Received: 10 November 2022 | Revised: 17 February 2023 | Accepted: 26 February 2023

DOI: 10.1002/qre.3311

CASE STUDY

WILEY

Unsupervised and supervised learning for the reliability analysis of complex systems

Maria Luz Gámiz[1] | **Fernando Navas-Gómez**[1] | **Rafael Nozal-Cañadas**[2] | **Rocío Raya-Miranda**[1]

[1]Department of Statistics and O.R., University of Granada, Granada, Spain

[2]Department of Computer Science, UiT—The Arctic University of Norway, Tromso, Norway

Correspondence
Maria Luz Gámiz
Email: mgamiz@ugr.es

Funding information
Spanish Ministry of Science and Innovation, Grant/Award Numbers: RTI2018-099723-B-I00, PID2020-120217RB-I00; Junta de Andalucía, Grant/Award Number: B-FQM-284-UGR20; IMAG-Maria de Maeztu, Grant/Award Number: CEX2020-001105-M/AEI/10.13039/501100011033

Abstract

In this paper, a strategy to deal with high-dimensional reliability systems with multiple correlated components is proposed. The goal is to construct a state function that enables the classification of the states of components in one of two categories, that is, failure and operative, in case of dealing with a large number of units in the system. To this end, it is proposed a new algorithm that combines a factor analysis algorithm (unsupervised learning) with local-logistic and isotonic regression (supervised learning). The reliability function is estimated and system failures are predicted in terms of the variables in the original state space. The dimensions in the latent state space are defined by blocks of units with a certain dependence structure. The flexibility of the model allows quantifying locally the effect that a particular unit has on the system performance and a ranking of components can be obtained under the philosophy of the Birnbaum importance measure. The good performance of the proposal is assessed by means of a simulation study. Also a real data case is considered to illustrate the method.

KEYWORDS

Birnbaum importance measure, dependent components, factor analysis, isotonic smoothing, logistic regression

1 | INTRODUCTION

Dependability Analysis is the most important task for reducing risk of failures and upgrading availability of manufacturing industries. Together with the increasing complexity of engineering systems, we have in recent years an explosion of the available amount of information related to condition monitoring. In maintenance of industrial systems, for decision making purposes, it is a key to master in techniques of classification of the state of performance of a machine based on datasets with a large number of features. Recently, several techniques based on soft computing have been demonstrated as a useful alternative for analyzing complex systems reliability. In[1] the authors discuss and review challenges in big data and traditional reliability analysis. The authors focus on how to use data with complicated structures to do reliability analysis. As stated in Hong et al.,[1] the increasing motivation to automate classification tasks leads to a reduced cost and time-consuming in the whole process of system maintenance that could derive from misclassification. That is one of the reasons of using artificial intelligence algorithms in reliability engineering. In Soltanali et al.,[2] soft computing and statistical techniques are used to provide a comparative structure for predicting the operational reliability in automotive manufacturing industry. In particular, the authors consider the adaptive neuro-fuzzy inference system (ANFIS) model for predicting operational reliability and safety of automotive manufacturing. However, in spite of the great contribution that soft computing techniques have added to the solution of this problem, there are still some drawbacks to overcome.

High-dimensional reliability analysis remains a grand challenge since most of the existing methods suffer from the curse of dimensionality. In this sense, Li and Wang[3] introduce a novel high-dimensional data abstraction (HDDA) framework for dimension reduction in reliability analysis. In their approach, the authors first formulate a strategy to reduce the dimensionality of the input space so creating a low-dimensional latent space for obtaining limit state function for failure prediction. Then they construct a feed-forward neural network to connect the high-dimensional input parameters with the low-dimensional latent variables. The high-dimensional reliability is estimated by capturing the limit state function in the latent space using Gaussian process regression. Focused on structural reliability in Yi and Du,[4] it is presented a new dimension reduction strategy based on the first-order reliability method (FORM) so that the contributions of unimportant input variables are also accommodated after dimension reduction.

From a different point of view, Cai et al.[5] and Si et al.[6] state that the traditional reliability analysis methods for a binary system are often oversimplified and insufficient for describing complex systems. In a multistate system, the components and system have more than two states. There are numerous examples in practical engineering, such as nuclear reactor systems, digital protection systems, oil transportation systems, and so forth. We will refer to continuous system when the range of performance of components and systems is an interval, which we will assume to be [0,1], for simplicity. Levitin and Lisnianski[7] explore an approximation based on the technique of the universal generating function to analyze multistate systems. Brunelle and Kapur[8] proposed a multivariate interpolation method to construct the structure function of a continuous system.

In addition to this, several proposals aimed to construct the structure function of a multistate system based on empirical data have arisen in the recent literature. In this scenario, uncertainty in the model has to be considered. Thus, system structure modeling has been generalized in the last years by introducing some sort of uncertainty in the model (see Ref. 9, 10 or 11 for different approaches). In Gámiz-Pérez and Martínez-Miranda,[11] a new methodology based on nonparametric statistics is presented to build the structure function of the system from empirical data. Uncertainty is modeled through a noise term thus presenting the structure function as a stochastic model rather than a deterministic model, which is the classical approach. Following these ideas, Gámiz et al.[12] propose a supervised learning algorithm that, among other things, allows to quantify the effect that each component has on the performance of the system. To do it, the concept of the Birnbaum importance measure is generalized and a new version of this measure is defined and is used to rank the components in the system in terms of the importance. It is worth to highlight that the new measure has a local character in the sense that it gives information on the impact that a unit change in the components state has on the system performance for different regions of the vector state domain.

Component importance measures are relevant to improve the system design and to develop optimal replacement policies, see, for example, Si et al.[6] and Miziula and Navarro.[13] One of the most popular measures is the Birnbaum's importance measure. In fact, as pointed out in Si et al.,[6] traditional importance measures may not be effective to evaluate the contribution of an individual component if its reliability value is measured such that it does not fall in the full range between 0 and 1. Moreover, if the components are (stochastically) independent, the Birnbaum measure can be defined using several equivalent expressions. However it turns out that in the case of dependent components, different Birnbaum measure definitions lead to different concepts. In Miziula and Navarro,[13] the authors extend this measure to the case of dependent components based on the contribution of the component to the system reliability.

In this paper, we propose a new strategy to deal with complex systems in the context of structural reliability. Our main purpose is to construct a limit state function in order to divide the space of states of components into the failure and safe regions when dealing with a large number of units in the system. To this end, we propose the use of unsupervised learning for dimensionality reduction. In concrete, we carry out a factor analysis to create a latent state space where classification techniques based on isotonic regression and local-logistic regression (supervised learning) are applied to obtain the reliability function.

When the number of units in the system is high certain correlation structure is expected to underlie the state space. The purpose of the factor analysis is to uncover such structure. Each variable of the latent space is defined from a subset of correlated components to which we will refer as *blocks*. Different blocks are independent. A back transformation allows to predict system failures in terms of the variables in the original state space.

The main purpose is to predict the probability that the system provides an acceptable level of performance as well as to evaluate the effect of every single component on the system behavior. To do it, Gámiz et a.[12] propose a machine learning procedure based on logistic regression, isotonic regression, and cross-validation.

Machine learning has recently demonstrated huge potential for industrial reliability analysis (see Alsina et al.[14] and Afshari et al.,[15] and references therein). Logistic regression is a probabilistic model considered as one of the most successful supervised algorithms of machine learning for classification problems (see among others Chen et al.[16] and Phillips et al.[17]). In Phillips et al.,[17] it is brought out that typically "black box" approaches such as Artificial Neural Networks (ANN) and Support Vector Machines (SVM) can be difficult to provide ease of interpretability. In contrast, the authors in Ref. 17 argue that logistic regression offers easy interpretability to industry experts, providing insight to the classification process and preventing potential misclassification. A comparative study is provided in Phillips et al.[17] based on predictive performance of logistic regression, ANN, and SVM through a real data set from engines on mining trucks, and it is shown that logistic regression outperforms the ANN and SVM approaches at predicting the true state of the engines. Decision trees and their generalization as Random Forest (RF) are also intuitive and easy to implement but do not provide a probability for the outcome as a direct result. However, when further inferences about the model are needed, as is the case in this paper, logistic regression is the recommended method.[1,18]

This paper supposes a natural extension of the paper[12] for more complex systems with high dimension in the sense of number of components, which are arranged in blocks of (possibly) correlated units. Our main goal is to build a statistical model able to predict system reliability from data. The contribution of this paper can be summarized as follows.

1. The reliability model is built under the assumption of dependence of components. This is a very important issue. However, for simplicity, a very common assumption is that the units of the system are statistically independent, which might not be realistic in many practical situations. In this paper, we assume that the units in the system can be correlated.
2. We propose a new algorithm where we introduce a dimension reduction step in order to create a convenient state space (latent space) to model fitting. One of the difficulties that arise here has to do with model selection, that is, our procedure requires selecting the adequate number of features of the latent space. We discuss this point in the simulations.
3. Unlike the work in Gámiz et al.,[12] the local logistic model is built on a latent space instead of taking the states of the components of the system as input variables. One of the consequences is that the issue of the system coherence must be addressed very differently.
4. The asymptotic properties of the new estimator are obtained, in particular an expression of the variance of the estimator is presented.
5. The importance measure obtained in this paper takes into account the correlation structure between the components of the system.

The structure of the paper is as follows. In Section 2, the different statistical tools that are required in our algorithm are presented in a general context. In particular, the basics on factorial analysis and nonparametric regression are briefly recalled. In Section 3, we specifically describe in details all the steps of the new algorithm aimed at building the reliability function of complex system with a large number of components that are not assumed to be statistically independent. In Section 4, an extensive simulation study is presented to evaluate the finite sample performance of the method. A real dataset is analyzed in Section 5. Finally, Section 6 concludes this article.

2 | METHODOLOGY

A mathematical representation of the logic of a system is one of the main objectives pursued in reliability analysis. Since system performance assessment can be a complex problem in practice, even for simple structures, it seems reasonable that a skillful procedure for modeling the relationship between the state of the system and its components can help efficiently in complex systems classification problems. Thus the system structure function, that is, the link function between the state of the system and all its components, has been a major topic in the field of system reliability.

In this paper, we follow the work of Gámiz et al.[12] and extend the developments therein to analyze the case of complex multicomponent systems, with a large number of units that are not assumed to be statistically independent, which is a more realistic assumption specially when the number of components in the system is high.

2.1 | Factor analysis (FA) by means of principal components analysis (PCA)

With the explosion of information, it is very common to have many features in a dataset at hand. In our reliability analysis context, it means that we may deal with complex multi-component systems with a large number of units. It is not feasible to run the algorithm to fit the structure function developed in [12] on all the features as it will reduce the performance of the algorithm and it will not be easy to visualize many features in any kind of graph. So, one solution is to reduce the number of features in the dataset using for instance a factor analysis (FA) algorithm that creates factors from the observed variables to represent the common variance i.e. variance due to correlation among the observed variables.

