


## Group-Wise Principal Component Analysis for Exploratory Data Analysis

José Camacho , Rafael A. Rodríguez-Gómez & Edoardo Saccenti


To cite this article: José Camacho , Rafael A. Rodríguez-Gómez & Edoardo Saccenti (2016): Group-Wise Principal Component Analysis for Exploratory Data Analysis, Journal of Computational and Graphical Statistics, DOI: [10.1080/10618600.2016.1265527](https://doi.org/10.1080/10618600.2016.1265527)

To link to this article: <http://dx.doi.org/10.1080/10618600.2016.1265527>

 View supplementary material 

 Accepted author version posted online: 05 Dec 2016.  
Published online: 05 Dec 2016.




 Submit your article to this journal 

 Article views: 145

 View Crossmark data 



# Group-Wise Principal Component Analysis for Exploratory Data Analysis

José Camacho <sup>a</sup>, Rafael A. Rodríguez-Gómez <sup>a</sup>, and Edoardo Saccenti <sup>b</sup>

<sup>a</sup>Department of Signal Theory, Networking and Communication, University of Granada, Granada, Spain; <sup>b</sup>Laboratory of Systems and Synthetic Biology, Wageningen University & Research Center, Wageningen, The Netherlands

## ABSTRACT

In this article, we propose a new framework for matrix factorization based on principal component analysis (PCA) where sparsity is imposed. The structure to impose sparsity is defined in terms of groups of correlated variables found in correlation matrices or maps. The framework is based on three new contributions: an algorithm to identify the groups of variables in correlation maps, a visualization for the resulting groups, and a matrix factorization. Together with a method to compute correlation maps with minimum noise level, referred to as missing-data for exploratory data analysis (MEDA), these three contributions constitute a complete matrix factorization framework. Two real examples are used to illustrate the approach and compare it with PCA, sparse PCA, and structured sparse PCA. Supplementary materials for this article are available online.

## ARTICLE HISTORY

Received August 2015  
Revised August 2016

## KEYWORDS

Exploratory data analysis;  
Matrix factorization;  
Missing-data; Sparsity;  
Visualization

## 1. Introduction

Exploratory data analysis (EDA) has been employed for decades in many research fields, including social and life sciences, psychology, education, medicine, and chemometrics (Han and Kamber 2006). Information to understand complex data can be obtained through matrix factorization methods and associated visualization tools. Factorization methods decompose data into a product of matrices able to highlight special observations (outliers), clusters of similar observations, groups of related variables, and crossed relationships between observations and variables.

Principal component analysis (PCA; Jolliffe 2002; Jackson 2003) is a valuable tool within EDA. PCA provides a factorization based on the criterion of maximizing variance. For data understanding, however, PCA presents two main shortcomings: (i) PCA does not distinguish between unique variance in each variable and shared variance, that is, the variance that is common among a set of variables. This proved to be a serious limitation to unveil relationships among variables (Jolliffe 2002). For this task, a factorization method that focuses on shared variance rather than on any type of variance, like factor analysis (FA; Fabrigar et al. 1999; Costello and Osborne 2005), can be used. (ii) The PCA factorization is often poorly interpretable because the principal components (PCs) are linear combinations of all the variables simultaneously. To overcome this limitation, easier-to-interpret factorizations can be obtained imposing a simple structure on the loadings so that they are combinations of a limited number of variables. This can be achieved by means of rotation (Jolliffe 2002) or sparse methods like sparse principal component analysis (SPCA; Jolliffe et al. 2003; Zou et al. 2006).

Rotation and sparse methods search for an optimum trade-off between simplicity and the amount of variance captured by


the factorization model. This is of clear benefit in a predictive context, where well-defined strategies, such as cross-validation (Zhang 1993), can be implemented to infer the optimum trade-off between variance and model complexity in terms of predictive error. Still, this operation is computationally intensive since a complete space of possible solutions needs to be inspected. The applicability of rotation and sparse methods in the context of EDA, however, is not straightforward. This is because the optimality of the model in the exploratory set-up does not have an explicit definition, like it does in the predictive set-up with the predictive error. Consequently, the same strategy as the predictive set-up is often used in EDA: this is not always fully suitable, since prediction and interpretation are very different goals. Moreover, imposing sparsity to arrive at simplicity brings the risk of simplifying the true relationships in the data, missing part or the whole structure (Camacho 2010).

To overcome this problem, Camacho (2010) proposed the missing-data for exploratory data analysis (MEDA) approach. MEDA consists in the application of a post-processing step after factorization through PCA or partial least square (PLS) to infer the structure among variables. With MEDA, a map of variables, similar to a correlation matrix, is computed. In contrast with standard correlations, the degree of relationship between each pair of variables is estimated using missing data imputation (Nelson, Taylor, and MacGregor 1996; Arteaga and Ferrer 2002, 2005). This has the substantial advantage of filtering out the noise in the computation of correlations (see Camacho 2010, for a comparison), reducing the risk of including spurious or chance associations among variables as often is the case in high-dimensional data (Saccenti et al. 2011a,b).

In this article, we propose a new framework for matrix factorization based on the identification of groups of correlated

**CONTACT** José Camacho  [josecamacho@ugr.es](mailto:josecamacho@ugr.es)

Color versions of one or more of the figures in the article can be found online at [www.tandfonline.com/r/JCGS](http://www.tandfonline.com/r/JCGS).

 Supplementary materials for this article are available online. Please go to [www.tandfonline.com/r/JCGS](http://www.tandfonline.com/r/JCGS).

© 2017 American Statistical Association, Institute of Mathematical Statistics, and Interface Foundation of North America

variables starting from a correlation map. The framework includes the following methods:

- MEDA, which is used to compute the correlation maps with minimum noise level.
- A new algorithm termed the Group Identification Algorithm (GIA) for the identification of (possibly overlapping) groups of variables in the correlation map (MEDA map).
- A new matrix factorization approach referred to as group-wise PCA (GPCA) defined from the identified groups of variables, where loading vectors are restricted to present nonzero values for a single group of variables.
- A new visualization proposed to inspect the groups of variables.

This framework has a number of advantages in the context of EDA. First, parameter setting can be easily made through interactive data visualization and analysis, and datasets for which the assumption of sparsity does not hold can be easily identified. We claim that this is a main advantage over SPCA, where parameter setting is complex, computationally intensive, and may lead to missing part of the structure in the data. Second, the matrix factorization and visualization are fully interpretable. Third, the fitting algorithm of GPCA is fast and simple, since it consists of a set of nested PCA together with a suitable deflation procedure. Our approach is applicable on a wide range of different datasets, ranging from network security to biological *omics* data, for which the proposed factorization greatly improves understanding.

The remaining of the article is organized as follows. [Section 2](#) contains an introduction to matrix factorization approaches, with special focus on the sparse PCA approaches. [Section 3](#) introduces the MEDA approach. [Section 4](#) introduces the new group-wise PCA approach, including the GIA algorithm to identify the groups of variables in correlation maps. The new associated visualization, the Treemap plot, is included in supplementary materials. Two experimental datasets are analyzed in [Section 5](#). [Section 6](#) presents some concluding remarks.

