA-MB stage. To develop methodological work on how to detect and how to treat outliers, it is often useful to simulate this type of data.

Most approaches used to simulate outliers assume the paradigm of row-wise outliers. This paradigm defines an outlier as a whole observation or row in a matrix. Probably, the most famous model in order to define this situation is the classical Tukey-Huber Contamination Model (THCM) [4]. In these scenarios the observed data $\mathbf{X}$ is thus a mix of unobserved distributions defining two different submatrices $\mathbf{Y}$ and $\mathbf{Z}$, representing data from two different populations. As one could expect, the election of the distributions to simulate both the contaminated and clean parts of the data is a critical procedure step, as it creates the conditions under which the performance of different methods are evaluated and compared. Examining literature, one can notice that the task of simulating the data sets and outliers in the framework of PCA-MB has been addressed differently [5, 6, 7, 3]. In general terms, what remains in common among most proposals is that outliers are defined by setting the parameters of the population to which they belong. Thus, observations are classified as outliers because they are drawn from a distribution that is different from the one which describes the clean data. However, it is not straightforward to stablish the relationship between the chosen parameters for the distribution of the outliers and the resulting properties of the simulated observations. As a result, simulating observations with the desired distance from the reference data set by setting different parameters of the data distribution, becomes practically unfeasible. Moreover, working with this simulation paradigm means to make assumptions about the distributions that describe both the reference and outlying data set. Usually, a multivariate normal distribution is assumed and the mean vector or the covariance matrix are altered in order to generate outlying observations. Yet, assuming a particular probability distribution might not be that simple in case that one wants to simulate outliers for a real reference data set. For these reasons, though the traditional paradigm is technically correct, our belief is that one could further exploit the information offered by a PCA model in order to generate outliers with more control of their properties based on two statistics: the Squared Prediction Error (SPE) and the Hotelling T^2 (T^2). In this work we propose a standard framework for outliers definition and simulation based on its characterization in terms of these statistics.

Firstly, the conceptual framework is introduced, defining the PCA model and the aforementioned pair of statistics. Afterwords, the methodology to generate moderate and severe perturbations, based on shift directions of the SPE and the T^2, is explained. Later on, the proposed variants of the algorithm to simulate outliers are introduced, and some examples of how to simulate controlled outliers are shown. Moreover, some practical applications will be provided to illustrate the potential of the proposed method as standard framework to simulate outliers. In these examples, our procedure to simulate controlled outliers will be configured to emulate other strategies of outliers generation from literature on PCA models. Additionally, the consistency of the outlying properties will be assessed by projecting our simulated outliers onto a robust PCA model. Finally, a summary of the main conclusions is provided. The Matlab code and documentation for outliers generation are available in the GitHub repository `https://github.com/albagc/SCOUTer.git`. Detailed code lines to reproduce the results from Section 3 are available in the *howto.pdf* document on the repository and further details about references for the outliers simulation are provided in Appendix A.

2. Materials and methods

2.1. The PCA model framework

Let $\mathbf{X}$ be a matrix with N observations on K variables. After some preprocessing such as mean-centering and/or unit variance scaling, a PCA model is estimated. This is done by compressing the high-dimensional $\mathbf{X}$ matrix into a low-dimensional subspace of dimension A (with $A \leq rank(\mathbf{X})$). PCA is based on the bilinear decomposition of $\mathbf{X}$ in $\mathbf{X} = \mathbf{T}\mathbf{P}^\top + \mathbf{E}$, where $\mathbf{T}$ is an $N \times A$ matrix of *scores* and $\mathbf{P}$ is a $K \times A$ matrix of *loadings* (Figure 1).

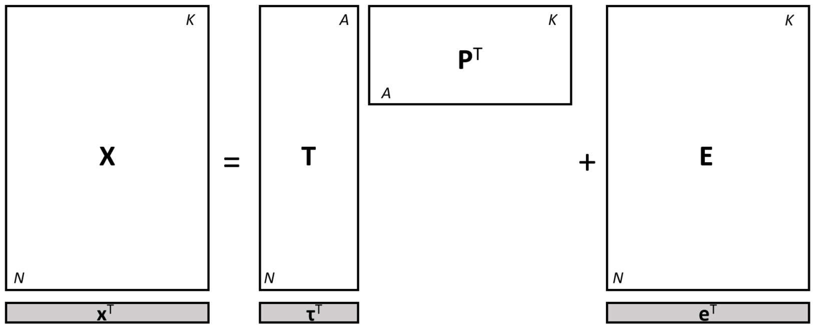


Figure 1: Visual representation for the PCA model.

The A columns of the loading matrix $\mathbf{P}$ are the *loading* vectors $\mathbf{p}_a$, with $a = 1, 2, \ldots, A$. The *score* matrix $\mathbf{T}$ can be considered as a collection of row vectors $\boldsymbol{\tau}^\top$ (scores of an observation) or column vectors $\mathbf{t}_a$ (latent variables, with $\mathbf{t}_a = \mathbf{X}\mathbf{p}_a$ and $a = 1, 2, \ldots, A$). The score matrix can be obtained as $\mathbf{T} = \mathbf{X}\mathbf{P}$, that is, as the projection of the $\mathbf{X}$ matrix on the $A-$dimensional space of the PCA model (i.e., columns of $\mathbf{P}$ matrix). Analogously, given an observation $\mathbf{x}$ of the original $K-$dimensional space, its projection $\boldsymbol{\tau}$ onto the subspace of the model can be obtained using the projection matrix $\mathbf{P}$ as well by $\boldsymbol{\tau} = \mathbf{P}^\top\mathbf{x}$.

From the scores matrix one can recall the explained part of $\mathbf{X}$ in the PCA model as $\hat{\mathbf{X}} = \mathbf{T}\mathbf{P}^\top$. This notation can be used as well for individual observations, where $\hat{\mathbf{x}} = \mathbf{P}\boldsymbol{\tau}$. The original observation can be decomposed by the part explained (i.e., predicted) by the model (signal or $\hat{\mathbf{x}}$) and the error not considered in any of the A latent variables (noise or $\mathbf{e}$). Thus, for a given observation we have $\mathbf{x} = \mathbf{P}\boldsymbol{\tau} + \mathbf{e} = \hat{\mathbf{x}} + \mathbf{e}$. From the previous expressions it can be seen that $\mathbf{E} = \mathbf{X}\left(\mathbf{I} - \mathbf{P}\mathbf{P}^\top\right)$ and then $\mathbf{e} = \left(\mathbf{I} - \mathbf{P}\mathbf{P}^\top\right)\mathbf{x}$.

2.2. Outliers in the PCA model

An observation can be considered an outlier in terms of a PCA model, according to its values for the Squared Prediction Error (*SPE*) and the Hotelling's T^2 (T^2, or more specifically, T^2_A for a PCA model with A components). These statistics, obtained from the residuals and the scores respectively, offer complementary information about the distance of an observation to the PCA model and the majority of data. In [8], there is a comprehensive explanation about the properties of these statistics and their use to detect outlying observations. Following their work, some mathematical aspects of SPE and the T_A^2 are given in this section.

The SPE is the squared Euclidean (perpendicular) distance from the observation $\mathbf{x}$ to the $A-$dimensional subspace of the model, that is $SPE = \mathbf{e}^\top\mathbf{e}$, where $\mathbf{e}$ is the error vector of the observation $\mathbf{x}$. From the previous expression, the SPE can be rewritten as $SPE = \mathbf{x}^\top(\mathbf{I} - \mathbf{P}\mathbf{P}^\top)^\top(\mathbf{I} - \mathbf{P}\mathbf{P}^\top)\mathbf{x}$. Since $(\mathbf{I} - \mathbf{P}\mathbf{P}^\top)$ is symmetric and idempotent matrix:

$$SPE = \mathbf{x}^\top \left(\mathbf{I} - \mathbf{P}\mathbf{P}^\top\right) \mathbf{x} \tag{1}$$

Assuming that residuals follow a multivariate normal distribution, [9], [10] and [11], derived approximate distributions for such quadratic forms.

On the other hand, the Hotelling-T_A^2 statistic for an observation is defined as

$$T_A^2 = \boldsymbol{\tau}^\top \boldsymbol{\Theta}^{-1} \boldsymbol{\tau} = \sum_{a=1}^{A} \left(\tau_a^2/\lambda_a\right) \tag{2}$$

where $\boldsymbol{\Theta}(A \times A)$ is the covariance matrix of $\mathbf{T}$ (diagonal matrix of the highest A eigenvalues $\{\lambda_1, \ldots, \lambda_A\}$). It represents the estimated squared Mahalanobis distance from the center of the latent subspace to the projection of an observation onto this subspace.

When diagnosing which variables yield the obtained values for the SPE and the T^2 it can be useful to check the contributions of each variable to each statistic [8].

From these two statistics (the SPE and the T^2), two complementary control metrics are obtained. Firstly, with an appropriate reference set of data, the in-control PCA model is built. The control limits are defined as well using the reference distributions for each statistic.

Regarding the Upper Control Limit (UCL) for the SPE, several procedures can be used. In [10] it is shown that an approximate SPE critical value at significance level α is given by

$$UCL(SPE)_\alpha = \theta_1 \left[z_\alpha \sqrt{2\theta_2 h_0^2}/\theta_1 + 1 + \theta_2 h_0 (h_0 - 1)/\theta_1^2 \right]^{1/h_0} \tag{3}$$

where $\theta_k = \sum_{j=A+1}^{rank(\mathbf{X})} (\lambda_j)^k$, $h_0 = 1 - 2\theta_1\theta_3/3\theta_2^2$, λ_j are the eigenvalues of the PCA residual covariance matrix $\mathbf{E}^\top\mathbf{E}/(N-1)$, and z_α is the $100(1-\alpha)\%$ percentile of a standard normal variable.