The method that we will use for factor analysis is based on principal components analysis that was developed by H. Hotelling and we summarize in the following.

Principal component analysis (PCA) is a linear projection method giving as an output a sequence of nested linear subspaces, which are adapted to the data at hand. In Machine learning, PCA is one of the most used unsupervised algorithms. It is a widely used preprocessing method with diverse applications, ranging from dimensionality reduction to denoising.[19] The method performs an eigen-decomposition of the empirical covariance matrix and considers the space generated by the eigenvectors corresponding to the leading eigenvalues.

Now we briefly recall one solution to PCA using linear algebra (see Refs. 20–22). Let $\mathbf{X}$ be an $n \times p$ matrix of data, where p is the number of features (covariates) and n is the number of samples.

The data must be centered with 0 mean and standard deviation 1. The goal is to find some orthonormal matrix $\mathbf{\Gamma}$, that is $\mathbf{\Gamma}^t\mathbf{\Gamma} = \mathbf{I}$, such that the following linear representation is achieved:

$$\mathbf{Z}_{n\times p_0} = \mathbf{X}_{n\times p}\mathbf{\Gamma}_{p\times p_0} + \epsilon,$$

where the matrix $\mathbf{Z}_{n\times p_0}$ is called the *score* matrix and contains all the information of the sample. The matrix $\mathbf{\Gamma}$ is called *loadings*-matrix and gives a measure of the strength of the relation between each variable in $\mathbf{X}$ and $\mathbf{Z}$. The columns of $\mathbf{\Gamma}$ are the principal components of the data. Since a lower number of principal components (p_0) than number of variables (p) are kept in the model, that is $p_0 < p$, there is an error term that is accumulated in the matrix ϵ.

As already mentioned, PCA is based on the decomposition in eigenvectors and eigenvalues of the covariance matrix of the data $\mathbf{\Sigma} = \frac{1}{n-1}\mathbf{X}^t\mathbf{X}$. Then, we can write $\Sigma\Gamma_j = \lambda_j\Gamma_j$, $\sum_{j=1}^{p_0}\lambda_j = 1$, with λ_j denoting the eigenvalue related to the eigenvector $\mathbf{\Gamma}_j$. Finally, $\mathbf{Z}_j = \mathbf{X}\mathbf{\Gamma}_j$, for $j = 1, \dots, p_0$.

This equation is understood as follows: $\mathbf{Z}_j$ are the projections of $\mathbf{X}$ in the space of eigenvectors $\{\mathbf{\Gamma}_j, j = 1, \dots, p_0\}$, with the eigenvalues λ_j measuring the amount of variance explained, that is, the information kept by each of the principal components, which are sorted in descendant order according to this amount of information captured. See Jackson[20] for details on principal components analysis.

2.2 | The transformed model

The state of the jth component of the system is a random variable X_j $(j = 1, 2, \dots, p)$, which is directly observable and that takes values in the interval [0,1]. Let then $\mathbf{X} = (X_1, X_2, \dots, X_p)$ denote the input vector of variables that describes the internal development of the system. As in Gámiz et al.,[12] the vector $\mathbf{X}$ gives the states of all components in the system and the variable Y takes value 1 when the system is operative and 0 otherwise.

The main objective is to built a model able to predict the probability that the system works known the vector of levels of performance of its components. When the number of inputs in the regression model is very big, certain problems may arise, such as overfitting and an increase of the prediction error (see Ref. 23). So in this paper, we propose to reduce the number of features before building the regression model.

After running the FA algorithm, we get a (reduced) set of variables $Z_1, \dots, Z_{p_0}$, which are linearly related to the original ones. Each variable Z_j could be identified with a (latent) block made of correlated variables of the original state space.

We can write the reliability of the system following the reasoning in Gámiz et al.[12] as

$$R(\mathbf{x}) = P(Y = 1 \mid \mathbf{X} = \mathbf{x}),$$

which, using $\mathbf{Z} = \mathbf{X\Gamma}$, can be written

$$\widetilde{R}(\mathbf{z}) = P(Y = 1 \mid \mathbf{Z} = \mathbf{z}),$$

where we denote $\mathbf{z} = \mathbf{x\Gamma}$.

In other words, given $\mathbf{Z} = \mathbf{z}$ the random variable Y follows a binomial distribution with parameter $\widetilde{R}(\mathbf{z})$. Notice that $\widetilde{R}(\cdot)$ is not the reliability function of the system, instead, the reliability function R gives the probability that the system is operative as a function of the states of its components, that is, given $\mathbf{X} = \mathbf{x}$. Then, to estimate the reliability function, we propose the following. First, fit the local-logistic model based on the latent features $(Z_1, \dots, Z_{p_0})$, that is, obtain an estimator of the function $\widetilde{R}(\mathbf{z})$. Second, define $R = \widetilde{R} \circ \varphi$, where $\varphi : \mathbb{R}^p \to \mathbb{R}^{p_0}$, is a linear function that transforms the original variables to the latent features space provided by the FA algorithm.

2.3 | Local-logistic regression

We fit a local-logistic model in the space of the factors $(Z_1, \dots, Z_{p_0})$ as we explain briefly in the following (see Gámiz et al.[12] for details). Let us define

$$g(\mathbf{z}) = \log \frac{\widetilde{R}(\mathbf{z})}{1 - \widetilde{R}(\mathbf{z})}, \tag{1}$$

where $\mathbf{z}^t = (z_1, z_2, \dots, z_{p_0})$ and g is a smooth function in the sense of derivability.

Defining $b_0 = g(\mathbf{z}_0)$, y $b_j = \frac{\partial g(\mathbf{z}_0)}{\partial z_j}$, $j = 1, \dots, p_0$, it can be written

$$g(\mathbf{z}) \approx b_0 + b_1(z_1 - z_{01}) + \cdots + b_{p_0}\left(z_{p_0} - z_{0p_0}\right), \tag{2}$$

for $\mathbf{z}$ in a neighborhood of $\mathbf{z}_0$ conveniently chosen, that is for all $\mathbf{z}$ such that $\|\mathbf{z} - \mathbf{z}_0\| < h$, with $\|\cdot\|$ the Euclidean norm and $h > 0$ sufficiently small.

Then, we can approximate the function $\widetilde{R}$ by

$$\widetilde{R}_0(\mathbf{z}) = \frac{e^{(1,(\mathbf{z}-\mathbf{z}_0)^t)\mathbf{b}}}{1 + e^{(1,(\mathbf{z}-\mathbf{z}_0)^t)\mathbf{b}}}, \tag{3}$$

for all $\mathbf{z}$ such that $\|\mathbf{z} - \mathbf{z}_0\| < h$ and $h > 0$ small enough, and $\mathbf{b} = (b_0, b_1, \dots, b_{p_0})^t$.

The vector of parameters $\mathbf{b}$ can be estimated locally by maximum-likelihood techniques. In other words, based on a dataset $\{(\mathbf{z}_1, Y_1), \dots, (\mathbf{z}_n, Y_n)\}$, the value of $\mathbf{b}$ maximizes the local likelihood function given by

$$\ell_0(\mathbf{b}) = \sum_{i=1}^{n} w_{h,0}(\mathbf{z}_i)\Big(Y_i(\mathbf{z}_i - \mathbf{z}_0)^t\mathbf{b}\Big) - \log\Big(1 + \exp((\mathbf{z}_i - \mathbf{z}_0)^t\mathbf{b})\Big),$$

where $\{w_{h,0}(\mathbf{z}_i), i = 1, \dots, n\}$ determine the weights of the observations around the estimation point $\mathbf{z}_0$. This is to ensure that the linear approximation of g given in Equation (2) is considered only into the window where it is valid. We define

$$w_{h,0}(\mathbf{z}_i) = K_h(\|\mathbf{z}_i - \mathbf{z}_0\|),$$

with $K_h(\cdot) = (1/h)K(\cdot/h)$, being h the bandwidth parameter that controls the amount of smoothing, that is, it determines the size of the window around the point $\mathbf{z}_0$ where the local linear approximation is valid. Usually, K is a density function with a compact support.

Solving the problem of nonlinear equations, we can obtain a local estimate of the function g, that is, the estimated value of b_0 provides an estimate of $g(\mathbf{x}_0)$ and we also obtain estimations of the corresponding partial derivatives.

Given the relationship between $\mathbf{Z}$ and $\mathbf{X}$ provided by the FA, the model in Equation (3) involves a local-logistic model in the space of the original variables. This point will be detailed in Section 3.1, where the algorithm is fully explained.

The asymptotic properties of the local estimator are studied in the Appendix of Gámiz et al.[12]

2.4 | Isotonization

The local regression model is not built directly on the original data but using the scores-matrix provided by the FA algorithm. Nonparametric regression produces estimators with good asymptotic properties such as consistency and normal distribution that are reached at a reasonable convergence rate. However, these estimators are not necessarily monotone. In this context of coherent system, the response variable of the regression model (the system state) has to be monotone in each original variable (component state). In principle and in general, the features that we use to feed the local-logistic model do not have a clear physical interpretation in our problem, then we can not assume that the response is monotone as a function of them. Therefore, we propose the isotonization step to be implemented once the back transformation process is fulfilled and the model is expressed in terms of the original variables.

The isotonization procedure that we use in this paper is the **hinge** algorithm presented in Meyer,[24] which is simple, fast, and intuitive (see Meyer[24]). As mentioned above, we will apply this algorithm once the local-logistic model has been fitted to the data, that is, we run the isotonization algorithm with $v_i = \widehat{R}_i^*$, where for $i = 1, 2, \ldots, n$, $\widehat{R}_i^*$ is the estimated value of $R(\mathbf{x}_i) = P(Y = 1|\mathbf{x}_i)$ by the local-logistic model expressed in terms of the original variables after running the corresponding back transformation from the space of factors as explained in Section 2.3. Notice that each factor is identified with a block in the system design. Then, the value $w_i = R_i$, for $i = 1, 2, \ldots, n$, is an estimate reliability of the system at the profile of components determined by $\mathbf{x}_i$ that meets the coherent conditions of the system.

3 | RELIABILITY ANALYSIS OF COMPLEX SYSTEMS

Let $\{\mathbf{X}, Y\}$ be the observed data where $\mathbf{X}$ is a matrix of dimension $n \times p$. Each input of the matrix for $i = 1, \ldots, n$, is a configuration of components states of a system of dimension p. The state of a system is given by a random variable that is assumed to range in the interval [0,1]. The components of the system are not necessarily independent. Y is a vector of dimension n. For $i = 1, 2, \ldots, n$, $Y_i = 1$ if the system is operative and 0 otherwise.