## 2. Exploring Relationships Among Variables Using PCA

PCA is one of the most used tools to explore relationships among variables (Fabrigar et al. 1999; Costello and Osborne 2005). PCA is applied to two-way datasets, where  $M$  variables (or features) are measured/computed for  $N$  observations (or objects). The goal of PCA is to find the subspace of maximum variance in the  $M$ -dimensional variable space. This is done by finding linear transformations of the original variables, called principal components (PCs), which are orthogonal and explain decreasing amounts of variance in the original data. The PCA model follows the expression:

$$\mathbf{X} = \mathbf{T}_A \mathbf{P}_A^T + \mathbf{E}_A, \quad (1)$$

where  $\mathbf{X}$  is an  $N \times M$  data matrix,  $\mathbf{T}_A$  is the  $N \times A$  score matrix containing the projection of the objects onto the  $A$  principal components (PCs) subspace,  $\mathbf{P}_A$  is the  $M \times A$  loading matrix containing the linear combination of the variables represented in each of the PCs, and  $\mathbf{E}_A$  is the  $N \times M$  matrix of residuals.

PCA has several limitations that may hamper the retrieval of the structure among variables. In particular (i) it does not distinguish between unique and shared variance (Jolliffe 2002), and (ii) the principal components are a linear combination of all the variables simultaneously (Jolliffe, Trendafilov, and Uddin 2003). Hence, every component typically compresses variance for several and interdependent groups of related variables. This greatly complicates interpretation especially for high-dimensional data (Camacho 2010). Moreover, outliers, that is, observations with large scores, need to be investigated with non-trivial tools (Alcala and Joe Qin 2011; Camacho 2011).

To overcome the limitation of PCA making no distinction between shared and unique variance in the data, factor analysis (FA) has been proposed. FA focuses on the shared variability by several variables, referred to as the communalities. Still, it shares with PCA the same limitation that the loading vectors may be difficult to interpret because they are linear combinations of all the variables. For this reason, when PCA or FA are used to explore the relationships among variables, a two-step procedure is typically followed (Jolliffe 2002; Jackson 2003). First, the model is calibrated from the available data, second, the loadings are rotated to facilitate the interpretation of the model. The transformation matrix is usually found using a particular rotation criterion such as the varimax criterion (Kaiser 1958), which is a member of the orthomax family (Crawford and Ferguson 1970) (see Browne 2001 for a review). In general, oblique transformations are preferred to more simple orthogonal transformations (Fabrigar et al. 1999; Costello and Osborne 2005), although in many situations the results are similar (Jolliffe 2002). The transformation criteria aim at a particular form of simplicity of the rotated loading matrix, either in terms of a predetermined, fixed rotation criterion or in terms of optimal interpretability. Usually, a combination of both high and low loadings within each component is perceived as a simple structure (Timmerman, Kiers, and Smilde 2007). Alternatively, Procrustes rotation of the loading matrix toward a given target matrix can be used, for which different approaches exist (Cliff 1966).

Jolliffe (1995) listed a number of drawbacks of rotation methods, including the fact that the rotated model depends greatly on the normalization of the data and the number of components used to fit the model. To overcome these limitations, alternative approaches have been suggested (Jolliffe 2002) to define a calibration algorithm for factorization where a trade-off between variance explained and model simplicity is pursued. A subset of these techniques that constitutes the state-of-the-art are the so-called SPCA methods.

### 2.1 Sparse Principal Component Analysis

SPCA (Jolliffe, Trendafilov, and Uddin 2003; Zou, Hastie, and Tibshirani 2006) is an extension of PCA where the loss function is modified to incorporate constraints similarly to what is done in nonnegative matrix factorization (Lin 2007) and multivariate curve resolution (de Juan and Tauler 2006). Commonly, constraints are imposed by means of regularization terms leading to a simple structure, so that the number of nonzero loadings in a single PC is reduced or constrained. Jolliffe, Trendafilov, and Uddin (2003) incorporated the LASSO constraint (Tibshirani 1994) in the PCA calibration. In the LASSO, the  $l_1$  norm

(absolute values) of the loadings is penalized. The LASSO criterion to be minimized in the regression paradigm is

$$\boldsymbol{\beta}_{\text{lasso}} = \arg \min_{\boldsymbol{\beta}} \|\mathbf{y} - \sum_{m=1}^M \mathbf{x}_m \beta_m\|^2 + \lambda \sum_{m=1}^M |\beta_m|, \quad (2)$$

where  $\beta_m$  are the coefficients in  $\boldsymbol{\beta}$  and  $\mathbf{x}_m$  correspond to the columns of  $\mathbf{X}$ .

Zou et al. (2006) introduced an alternative formulation, which was actually referred to as the sparse PCA algorithm, redefining PCA as a regression problem with the ridge penalty, where  $\mathbf{y}$  equals the PCA scores. In particular, they showed that the  $a$ th loading vector of PCA,  $\mathbf{p}_a$ , equals  $\frac{\boldsymbol{\beta}_{\text{ridge}}}{\|\boldsymbol{\beta}_{\text{ridge}}\|}$ , where

$$\boldsymbol{\beta}_{\text{ridge}} = \arg \min_{\boldsymbol{\beta}} \|\mathbf{t}_a - \sum_{m=1}^M \mathbf{x}_m \beta_m\|^2 + \lambda_2 \sum_{m=1}^M \|\beta_m\|^2 \quad (3)$$

and  $\mathbf{t}_a$  is the  $a$ th score vector. Using this redefinition, the LASSO can be implemented using a criterion close to the (naive) elastic net (Zou and Hastie 2005), which is a combination of both the LASSO and ridge constraint:

$$\begin{aligned} \boldsymbol{\beta}_{\text{SPCA}} = \arg \min_{\boldsymbol{\beta}} & \|\mathbf{t}_a - \sum_{m=1}^M \mathbf{x}_m \beta_m\|^2 + \lambda_2 \sum_{m=1}^M \|\beta_m\|^2 \\ & + \lambda_1 \sum_{m=1}^M |\beta_m|. \end{aligned} \quad (4)$$

In the SPCA algorithm, loadings and the corresponding scores are obtained using an alternating approach, where  $\boldsymbol{\beta}_{\text{SPCA}}$  terms are obtained from the scores  $\mathbf{T}_A$  and the latter are recomputed from the singular value decomposition (SVD) of the covariance matrix post-multiplied by  $\boldsymbol{\beta}_{\text{SPCA}}$ , that is,  $\mathbf{X}^T \mathbf{X} \boldsymbol{\beta}_{\text{SPCA}}$ .