Alternatively, one can use an approximation based on the weighted chi-squared distribution ($g\chi^2_h$) proposed by [9]. In [12] authors suggested a simple and fast way to estimate parameters g and h which is based on matching moments between a $g\chi^2_h$ distribution and the sample distribution of SPE. The mean ($\mu = gh$) and variance ($\sigma^2 = 2g^2h$) of the $g\chi^2_h$ distribution are equated with the sample mean (b) and variance (v) of the SPE sample. Hence, the Upper SPE Control Limit at significance level α is given by

$$UCL(SPE)_\alpha = v\chi^2_{(2b^2/v),\alpha}/(2b) \tag{4}$$

where $\chi^2_{(2b^2/v),\alpha}$ is the 100(1-α)% percentile of the corresponding chi-squared distribution with $2b^2/v$ degrees of freedom.

Upper Control Limits (UCL) for the T^2_A at a significance level (type I) risk α can be obtained assuming that the statistic follows an F distribution

$$T^2_A \sim A\left(N^2 - 1\right) F_{A,(N-A)}/(N\left(N - A\right)) \tag{5}$$

Thus, the corresponding UCL from Equation 5 is given by

$$UCL(T^2_A)_\alpha = A\left(N^2 - 1\right) F_{(A,(N-A)),\alpha}/(N\left(N - A\right)) \tag{6}$$

According to the aforementioned conceptual meaning of these multivariate statistics (SPE and T^2_A), observations above their associated UCL will be representing different types of outliers.

The first type of outliers, with high SPE, occurs when the correlation structure between variables is different from the observed one during the model fitting with the clean data set. Using these observations to fit the model can lead to dramatic distortions on the correlation structure captured by the PCs. These perturbations are named “moderate outliers” or “anomalous observations” and are caused by unusual variations outside the model.

The second type of outliers, with high values of the T^2_A, appears when the correlation structure between measured variables remains constant but their absolute values differ from the expected ones. These perturbations are named “severe outliers” or “extreme observations” and they are usually representing unusual shifts in the model (i.e. shifts that respect the correlation structure of the model). This leads to extreme values in the projection of these observations with respect to the ones obtained for the clean data set.

These links between distances and types of outliers, or outlying properties, can be described using the Squared Prediction Error and Hotelling’s T^2_A. On

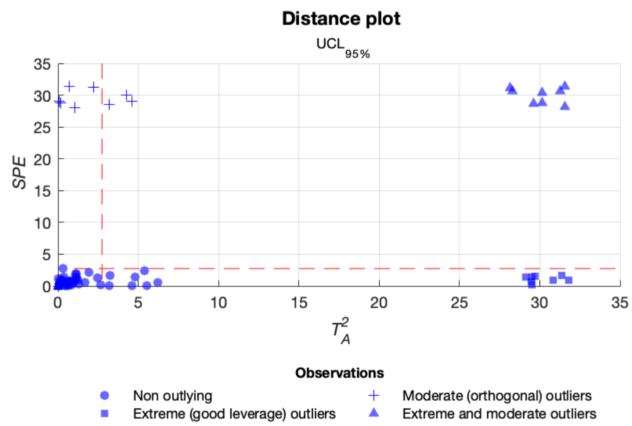


Figure 2: Types of outliers according to the PCA model built with a reference data set (blue dots). Red discontinuous lines are the 95% control limits for the SPE and Hotelling's T^2.

one hand, moderate outliers will present high values for the SPE statistic, which is the reason why they are also known as orthogonal outliers. On the other hand, extreme outliers are observations with high values for the T_A^2. They are named as well good leverage observations, since their presence does not distort the correlation structure of the model. There can be as well observations that are both moderate and extreme outliers (Figure 2).

In conclusion, outliers in the context of a PCA model will be associated with large values of the SPE, the T_A^2 or both distances. Using this pair of statistics to describe observations provides more meaningful criteria to define outliers than setting their distribution parameters in the original or latent space. Setting not only the type, but also how far outliers will be from the reference data set, is something plausible when using the SPE and T_A^2 as targets to simulate outliers. This idea of representing all types of outliers as combinations of SPE and T_A^2 values, is the basis of the simulation approach presented in this work.

2.3. Framework to generate outliers

Our proposal for the generation of outliers is to transform an observation $\mathbf{x}$, with given SPE and T^2 values ($SPE_{\mathbf{x}}$ and $T^2_{\mathbf{x}}$, respectively), into a new observation with an SPE and/or T^2 values specified by the user ($SPE_{\mathbf{y}}$ and

$T^2_{\mathbf{y}}$, respectively). The transformation will consist in a shift of the observation following certain direction in the space of the original variables.

Moving the observation $\mathbf{x}$ in the direction $\mathbf{v}$ to obtain a new observation $\mathbf{y} = \mathbf{x} + \mathbf{v}$, we can calculate the new value of the SPE and the T^2 statistics, based on the original values:

$$SPE_{\mathbf{x}+\mathbf{v}} = (\mathbf{x}+\mathbf{v})^\top \left(\mathbf{I} - \mathbf{P}\mathbf{P}^\top\right)(\mathbf{x}+\mathbf{v}) = SPE_{\mathbf{x}} + \mathbf{v}^\top \left(\mathbf{I} - \mathbf{P}\mathbf{P}^\top\right)(2\mathbf{x}+\mathbf{v}) \tag{7}$$

$$T^2_{\mathbf{x}+\mathbf{v}} = (\mathbf{x}+\mathbf{v})^\top \mathbf{P}\mathbf{\Theta}^{-1}\mathbf{P}^\top (\mathbf{x}+\mathbf{v}) = T^2_{\mathbf{x}} + \mathbf{v}^\top \mathbf{P}\mathbf{\Theta}^{-1}\mathbf{P}^\top (2\mathbf{x}+\mathbf{v}) \tag{8}$$

The next issue is how to choose the direction $\mathbf{v}$. An obvious choice is to shift the observation in the direction that joins it with the origin of coordinates in the original data space, taking $\mathbf{v} = c\mathbf{x}$. In this case, it is easy to calculate the change in both statistics:

$$SPE_{\mathbf{x}+c\mathbf{x}} = (1+c)^2 SPE_{\mathbf{x}} \tag{9}$$

$$T^2_{\mathbf{x}+c\mathbf{x}} = (1+c)^2 T^2_{\mathbf{x}} \tag{10}$$

However, we are interested in finding directions in which we can control the change that occurs in each statistic. For example, there are specific directions that allow the change in one of both statistics, without affecting the other. In particular, we can move the observation in the direction of its residual vector in the PCA model: $\mathbf{e} = \left(\mathbf{I} - \mathbf{P}\mathbf{P}^\top\right)\mathbf{x}$, so that a change in the SPE will occur, without modifying the T^2. Similarly, we can move it in the direction that joins the projection of the observation on the model with the origin (i.e. the direction of the predicted observation $\hat{\mathbf{x}}$): $\mathbf{P}\mathbf{P}^\top\mathbf{x}$, so that there will be a change in T^2, without modifying the SPE. As both directions are orthogonal, we can compose both displacements in one operator, with control over the amount by which each of them increases. This will be illustrated in following sections.

2.3.1. Shift of the SPE statistic

If we move the observation $\mathbf{x}$ in the direction given by its residual vector (according to the PCA model): $\mathbf{e} = \left(\mathbf{I} - \mathbf{P}\mathbf{P}^\top\right)\mathbf{x}$, multiplied by a scalar a, we get, from Equation 7 and Equation 8:

$$SPE_{\mathbf{x}+a\left(\mathbf{I}-\mathbf{P}\mathbf{P}^\top\right)\mathbf{x}} = SPE_{\mathbf{x}} + a\mathbf{x}^\top \left(\mathbf{I} - \mathbf{P}\mathbf{P}^\top\right)\left(2\mathbf{x} + a\left(\mathbf{I} - \mathbf{P}\mathbf{P}^\top\right)\mathbf{x}\right) = (1+a)^2 SPE_{\mathbf{x}} \tag{11}$$

$$T^2_{\mathbf{x}+a\left(\mathbf{I}-\mathbf{P}\mathbf{P}^\top\right)\mathbf{x}} = T^2_{\mathbf{x}} + a\mathbf{x}^\top\left(\mathbf{I}-\mathbf{P}\mathbf{P}^\top\right)\mathbf{P}\boldsymbol{\Theta}^{-1}\mathbf{P}^\top\left(2\mathbf{x}+a\left(\mathbf{I}-\mathbf{P}\mathbf{P}^\top\right)\mathbf{x}\right) = T^2_{\mathbf{x}} \tag{12}$$

We can choose the value a to achieve a target value for the SPE statistic, say $SPE_{\mathbf{y}}$:

$$(1+a)^2 SPE_{\mathbf{x}} = SPE_{\mathbf{y}} \rightarrow a = \sqrt{SPE_{\mathbf{y}}/SPE_{\mathbf{x}}} - 1 \tag{13}$$

Note that the selected direction is the one that maximizes the change in the SPE, because the gradient of this statistic is: $\nabla(SPE)(\mathbf{x}) = 2\left(\mathbf{I}-\mathbf{P}\mathbf{P}^\top\right)\mathbf{x}$.