3.1 | The FA–LR–IS algorithm

1. Center the columns of matrix $\mathbf{X}$ that is, for each $j = 1, 2, \ldots, p$, define $U_j = (X_j - \mu_j)/\sigma_j$, with $\mu_j = \frac{1}{n}\sum_{i=1}^{n} X_{ij}$ and $\sigma_j = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(X_{ij} - \mu_j)^2}$, for $j = 1, \ldots, p$. Let $\boldsymbol{\Sigma}_0$ a diagonal matrix with jth element σ_j, and $\mathbf{M} = \mathbf{1}_p(\mu_1, \mu_2, \ldots, \mu_p)$, with $\mathbf{1}_p$ a unitary column-vector of size p. Let us denote $\mathbf{U} = (\mathbf{X} - \mathbf{M})\boldsymbol{\Sigma}_0^{-1}$, then $\mathbf{U}$ is a matrix with dimension $n \times p$.
2. Apply an algorithm for FA to the scaled dataset $\mathbf{U}$ and transform the data to a reduced set denoted $\{\mathbf{Z}, Y\}_{n\times p_0}$, with $p_0 < p$, and $\mathbf{Z} = \mathbf{U}\boldsymbol{\Gamma}$, being $\boldsymbol{\Gamma}$ the FA coefficient-matrix of dimension $p \times p_0$ (see Section 2.1).
3. Fit a local-logistic model based on the reduced dataset $\{\mathbf{Z}, Y\}$ (see Section 2.3) using the leave-one-out-cross-validation (LOOCV) criterion for bandwidth selection.

   LOOCV:

   3.1. Let h a bandwidth varying in a grid $\{h_1, h_2, \ldots, h_m\}$.

   3.2. Build the local-logistic model based on $\{\mathbf{Z}, Y\}$ and estimate $\widetilde{R}_h(\mathbf{z}_i)$ for all $i = 1, 2, \ldots, n$.

   3.3. For $i = 1, 2, \ldots, n$, let the leave-one-out (*loo*) dataset $\{\mathbf{Z}, Y\}^{(-i)}$, the dataset without the ith input. Let $\widetilde{R}_h^{(-i)}(\mathbf{z})$ the fitted model using the *loo* dataset and estimate $\widetilde{R}_h^{(-i)}(\mathbf{z}_i)$ for all $i = 1, 2, \ldots, n$.

   3.4. Define the cross-validation score as

$$Q(h) = \sum_{i=1}^{n} \widetilde{R}_h(\mathbf{z}_i)^2 - 2\sum_{i=1}^{n} Y_i \widetilde{R}_h^{(-i)}(\mathbf{z}).$$

   3.5. Define $h_{CV} = \arg\min_h Q(h)$.

4. Let $\mathbf{x}_0$ a particular configuration of states of the components of the system. Let $\mathbf{z}_0 = (\mathbf{x}_0 - (\mu_1, \mu_2, \dots, \mu_p))\boldsymbol{\Sigma}_0^{-1}\boldsymbol{\Gamma}$, and let us build the local-logistic model as above. Then, for $\mathbf{z}$ such that $\|\mathbf{z} - \mathbf{z}_0\| < h_{CV}$, the fitted model is given by

$$\widetilde{R}_{CV}(\mathbf{z}) = \frac{e^{(1,(\mathbf{z}-\mathbf{z}_0)^t)\widehat{\mathbf{b}}}}{1 + e^{(1,(\mathbf{z}-\mathbf{z}_0)^t)\widehat{\mathbf{b}}}},$$

   with $\widehat{\mathbf{b}} = (\widehat{b}_0, \widehat{b}_1, \dots, \widehat{b}_{p_0})^t$ vector of coefficients obtained using h_{CV}.
5. Let $\mathbf{x}$ such that $\|(\mathbf{x} - \mathbf{x}_0)\boldsymbol{\Sigma}_0^{-1}\boldsymbol{\Gamma}\| < h_{CV}$, then the local-logistic model in the state space of components is estimated by

$$\widehat{R}^*_{CV}(\mathbf{x}) = \frac{e^{(1,(\mathbf{x}-\mathbf{x}_0)^t)\widehat{\boldsymbol{\beta}}}}{1 + e^{(1,(\mathbf{x}-\mathbf{x}_0)^t)\widehat{\boldsymbol{\beta}}}},$$

   where the vector of estimated coefficients $\widehat{\boldsymbol{\beta}}$ is

$$\widehat{\beta}_0 = \widehat{b}_0,$$
$$\widehat{\beta}_j = \sigma_j^{-1}\boldsymbol{\Gamma}_{j.}\widehat{\mathbf{b}}_{-0}, \quad j = 1, 2, \dots, p; \tag{4}$$

   where we denote $\boldsymbol{\Gamma}_{j.}$ the jth row of matrix $\boldsymbol{\Gamma}$, and $\widehat{\mathbf{b}}_{-0} = (b_1, \dots, b_{p_0})^t$.
6. Define $\widehat{R}^*_{CV}(\mathbf{x}_i) = \frac{e^{b_0}}{1+e^{b_0}}$, for $i = 1, 2, \dots, n$. Use the algorithm described in Section 2.4 to isotonize these estimated values and then obtain the estimated reliability for the ith configuration of states of components, that is $\widehat{R}_{CV}(\mathbf{x}_i)$, $i = 1, 2, \dots, n$.

3.2 | Importance measure of components

For $j = 1, \dots, p$, we aim at measuring the effect that component jth has on the system performance from the point of view of the Birnbaum measure as defined in Gámiz et al.,[12] which we recall for clarity.

Let $\Phi(X_1, \dots, X_p)$ the structure function of a coherent system of size p. Let us assume that $Y = \Phi(X_1, \dots, X_p) + \epsilon$, the state of the system, where ϵ represents noise. Let $R(x_1, \dots, x_p) = E[Y \mid X_1 = x_1, \dots, X_p = x_p)$, the probability that the system is operative when $X_1 = x_1, \dots, X_p = x_p$. For each $j = 1, 2, \dots, p$, the importance of component jth in the sense of Birnbaum is defined as $I^B(j) = \frac{\partial R(x_1,\dots,x_p)}{\partial x_j}$.

Following the conclusions in [12], the importance of a component with respect to the good performance of the system can be established according to the magnitude of the corresponding partial derivative of the *logit* function, i.e. the value of the corresponding β-coefficient. Given that in expression

$$\left.\frac{\partial \widehat{R}_{CV}(\mathbf{x})}{\partial x_j}\right|_{\mathbf{x}_i} = (\widehat{R}_{CV}(\mathbf{x}_i))(1 - \widehat{R}_{CV}(\mathbf{x}_i))\widehat{\beta}_j(\mathbf{x}_i). \tag{5}$$

the factor $\widehat{V}(Y_i) = \widehat{R}(\mathbf{x}_i)(1 - \widehat{R}(\mathbf{x}_i))$ is a common factor, then $\widehat{I}^B(j) = \widehat{\beta}_j$, for $j = 1, \dots, p$. The importance of a particular component is estimated locally.

3.3 | Inference about the model

In this section we address two important questions. On the one hand, we aim at quantifying the accuracy of the fitted model. In this sense we propose to build point confidence intervals around the reliability function. To do it, we first obtain asymptotic distribution of the local-maximum estimator that we have obtained in the previous sections. On the other hand, we can evaluate and compare different units in the system in terms of the impact they have on the system performance. To do it we propose to formulate the corresponding testing problem and solve it using Montecarlo techniques.

3.3.1 | Asymptotic confidence interval for the reliability function

To build confidence intervals, we first obtain the asymptotic distribution of $\widehat{\boldsymbol{\beta}}$.

Local-likelihood estimates are asymptotically normally distributed, see for example.[25] Then, using the results in Gámiz et al.,[12] we have that $\widehat{\mathbf{b}} \to N(0, \boldsymbol{\Sigma}_{\mathbf{b}})$, with variance

$$\boldsymbol{\Sigma}_{\mathbf{b}} = \left(\mathbf{Z}^t\mathbf{WVZ}\right)^{-1}\mathbf{Z}^t\mathbf{WVWZ}\left(\mathbf{Z}^t\mathbf{WVZ}\right)^{-1},$$

where $\mathbf{W}$ is a diagonal matrix whose ith element is $W(i,i) = K_h(\|\mathbf{z}_i - \mathbf{z}_0\|)$, for $i = 1, \dots, n$; and, $\mathbf{V}$ is a diagonal matrix whose ith element is $V(i,i) = R_0(\mathbf{z}_i)(1 - R_0(\mathbf{z}_i))$, for $i = 1, \dots, n$.

Then we deduce that $\boldsymbol{\beta} \to N(0, \boldsymbol{\Sigma}_{\boldsymbol{\beta}})$, with

$$\boldsymbol{\Sigma}_{\boldsymbol{\beta}} = \boldsymbol{\Sigma}_0^{-1}\boldsymbol{\Gamma}\boldsymbol{\Sigma}_{\mathbf{b}}\boldsymbol{\Gamma}^t\boldsymbol{\Sigma}_0^{-1}, \tag{6}$$

where $\boldsymbol{\Sigma}_0^{-1}$ and $\boldsymbol{\Gamma}$ have been defined previously.

With this we can build confidence intervals around $R(\mathbf{x}_0)$. Let $\mathbf{x}_0 \in \mathbb{R}^p$ a particular and fixed configuration of states of the components of the system. Using the theoretical properties of the local-likelihood estimator defined in Section 2.3 and given that $\widehat{\beta}_0 = \widehat{b}_0$ as deduced in Equation (4), a confidence level $(1-\alpha)100\%$ for the reliability of the function at the profile of components $\mathbf{x}_0$ is obtained from the confidence interval obtained for β_0, that is

$$\widehat{\beta}_0(\mathbf{x}_0) \pm q_{\frac{\alpha}{2}}\widehat{\sigma}_{\beta_0}(\mathbf{x}_0), \tag{7}$$

with $\widehat{\sigma}_{\beta_0}$ the (1,1)-element of matrix $\boldsymbol{\Sigma}_{\beta}$, and $q_{\frac{\alpha}{2}}$ denoting the quantile of order $1 - \frac{\alpha}{2}$ of the standard Normal law. Then, the confidence interval for $R(x_0)$ is obtained applying the inverse *logit* transformation $f(u) = 1/(1 + e^{-u})$ to the extremes of the interval (7).

3.3.2 | Hypothesis testing for components comparison

Let us test the null hypothesis:

$$H_0 : \beta_j(\mathbf{x}_0) = \beta_k(\mathbf{x}_0)$$

$$H_1 : \beta_j(\mathbf{x}_0) \neq \beta_k(\mathbf{x}_0),$$

for a particular vector of components' states, $\mathbf{x}_0 = (x_{01}, x_{02}, \dots, x_{0p})^t$.

To solve the above testing problem, we consider the statistic $T_{jk} = \mathbf{E}_{jk}\widehat{\boldsymbol{\beta}}$, with $\mathbf{E}_{jk}$ vector of 0s, except for positions j and k, which are, respectively, 1 and −1. From Equation (6), we obtain that the variance of this statistics is $Var(T_{jk}) = \mathbf{E}_{jk}Var(\widehat{\boldsymbol{\beta}})\mathbf{E}_{jk}^t$.