Several alternative constraints have been proposed to incorporate other types of a priori information in the matrix factorization. Of special interest for this article are those based on the group LASSO (Bach 2007; Jacob, Obozinski, and Vert 2009), where groups of variables are set to 0 together. SPCA on the group LASSO applies to a predefined set of groups of variables rather than to individual variables. Further extensions (Jenatton, Audibert, and Bach 2009) allow the definition of a prespecified structure (or set of shapes) within the SPCA formulation (Jenatton, Obozinski, and Bach 2009). This approach is referred to as structured SPCA (SSPCA). Following the notation of this article, the SSPCA approach follows:

$$\boldsymbol{\beta}_{\text{SSPCA}} = \arg \min_{\boldsymbol{\beta}} \|\mathbf{t}_a - \sum_{m=1}^M \mathbf{x}_m \beta_m\|^2 + \lambda_2 \sum_{m=1}^M \|\beta_m\|^2 + \lambda_1 \Omega^\alpha(\boldsymbol{\beta}), \quad (5)$$

where  $\Omega^\alpha$  are a set of quasi-norms for  $\alpha \in (0, 1)$  and the  $l_1$  norm for  $\alpha = 1$  as defined in Jenatton, Audibert, and Bach (2009) and Jenatton, Obozinski, and Bach (2009). The SSPCA approach is suitable for analyzing datasets with a certain ordering among variables, for instance, the grid of pixels in a multivariate image, so that a set of possible grouping structures among the variables can be defined a priori. Thus, a main limitation of the SSPCA approach for exploratory analysis is the need to define a priori this grouping structure. This predefinition requires domain knowledge, so that it is not generally applicable in the context

of EDA. However, we can follow the same approach proposed in this article, that is, defining this structure from correlation, to use SSPCA in EDA. This will be shown in the experimental part of the article.

It should be noted that neither SPCA nor SSPCA were developed for EDA. However, at least the former has been used for that purpose by Rasmussen and Bro (2012) and they are, essentially, the most similar approaches to the one proposed in this article.

### 3. Missing-Data for Exploratory Data Analysis

As explained in the introduction, an alternative to sparse matrix factorization for EDA is the MEDA approach, where standard factorizations are post-processed to construct correlation matrices that can be used to unveil the relationships among variables.

The procedure to create the MEDA map  $\mathbf{M}$  consists of the following steps:

Step 1: Factorize the data matrix  $\mathbf{X}$  such that

$$\mathbf{X} = \mathbf{T}_A \mathbf{P}_A^T + \mathbf{E}_A, \quad (6)$$

where  $\mathbf{T}_A$  is the  $N \times A$  matrix of scores containing the projection of the objects in the  $A$  latent variables sub-space,  $\mathbf{P}_A$  is the  $M \times A$  matrix of loadings containing the linear combination of the original variables, and  $\mathbf{E}_A$  is the  $N \times M$  matrix of residuals.

The factorization in Equation (6) is general, and can be obtained, for instance, via PCA, FA, or PLS.

Step 2: For each variable  $i$  from 1 to  $M$

Step 2.1: Build  $\tilde{\mathbf{X}}_i$ , an  $N \times M$  matrix with all zeros except in the  $i$ th column containing the  $i$ th column of  $\mathbf{X}$

$$\tilde{\mathbf{X}}_i = [\mathbf{0} \dots \mathbf{0} \mathbf{X}_i \mathbf{0} \dots \mathbf{0}]. \quad (7)$$

Step 2.2: Estimate the scores  $\hat{\mathbf{T}}_A$  with  $A$  latent variables from  $\tilde{\mathbf{X}}_i$  using a missing data approach MD

$$\hat{\mathbf{T}}_A = \text{MD}(\tilde{\mathbf{X}}_i). \quad (8)$$

Step 2.3: Reconstruct the original measurements

$$\hat{\mathbf{X}}_A = \hat{\mathbf{T}}_A \mathbf{P}_A^T, \quad (9)$$

and estimate the error:

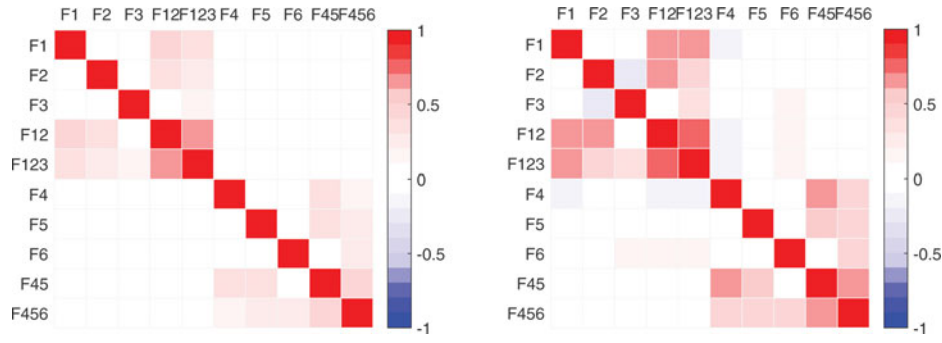
$$\hat{\mathbf{R}}_A = \mathbf{X} - \hat{\mathbf{X}}_A. \quad (10)$$

Step 2.4: For each variable  $i$  different to  $j$ ,  $m_{ij}$  (where we drop the subscript  $A$  for simplicity) is computed as

$$m_{i,j} = 1 - \frac{\sum_{n=1}^N (\hat{\mathbf{R}}_{A,(n,i)})^2}{\sum_{n=1}^N (\mathbf{X}_{n,j})^2}, \quad \forall i \neq j, \quad (11)$$

where the right term of Equation (11) is the goodness-of-prediction index proposed by Wold (1978). The closer the value the index is to 1, the more related the  $i$ th and  $j$ th variables are.

Step 3: The MEDA map  $\mathbf{M}$  is built from elements  $m_{ij}$ . The diagonal elements  $m_{ii}$  are set to the square of the variance in each variable captured by the first  $A$  components in model (6).



**Figure 1.** Comparison of the MEDA map  $\mathbf{M}$  (a) and the correlation map (b) for the pipeline system simulated data. The PCA model is fitted with six components. The structure underlying the data is clearly visible.

The calibration method in Equation (6) provides the component structure used to estimate the missing data in Step 2.2. There is a number of possibilities to perform this step. The known data regression method is the default method to construct MEDA in the MEDA toolbox (The MEDA toolbox is available at <https://github.com/josecamachop/MEDA-Toolbox>) (Camacho et al. 2015). This method is statistically superior to the other approaches (Arteaga and Ferrer 2002) and it is equivalent to the conditional mean replacement method (Nelson, Taylor, and MacGregor 1996).

Let us now illustrate the use of MEDA with a simulated example, which we will use as guiding example through the article. We consider a pipeline system where six different input liquids are mixed into two output pipes. For each pipeline, data are simulated according to the following rules:

$$F_{12} = F_1 + F_2 \quad (12)$$

$$F_{123} = F_{12} + F_3$$

$$F_{45} = F_4 + F_5$$

$$F_{456} = F_{45} + F_6,$$

where  $F_1, F_2, \dots, F_6$  represent input liquid flows that are randomly and independently generated following a normal  $\mathcal{N}(0, 1)$  distribution. The pipeline system is shown in supplementary Figure S5. Additionally, a 30% Gaussian  $\mathcal{N}(0, 1)$  noise is added to each variable  $Fx$  to simulate measurement noise:

$$Fx' = \frac{(Fx + \sqrt{0.3} \cdot \Sigma)}{\sqrt{1.3}}, \quad (13)$$

where  $Fx'$  is the contaminated variable,  $Fx$  is the noise-free variable generated according to Equation (12) and  $\Sigma$  is the noise generated.