2.3.2. Shift of the T^2 statistic

If we move the observation $\mathbf{x}$ in the direction $\mathbf{P}\mathbf{P}^\top\mathbf{x}$, multiplied by a scalar b, we get, from Equation 7 and Equation 8:

$$SPE_{\mathbf{x}+b\mathbf{P}\mathbf{P}^\top\mathbf{x}} = SPE_{\mathbf{x}} + b\mathbf{x}^\top\mathbf{P}\mathbf{P}^\top\left(\mathbf{I}-\mathbf{P}\mathbf{P}^\top\right)\left(2\mathbf{x}+b\mathbf{P}\mathbf{P}^\top\mathbf{x}\right) = SPE_{\mathbf{x}} \tag{14}$$

$$T^2_{\mathbf{x}+b\mathbf{P}\mathbf{P}^\top\mathbf{x}} = T^2_{\mathbf{x}} + b\mathbf{x}^\top\mathbf{P}\boldsymbol{\Theta}^{-1}\mathbf{P}^\top\left(2\mathbf{x}+b\mathbf{P}\mathbf{P}^\top\mathbf{x}\right) = (1+b)^2 T^2_{\mathbf{x}} \tag{15}$$

We can choose the value b to achieve a target value for the T^2 statistic, say $T^2_{\mathbf{y}}$:

$$(1+b)^2 T^2_{\mathbf{x}} = T^2_{\mathbf{y}} \rightarrow b = \sqrt{T^2_{\mathbf{y}}/T^2_{\mathbf{x}}} - 1 \tag{16}$$

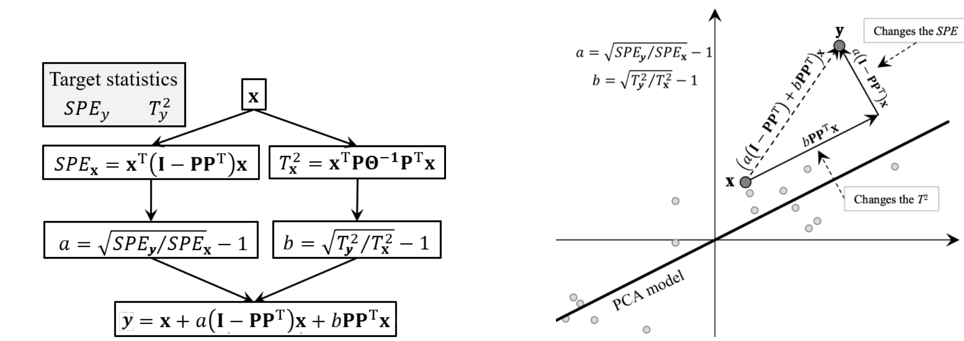
We can also select the direction that maximizes the change in the T^2 statistic, without changing the SPE statistic, choosing the gradient of the T^2 statistic: $\nabla(T^2) = 2\mathbf{P}\boldsymbol{\Theta}^{-1}\mathbf{P}^\top\mathbf{x}$. We do not use this direction because it is difficult to parametrise the amount of change in the T^2 statistic.

2.3.3. Shift both statistics simultaneously

If we have an observation $\mathbf{x}$ with statistics $SPE_{\mathbf{x}}$ and $T^2_{\mathbf{x}}$, we can transform it into a new observation with statistics $SPE_{\mathbf{y}}$ and $T^2_{\mathbf{y}}$ combining the aforementioned transformations:

$$\mathbf{y} = \mathbf{x} + a\left(\mathbf{I}-\mathbf{P}\mathbf{P}^\top\right)\mathbf{x} + b\mathbf{P}\mathbf{P}^\top\mathbf{x} \tag{17}$$

With $a = \sqrt{SPE_{\mathbf{y}}/SPE_{\mathbf{x}}} - 1$ and $b = \sqrt{T^2_{\mathbf{y}}/T^2_{\mathbf{x}}} - 1$, as seen in Equation 13 and Equation 16. The procedure to build a new observation with desired SPE and T^2 statistics, based on an arbitrary prior observation $\mathbf{x}$,



(a) Scheme to generate a new observation with target SPE and T^2 statistics.

(b) Simple representation of the transformation from $\mathbf{x}$ to $\mathbf{y}$.

Figure 3: Representation of the algorithm to generate $\mathbf{y}$.

is illustrated in Figure 3a. The visual representation of the algorithm with a model of only one PC, for an original space with only two variables is represented in Figure 3b.

Furthermore, there is another aspect that can be used to control the outlying behaviour of the new observations. Given the reference and target values of a statistic, one can generate a series of $M-1$ intermediate observations between the reference and the target one: $\{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_{M-1}\}$. Mathematically, the expected value of a statistic H_m as a result of a transition from the reference H_0 to the target value H_M:

$$H_m = H_0 + (m/M)^\gamma (H_M - H_0) \qquad m = 1, 2, ..., M-1 \tag{18}$$

Thus, SPE_m and T_m^2 will follow a pattern of gradual change according not only to the number of steps, but also to the spacing between them. This spacing is regulated in Equation 18 by the γ parameter. As it can be appreciated in Figure 4, when this parameter is set to 1, the spacing between steps is linear, shifting towards a non-linear dynamic as it drifts from 1.

Given that both parameters (γ_{SPE} and γ_{T^2}) can be shifted simultaneously, this gives to the user the flexibility to simulate a wider variety of trajectories for each possible combination of values along the spacing of the two parameters. Performing simultaneous shifts with some values for the parameters, results in the curves of Figure 5.

This framework, including the possibility of controlling the distance between intermediate observations in series of outliers, can be useful in order to study and compare the sensitivity of different robust PCA approaches

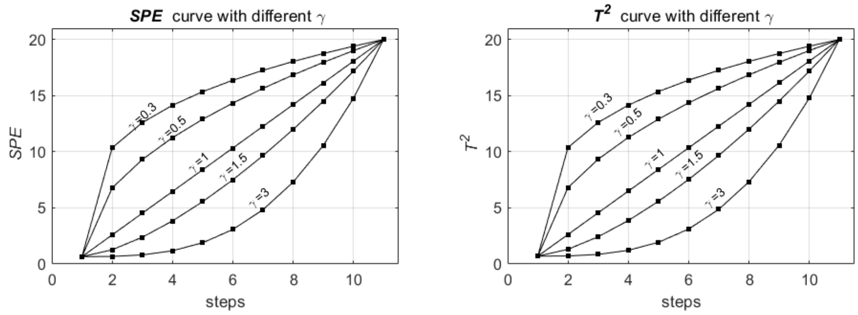


Figure 4: Curves for the SPE (left) and T^2 (right) statistics along the shift in 20 steps for different values of their spacing parameters γ.

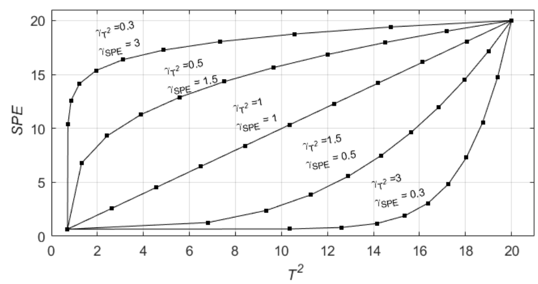


Figure 5: Curves for the SPE and T^2 statistics along the shift in 20 steps for different combinations of their γ parameters.

or methods for outlying detection. Thus, one could know not only for what type of outliers, but also at which step, one method performs differently from others. Finally, considering all these parameters one has the complete flux diagram of the procedure in Figure 6.

If a given observation $\mathbf{x}$ is moved in different directions, it will be appreciated both in the SPE and T^2 statistics, and also in the scores. Figure 7 illustrates different shifts on a five dimensional observation $\mathbf{x}$ according to a reference PCA model.

In Figure 7a, red dashed lines represent the UCL for the T^2 and SPE statistics. The ellipse represented in the score plot from Figure 7b, is the contour curve of the confidence ellipsoid for the T^2 statistic, calculated for a confidence level of $(1-\alpha)\times 100\%$. From Equation 6, it is obtained an ellipsoid delimited in each dimension (i.e. PC) of the latent subspace. The contour of that ellipsoid represents a region of the space which holds $T^2 = T^2_{100(1-\alpha)\%CL}$ for each observation lying on that contour. Since the score plot is a bi-

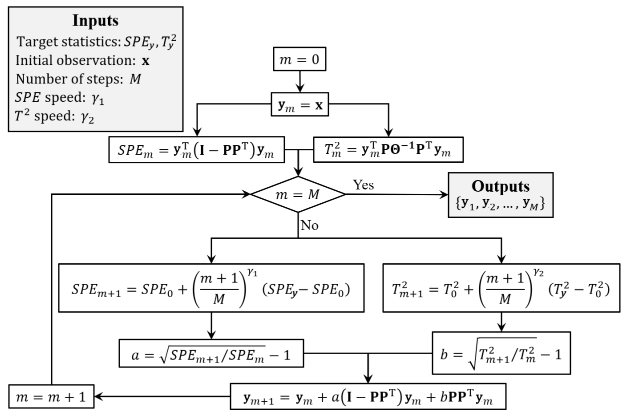
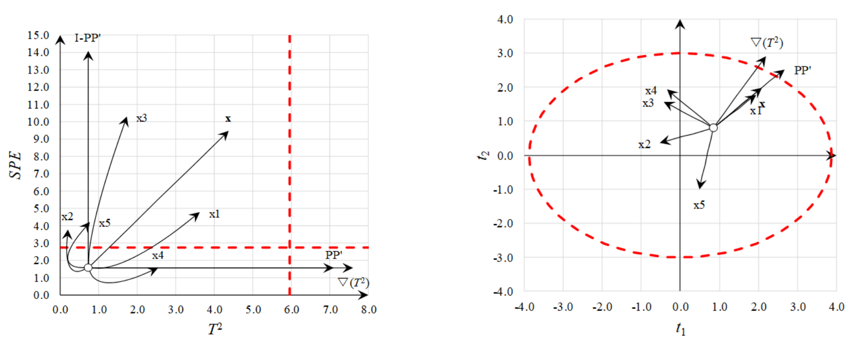


Figure 6: Flux diagram of simulation algorithm including all the parameters.



(a) Distance plot with five different shift directions for an observation.

(b) Score plot with five different shift directions for an observation.

Figure 7: Illustration of how moving in different directions would affect the SPE and T^2 (a) and the scores (b).

dimensional plot, the bi-dimensional representation of the confidence ellipsoid turns into a confidence ellipse. Therefore, observations lying outside the ellipse will be over passing the UCL for the T^2 statistic.

The first set of directions are those that correspond to the five variables (labelled as x1, ..., x5). The trivial direction ($\mathbf{v} = \mathbf{x}$) is also considered. The direction corresponding to the residual vector ($\mathbf{v} = (\mathbf{I} - \mathbf{P}\mathbf{P}^\top)\mathbf{x}$) is easy to recognize, since it causes an increase in the SPE without affecting the T^2 statistic. In the distance plot (Figure 7a) it is represented as a vertical arrow, whereas it does not appear in the score plot (Figure 7b), given that the projection of $\mathbf{x} + a(\mathbf{I} - \mathbf{P}\mathbf{P}^\top\mathbf{x})$ in the model space is the same as that of $\mathbf{x}$, for all a values.