4 | SIMULATIONS

To illustrate our methodology, we use in this section simulated systems that, among other things, allow us to evaluate the goodness of fit of the estimators proposed in the previous sections.

4.1 | Reliability block diagrams

To generate data from reliability systems, we assume that the logic of the system structure can be represented by means of a block diagram. Static or dynamical systems consisting of multiple components that can be partitioned into distinct and interacting blocks arise in many scientific areas,[26,27] for example. Recently, in Catelani et al.,[28] the authors introduce a new approach to assess system reliability prediction in presence of redundant and stand-by architecture, to do it they develop a software tool based on block-diagrams, *RBDesigner*, a project for engineers to be able to achieve a reliability prediction of very complex systems in the early stages of product development.

A reliability block diagram (RBD) is a hierarchical model aimed at modeling the failure relationships of complex systems and their subcomponents (or blocks) and is extensively used for system reliability, availability, and maintainability analysis.[26]

An RBD is not a physical design diagram but a logical diagram illustrating what is required for the system to work. The RBD has a single starting point and a single ending point, and is made up of functional blocks connected by lines.

The constructions of an RBD can follow any of the basic patterns of component connections series or parallel. For binary systems, that is, when only two levels of performance are considered for describing the system state and the component's states, perfect functioning (1) and failure (0), in the series connection, all components have to be operative for the system to remain functional. Whereas, in a parallel structure at least one of the components must be functioning for the system to perform well. In this paper, we consider that the state of each component is a random variable ranging in the interval [0,1], where, for simplicity, we assume that 0 represents the failure state and 1 gives the full operational mode.

Using this approach, we extend the binary conception of RBDs as follows. For a structure with components connected in series, the structure function Φ returns the minimum value of the states of its components, and for a parallel structure, it returns the maximum value of the states of the components. Then, the state of the system is a binary random variable

$$Y = \begin{cases} 1, & \Phi(x_1, \dots, x_p) \geq \phi_0 \\ 0, & \Phi(x_1, \dots, x_p) < \phi_0 \end{cases}$$

with ϕ_0 a prespecified threshold. The region of the components state-space, that is, $\mathcal{X} = [0, 1]^p$, where $Y = 1$ is called the *safe* region. As explained before in this paper we assume that Y also depends on a random term ϵ.

In many cases, real-world systems involve subsystems or blocks, which themselves form a nested RBD configuration. Such systems can be modeled by nested RBD configurations. For instance, if a system and its components both are modeled by the series–parallel RBDs, then the complete system can be modeled by using a nested series–parallel RBD configurations. This makes RBDs a powerful tool that allows to easily construct the reliability models of many real-world systems and in particular highly complex engineering structures.[26,27]

4.2 | Numerical examples

4.2.1 | Finite sample properties of the estimator

As already mentioned, to evaluate our procedure, we run a simulation study where the systems considered are based on different RBD configurations. Figure 1 gives a graphical representation of the RBDs corresponding to the three cases analyzed in this section.

The data for each case have been generated as follows. We have simulated samples of $n = 50, 100, 500$. Let p denote the size of the system, then $p = 9, 10, 15$, respectively, in our examples. The data consist of a matrix with $p + 1$ columns. The first p columns report the state of the components $X_1, X_2, \dots, X_p$, while the $p + 1$ column refers to the state of the system Y, which takes value 1 if the system is working and 0 otherwise. Any two components in the same block have a correlation of 0.9, while components in different blocks are independent. This point is considered in Figure 1, where the blocks of dependent components have been highlighted using dashed lines.

The state of the system has been simulated considering a latent variable that is not directly observed and is assumed to be $\widetilde{Y} \rightarrow N(\Phi(\mathbf{X}), \sigma)$, with $\mathbf{X} = (X_1, X_2, \dots, X_p)$ being a particular configuration of the state vector. We fix $\sigma = 0.2$ and finally, the information about the state of the system Y is simulated from a Binomial distribution with event probability given by $R(\mathbf{x}) = P(\widetilde{Y} > y_0)$, with $y_0 = 0.5$.

For each model and sample size, we have simulated $M = 500$ repetitions of the experiment. Tables 1–4 show the results of model fitting. A discussion about the appropriate number of factors (principal components or blocks) to be extracted is also presented in Table 5.

TABLE 1 Area under the ROC curve (AUC).

	FA–LR–IS algorithm			**Parametric logistic model**		
	Sample size, *n*			**Sample size, *n***		
System	**50**	**100**	**500**	**50**	**100**	**500**
1	0.909	0.903	0.896	0.838	0.831	0.822
2	0.912	0.903	0.901	0.813	0.799	0.785
3	0.895	0.906	NA	0.788	0.817	NA

TABLE 2 Average mean squared error (AMSE) of the local-logistic model without isotonization, for the three systems, sample size n, and different number of blocks, p_0.

$S = 1$		***n***	**50**	**100**	**500**
p_0	2		0.0302	0.0304	0.0298
	3		0.0183	0.0134	0.0076
	4		0.0208	0.0148	0.0081
$S = 2$		***n***	**50**	**100**	**500**
p_0	3		0.0289	0.0272	0.0241
	4		0.0243	0.0189	0.0118
	5		0.0246	0.0191	0.0121
$S = 3$		***n***	**50**	**100**	**500**
p_0	4		0.0418	0.0372	NA
	5		0.0378	0.0293	NA

TABLE 3 Average mean squared error (AMSE) of the local-logistic estimator with isotonization, for the three systems, sample size n, and different number of blocks, p_0.

$S = 1$		***n***	**50**	**100**	**500**
p_0	2		0.0291	0.0290	0.0283
	3		0.0182	0.0133	0.0074
	4		0.0204	0.0146	0.0079
$S = 2$		***n***	**50**	**100**	**500**
p_0	3		0.0284	0.0266	0.0234
	4		0.0240	0.0188	0.0117
	5		0.0243	0.0190	0.0120
$S = 3$		***n***	**50**	**100**	**500**
p_0	4		0.0417	0.0371	NA
	5		0.0377	0.0292	NA

TABLE 4 Average mean squared error (AMSE) of the parametric logistic model, for the three systems, sample size n, and different number of blocks, p_0.

$S = 1$		***n***	**50**	**100**	**500**
p_0	2		0.0961	0.0967	0.0952
	3		0.0476	0.0469	0.0467
	4		0.0250	0.0203	0.0166
$S = 2$		***n***	**50**	**100**	**500**
p_0	3		0.0625	0.0605	0.0592
	4		0.0384	0.0357	0.0346
	5		0.0317	0.0266	0.0229
$S = 3$		***n***	**50**	**100**	**500**
p_0	4		0.0681	0.0656	NA
	5		0.0631	0.0584	NA

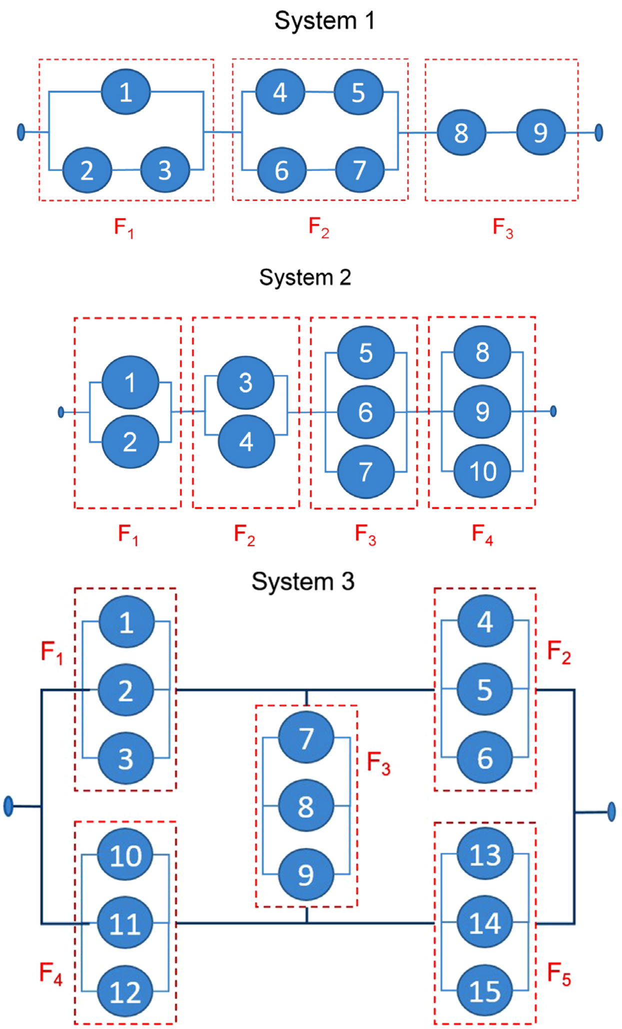

FIGURE 1 Reliability block diagrams for the simulated systems.

TABLE 5 *PVAL* (p_0) is the p-value reported by the significance testing of the number of factors equal to p_0.

$S = 1$		n	50	100	500
p_0	2		8.135e−7	3.878e−17	7.045e−138
	3		0.4660	0.4603	0.4559
	4		0.6372	0.6567	0.6829
$S = 2$		n	50	100	500
p_0	3		1.26e−6	1.225e−17	3.815e−133
	4		0.4525	0.4463	0.4580
	5		0.5430	0.5815	0.5944
$S = 3$		n	50	100	500
p_0	4		4.31e − 09	5.76e-30	NA
	5		0.499	0.490	NA

The structure function for each model is given in the following:

- **System 1**. We consider a series–parallel system with $p = 9$ components, as shown in Figure 1 (top plot). The system is composed of three blocks connected in series. The first two blocks are arranged in parallel and have three and four components, respectively. The third block consists of two components in series. In this context, the structure function of the system is given by the following expression:

$$\phi(\mathbf{x}) = \min(\max(x_1, \min(x_2, x_3)), \max(\min(x_4, x_5), \min(x_6, x_7)), \min(x_8, x_9)),$$

where x_j denotes the state of the jth component, $j = 1, 2, \ldots, 9$.

- **System 2**. We consider a series–parallel combination system with $p = 10$ components, as displayed in Figure 1 (second plot). The system is composed of four parallel blocks connected in series. The first two blocks have two components each and the next two blocks have three components each. In this case, the structure function of the system is given by the following expression:

$$\phi(\mathbf{x}) = \min(\max(x_1, x_2), \max(x_3, x_4), \max(x_5, x_6, x_7), \max(x_8, x_9, x_{10})),$$

where x_j denotes the state of the jth component, $j = 1, 2, \ldots, 10$.

- **System 3**. We consider a bridge system with $p = 15$ components as displayed in the bottom plot of Figure 1. A simple bridge structure has been modified introducing redundancy.[29] That is, each component has been replaced by a block consisting of three units connected in parallel.

$$\phi(\mathbf{x}) = \max(\min(\mathbf{x}_1, \mathbf{x}_4), \min(\mathbf{x}_1, \mathbf{x}_3, \mathbf{x}_5), \min(\mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4), \min(\mathbf{x}_2, \mathbf{x}_5)) \tag{8}$$

where $\mathbf{x}_k = \max(x_{3k-2}, x_{3k-1}, x_{3k})$, for $k = 1, \ldots, 5$, and x_j denoting the state of the jth component, $j = 1, 2, \ldots, 8$.