The input flows ( $F_1, F_2, \dots, F_6$ ), the intermediate ( $F_{12}$  and  $F_{45}$ ), and the output flows  $F_{123}$  and  $F_{456}$  are monitored each by a single variable labeled with the name of the corresponding pipe, for a total of 10 variables built from six common factors (input pipelines,  $F_1, F_2, \dots, F_6$ ).

To create an MEDA map  $\mathbf{M}$  from the data  $\mathbf{X}$ , we factorized the data using PCA as for step 1 in Section 3. The optimal number of components (which resulted to be 6) is chosen by means of cross-validation using the column-wise  $k$ -fold ( $ckf$ ) algorithm that has shown to be an adequate choice when the PCA model is used for exploratory data analysis (Saccenti and Camacho 2015).

The  $ckf$  is available in the MEDA toolbox (Camacho et al. 2015). Once the MEDA algorithm is run, the resulting MEDA map can be easily visualized as a color map as shown in Figure 1(a). Inspecting the MEDA map, the structure underlying the data can be easily observed. For example, the independence of the input flows variables  $F_1, F_2$ , and  $F_3$  from the input flows  $F_4, F_5$ , and  $F_6$  is evident, as well as the correlation of  $F_{12}$  with the inputs  $F_1$  and  $F_2$ . Although the structure in the data is easily predictable in this simple example, it serves to illustrate the usefulness of the MEDA plot in the first stage of data exploration. Figure 1(b) shows the standard correlation map, where several spurious correlations that are filtered out by MEDA are observed.

## 4. Group-Wise Principal Component Analysis

In this section, we propose a sparse factorization method based on PCA where sparsity is defined in terms of groups of (correlated) variables identified from the MEDA map. We refer to this approach as group-wise PCA (GPCA). In the GPCA, the factorization of the data matrix  $\mathbf{X}$  is such that every component has loadings different from zero only for a group of variables. Thus, here sparsity is different to the one used in the regularization setting, where it is obtained by forcing to zero the loadings corresponding to variables or groups of variables.

### 4.1 Identification of Groups of Correlated Variables

GPCA is based on the identification of  $K$  (possibly overlapping) groups  $S_1, S_2, \dots, S_k, \dots, S_K$  of correlated variables in the MEDA map (or in any correlation matrix). To create the groups  $S_k$ , we propose a new algorithm, referred to as the group identification algorithm (GIA) for which we give here the general idea: details are given in the supplementary materials.

Let  $\mathbf{M}$  ( $M \times M$ ) be an MEDA map or a correlation matrix with elements  $m_{i,j} \in [-1, 1]$  and let  $0 < \gamma < 1$  be a threshold on the correlation values. The group  $S_k$  is built in such a way that variables satisfy the conditions

$$\forall i, j \in S_k \rightarrow |m_{i,j}| > \gamma, \quad (14)$$

and

$$\forall j \notin S_k / \exists i \in S_k \rightarrow |m_{i,j}| \leq \gamma \quad (15)$$

so that if the  $j$ th variable is not in group  $S_k$ , it has a correlation  $< \gamma$  with at least one of the other variables in the group. This is equivalent to define groups of variables with maximum cardinality where all variables within the group present a correlation larger than  $\gamma$ .

The user-defined threshold  $\gamma$  can be interactively adjusted by inspecting the visualization of MEDA and the output of the GIA. This is coherent with the EDA philosophy, and it is easier to tune than the regularization parameters in the SPCA implementation. The GIA algorithm is available in the MEDA toolbox (Camacho et al. 2015).

## 4.2 The GPCA Algorithm

The algorithm to arrive at the GPCA model consists of a set of nested PCAs together with a suitable deflation procedure. Given the data matrix  $\mathbf{X}$ , the procedure is as follows:

Step 1: Initialize the following matrices:

$$\mathbf{C} = \mathbf{X}^T \mathbf{X} \quad (16)$$

$$\mathbf{B} = \mathbf{I}, \quad (17)$$

where  $\mathbf{I}$  is the identity matrix.

Step 2: For each component  $a$  from 1 to  $A$

Step 2.1: For each group  $S_k$  in the set of groups  $S$

Step 2.1.1: Create  $\mathbf{C}^k$  from  $\mathbf{C}$  and setting elements out of  $S_k$  to zero.

$$\mathbf{C}^k = \mathbf{C} \quad (18)$$

$$c_{lm}^k = 0, \quad \forall l \notin S_k \text{ or } \forall m \notin S_k. \quad (19)$$

Step 2.1.2: Compute the eigendecomposition of  $\mathbf{C}^k$  and select the first eigenvector.

$$\mathbf{C}^k = \mathbf{p}^k (\sigma^k)^2 (\mathbf{p}^k)^T + \mathbf{E}^k. \quad (20)$$

Step 2.2: Choose the loadings and scores of component  $a$  from the group capturing the most variance.

$$\mathbf{p}_a = \arg \min_{\mathbf{p}^k} \|\mathbf{E}^k\|_F \quad (21)$$

$$\mathbf{t}_a = \mathbf{X} \mathbf{p}_a. \quad (22)$$

Step 2.3: Perform the deflation according to Mackey (2008).

$$\mathbf{q} = \mathbf{B} \mathbf{p}_a \quad (23)$$

$$\mathbf{C} = (\mathbf{I} - \mathbf{q} \mathbf{q}^T) \mathbf{C} (\mathbf{I} - \mathbf{q} \mathbf{q}^T) \quad (24)$$

$$\mathbf{B} = \mathbf{B} (\mathbf{I} - \mathbf{q} \mathbf{q}^T). \quad (25)$$

The GPCA algorithm first computes  $K$  loading vectors, each of them considering only the set of variables corresponding to group  $S_k$ . From these, it chooses the one with the highest variance, discarding the rest. Using this loading vector, the complete Gram matrix  $\mathbf{C}$  is deflated following Mackey (2008). To select the number of PCs in group-wise PCA, the *ckf* algorithm (Saccenti and Camacho 2015) can be directly employed.

When in the correlation map there is a clear structure in groups, the metaparameter  $\gamma$  can be easily set, and the GIA yields the set of groups of variables  $S_k$ . In accordance with Equations (14) and (15), and with the restatement of the definition of sparsity in PCA proposed here, the GPCA yields PCs with loading vectors different to 0 for a single group of related variables.

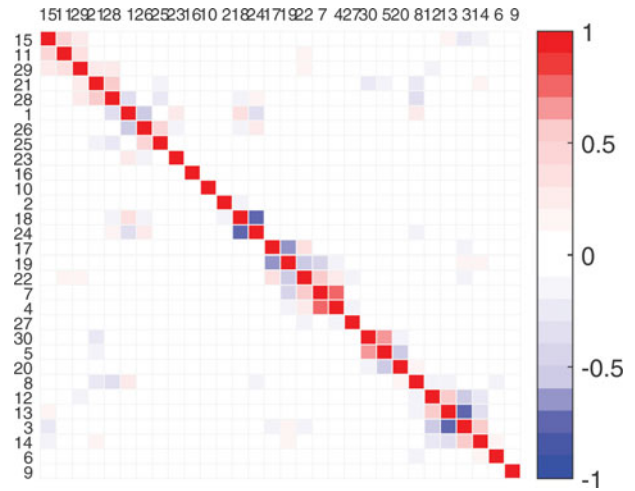


Figure 2. MEDA of a random matrix obtained with ADICOV (Camacho et al. 2011).