The last two directions are $\mathbf{P}\mathbf{P}^\top\mathbf{x}$ and $\mathbf{P}\boldsymbol{\Theta}^{-1}\mathbf{P}^\top\mathbf{x}$ (labelled as $\nabla(T^2)$ in Figure 7). These two directions are in the model plane and this means that the SPE will not be affected, which can be appreciated by the horizontal arrows in Figure 7a. The magnitude of the shift in the T^2 value is bigger for the $\mathbf{P}\boldsymbol{\Theta}^{-1}\mathbf{P}^\top\mathbf{x}$ direction, since it corresponds to the gradient of the T^2 statistic. The trajectory described by the scores when the direction $\mathbf{P}\mathbf{P}^\top\mathbf{x}$ is chosen, is an extension of the segment that joins the origin (0,0) with the scores of $\mathbf{x}$ (i.e. the direction of the predicted observation $\hat{\mathbf{x}}$). The trajectory followed when the shift is performed in the direction $\mathbf{P}\boldsymbol{\Theta}^{-1}\mathbf{P}^\top\mathbf{x}$ ($\nabla(T^2)$) is perpendicular to the $(1-\alpha)\times 100$ confidence level Hotelling's T^2 ellipse, which is defined as the level curve for the T^2 statistic.

3. Results

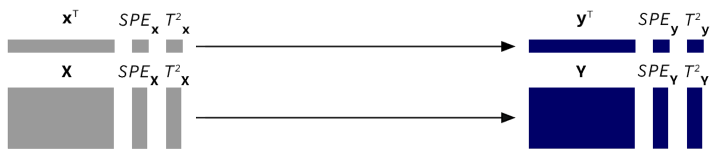
In this section, some examples of how to simulate outliers with the desired properties are shown. This section is divided in two main parts. The first part will present results for three different scenarios of outliers simulation. Afterwards, four examples of outliers simulation extracted from literature are emulated using the framework proposed in this work. The aim of this exercise is to show how the technique described in this work can comprise other particular simulation settings. Finally, an assessment about the properties of the simulated outliers in terms of a robust PCA model is provided as well.

3.1. Cases of use of the proposed method.

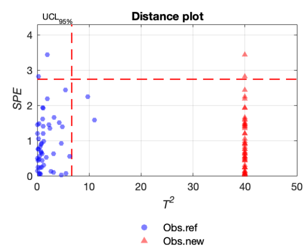
These results illustrate three generic simulation scenarios: generating outliers in one step, generating a sequence of outliers, and generating a grid of outliers. For this purpose, a reference matrix $\mathbf{X}$ of $n = 50$ observations and $k = 5$ normally distributed variables is simulated. The PCA model based on $\mathbf{X}$ is built with two PCs, assuming a type I risk α of 5% and performing a mean centering. All functions along their documentation and the script to reproduce the following scenarios can be downloaded from the github repository `https://github.com/albagc/SCOUTer.git`. A detailed explanation abouth the obtention of the following results can be found in the *howto.pdf* file.

3.1.1. Case I: One-step simulation of outliers.

This is the simplest case, in which from an initial observation $\mathbf{x}$ with reference values $SPE_{\mathbf{x}}$ and $T^2_{\mathbf{x}}$, a new observation $\mathbf{y}$ is obtained, with the desired $SPE_{\mathbf{y}}$ and $T^2_{\mathbf{y}}$ values (8a). The aforementioned scheme can be easily generalised for a set of observations. In the following example, the original $\mathbf{X}$ matrix will be drifted from its initial coordinates. In this scenario a set of one-step outliers is generated by increasing only the T^2 value (i.e. extreme outliers). The SPE remains at its reference value.



(a)



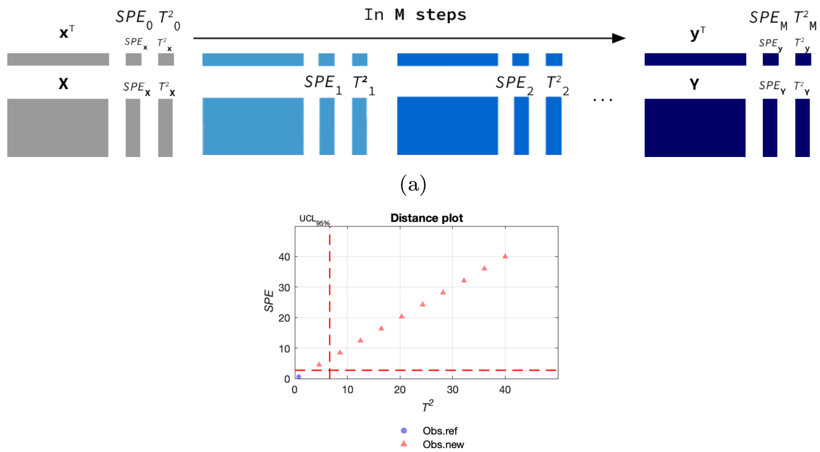
(b)

Figure 8: (a) Illustration of a one-step simulation of controlled outliers. (b) Distance plot with the reference (blue circles) and the shifted (red triangles) data sets, performing a single step keeping the initial $SPE_{\mathbf{X}}$ value, but setting a target value $T^2_{\mathbf{Y}} = 40$ for all the observations.

As it can be seen in Figure 8b, all observations have been shifted in their distance to the center on the model plane, drawing a contour on the score plot for the value $T^2_A = 40$, whereas they have kept their values on the SPE statistic. In other words, this is an example of a how to simulate a set of extreme observations.

3.1.2. Case II: Step-wise simulation of outliers

In this scenario, the transition between the reference and the target values for the statistics is performed with a spacing of n steps between them. From a reference observation $\mathbf{x}$ (or set of observations $\mathbf{X}$) with reference values $SPE_{\mathbf{x}}$ and $T^2_{\mathbf{x}}$ (or $SPE_{\mathbf{X}}$ and $T^2_{\mathbf{X}}$), a series of $M-1$ new sets of observations up to $\mathbf{y}$ (or $\mathbf{Y}$) with the desired $SPE_{\mathbf{y}}$ and $T^2_{\mathbf{y}}$ (or $SPE_{\mathbf{Y}}$ and $T^2_{\mathbf{Y}}$) values is generated (9a).



(a)

(b)

Figure 9: (a) Illustration of a M-step simulation of controlled outliers. (b) Distance plot after performing a 10-step shift both in the SPE_x and the T^2 values from one initial observation $\mathbf{x}$ (blue circle).

In the example from above (Figure 9b), there is a linear spacing between steps for the SPE and the T^2. However, the spacing between steps can be tuned, as seen in Figure 4 and Figure 5 from Section 2.3.

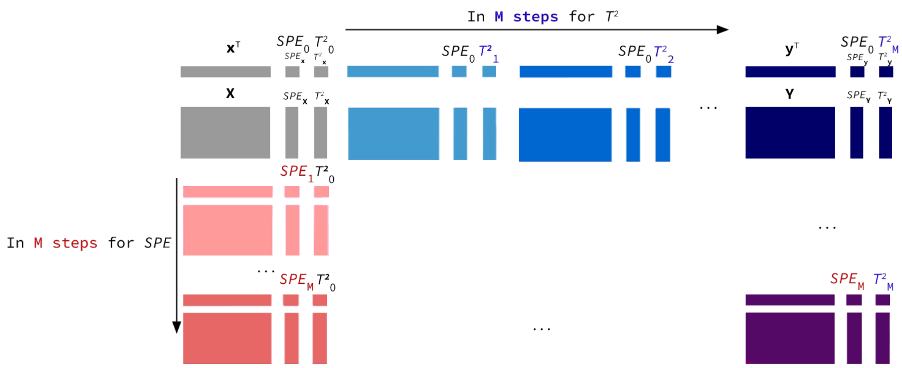
3.1.3. Case III: Grid-wise simulation of outliers

With the step-wise approach the same number of steps is performed for both statistics. Finally, the grid-wise case enables a different number of steps for each statistic. Starting from an initial data set $\mathbf{x}$ (or $\mathbf{X}$) with reference values $SPE_{\mathbf{x}}$ and $T^2_{\mathbf{x}}$ (or $SPE_{\mathbf{X}}$ and $T^2_{\mathbf{X}}$), a grid of new observations combining each step of the statistics is obtained (Figure 10a). As a result, there are as many data sets simulated as combinations between the steps of the statistics.

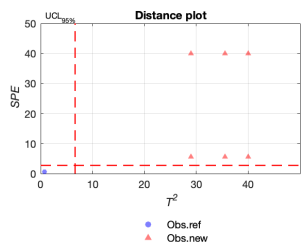
In this last case, a grid with 2 steps for the SPE and 3 steps for the T^2 has been produced, setting different spacing parameters for each parameter as well (Figure 10b).

3.2. Comparison to other simulation methods and PCA frameworks

The aim of this section is to address two important questions about the simulation method proposed in this work: i) the proposed simulation framework can encompass other existing simulation strategies, and ii) if the prop-



(a)



(b)

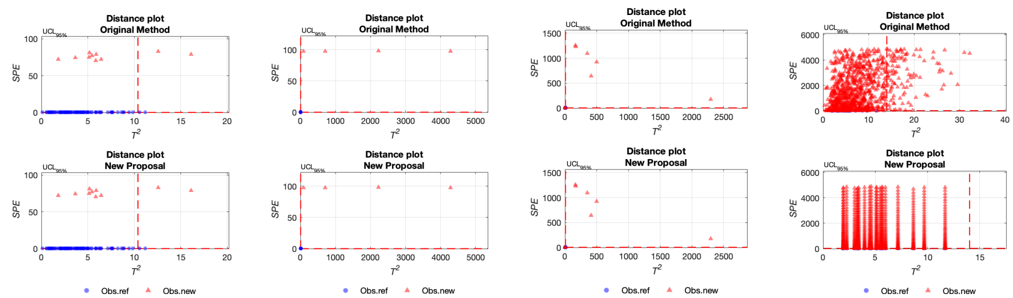
Figure 10: (a) Illustration of a grid-case simulation with M-step shifts for the SPE and the T^2. (b) Distance plot after performing two steps for the SPE with $\gamma_{SPE} = 3$ and three steps for the T^2 with $\gamma_{T^2} = 0.3$ from one reference observation $\mathbf{x}$ (blue circle).

erties of the simulated outliers will be maintained when they are projected onto PCA models fitted with other algorithms rather than the classical least squares version.