A graphical description of the algorithm we have implemented to run the simulations is presented in Figure 2.

Table 1 presents the results of goodness of fit of the model obtained running the FA–LR–IS algorithm. The bandwidth parameter has been selected using cross-validation techniques as explained in Section 3.1. To assess the properties of the method, we have compared the nonparametric model with the results obtained using a parametric logistic model to estimate the reliability function with the same data. We do not provide here the details of the parametric model that are detailed in Gámiz et al.[12] To check the goodness of fit of the estimator, we have calculated the area under the ROC curve (AUC) following the guidelines in Gámiz et al.[12] Then the results displayed in Table 1 have been calculated using the estimated reliabilities for each system model ($S = 1, 2, 3$), sample size ($n = 50, 100, 500$), and number of latent factors, which takes values $p_0 = 3$ for System 1, $p_0 = 4$ for System 2 and $p_0 = 5$ for System 3, depending on the particular model being analyzed.

In our context, the AUC is calculated as the probability that the reliability predicted by the estimated model for an operative system is higher than the reliability predicted for a failed system.

As can be seen from the results in Table 1, the nonparametric models outperform the parametric model for all systems and all samples sizes.

Tables 2–4 give the results of accuracy of the estimator provided by the FA–LR–IS algorithm. For comparison purposes, we provide three tables with the results obtained using different methods. The results in the table show the estimation error calculated as follows. For each system (S) and sample size (n), we fix the number of factors we want to extract (p_0) and build the local-logistic model in the space of factors that represent blocks of components in the system design. We obtain the estimated values $\{\widetilde{R}_i, i = 1, \ldots, n\}$. Then we transform the model to the space of original variables, so we obtain the unrestricted estimation of the reliability for each data point as explained in Step 5 of the algorithm in Section 3.1, that is $\{\widehat{R}_i^*, i = 1, \ldots, n\}$ (Table 2). Finally, we obtain the isotonized responses $\{\widehat{R}_i^*, i = 1, \ldots, n\}$ as in Step 6 of the algorithm (Section 3.1), which are the final estimation of the reliability at each item of the dataset (Table 3). Additionally, we also obtain an estimation of the reliability for each data point using the parametric logistic model following.[12] We repeat this procedure for M samples as displayed Figure 2.

For each system, given a particular sample $\{\mathbf{x}_i^m, y_i; i = 1, \ldots, n\}$, with $m = 1, \ldots, M$, the mean squared error MSE_m can be calculated as

$$MSE_{S,m}^{\bullet} = \frac{1}{n}\sum_{i=1}^{n}\left(\widehat{R}_{S,i}^{\bullet,m} - R_S(\mathbf{x}_i^m)\right)^2,$$

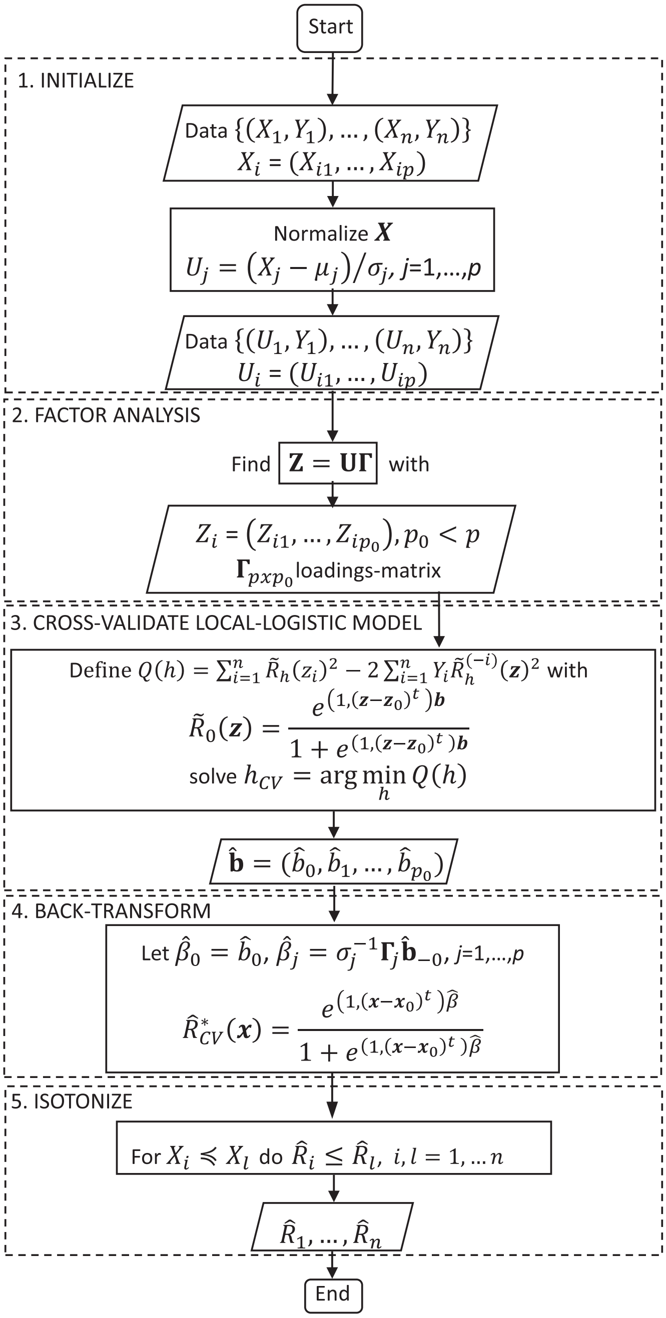
Start
1. INITIALIZE
Data $\{(X_1, Y_1), \dots, (X_n, Y_n)\}$
$X_i = (X_{i1}, \dots, X_{ip})$
Normalize $\boldsymbol{X}$
$U_j = (X_j - \mu_j)/\sigma_j$, j=1,…,p
Data $\{(U_1, Y_1), \dots, (U_n, Y_n)\}$
$U_i = (U_{i1}, \dots, U_{ip})$
2. FACTOR ANALYSIS
Find $\mathbf{Z} = \mathbf{U\Gamma}$ with
$Z_i = (Z_{i1}, \dots, Z_{ip_0}), p_0 < p$
$\boldsymbol{\Gamma}_{pxp_0}$ loadings-matrix
3. CROSS-VALIDATE LOCAL-LOGISTIC MODEL
Define $Q(h) = \sum_{i=1}^n \tilde{R}_h(z_i)^2 - 2\sum_{i=1}^n Y_i \tilde{R}_h^{(-i)}(\boldsymbol{z})^2$ with
$\tilde{R}_0(\boldsymbol{z}) = \frac{e^{(1,(\boldsymbol{z}-\boldsymbol{z}_0)^t)\boldsymbol{b}}}{1+e^{(1,(\boldsymbol{z}-\boldsymbol{z}_0)^t)\boldsymbol{b}}}$
solve $h_{CV} = \arg\min_h Q(h)$
$\hat{\mathbf{b}} = (\hat{b}_0, \hat{b}_1, \dots, \hat{b}_{p_0})$
4. BACK-TRANSFORM
Let $\hat{\beta}_0 = \hat{b}_0, \hat{\beta}_j = \sigma_j^{-1}\boldsymbol{\Gamma}_j \hat{\mathbf{b}}_{-0}$, j=1,…,p
$\hat{R}_{CV}^*(\boldsymbol{x}) = \frac{e^{(1,(\boldsymbol{x}-\boldsymbol{x}_0)^t)\hat{\beta}}}{1+e^{(1,(\boldsymbol{x}-\boldsymbol{x}_0)^t)\hat{\beta}}}$
5. ISOTONIZE
For $X_i \preccurlyeq X_l$ do $\hat{R}_i \le \hat{R}_l,\ i, l = 1, \dots n$
$\hat{R}_1, \dots, \hat{R}_n$
End

FIGURE 2 FA–LR–IS algorithm.

where $R_S(\mathbf{x}_i^m)$ denotes the true reliability function obtained for the structure S; and $\widehat{R}_{S,i}^{\bullet,m}$ can denote the corresponding reliability estimation using the unrestricted nonparametric estimator, $\widehat{R}_{S,i}^{\bullet,m} = \widehat{R}_{S,i}^{*,m}$; the isotonized nonparametric reliability estimator $\widehat{R}_{S,i}^{\bullet,m} = \widehat{R}_{S,i}^{m}$ or the parametric reliability estimation, $\widehat{R}_{S,i}^{\bullet,m} = \widehat{R}_i^{P,m}$.

Tables 2–4 give the averaged values of $MSE^{\bullet}_{S,m}$ along the M repetitions of the experiment, that is $AMSE^{\bullet} = \frac{1}{M}\sum_{m=1}^{M} MSE^{\bullet}_{S,m}$.

Finally, in Table 5, we provide relevant information about the model building process. In particular, at the stage of running FA, we need to decide the optimal number of factors (blocks) that we need to extract in order to get a model that conveniently reduces the dimension of the problem without a significant loose of information. We have considered the testing procedure implemented by the function *factanal* of the software *R*. For each value of p_0 indicated in the Table 5, the null hypothesis assesses that the number of blocks equal to p_0 is adequate to explain the variability of the data. As expected the p-value ($PVAL$) increases with the value of p_0. For system 1, the procedure suggests that a model with two

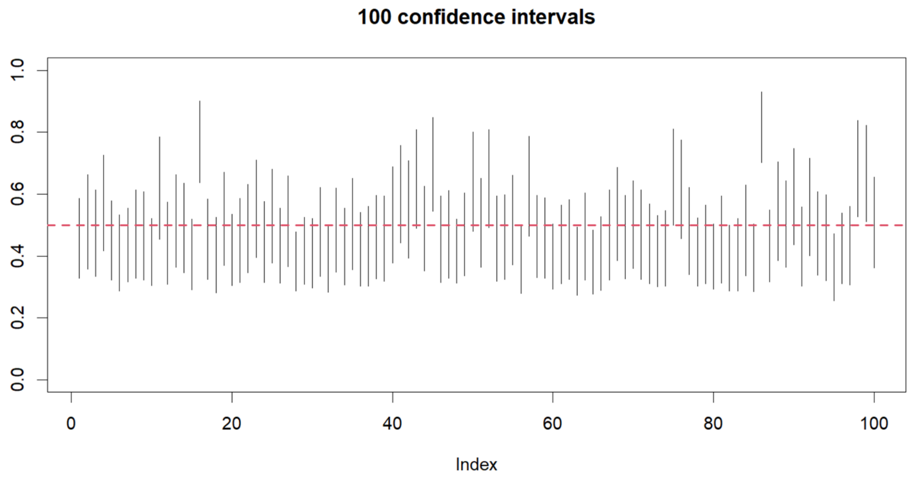

FIGURE 3 Inference about $R(\mathbf{x}_0)$.

blocks is adequate, which agrees with the system design as can be seen in Figure 1, where the units are arranged in two independent blocks of correlated units. For system 2, it is deduced that two blocks are not enough to explain the variability of the data. We need at least $p_0 = 3$ or $p_0 = 4$. Following the principle of parsimony of the model, we keep $p_0 = 3$, which again agrees with the system design. Moreover, although the results are not shown here, we have checked that the FA analysis correctly associate the components to blocks accordingly to the design given in Figure 1. Same conclusions have been obtained for system 3.