The GPCA approach resembles to some extent the goal of the SPCA using the group LASSO regularization (Jacob, Obozinski, and Vert 2009) or similar structure-based regularization (Jenatton, Obozinski, and Bach 2009), but differs in how the structure is defined. The GPCA method has a number of advantages in the context of EDA referred in the introduction. The most important is the fact that datasets where the application of sparse models is not admissible can be easily detected from the MEDA plot. Take for instance the MEDA plot in Figure 2, computed from a random dataset simulated with the ADICOV tool (Camacho et al. 2011), also available in the MEDA Toolbox. There is scarce structure in the data, and groups are limited to two variables at most. Therefore, in this dataset the use of GPCA is not recommended. Another advantage of GPCA is that adequate values for the metaparameter  $\gamma$  can be inferred from the MEDA visualization. The GPCA algorithm is available in the MEDA toolbox (Camacho et al. 2015).

Figure 3 shows the GPCA loadings for the six PCs of the model in the pipelines example. We can see the influence of the different variables to the components of the model. In this case, the groups observed in the visualization can be easily identified

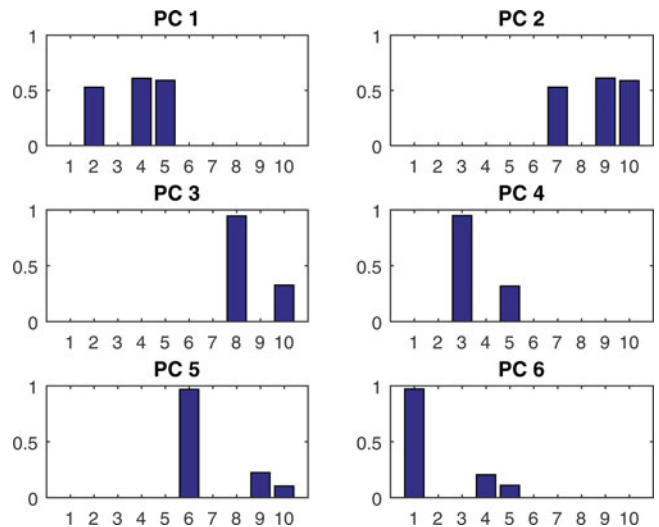


Figure 3. GPCA loadings for the first six principal components of the simulated pipeline example.

in the loadings. In our simple example, each component captures the variability corresponding to a single group.

## 5. Experimental Examples

In this section, we compare the proposed framework based on GPCA with the standard PCA, the SPCA (Zou, Hastie, and Tibshirani 2006) and the SSPCA (Jenatton, Obozinski, and Bach 2009) approaches. The results are compared in terms of visualization and, following Jolliffe et al. (2003), in terms of captured variance (Zou, Hastie, and Tibshirani 2006) and simplicity, the latter using the Varimax index (Jolliffe, Trendafilov, and Uddin 2003):

$$V = \sum_{a=1}^A \left[ \sum_{m=1}^M b_{ma}^4 - \frac{1}{M} \left( \sum_{m=1}^M b_{ma}^2 \right)^2 \right], \quad (26)$$

where  $b_{ma}$  is the loading of the  $m$ th variable in the  $a$ th component.

PCA and GPCA are computed using the MEDA toolbox (Camacho et al. 2015). The SPCA results are computed with the SpaSM toolbox (<http://www2.imm.dtu.dk/projects/spasm/>) (Sjöstrand and Clemmensen 2012). In that implementation, input arguments for SPCA are the value for  $\lambda_2$  in Equation (4) and either a bound  $u$  for the  $l_1$  norm of the loading vectors or the number of nonzero elements in those vectors. It is well known (Zou and Hastie 2005) that such a definition is equivalent to the use of the regularization term with  $\lambda_1$  in Equation (4). The SSPCA results are computed with the toolbox supplied with (Jenatton et al. 2009) [http://rodolphejenatton.com/software/SparseStructuredPCA\\_MatlabToolbox\\_V1.0\\_rjenatton.tar](http://rodolphejenatton.com/software/SparseStructuredPCA_MatlabToolbox_V1.0_rjenatton.tar). In absence of grouping structure, this was identified using MEDA and GIA and the weights of the variables in the group were set to 1. Input arguments for SSPCA are  $\gamma$  in GIA and  $\alpha$  and  $\lambda_1$  in Equation (5).

Two datasets of very different nature are analyzed. In both cases, the general goal is to understand/investigate the data. However, the specific findings that are looked for are very different. The first dataset corresponds to data collected from a communication network, where the goal is to find cybersecurity issues in the network during the data collection interval. The second dataset originates from a plant metabolomics study where the goal is to investigate the response of the plant metabolome to a chemical toxin.

### 5.1 Network Security Data

The VAST 2012 2nd mini challenge is a benchmark for visualization in cybersecurity. The goal is to identify cybersecurity issues in the data collected during 2 days from a computer network. During those days, a number of nonlegitimate programs were found to be running on several computers, slowing them down. A cyber-forensics operation is required to discover the root causes for this strange behavior.

Two typical sources of data are collected from the network: firewall and Intrusion Detection System (IDS) logs. The firewall analyzes the incoming and outgoing data traffic in the network, and records in a log file all connection attempts that are blocked according to security policies. The IDS employs higher level intelligence to identify cybersecurity incidents in data traffic. It also stores the results in a log file, though it does not block any traffic connection. Also, typically, it only analyzes a sub-set (sample) of the total traffic.

A total of 2350 observations, each one with the information for 1 min, are obtained. For every sampling period of 1 min, we defined a set of 112 variables that represent the information from the two data sources: 69 variables for the firewall log and 43 for the IDS log (see supplementary Table S1). The number of variables was reduced to 95 by discarding those with constant value throughout the capture period. The resulting dataset with 2350 observations and 95 variables has been previously used for Multivariate Statistical Network Monitoring (MSNM) in Camacho et al. (2014, 2016) where standard PCA is used.

Let us start the exploration of the data with PCA. For the sake of simplicity in the comparison between methods, we will restrict ourselves to the first two PCs. Figure 4 shows the score plot and loading plot of the model, using a typical bi-plot (or scatterplot) visualization. To improve visualization and for the sake of easy comparison with the other methods, the variables have been reordered using MEDA. Thus, correlated variables are arranged together. According to Figure 4(a), the first PC captures a general trend in the data while the second PC reflects the excursion of a number of outliers. Inspecting the loading plot in Figure 4(b), we can hypothesize that the trend is related to the group of variables at the left (vars #40-57) and the excursion to the group of variables at the bottom (vars #66-79). However, care should be taken to double-check these hypotheses extracted from PCA bi-plots, as shown in Camacho (2010).