3.2.1. Simulation of other outlier generation strategies

With the aim of assessing if the proposed method can be seen as a general simulation framework, four strategies to simulate outliers extracted from literature [5, 6, 13, 3] will be redefined in terms of the proposed simulation framework. Figure 11 provides a graphical comparison between the simulated outliers following the original strategy from the aforementioned works and using the algorithm proposed in this article. Each simulation procedure and all the details to get the results presented in this section, are further explained in the Appendix A.

At first glance, one can notice in Figure 11 that despite sharing the purpose of simulating outliers, each strategy leads to very different outliers in



(a) (b) (c) (d)

Figure 11: Distance plots of the observations simulated as in [5] (a), as in [6] (b), as in [13] (c) and as in [3] (d) using the approach from the original work (first row) and the proposed algorithm controlling the outlier properties (second row).

qualitative and quantitative terms. In Figures 11a and 11d outliers are far in terms of their orthogonal distance but their projection onto the model plane seems still under control limits. These plots differ from the ones reported in Figures 11b and 11c, where outliers are distant both in terms of the T^2 and in the SPE.

Furthermore, the simulation procedure from Figure 11c differs strategically from the others, since the same set of observations is shifted apart in 50 steps from their reference values. In Figure 11d, it can be appreciated the gradual shift of the same set of observations increasing their SPE and randomly shifting the T^2. This can be seen as well in Figure A.13. It also stands out the difference between the upper and lower distance plots in Figure 11d. This is because we considered that variations of the T^2 in their simulated outliers were not a strategical feature of the simulation. This is explained in detail in the Appendix A.

Comparing the original methods to simulate outliers (upper row of plots in Figure 11), it can be seen that all of them increase the SPE of the outliers. This is because in the end, despite following different strategies, all procedures to simulate outliers rely on breaking the correlation structure described by the reference data set. This is done differently by each author. In [5] the simulation strategy relies on adding noise to the outlying observations, whereas in [3], the noise is introduced as the new mean vector of the outlying distribution. This results in outliers with an increased SPE but with a moderate T^2, as it can be seen in Figures 11a and 11d. In [6], outliers are generated by altering the variance of variables, which leads to an increase in the T^2 (Figure 11b). The mean vector of the outliers distribution is changed as well, in such a way that the correlation pattern is not respected anymore, which leads to the increase of the SPE. Finally, in [13], authors shift the sign of randomly selected cells. As a consequence, they are clearly breaking the correlation structure and this can lead as well to an increase in the T^2 of the outlying observations (Figure 11c).

The comparison between plots from the upper and lower row in Figure 11, shows that results obtained by the proposed algorithm to simulate outliers with the desired properties, are fairly similar to the ones obtained by other simulation settings. Furthermore, some limitations of the traditional paradigm to simulate outliers can be seen as well. This traditional framework relies on changing the parameters of the distribution that describes the outlying population, but there is not a direct and clear relationship between the new parameters of the outlying distribution and their effect on the SPE or the T^2. Consequently, it is difficult to control how this new distribution will affect to the outlying properties of the outliers when they are projected onto the reference PCA model. This can be appreciated in the fact that most simulation strategies easily increase the SPE of their observations, but without controlling its value and without having the same control over the T^2 of the outliers. In fact, the T^2 seems to be a more uncontrolled parameter and any of the proposals includes specific outliers for the T^2. This is probably because it is not trivial how to find a new mean vector for the distribution of the outliers that still respects the correlation structure of the reference data set.

The change from the traditional simulation paradigm, to the new one proposed in this work, simplifies the relationship between the simulation setup and the properties of the resulting outliers.The algorithm proposed in this work does not rely on the distribution of the reference and the outlying observations and it has an independent control over the SPE and the T^2. This results in a new simulation approach that is versatile enough to encompass other particular simulation strategies (Figure 11). Besides, differences between simulation settings can be directly measured in terms of the target SPE and T^2 of the outliers.

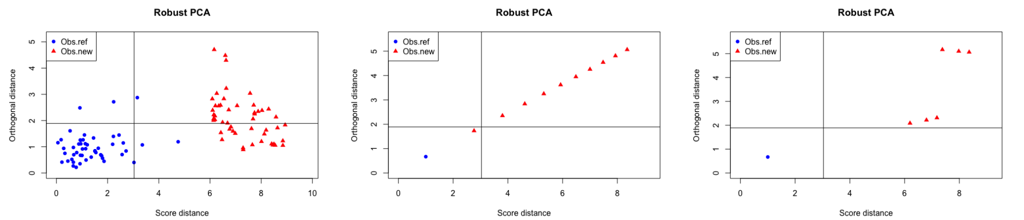
3.2.2. Properties of the simulated outliers in a robust PCA model

The second aspect to assess in this comparison is to what extent (just quantitative or also qualitative) outliers simulated by the proposed algorithm behave as outliers in terms of other detection techniques. In this sense, it is also interesting to assess if the properties of simulated outliers change when they are expressed in terms of different distance metrics. For instance, some robust PCA techniques differ not only in the core algorithm to calculate the principal components, but also in terms of the statistics that measure the distance of an observation to the model. Hence, the whole basis used by our proposed framework to define the outliers, is different in these cases. This may affect the properties of simulated observations when they are defined in these new terms.

For this purpose, simulation scenarios from sections 3.1.1, 3.1.2 and 3.1.3, will be projected onto a robust PCA model calculated with MacroPCA[3]. This technique can be considered as an ensemble of several outlier detection methods. It includes the *Detect Deviating Cells* (DDC) [14] algorithm as first step in order to detect outlying cells, which itself, can be regarded as an outlier detection technique. Later on, MacroPCA algorithm fits a robust PCA model using a version of the ROBPCA algorithm [15]. In this ROBPCA step, they include the detection of outlying observations in several steps. Firstly, in the Projection Pursuit step, to rank rows according to their outlyingness. Secondly, after the iterative subspace estimation, they apply a filter on observations based on their orthogonal distance to the model. Thirdly, they apply the DetMCD method [16], for the covariance matrix estimation, which also includes intermediate distance calculations to use the least distant observations for the covariance matrix computation. Finally, when the PCA model has been estimated, they perform a last outlier detection based on two robust distance metrics: the orthogonal distance and the score distance.

It is worth to highlight that although the distance metrics used in [3] do not coincide with the SPE and T^2, their conceptual meaning is equivalent, since they represent the orthogonal distance and the Mahalanobis distance on the model plane, respectively. Thus, we considered that MacroPCA was clearly representative as a state-of-the-art outlier detection method and as a robust PCA model building algorithm. Moreover, its good performance in outliers detection and the comparable meaning of its distance metrics (orthogonal and score distances) to ours (the SPE and the T^2), were considered as interesting factors for the comparison. Results shown in Figure 12 were obtained using the *cellWise* package in R (available in `https://CRAN.R-project.org/package=cellWise`).

As it can be seen in Figure 12, qualitative properties of the simulated outliers are still met in terms of alternative PCA models and distance metrics. However, there are some differences in the distance values and their Upper



(a) (b) (c)

Figure 12: Distance plots of the observations simulated in Figure 8b (a), in Figure 9b (b) and in Figure 10b (c), when they are projected onto the PCA model fitted using MacroPCA with the reference data set. Blue circles represent reference observations, whereas red triangles represent the simulated outliers. Black lines represent the Upper Control Limits for the Orthogonal Distance (ordinate) and the Score Distance (abscissa).

Control Limits, which is reasonable given that the Orthogonal Distance and Score Distance are not exactly the SPE nor the T^2. Results in Figure 12a show also an increase of the simulated outliers in terms of the orthogonal distance. Given the robust estimation of the covariance determinant in the detMCD step of MacroPCA, extreme observations in the T^2 were detected as outliers, and excluded for the computation of the final PCA model parameters. As a result, since these observations were excluded at some point for the PCA model building, we find reasonable that they also increased their distance to the model. Nonetheless, in all cases simulated outliers keep the outlying character that they were asked to represent in first instance. This can be appreciated by their position above the cut-off values for the distances in all distance plots, indicating the persistence of their outlying properties.

4. Conclusion

In this work, a new framework to simulate outliers directly controlling their outlying properties has been proposed. This approach is based on the use of a well-known pair of statistics, the SPE and the Hotelling-T^2 from a PCA model, which evaluate in a complementary way how far an observation is from the majority of the data set (i.e. the outlyingness degree).

Given an observation with initial values for the statistics, a PCA model and target values for the pair of statistics, our simulation method drifts the aforementioned observation in a direction that shifts the initial SPE and Hotelling-T^2 until reaching their target values. This shift direction is a combination of two orthogonal directions, each one independently controlling the shift on the SPE and the Hotelling-T^2.

This feature is a key factor, since it enables a specific control over the two properties that define multivariate outliers in terms of a PCA model. This becomes critical specially when simulating anomalous data, which is a extremely common procedure when testing the performance of different statistical methods handling datasets with outlying observations. However, the outliers generation is usually an ad hoc procedure, with a lack of standard protocols and being based most of the times, even when working with PCA models, on distributions and parameters that do not tune neither how nor how much an observation is outlying. This makes the supposed benefits of the different statistical methods depend on the nature of the simulated outliers and consequently, the comparison of the different methods reported in the literature becomes difficult or impossible. Moreover, most simulation methods require an assumption about the distribution of the reference data set, and simulate outliers by changing one of its parameters, such as the mean or the covariance matrix. This simulation paradigm might not be feasible to implement with real data sets, when the distribution is unknown. Furthermore, the relationship between the new parameters of the distribution and the outlying properties of the simulated observations is not simple and direct.