Remark. In the analysis of the system $S3$, the case $n = 500$ has not been considered due to the high computational cost associated with this simulation. Although after examining a few replications of the experiment, we have observed that, as expected, the results obtained using the nonparametric model overcome the parametric model, as shown in Table 1 for systems $S1$ and $S2$. Also the trend of the accuracy of the estimation is observed to decrease with sample size. Since we are interested in high dimensionality with respect to system size more than to sample size, we do not find it so illustrative for our purposes to consider this last example with so big sample sizes.

4.2.2 | Some inference issues

In this section, we focus on system 1 but similarly systems 2 and 3 could be used for discussion.

- *Confidence interval for the reliability at a given state vector.*

  For a fixed level of components states, first we are interested in building confidence intervals for the reliability index. In particular, we take $\mathbf{x}_0$ a vector with elements $x_i = 0.5$, for $i = 1, 2, \dots, 9$. Using the results of Section 3.3.1, we can deduce the asymptotic distribution of the estimator $\widehat{\beta}_0(\mathbf{x}_0)$. In this case, we have simulated samples of size $n = 200$ and for each we have constructed a confidence interval at the level of confidence of 95%, that is, $\widehat{\beta}_0(\mathbf{x}_0) \pm q_{0.025}\widehat{\sigma}_{\beta_0}(\mathbf{x}_0)$. From here, a confidence interval for the reliability of the system at $\mathbf{x}_0$ can be easily calculated applying the inverse logit transformation.

  We have run $M = 100$ repetitions of this experiment, so we have 100 confidence intervals for the reliability at $\mathbf{x}_0$. The results are displayed in Figure 3 where we have represented the 100 confidence intervals for the reliability at $\mathbf{x}_0$. We have obtained that the mean size of the intervals is 0.2628, which is reasonable given that the range of R is (0,1). For this configuration of components and under the conditions specified in Section 4.2 for the model from which we simulate the data, the true reliability is $R(\mathbf{x}_0) = 0.5$. In Figure 3, a dashed line is added at the true reliability value. Finally, we have that with this data, the empirical coverage has been calculated as 88%.
- *Testing the importance of components.*

  Using the *FA–LR–IS*, we have considered the following testing problem. We focus on units 7 and 8. As for the system design, it is expected that a change of a certain magnitude in the state of unit 8 has a higher impact in the system performance than a change of the same magnitude in the state of unit 7. In other words, we consider the following

TABLE 6 Recorded execution time of the algorithm in terms of the complexity of the system p, sample size n, and number of blocks p_0.

$\boldsymbol{S = 1}$		$\boldsymbol{n}$	**50**	**100**	**500**
p_0	2		4.1751 s	10.9941 s	3.2526 min
	3		7.0606 s	18.6147 s	5.1491 min
	4		12.9206 s	34.0204 s	8.9630 min
$\boldsymbol{S = 2}$		$\boldsymbol{n}$	**50**	**100**	**500**
p_0	3		6.8160 s	18.4671 s	4.9339 min
	4		11.2588 s	29.5362 s	7.8049 min
	5		13.5407 s	37.5550 s	9.6279 min
$\boldsymbol{S = 3}$		$\boldsymbol{n}$	**50**	**100**	**500**
p_0	4		11.2356 s	29.7303 s	7.4008 min
	5		13.4553 s	37.1375 s	9.4367 min

testing problem:

$$H_0 \,:\, I_7(\mathbf{x}_0) = I_8(\mathbf{x}_0)$$

$$H_1 \,:\, I_7(\mathbf{x}_0) < I_8(\mathbf{x}_0),$$

for $\mathbf{x}_0$, as defined above.

To solve this problem, we have constructed a confidence interval based on Montecarlo methods. Specifically, we have simulated $M = 1000$ samples of size $n = 200$ from system 2. For each sample, we have used the FA–LR–IS algorithm to fit the nonparametric model and obtain an estimation of the vector of parameters $\widehat{\boldsymbol{\beta}}^m$, for $m = 1, \ldots, M$. Then for each sample, we can obtain an estimation of the difference $\beta_7 - \beta_8$. As a result, we obtain a sequence of 1000 estimations of $\beta_7 - \beta_8$. Finally, we use the empirical quantiles of order 0.025 and 0.975, respectively, based on the estimated sequence to construct a confidence interval for the parameter $\beta_7 - \beta_8$. We reject the null hypothesis if the resulting interval is negative. We have obtained that the 95% confidence interval is smaller than −0.1234.

The conclusion is that component 7 has a lower impact on the system performance than component 8 at the significance level of $\alpha = 0.05$ when the system is running at a profile of components given by $\mathbf{x}_0$.

Let us consider a different situation. In concrete, we focus on units 1 and 4

$$H_0 \,:\, I_1(\mathbf{x}_0) = I_4(\mathbf{x}_0)$$

$$H_1 \,:\, I_1(\mathbf{x}_0) \neq I_4(\mathbf{x}_0),$$

for $\mathbf{x}_0$ as above. We have carried out the same procedure as explained in the previous case and the corresponding Montecarlo confidence interval for the parameter $\beta_1 - \beta_4$ has been obtained $(-0.6871423, 0.8941527)$. The conclusion is that we cannot reject that component 1 is as important as 4 for the system performance at the significance level of $\alpha = 0.05$.

4.2.3 | Scalability analysis

In order to analyze how the complexity of the systems affects the performance of our method, we have measured the execution time as the number of units in the system increases. We have considered the three cases depicted in Figure 1. For each system, we have simulated samples of size 50, 100 and 500, respectively. The execution time of one single simulation for each case has been recorded and is presented in Table 6, for all systems and sample sizes. The simulations have been performed using a *3.60 -GHz Intel Core i5-8600 K* processor.

Figure 4 shows the execution time for each case as a function of the sample size n and only for the case of optimal number of blocks. In the graph, we can see that the execution time grows up exponentially as the sample size increases. Also, it can be seen from the plot that the higher the complexity in the system, the longer execution time is recorded.

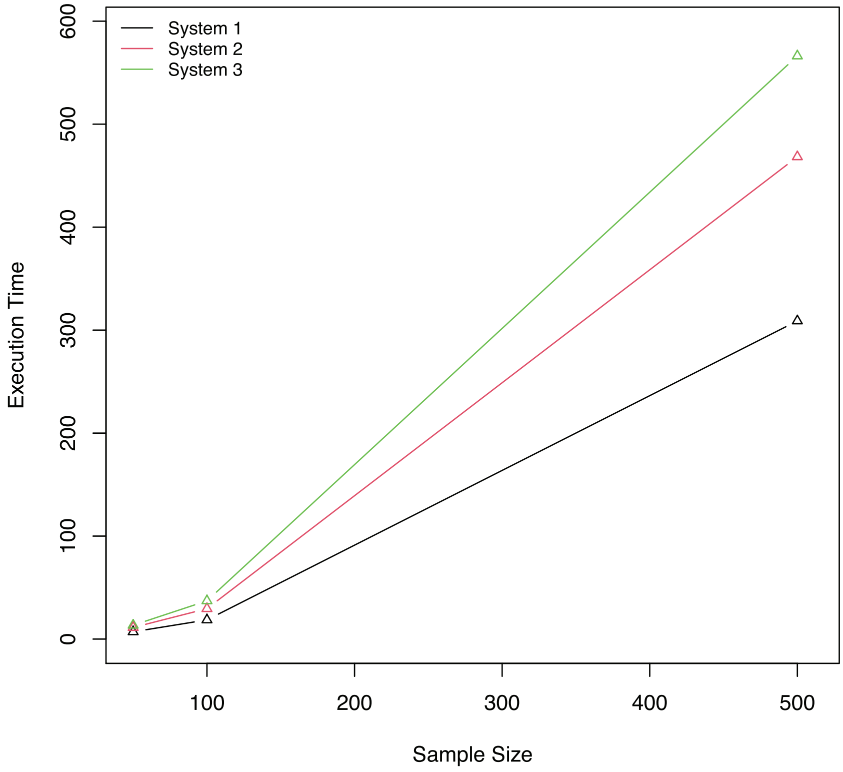

FIGURE 4 Recorded execution time for the three systems, for samples of size n, and optimal number of blocks p_0.

5 | A REAL CASE STUDY: WATER PUMP SENSOR DATA

We analyze a dataset related to the functioning of a water pump of a small area. The data have been taken from the data platform www.kaggle.com/ and a further analysis is presented in Alagarsamy.[30]

Very few technical details about the system are available from the website. The information provided consists of timestamp measurements recorded by 51 sensors and machine status every minute from April 1, 2018 to August 31, 2018. In total, there are 220,320 datapoints and 54 variables. In this example, we do not observe directly the status of each part of the system. Instead, we have some measurement recorded by the corresponding sensor that gives certain information about the state of the corresponding part that this sensor is controlling. Then sensors are used to record temperature, pressure, vibration, load capacity, volume, flow density, and so forth. Although we are aware that in this case we do not observe directly the state of components of the system, we are going to assume in this example that each sensor identifies a particular component in the system and that the value recorded at each time is an observation of the state of that component. In our setting, we assume that we observe a sample of independent systems. In this case, we rather have a longitudinal follow up with one single system with observations taken with a time span of just 1 min between two consecutive datapoints. In order to avoid this inconvenience, we do not take all the records in the sample but consider more distance in time between the data analyzed. We have a sample of size $n = 153$ that consists of all records taken at 00:00 every day, and with this, we feed our model with the aim of determining which are the sensors that give information more relevant to the system behavior in the sense considered in this paper.