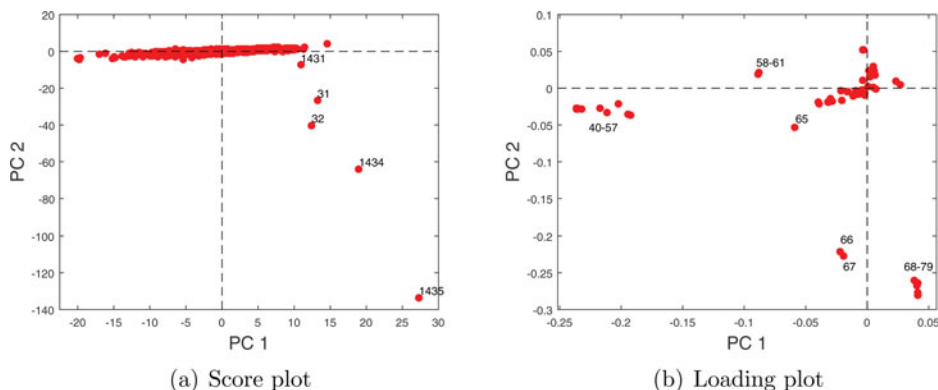
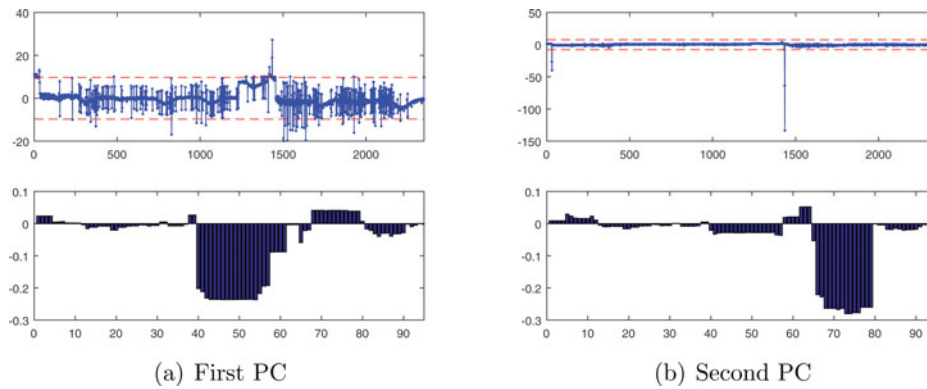


Figure 4. Scatterplots for the first two PCs in PCA for the Network security example.



**Figure 5.** Line/bar plots for the first two PCs in PCA for the Network security example: scores are shown on the top, between 99% control limits, and loadings are shown on the bottom.

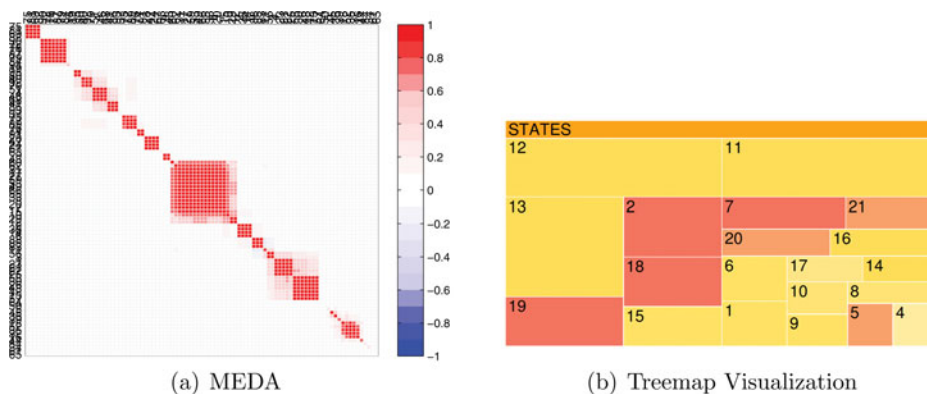
An alternative visualization of the problem is presented in [Figure 5](#), in which each subfigure shows a line/bar plot of scores and loadings of a single component. On the top, the scores corresponding to the PCs are shown between 99% control limits (Nomikos and MacGregor 1995). The limits help identify outliers and trends in the observations. On the bottom, the loadings are shown. This visualization is very useful in this example, because the patterns described by the PCs (trend and excursion) are not interrelated. In the figure, we can easily identify outliers, shifts, etc., with better detail than in bi-plots. However, bi-plots are by far more popular for PCA inspection, especially because some patterns in the data are more apparent in the sub-space defined by a few PCs and not just one PC. This is a consequence of the problem stated in [Section 2](#) according to which one PC captures information for several and interdependent groups of related variables, which in turn means that a specific data pattern may span several PCs, a problem that makes EDA with PCA a hard task for some datasets.

Let us continue with the EDA performed using GPCA. The MEDA plot of the data for 15 PCs (selected using *ckf*) is shown in [Figure 6\(a\)](#). The plot shows a clear group-wise pattern of correlation, which tell us that GPCA can be a useful analysis tool. From the inspection, we can see that setting  $\gamma$  between 0.6 and 0.8 could be a good choice to capture the information in the squares with the GIA algorithm. If  $\gamma = 0.7$  is selected, the corresponding treemap visualization (details in supplementary materials) is shown in [Figure 6\(b\)](#). A total of 21 states are identified in the plot by GIA and shown as colored rectangles. Relevant

(security-related) variables are enclosed in red-color states, to focus the inspection. For that, the color information is independently introduced in the visualization. Expert knowledge on the data is needed for that. For datasets where the relevance of the variables cannot be established a priori, like in the metabolomics example, the treemap visualization cannot be used.

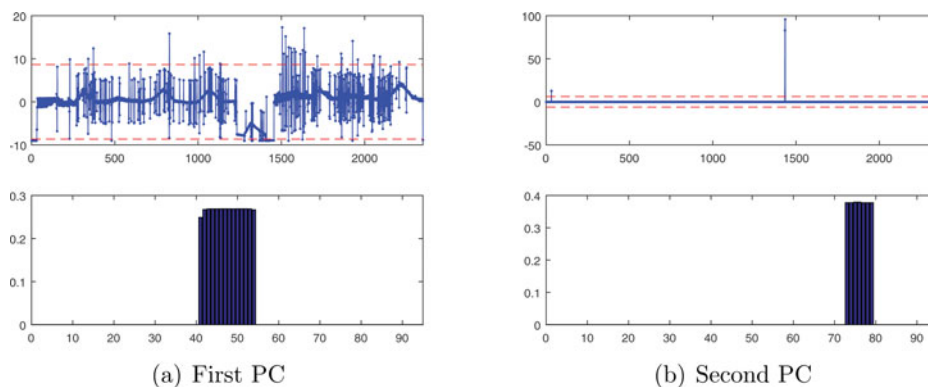
The first two PCs of the GPCA model for  $\gamma = 0.7$  are shown in [Figure 7](#). We can use independent plots for each PC since we know that PCs are not interrelated in GPCA. In this example, the GPCA loadings of the first two PCs are a direct simplification of the corresponding in the PCA model. However, in the GPCA PCs, low loadings out of the groups are filtered out.

In the network security problem, for which the current dataset is an example, the combination of the treemap and GPCA is very powerful: the treemap visualization leads the analyst to the events of interest, represented by the groups of variables in red color and/or of large size, and the GPCA provides a description of the distribution (scores) of those events in the observations, to see whether this group is the consequence of a general trend in the data or of anomalous observations. As an illustrating example, take group number 19 in the treemap of [Figure 6\(b\)](#). This group, among others, is marked in red color due to its relevance to attract the attention of the cybersecurity analyst. The group includes variables numbered from #73 to #79. These variables are counters of specific cybersecurity events detected by the firewall. The correlation in time of several cybersecurity events, correlation that conform groups in MEDA, help understanding what is happening in the network.