In Section 3.2.1 we showed how the methodology proposed in this article, successfully encompasses particular simulation strategies proposed in the literature in a common framework. Consequently, the comparison between approaches can be easily measured in terms of target specifications or in terms of the strategy followed to shift the outliers, i.e.: one step, step-wise of grid-wise. Besides, we also illustrated the shortage of extreme (good leverage) outliers simulated in the literature given the difficulty of modifying the reference distribution while respecting its covariance structure, which is easily achieved by the simulation framework proposed in this paper (Figure 8b). Moreover, in Section3.2.2 we also showed how the outlying properties are, at least, qualitatively consistent when the simulated outliers are projected on a robust PCA model.

However, the proposed method has some limitations, which are further addressed in Appendix B. The simulation procedure does not set any restriction in case that binary or categorical variables are present in the matrix. Naturally, this framework is also restricted by the same limitations as the PCA model is, such as the inability to model non-linear relations between variables (see Appendix B.2).

In summary, the framework proposed in this paper offers the possibility of generating outlying observations with a wide range of desired properties, given that the user can control the pair of statistics that essentially define the outlyingness degree: the SPE and the Hotelling-T^2. This procedure has been implemented in Matlab, providing a set of functions to perform the PCA Model Building and the simulation of controlled outliers. Further details about the Matlab code can be found in the documentation file available in the GitHub repository.

Computational details

The results have been obtained executing the functions from `https://github.com/albagc/SCOUTer.git` in Matlab version R2020a 9.8.0.1323502. Further information about the functions can be found in the *documentation* and *howto* documents on the repository.

Acknowledgements

Financial support was granted by the UPV as a part of the FPI Grant (Subprogramme I) under the UPV Research and Development support programme and also by the Spanish Ministry of Economy and Competitiveness under the project DPI2017-82896-C2-1-R.

References

[1] F. Arteaga, A. Ferrer, Dealing with missing data in MSPC: Several methods, different interpretations, some examples, Journal of Chemometrics 16 (2002) 408–418. doi:10.1002/cem.750.

[2] A. Smoliński, B. Walczak, J. W. Einax, Exploratory analysis of data sets with missing elements and outliers, Chemosphere 49 (2002) 233–245. doi:10.1016/S0045-6535(02)00326-0.

[3] M. Hubert, P. J. Rousseeuw, W. Van den Bossche, W. V. den Bossche, MacroPCA: An all-in-one PCA method allowing for missing values as well as cellwise and rowwise outliers, Technometrics (2018) 1–18. URL: https://doi.org/10.1080/00401706.2018.1562989. doi:10.1080/00401706.2018.1562989. arXiv:1806.00954.

[4] P. J. Huber, Robust Estimation of a Location Parameter, The Annals of Mathematical Statistics 35 (1964) 73–101. doi:10.1214/aoms/1177703732.

[5] I. Stanimirova, M. Daszykowski, B. Walczak, Dealing with missing values and outliers in principal component analysis, Talanta 72 (2007) 172–178. doi:10.1016/j.talanta.2006.10.011.

[6] S. Serneels, T. Verdonck, Principal component analysis for data containing outliers and missing elements, Computational Statistics and Data Analysis 52 (2008) 1712–1727. doi:10.1016/j.csda.2007.05.024.

[7] C. Agostinelli, A. Leung, V. J. Yohai, R. H. Zamar, Robust estimation of multivariate location and scatter in the presence of cellwise and casewise contamination, Test 24 (2015) 441–461. doi:10.1007/s11749-015-0450-6. arXiv:1406.6031.

[8] A. Ferrer, 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 (2007) 311–325. doi:10.1080/08982110701621304.

[9] G. E. P. Box, Some Theorems on Quadratic Forms Applied in the Study of Analysis of Variance Problems, I. Effect of Inequality of Variance in the One-Way Classification, Ann. Math. Statist. 25 (1954) 290–302. URL: https://projecteuclid.org:443/euclid.aoms/1177728786. doi:10.1214/aoms/1177728786.

[10] J. E. Jackson, G. S. Mudholkar, Control Procedures for Residuals Associated with Principal Component Analysis, Technometrics 21 (1979) 341–349. URL: http://www.jstor.org/stable/1267757. doi:10.2307/1267757.

[11] L. Eriksson, T. Byrne, E. Johansson, J. Trygg, C. Vikström, Multi- and Megavariate Data Analysis: Principles and Applications., 3rd ed., Umetrics Academy, 2001.

[12] P. Nomikos, J. F. MacGregor, Multivariate SPC Charts for Monitoring Batch Processes, Technometrics 37 (1995) 41–59. URL: https://www.tandfonline.com/doi/abs/10.1080/00401706.1995.10485888. doi:10.1080/00401706.1995.10485888.

[13] A. Folch-Fortuny, A. F. Villaverde, A. Ferrer, J. R. Banga, Enabling network inference methods to handle missing data and outliers, BMC Bioinformatics 16 (2015) 1–12. URL: http://dx.doi.org/10.1186/s12859-015-0717-7. doi:10.1186/s12859-015-0717-7.

[14] P. J. Rousseeuw, W. V. D. Bossche, Detecting Deviating Data Cells, Technometrics 60 (2018) 135–145. URL: https://www.tandfonline.com/doi/full/10.1080/00401706.2017.1340909. doi:10.1080/00401706.2017.1340909.

[15] M. Hubert, P. J. Rousseeuw, K. Vanden Branden, ROBPCA: A New Approach to Robust Principal Component Analysis, Technometrics (2005). URL: https://pdfs.semanticscholar.org/250b/4f05982b491ad80ba8b986d958eedb69a6be.pdf. doi:10.1198/004017004000000563.

[16] M. Hubert, P. J. Rousseeuw, T. Verdonck, A deterministic algorithm for robust location and scatter, Journal of Computational and Graphical Statistics 21 (2012) 618–637. doi:10.1080/10618600.2012.672100.

[17] N. Locantore, J. S. Marron, D. G. Simpson, N. Tripoli, J. T. Zhang, K. L. Cohen, Robust principal component analysis for functional data, Test 8 (1999) 1–73.

[18] F. Arteaga, A. Ferrer, How to simulate normal data sets with the desired correlation structure, Chemometrics and Intelligent Laboratory Systems 101 (2010) 38–42. URL: http://dx.doi.org/10.1016/j.chemolab.2009.12.003. doi:10.1016/j.chemolab.2009.12.003.

Appendix A. Comparison of the simulation method to other simulation strategies

This section contains information about how to replicate the strategies to simulate outliers present in different articles of PCA-MB dealing with outliers. In each case there are two main items to simulate: the reference data set used to fit the PCA model and the outlying observations.

The aim of this section is to provide a brief summary about the simulation strategies followed in each reference and to give the details about the set up of our proposed algorithm to imitate them, getting the results of Figure 11. The following table provides information about the method used in each referred work to simulate the reference data set and the outlying observations. Some notation has been changed from the original works to avoid potential confusions with other terms used in this paper.

Ref.	Simulation of reference data set	Simulation of outliers
1 - [5]	$\mathbf{X}_0 \sim N_n(\mathbf{0}_n, \mathbf{I}_n) \rightarrow \mathbf{X}_0 = \mathbf{T}_A\mathbf{P}_A^\top + \mathbf{E}_0$ $\mathbf{E}_1 \sim N(\mathbf{0}, \mathbf{1}) \cdot 0.1$ $\mathbf{X}_1 = \mathbf{T}_{1,A}\mathbf{P}_A^\top + \mathbf{E}_1$ $n = 98; k = 20; A = 4$	$\mathbf{X}_2 = \mathbf{T}_{2,A}\mathbf{P}_A^\top + \mathbf{E}_2$ $\mathbf{E}_2 \sim N(\mathbf{10}, \mathbf{1})$
2 - [6]	$\mathbf{T}_A \sim N_A\ (\mathbf{0}_A, \mathbf{I}_A)$ $\mathbf{P}_A :\perp k \times A$ uniformly distributed pseudorandom numbers $\mathbf{E}_k \sim N_k\ (\mathbf{0}_k, \mathbf{1}_k)/100$ $\mathbf{X}_1 = \mathbf{T}_A\ \mathbf{P}_A^\top + \mathbf{E}$	$\mathbf{X}_2 \sim N_A\ (\mathbf{15}_A, 8 * \mathbf{I}_A)$
3 - [13]	$\mathbf{X}_1$: Data reconstructing the metabolic network of thebenchmark problem 4 from the original work	$\mathbf{X}_2 : outliers$ $\forall i \in 1, \ldots, n_2$ $\forall j \in 1, \ldots k$ $m_j = mean(\mathbf{x}_j)$ $s_j = std(\mathbf{x}_j)$ if $\lvert x_{ij,1} \rvert \leq m_j + 1.5s_j$: $x_{ij,2} = -x_{ij,1}$
4 - [3]	$\mathbf{X}_1 \sim N(\mathbf{0}, \mathbf{\Sigma}_{A09})$ $A = 6PCs, N = 100, K = 200$ $\mathbf{\Sigma}_{A09} = \mathbf{V}_{A09}\mathbf{D}_{A09}\mathbf{V}_{A09}^\top$ $\mathbf{D}_{A09} = diag(30, 25, \ldots, 5, 0.098, 0.0975, \ldots, 0.005)$	$\mathbf{X}_2 \sim N(m\boldsymbol{\nu}_{A+1}, \mathbf{\Sigma}_{A09})$ $m \in 1, \ldots, 50$ $\boldsymbol{\nu}_{A+1} = \mathbf{V}_{A09}\ [:, A+1]$

Table A.1: Strategies followed by different authors to simulate the reference data sets and the outlying observations.