1. **Normalize the data**. The values measured by each sensor are related to different magnitudes and thus given in different scales, so it is necessary to normalize the data in order to have values in the same scale that make it possible the comparison.
2. **Examine possible correlations**. In Figure 5, the correlation matrix of all the variables (i.e., sensors measurements) is represented. As can be seen, some groups of variables can be detected with high correlation between the variables in the group. As we have mentioned before, a physical description of the system is not provided in the website that provides the data. However, some experts in the field have contributed through the usual discussion forums available at the platform trying to find up the relationship between each sensor and the corresponding part of the water pump that this sensor is monitoring. According to one of the experts' opinion, the first 14 sensors are monitoring certain

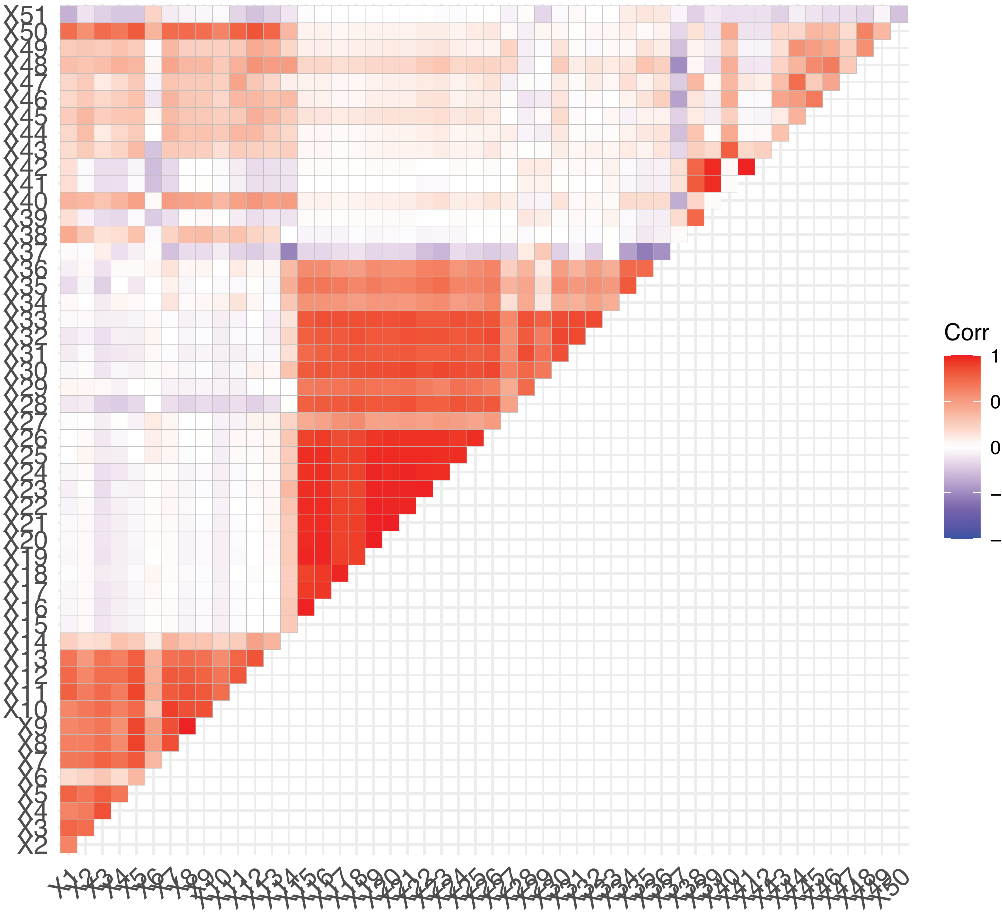

FIGURE 5 Correlation matrix.

aspects all related to the motor. The middle group of sensors are monitoring the performance of two impellers. This can explain in part the correlation structure displayed in Figure 5.

3. We have used the package *psych* of R, to carry out the factorial analysis with these data. First of all, we determine the appropriate number of factors to be extracted by means of a *scree* plot that is shown in Figure 6.
4. **Factor analysis**. We use the function fa included in the package **psych** to carry out latent variable exploratory factor analysis based on maximum likelihood. Then the correlation matrix is decomposed based on eigen values and eigen vector. The communalities for each variable are estimated by the first five factors. The loadings and the interfactor correlations are obtained. Bellow we give the loading matrix that have been estimated by this procedure.

   As can be seen from the results presented in Table 7, the structure is very clear for almost all the sensors. Only three cases, that is, 40, 43, and 45, seem not to present a clear contribution to just one factor.
5. **Local-logistic estimation**. The next step is to fit a local-logistic model in the space of $p_0 = 5$ factors and then we back-transform the results to write the model in the space of original features of dimension $p = 51$. The bandwidth parameter has been estimated by cross-validation.

   With these data, we do not consider the isotonization step. In this case, the regressors considered do not directly correspond to the states of units inside a system. Rather, the observations we have, give partial information about the components states. Then, the coherence property cannot be considered here and then isotonization of the estimated responses makes no sense in this real data study and will not be done. Then, we have a model that predicts the probability that the machine is working (reliability function) as a function of the values reported by each of the 51 sensors installed.
6. **Importance measures**. With the fitted local-logistic model, we can obtain estimations of the first derivatives of the reliability function with respect to all its arguments. We have evaluated these derivatives at all estimation points that are the observation time points (at 00:00 every day). We can thus evaluate what the change in reliability is caused by a unit change at a particular sensor-location. Then we can build a rank of components (sensors) in terms of the effect that

TABLE 7 Loadings matrix.

Sensor	Factor 1	Factor 2	Factor 4	Factor 3	Factor 5
X_1		0.786		0.214	0.388
X_2		0.730			0.120
X_3	−0.111	0.859			0.254
X_4	−0.106	0.708	0.228		0.287
X_5		0.940	0.113		
X_6		0.457	−0.111	−0.211	−0.349
X_7		0.861	0.285		
X_8		0.927	0.150	0.108	−0.308
X_9		0.911	0.149	0.126	−0.356
X_{10}		0.828	0.101		−0.198
X_{11}		0.880	0.208		
X_{12}		0.856	0.313		0.129
X_{13}		0.804	0.231		0.141
X_{14}	0.213	0.257	0.580		
X_{15}	0.969		0.115		
X_{16}	0.977		0.100		
X_{17}	0.917				
X_{18}	0.921				
X_{19}	0.992		0.106		
X_{20}	0.991				
X_{21}	0.991				
X_{22}	0.973		0.201		
X_{23}	0.960		0.261		
X_{24}	0.977				
X_{25}	0.972		0.109		
X_{26}	0.941		0.163		
X_{27}	0.485		0.101		
X_{28}	0.857	−0.107	−0.203		
X_{29}	0.746		−0.360		0.168
X_{30}	0.860		0.145		
X_{31}	0.825				
X_{32}	0.836				
X_{33}	0.871				
X_{34}	0.499		0.389		−0.102
X_{35}	0.604	−0.130	0.507		−0.169
X_{36}	0.534		0.483		−0.153
X_{37}			−0.833	0.128	0.179
X_{38}		0.274		0.809	
X_{39}				0.968	
X_{40}		0.432	0.480		
X_{41}				0.983	
X_{42}				0.987	
X_{43}		0.227	0.317	0.253	
X_{44}		0.261	0.423		
X_{45}	0.101	0.318	0.344		0.174
X_{46}		0.262	0.552		0.124

(Continues)

TABLE 7 (Continued)

Sensor	Factor 1	Factor 2	Factor 4	Factor 3	Factor 5
X_{47}		0.244	0.348	0.131	
X_{48}	0.149	0.373	0.630		0.181
X_{49}		0.285	0.403		0.208
X_{50}		0.818	0.252		0.163
X_{51}		−0.177		−0.121	−0.429

a unit increment in the corresponding sensor-measure has on the system reliability. For each data point $i = 1, \ldots, 153$, we have calculated the vector whose entries are the corresponding partial derivatives for all components, $j = 1, \ldots, 51$, that are given by

$$\mathrm{d}R_{ij} = \widehat{\beta}_{ij}\widehat{R}_i(1-\widehat{R}_i)$$

where $\widehat{\beta}_{ij}$ is the $j+1$th element of the vector of coefficients β estimated at the ith estimation point, and $\widehat{R}_i$ is the estimation of the reliability at the ith estimation point. As a summary measure, we consider the average along all the estimation points. That is, we define

$$\overline{\mathrm{d}R}_j = \frac{1}{153}\sum_{i=1}^{153}\mathrm{d}R_{ij}. \tag{9}$$

The results are presented in Figure 7. From the plot, we can appreciate the different effect that each component has on the system behavior. Considering the absolute value reported by $\overline{\mathrm{d}R}_j$ defined in equation $\overline{\mathrm{d}R}_j$, it is possible to establish a rank of components detect those with higher impact on the system performance, in the sense of the Birnbaum importance measure defined in Gámiz et al.[12] In this case, units monitored by sensors with numbers 04, 03, 12, 50, and 13 are identified as the most important, then, for instance, for maintenance purposes, these elements are recommended to be inspected more carefully.

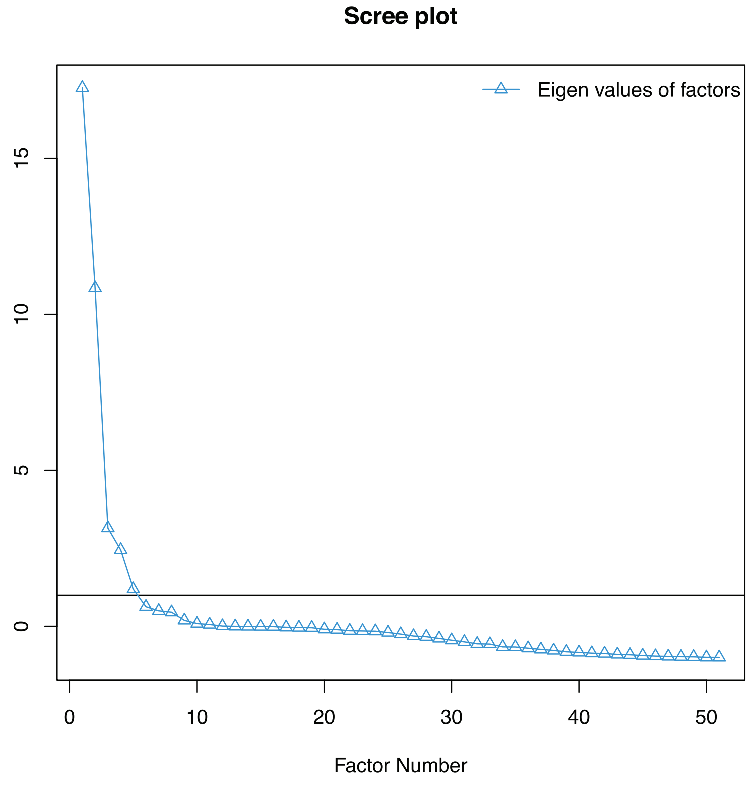

FIGURE 6 Scree plot. Parallel analysis suggests that number of factors is 5.