**Figure 6.** MEDA plot of the data for PCA with 15 PCs (a) and corresponding treemap visualization for GIA to  $\gamma = 0.7$  and a relevance qualification of the variables for the Network security example.





**Figure 7.** Line/bar plots for the first two PCs in GPCA for the Network security example: scores are shown on the top, between 99% control limits, and loadings are shown on the bottom.

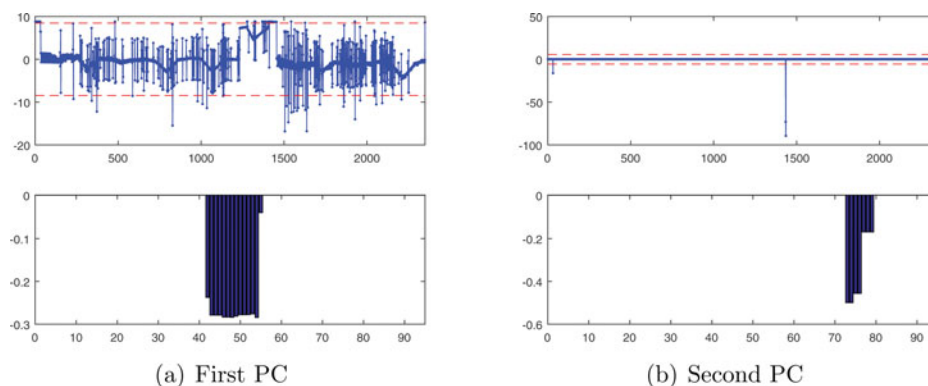
To obtain more information, the analyst may inspect the PC in GPCA related to group 19, which corresponds to the PC shown in Figure 7(b). The scores plot shows that the cybersecurity events corresponding to variables in group 19 occur in two specific time intervals, one at the beginning and one close to observation #1500. Indeed, this is confirmed by directly inspecting the data: all the variables in the group remain to 0 during the whole measured interval, except for the specific intervals highlighted in the score plot of the second PC of GPCA. Thus, a large amount of rich security information, including which cybersecurity events take place at the same time and when does this happen, has been provided with just two plots, the treemap visualization and a line score plot, which are easily interpretable.

To compare the results with that of SPCA, we set  $\lambda_2$  to  $\infty$  and restricted the first PC to 14 nonzero loadings (NZLs) and the second PC to 7 NZLs, just like in the GPCA results of Figure 7. Results are shown in Figure 8. Interestingly, the loadings selected by SPCA exactly match those identified in GPCA. This shows an interesting connection between GPCA and SPCA. However, it should be noted that the shape of SPCA loadings is somehow affected by the regularization. In particular in the second PC, the shape of NZL in GPCA (Figure 7(b), constant loadings) closely resembles the corresponding loadings of PCA (Figure 5(b), quasi-constant loadings for the same variables). This is not the case in SPCA (Figure 8(b), some loadings attain half the value than others). Apart from that, it should be highlighted that a main difference of GPCA and SPCA in this example is that while the solution by GPCA is driven from the MEDA result, there is no clue to select 14 or 7 NZLs in SPCA, and

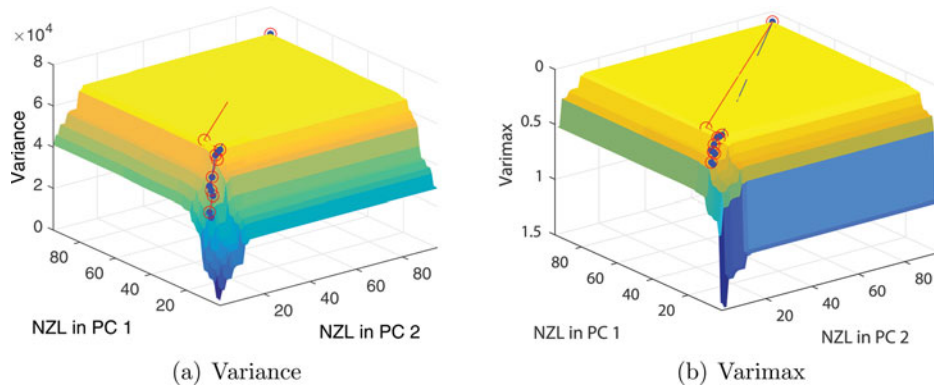
therefore a full exploration of the parameters space is necessary in the latter (see Zou and Hastie 2005).

In Figure 9, we show the result, in terms of variance and Varimax, of such an exploration of SPCA parameters. For simplicity, we set  $\lambda_2$  to infinity and make a 2D grid for different values of the number of NZLs in the first two PCs. Notice that such an experiment gets more complex with an increasing number of PCs. The surface of variance is shown in Figure 9(a) and the surface on (inverse) Varimax in Figure 9(b). The flat region on top of both figures corresponds to SPCA models very close to PCA. As the number of NZLs in either PC 1 or PC 2 gets very restrictive (below 20), there is a noticeable decrement of variance and increment of Varimax, reaching the extreme lowest value in the figures for both loading vectors restricted to one single NZL.

On top of surfaces in Figure 9(a), we have added the result (again in terms of variance and Varimax) of GPCA models for values of  $\gamma$  from 0 to 0.99. Blue large dots represent GPCA where the structure is captured from the MEDA matrix. Red circles represent GPCA where the structure is captured from the (standard) correlation matrix. GPCAs models for  $\gamma = 0$  are located at the top corner of the plot, being equivalent to standard PCA (no simplicity imposed). Slight modifications of  $\gamma$  lead GPCA very quickly to improve simplicity. However, simplicity in GPCA does not actually destroy structure to the same extent as in SPCA. This is seen in the fact that with GPCA we do not reach to the extreme values in the bottom corner. Stated otherwise, GPCA does not lead to over-simplifying the reality like SPCA does. Furthermore, GPCA provides a feedback on the choice of  $\gamma$  by visually comparing the MEDA plot and the



**Figure 8.** Line/bar plots for the first two PCs in SPCA for the Network security example: scores are shown on the top, between 99% control limits, and loadings are shown on the bottom.



**Figure 9.** Variance (a) and simplicity (Varimax) (b) surfaces for different values of nonzero loadings (NZLs) in the first two PCs in SPCA for the Network security example. On top, the same for GPCA models from MEDA (large dots) and the correlation matrix (circles) and  $\gamma$  from 0 to 0.99.

treemap visualization. Therefore, we conclude that the GPCA approach is more suitable for EDA, more simple to use and does not have the risk of over-simplifying reality like SPCA. Finally, comparing the GPCA from the correlation matrix and MEDA, we can see some slight differences, motivated by the level of noise in the former. However, for this present example, differences are not of practical relevance.