Appendix A.1. Reference 1

In [5], an adaptation of the classical Expectation Maximization PCA (EM-PCA)is provided. The core least squares PCA is substituted by the robust spherical PCA. [17] Their reference data set is a matrix of dimensions $n = 98$ and $k = 20$, with its variables following a multivariate normal distribution. After its reconstruction with A principal components, an error term following a $N(0, 1)$ distribution is added to regular observations. A certain percentage of observations is randomly sampled and instead, their added error term follows a $N(10, 1)$ distribution. Before its addition to the reconstructed matrix, the error term is multiplied by a 0.2 factor.

Following one of the simulation settings from the original work, we set a number of $A = 4$ PCs and selected 10% of the observations to transform them into outliers. After building a PCA on the clean data, the outliers simulated as in the original work, were projected onto the model, obtaining their SPE and T^2. These metrics were used as an input to our simulation function, setting them as target values for the algorithm.

The code used to generate the outliers is the following one:

```
pcamodel_ref1 = pcamb_classic(Xref1, 4, 0.05, 'cent');
pcaxout = pcame(Xoutref1, pcamodel_ref);
Xout1 = scout(Xoutref1_0, pcamodel_ref, 'simple', 'spey',
pcaxout.SPE, 't2y',pcaxout.T2);
```

The elements `Xref1, Xoutref1` and `Xoutref1_0` are matrices containing the reference data set used to build the PCA model, the outliers simulated as in the original work and the outlying observations before being transformed into outliers, respectively.

Appendix A.2. Reference 2

In [6], the simulation procedure begins by simulating the latent subspace. On one hand, scores are simulated as independent normally distributed variables with zero mean and unitary variance. On the other hand, loadings are simulated as orthogonal vectors with pseudo-random uniformly distributed pseudo-random numbers. In third place, the error matrix is simulated as normally distributed random noise divided by 100. Using this terms, the reference matrix ($\mathbf{X}_1$) can be reconstructed.

Afterwards, the outlying data set is simulated as a matrix $\mathbf{X}_2$, with $n_2 = 0.1 \cdot n_1$ observations, and which follows a normal distribution $N_A(\mathbf{15}_A, 8\mathbf{I}_A)$, where $\mathbf{15}_A$ is a vector containing A elements equal to 15.

Following this simulation procedure to generate outliers, authors create three different setups (C_1, C_2 and C_3) varying the number of observations ($n_{C_1} = 100, n_{C_2} = 40, n_{C_3} = 40$), the number of variables ($k_{C_1} = 5, k_{C_2} = 10, k_{C_3} = 200$) and maintaining the number of PCs ($A_{C_1} = A_{C_2} = A_{C_3} = 2$). Results shown in Figure 11b are the ones obtained with the configuration B.

The following lines of code were used to replicate the PCA model of the reference data set and the outliers simulated in this work:

```
pcamodel_ref2 = pcamb_classic(Xref2, 2, 0.05, 'cent');
pcaxout= pcame(Xoutref2, pcamodel_ref2);
Xout2 = scout(Xoutref2_0, pcamodel_refB, 'simple', 't2y',
 pcaxout.T2,'spey',pcaxout.T2);
```

The elements `Xref2, Xoutref2` and `Xoutref2_0` are matrices containing the reference data set used to build the PCA model, the outliers simulated as in the original work and the outlying observations before being transformed into outliers, respectively. In this case, observations from `Xoutref2_0` were observations following the same distribution as the reference data set.

Appendix A.3. Reference 3

In [13], authors provide a solution based on Trimmed Scores Regression (TSR) to enable network inference methods work in presence of missing data and outliers. Outlying observations are simulated by shifting the sign of cells above the variable average plus 1.5 times the standard deviation, or below the mean minus 1.5 times the standard deviation. Thus, the correlation pattern between variables is broken for these outliers. In order to illustrate the results, we show the outliers generated for the benchmark problem 4, one of the five benchmark problems addressed in the original paper.

The original work provides the information about the data used in the article. After downloading it, we built a PCA model based on the `Xref3` matrix, and projected. Afterwards, a random selection of rows determined the observations that were transformed to outliers using the original procedure in [13] and the algorithm proposed in this article.

The process to achieve this simulation framework is very similar to the ones from previous references, where the SPE and the T^2 of the outliers generated following the procedure from the original work are used as target values in our simulation function.

```
pcamodel_ref3 = pcamb_classic(Xref3, 3, 0.05, 'cent');
pcaxout = pcame(Xoutref3, pcamodel_ref3);
Xout3 = scout(Xoutref3_0, pcamodel_ref3, 'simple', 'spey',
pcaxout.SPE, 't2y',pcaxout.T2);
```

The elements `Xref3, Xoutref3` and `Xoutref3_0` are matrices containing the reference data set used to build the PCA model, the outliers simulated as in the original work and the outlying observations before being transformed into outliers, respectively.

Appendix A.4. Reference 4

In [3] authors propose an adaptation of their previous robust PCA algorithm to deal with missing data and cellwise outliers. However, in this work we are focusing exclusively in the comparison between rowwise outliers, i.e. anomalous observations. In this case the reference data set is generated as a matrix whose columns follow a multivariate normal distribution $N(\mathbf{0}, \mathbf{\Sigma})$. Two different covariance matrices (A09 and ALYZ) are used and their singular values are adapted in order to reach over the 80% of explained variance with the first 6 principal components.

Later on, a certain percentage of rows is randomly sampled and changed by new observations that follow the distribution $N(m\boldsymbol{\nu}_{A+1}, \mathbf{\Sigma})$. In the previous expression, A is the number of principal components and the term $\boldsymbol{\nu}_j$ refers to the jth eigenvector of the covariance matrix. The factor m that multiplies the new mean vector ranges from 1 to 50, leading an increasing noise introduced in the outliers along with the increase in the m parameter. This is equivalent to make outleirs more distant to the model hyperplane.

The following code lines show the procedure used to imitate the simulation with the A09 covariance matrix. The matrix dimensions are $n = 100$ observations and $k = 200$ variables, with $A = 6$ principal components, $m = 1, \cdots, 50$ and 20 rows randomly selected to transform them into outliers.

```
pcamodel_ref4 = pcamb_classic(Xref4, 6, 0.05, 'cent');
pcaxout4 = pcame(Xout4, pcamodel_ref4);
```

The matrix `Xout3` contains the outlying rows for all the values of the step parameter m, i.e. it is a matrix of 1020 rows (20×51, the 20 original observations and their progressive 50 shifts). Vectors $\mathbf{SPE}_y^2$ and $\mathbf{T}_y^2$ contain the SPE and T^2 values of the 20 outliers along the 50 steps.

A characteristic aspect of this simulation is the gradual shift described by the outliers. In terms of our proposed procedure, this is equivalent to use the step-wise generation of outliers as in Section 3.1.2. For this purpose, we need the final values of the statistics at the $m = 50$ step for all the observations, but also the step parameter γ. In order to study the progression for the SPE and the T^2 along the m steps, we plotted their evolution. in Figure A.13.

After visualising Figure A.13, it stands out a clear difference between the growing patterns of the SPE and the T^2 along m . Whereas the SPE trajectory for the outliers draws a clear ascending pattern for all the observations, the T^2 does not seem to do so.

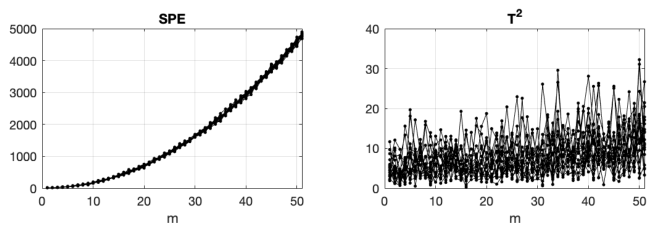


Figure A.13: SPE (left) and T^2 evolution of the outliers simulated in [3] along the m steps.

Moreover, when the outliers are visualised in the distance plot from Figure 11c, barely any of them is above the UCL for the T^2. This lead us to consider that changes in the T^2 among the outliers were more an artifact than a desired outcome of the simulation. Hence, we focused on calculating the γ_{SPE} parameter only.This parameter appears in Figure 3a and Equation 18 tunning the spacing between the SPE steps in the following expression.

In order to fit the γ_{SPE} parameter, we used the MatLab function `lsqnonlin.m`:

```
sperw = reshape(pcaoutrw4.SPE,20,51)';
xg_spe = nan(20,1);
for i = 1:20
    HM = sperw(end,i);
    H0 = sperw(1,i);
    Hm = sperw(:,i);
    M = 50;
    m = 0:50;
    gfun = @(gamma)H0 + (m./M).^gamma*(HM - H0) - Hm;
    xg_spe(i) = lsqnonlin(gfun,3);
end
g_spe_mean = mean(xg_spe);
```

After calculating the γ_{SPE} for each observation, the mean value is calculated and stored in the `g_spe_mean` variable. Figure A.14 shows the estimated trajectory using the average $\gamma_{SPE} = 2.6348$ value. This parameter used later as an input to the `scout.m` function:

```
Xout4 = scout(Xoutref4_0, pcamodel_ref4, 'steps', 'spey',
sperw(end,:)', 'nsteps', 50, 'gspe', g_spe_mean);
```

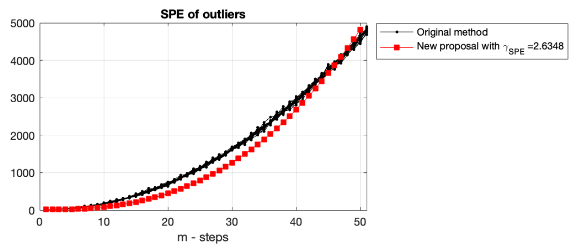


Figure A.14: SPE of outliers simulated by the strategy from the original work [3] (black) and SPE of the outliers simulated by the proposed algorithm (red) along the m steps.

Appendix B. Limitations of the proposed algorithm

This section addresses in further detail the results obtained with the method to simulate outliers with desired properties, when it is used on a matrix with non-linearities or with binary data.