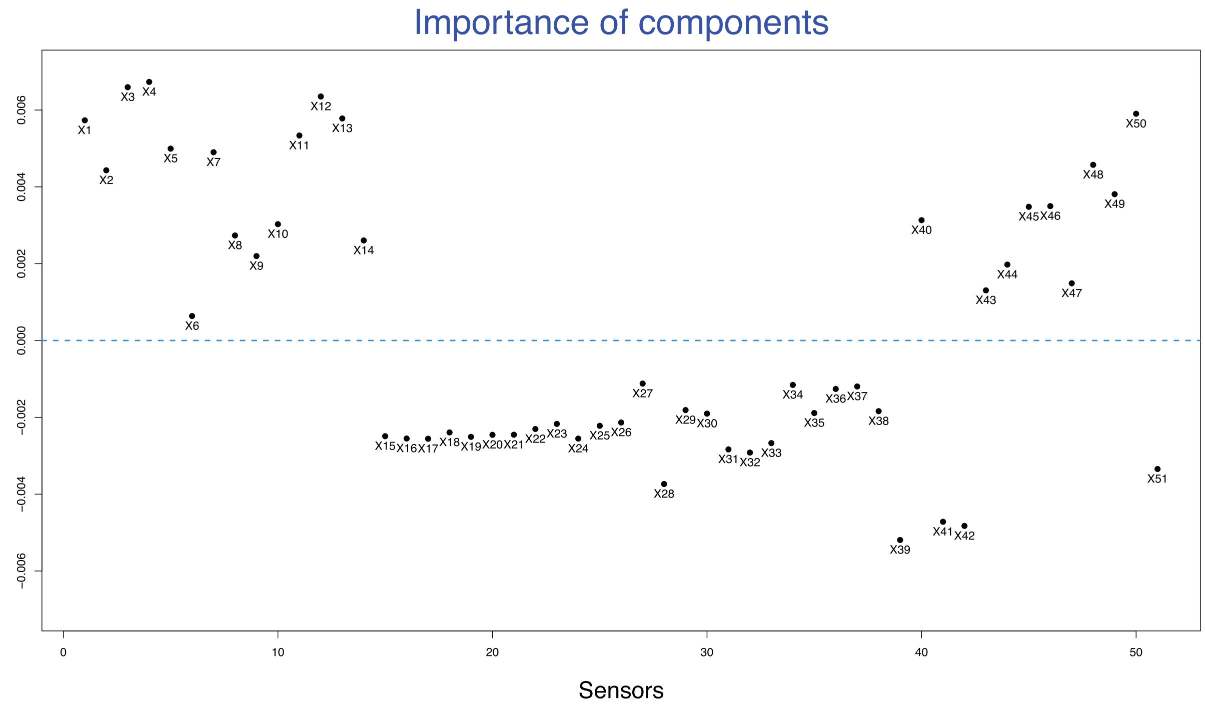

FIGURE 7 Averaged estimated partial derivatives of the reliability function, $\overline{\mathrm{d}R_j}$, for $j = 1, \ldots, 51$.

6 | CONCLUSIONS

In this paper, we propose a strategy to deal with high-dimensional reliability systems. Unless expertise reveals the rational of the system design that allows some strategies for model-fit, the estimation problem has too many features to deal with making the dimensions of the space to grow exponentially while the available data become sparse. We have suggested to carry out a procedure for dimension reduction. We have implemented an algorithm able to transform a large set of inputs into an ideal set of inputs as discussed. Another aspect to have into account when the number of features is very large is that certain correlations between the system units can arise in the latent structure underlying the data and then factor analysis developed using a principal component methodology suggested here is an appropriate tool.

To illustrate the method, we have carried out an extensive simulation study. We have considered three practical systems with different complexity, in terms of the size of the systems (number of units) and the configuration represented by different RBDs. In all cases, the FA–LR–IS provides good results, regarding the blocks recognition and accuracy of the model. Also, to test about the importance of components, we have constructed confidence intervals based on Montecarlo methods, which allows to detect weaknesses in the system configuration. A real dataset has also been considered for illustration purposes.

Future works include to investigate other machine learning techniques (e.g., artificial neural network) in combination with classic statistical models (e.g., nonparametric regression) to obtain a more efficient method both statistically and computationally.

ACKNOWLEDGMENTS

The authors are grateful for constructive comments from two anonymous Reviewers and the Associate Editor. This work was supported in part by the Spanish Ministry of Science and Innovation through Grant Numbers RTI2018-099723-B-I00 and PID2020-120217RB-I00, the Spanish Junta de Andalucía through Grant Number B-FQM-284-UGR20, and the IMAG-Maria de Maeztu Grant CEX2020-001105-M/AEI/10.13039/501100011033.

Open Access Funding provided by Universidad de Granada / CBUA.

DATA AVAILABILITY STATEMENT

Data sharing not applicable to this article as no datasets were generated or analyzed during the current study.

REFERENCES

1. Hong Y, Zhang M, Meeker WQ. Big data and reliability applications: the complexity dimension. *J Qual Technol*. 2018;50:135-149.
2. Soltanali H, Rohani A, Abbaspour-Fard MH, Torres Farinha J. A comparative study of statistical and soft computing techniques for reliability prediction of automotive manufacturing. *Appl Soft Comput*. 2021;98:106738.
3. Li M, Wang Z. Deep learning for high-dimensional reliability analysis. *Mech Syst Sig Process*. 2020;139:106399.
4. Yi J, Du X,. High-dimensional reliability method accounting for important and unimportant input variables. *J Mech Des*. 2022;144(4):041702.
5. Cai Z, Si S, Liu Y, Zhao J. Maintenance optimization of continuous state systems based on performance improvement. *IEEE Trans Reliab*. 2018;67(2):651-665.
6. Si S, Liu M, Jiang Z, Jin T, Cai Z. System reliability allocation and optimization based on generalized Birnbaum importance measure. *IEEE Trans Reliab*. 2019;68(3):831-843.
7. Levitin G, Lisnianski A. A new approach to solving problems of multi-state system reliability optimization. *Qual Reliab Eng Int*. 2001;17(2):93-104.
8. Brunelle RD, Kapur KC. Continuous-state system-reliability: an interpolation approach. *IEEE Trans Reliab*. 1998;47(2):181-187.
9. Coolen FPA, Coolen-Maturi T. The structure function for system reliability as predictive (imprecise) probability. *Reliab Eng Syst Saf*. 2016;154:180-187.
10. Levashenko VG, Zaitseva EN, Kvassay M. *Application of Structure Function in System Reliability Analysis based on Uncertain Data*. ICTERI; 2016.
11. Gámiz-Pérez ML, Martínez-Miranda MD. Regression analysis of the structure function for reliability evaluation of continuous-state system. *Reliab Eng Syst Saf*. 2010;95:134-142.
12. Gámiz ML, Navas-Gómez F, Raya-Miranda R. A machine learning algorithm for reliability analysis. *IEEE Trans Reliab*. 2021;70(2):535-546.
13. Miziula P, Navarro J. Birnbaum Importance Measure for Reliability Systems With Dependent Components. *IEEE Trans Reliab*. 2019;68(2):439-450.
14. Alsina EF, Chica M, Trawiński K, Regattieri A. On the use of machine learning methods to predict component reliability from data-driven industrial case studies. *Int J Adv Manuf Technol*. 2018;94:2419-2433.
15. Afshari SS, Enayatollahi F, Xu X, Liang X. Machine learning-based methods in structural reliability analysis: a review. *Reliab Eng Syst Saf*. 2022. https://doi.org/10.1016/j.ress.2021.108223
16. Chen B, Chen X, Li B, He Z, Cao H, Cai G. Reliability estimation for cutting tools based on logistic regression model using vibration signals. *Mech Syst Sig Process*. 2011;25(7):2526-2537.
17. Phillips J, Cripps E, Lau JW, Hodkiewicz MR. Classifying machinery condition using oil samples and binary logistic regression. *Mech Syst Sig Process*. 2015;60–61:316-325.
18. Lins ID, Chagas Moura M, Zio E, López Droguett E. A particle swarm-optimized support vector machine for reliability prediction. *Qual Reliab Eng Int*. 2011;28(2):141-158.
19. Cui Z, Park N, Chakraborty T. Incremental community discovery via latent network representation and probabilistic inference. *Knowl Inf Syst*. 2020;62(6):2281-2300.
20. Jackson JE. *A User's Guide to Principal Components*. Wiley; 2003.
21. Jolliffe IT. *Principal Component Analysis*. 2nd ed. Springer-Verlag; 2002.
22. Jolliffe IT, Cadima J. Principal component analysis: a review and recent developments. *Philos Trans R Soc A*. 2016;374:20150202.
23. James G, Witten D, Hastie T, Tibshirani R. *An Introduction to Statistical Learning*. Springer; 2013.
24. Meyer MC. Inference for multiple isotonic regression. Technical Report, Colorado State University; 2010.
25. Fan J, Gijbels I. *Local Polynomial Modelling and Its Applications*. Chapman & Hall; 1996.
26. Ahmed W, Hasan O, Tahar S. Formalization of reliability block diagrams in higher-order logic. *J Appl Logic*. 2016;18:19-41.
27. Xiao N, Wan J, Chen HP, Yang FC. Reliability block diagram methods for system reliability analysis of spatial structures. *Int J Lifecycle Perform Eng*. 2016;2(1/2):94-110.
28. Catelani M, Ciani L, Venzi M. RBD model-based approach for reliability assessment in complex systems. *IEEE Syst J*. 2019;13(3):2089-2097.
29. Sahoo L, Bhunia AK, Roy D. Reliability optimization with high and low level redundancies in interval environment via genetic algorithm. *Int J Syst Assur Eng Manag*. 2014;5:513-523.
30. Alagarsamy P. Predict pump failure before it happens using deep learning model. 2021. Accessed November 2022. https://becominghuman.ai/predict-pump-failure-before-it-happens-using-deep-learning-model-dc886bfa073e

How to cite this article: Gámiz ML, Navas-Gómez F, Nozal-Cañadas R, Raya-Miranda R. Unsupervised and supervised learning for the reliability analysis of complex systems. *Qual Reliab Eng Int*. 2023;1-22. https://doi.org/10.1002/qre.3311

AUTHOR BIOGRAPHIES

María Luz Gámiz is Associate Professor at the Department of Statistics and Operational Research of the University of Granada (Spain). She holds PhD degree in Mathematics and Statistics from the University of Granada. Her main areas of research are Nonparametric Statistics, Markov processes, Reliability, and Survival Analysis. She is a co-author of the book "Applied Nonparametric Statistics in Reliability," which is included in Springer Series in Reliability Engineering in 2011.

Fernando Navas-Gómez received his degree in Statistics from the University of Granada (Spain) in 2018, and a master's degree in Applied Statistics from the University of Granada in 2020. He is currently PhD student in Statistics and Post-Graduate Teaching Assistant at the Department of Statistics and Operational Research at the University of Granada. His research interests include Nonparametric Statistics and Reliability Analysis.

Rafael A. Nozal-Cañadas holds a master's degree in Bioinformatics and currently works at the UiT: The Artic University of Norway (Tromsø, Norway) in the Institute for Informatics. Currently, he is doing a PhD on epidemiology analyzing social networks of friends, and how they spread pathogens and they influence your health for better or worse.

Rocío Raya-Miranda is currently an Assistant Professor with the Department of Statistics and Operational Research of the University of Granada (Spain). She received her PhD degree in Statistics in 2003 from the University of Granada. Her research interests include nonparametric curve estimation, bootstrap methods, additive models, reliability analysis, and bandwidth selection.