The reference matrix $\mathbf{X}_0$ is simulated using the functions from [18]. The following code lines are the ones used to generate the reference matrix:

```
[X,S,srnd] = simdataset(100,10,[6,3],ones(1,10));
[X_0,srndn]=randnm(S,100,srnd);
```

The resulting matrix has 100 observations, 10 variables normally distributed and two principal components which explain above the 80% of the variance.

Appendix B.1. Non linearities

In this case, the matrix will present relations between variables that the classical PCA model will not be able to capture. In order to study to what extent this limitation of the PCA model would affect the simulations, we carried out a generation of outliers with a reference matrix that contained non-linearities and increasing only the T^2 of the outliers. This means that the generated observations should nor break the correlation pattern described by variables.

The new matrix $\mathbf{Y}$ is the result of concatenating the original matrix $\mathbf{X}_0$, and a set of non-linear variables generated from the original ones in $\mathbf{X}_0$. The non-linear relations included in each variable are:

```
rng(1101)
varind = randperm(10,8);
```

```
Y_11 = X_0(:,varind(1)).^2;
Y_12 = X_0(:,varind(2)).^3;
Y_13= exp(X_0(:,varind(3)));
Y_15 = rand(1,1) + X_0(:,varind(5)) + X_0(:,varind(5)).^2;
Y_16 = X_0(:,varind(2)).*X_0(:,varind(4));
Y_17 = X_0(:,varind(6)).*X_0(:,varind(7)).^2;
Y_18 =  exp(X_0(:,varind(3))).^(X_0(:,varind(7)) + X_0(:,varind(8)));
Y_19 = X_0(:,varind(3))*2;

Y = [X_0,Y_11,Y_12,Y_13,Y_14,Y_15,Y_16,Y_17,Y_18,Y_19];
```

As one can notice, the selection of the variables that were non-linearly combined was perform randomly. Also, a linearly generated variable ($\mathbf{y}_{19}$) was included in the set, to compare if the outliers on this variable still followed their analytic relation with the column. used to generate them.

As we aforementioned, some outliers on the T^2 were generated to keep the original correlation structure between variables. In order to do so, the PCA reference model based on $\mathbf{Y}$ had to be calculated. By setting "0" as the second input argument in the PCA-MB function, it returns a suggestion about the number of PCs to consider:

```
pcamodel_ref = pcamb_classic(Y, 0, 0.05, 'cent');

Sugested number of PCs:
 - Singular values of covariance matrix > 1 = 6
 - Minimum PCs to reach cummulative variance > 80 % = 3
Select the number of PCs: 3
```

A number of 3 PCs was selected. Then, outliers on the T^2 were generated setting the same target value for all of them in the *scout.m* function:

```
T2target = 60*ones(size(Y, 1), 1);
Yextreme = scout(Y, pcamodel_ref, 'simple', 't2y', T2target);
Yall = [Y; Yextreme.X];
```

The resulting outliers are represented in Figure, B.15 where it can be seen that the new observations accomplish the specified target values for the T^2.

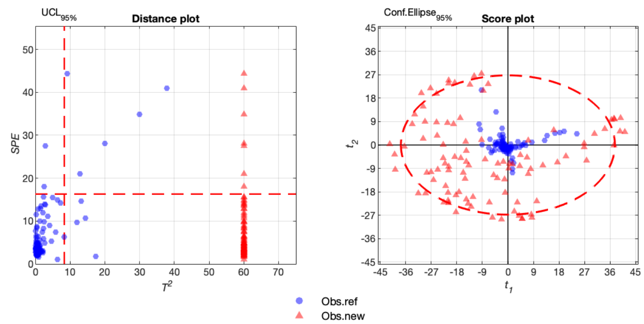


Figure B.15: Distance (left) and score (right) plot for the reference (blue circles) and the outliers (red triangles) generated.

However, the relations between the non-linear variables and the original columns use to generated have been distorted. In Figure B.16 there is a clear difference between blue and red observations.

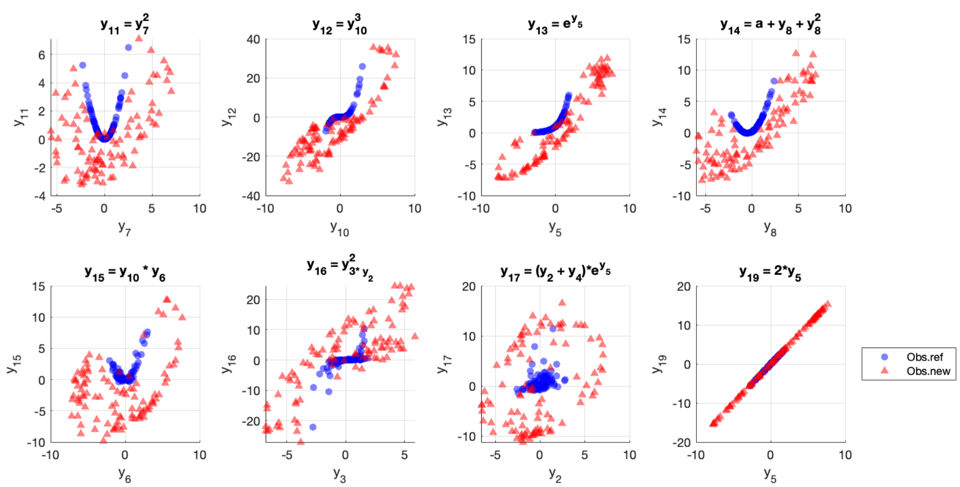


Figure B.16: Scatter plots with the reference (blue circles) and new (red triangles) observations for all the new variables in $\mathbf{Y}$ generated as combinations of the variables in $\mathbf{X}_0$.

Whereas the blue circles perfectly describe the analytical relation used to generate them, that is not the case for red triangles, since they clearly break the relative pattern between variables. This is not the case for the last variable ($\mathbf{x}_{19}$), which was generated as a linear combination. This result reinforces the limitation that is produced when the method has to take into account non-linear relations between the variables.

Appendix B.2. Binary variables

In this second example the purpose is to show the changes produced on categorical variables when the algorithm is used on a mixed matrix with continuous and categorical data.

In this case, four binary variables with different percentage of 0s and 1s are simulated. The resulting matrix $\mathbf{Y}$ has the original variables from $\mathbf{X}_0$ and the four additional binary columns.

```
rng(1101)
Y = [X_0,zeros(size(X_0,1),4)];
Y(randperm(size(X_0,1),round(0.2*size(X_0,1))),11) = 1;
Y(randperm(size(X_0,1),round(0.4*size(X_0,1))),12) = 1;
Y(randperm(size(X_0,1),round(0.6*size(X_0,1))),13) = 1;
Y(randperm(size(X_0,1),round(0.8*size(X_0,1))),14) = 1;
```

Similarly as in Appendix B.1, a PCA model is fitted with $\mathbf{Y}$, but in this case, two PCs were selected.

```
pcamodel_ref = pcamb_classic(Y, 0, 0.05, 'cent');
Sugested number of PCs:
 - Singular values of covariance matrix > 1 = 2
 - Minimum PCs to reach cummulative variance > 80 % = 2
```

In this case we generated outliers increasing the SPE and the T^2, imposing a target value of 50 for both of them and for all the data points. As it can be seen in Figure B.17, the set of new observations has the specified values for both statistics.

```
T2target = 50*ones(size(Ybin, 1), 1);
SPEtarget = 50*ones(size(Ybin, 1), 1);
Yout = scout(Ybin, pcamodel_ref, 'simple', 't2y', T2target,'spey',SPEtarget);
Yall = [Ybin; Yout.X];
```

Nonetheless, it is easy to see in Figure B.18 that new observations are outside the range of accepted values for binary variables. This artefact is produced because the simulation algorithm assumes to work with continuous variables. Consequently, it does not include any constraint in the data generation to respect the binary or qualitative nature of variables.

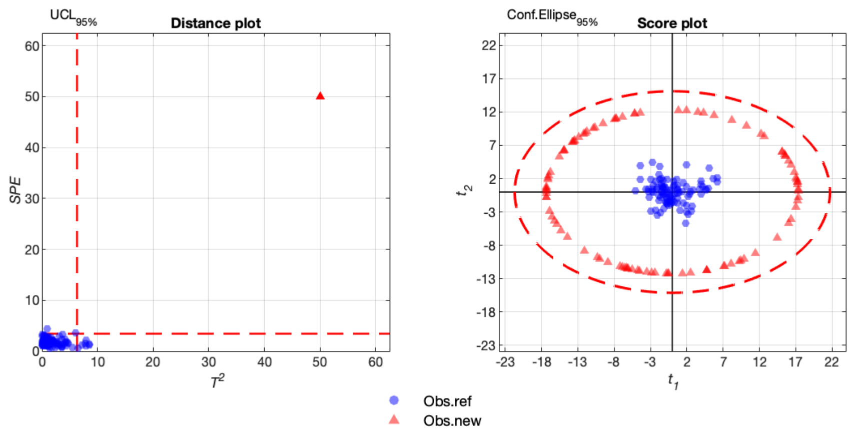


Figure B.17: Distance (left) and score (right) plot for the reference (blue circles) and the outliers (red triangles) generated.

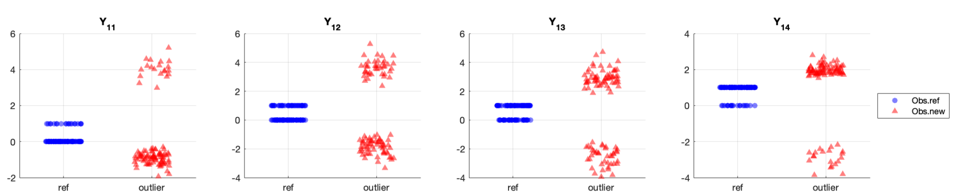


Figure B.18: Distance (left) and score (right) plot for the reference (blue circles) and the outliers (red triangles) generated.