Let us compare now the results with those obtained for SSPCA. Recall that SSPCA needs of a predefined set of possible groups of variables to perform the matrix factorization. We will use the groups obtained as part of our approach from MEDA and GIA. Still, unlike GPCA, SSPCA requires to set parameters  $\alpha$  and  $\lambda_1$  in Equation (5). To choose reasonable values for these parameters, a grid of values needs to be computed. This is shown in Figure 10. This figure should be compared to that of SPCA in Figure 9. Notice that Figures 9(a) and 10(a) have the same z-scale, but Figures 9(b) and 10(b) have not for the sake of proper visualization. Comparing the figures we can see that, using the groups information, SSPCA spans the same range of variance values than SPCA but does not reach to Varimax values above 0.5, which in turns means that SSPCA, like GPCA, does not lead to oversimplifying the structure in the data. This is the result of imposing the group structure identified from MEDA and GIA.

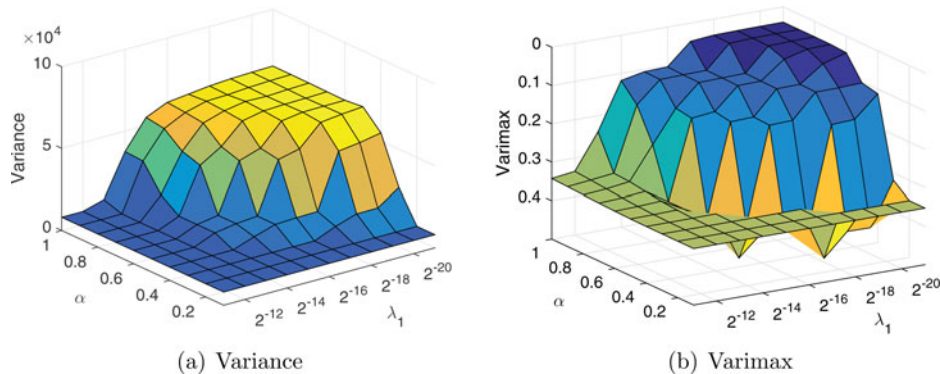
In Figure 11, the first two PCs for SSPCA are shown. The first one is similar to that for PCA, SPCA, and GPCA. SSPCA, unlike GPCA, tends to move the loadings of some of the groups of variables to zero, but the loading vector is not focused on a single group of variables. As a result, the loadings are not as easily interpretable as in GPCA. If simplicity is imposed further, by

**Table 1.** Comparison of several models in terms of variance and simplicity (Varimax).

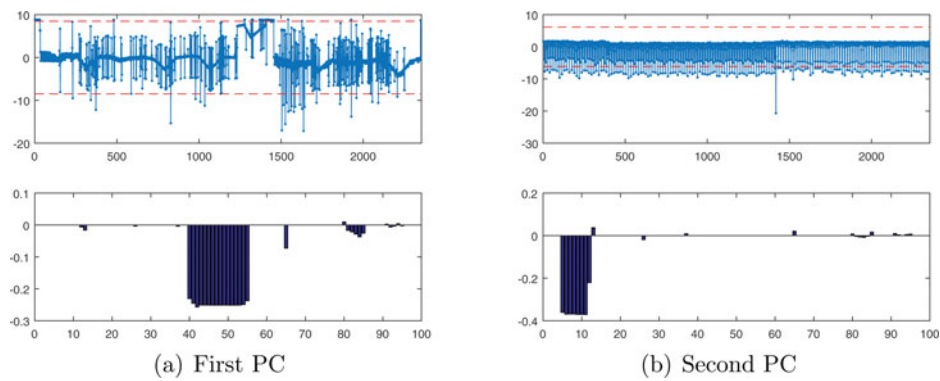
Method	c% Captured variance		
	PC 1	PC 2	PC 1&2
PCA	18.41%	11.64%	30.05%
GPCA ( $\gamma = 0.7$ )	14.54%	7.24%	21.78%
SPCA ([14, 7])	13.87%	6.07%	19.93%
SSPCA ( $\gamma = 0.7, \alpha = 0.5, \lambda_1 = 2^{-16}$ )	13.93%	7.35%	21.29%
Method	Simplicity (Varimax)		
	PC 1	PC 2	PC 1&2
PCA	0.0385	0.0569	0.0955
GPCA ( $\gamma = 0.7$ )	0.0610	0.1323	0.1933
SPCA ([14, 7])	0.0666	0.2027	0.2693
SSPCA ( $\gamma = 0.7, \alpha = 0.5, \lambda_1 = 2^{-16}$ )	0.0510	0.1204	0.1714

incrementing the value of  $\lambda_1$  or reducing  $\alpha$ , the amount of variance drops significantly. Stated otherwise, the result is very sensitive to the parameters setting with SSPCA. Also, we can see that the second PC in SSPCA is completely different to that for the other models.

To end the example, Table 1 compares the numerical results in terms of variance and simplicity (Varimax) for the models in Figures 5, 7, 8, and 11. As expected, the PCA model is the one with the highest variance and lowest Varimax index. Regarding the others, SPCA attains a higher Varimax. However, we know from the figures that the number of NZLs, and therefore the simplicity of the models, is the same for SPCA and GPCA, so



**Figure 10.** Variance (a) and simplicity (Varimax) (b) surfaces for different values of  $\alpha$  and  $\lambda_1$  in the first two PCs in SSPCA for the network security example.



**Figure 11.** Line/bar plots for the first two PCs in SSPCA for the network security example: scores are shown on the top, between 99% control limits, and loadings are shown on the bottom.

these differences in Varimax are not of practical use from the perspective of EDA. On the other hand, GPCA does a better job in capturing variance. SSPCA attains comparable results to that of GPCA, but we needed to set three parameters in the former for one in the latter.

## 5.2 Wheat Metabolomics Data

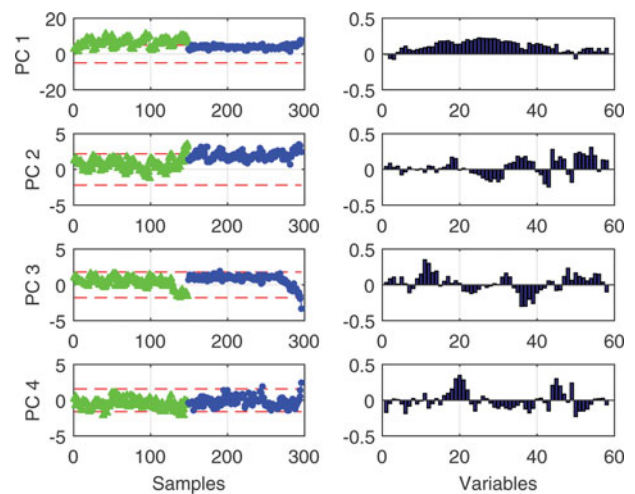
Experiments aimed at identifying changes in the metabolome of wheat (*Triticum aestivum*) induced by deoxynivalenol (DON), a mycotoxin produced by the infestant *Fusarium graminearum* and related species causing the devastating plant disease Fusarium head blight. In the study, six wheat genotypes with known varying resistance to Fusarium were treated with either DON or water control and harvested after 0, 12, 24, 48, and 96 hr after treatment. Target GC-MS profiling was used to quantify an array of 57 metabolites. The resulting data matrix  $\mathbf{X}$  has dimensions  $296 \times 57$ . Results are presented for the full dataset and not restricted to data collected at time 48 hr as in the original publication (Warth et al. 2014), to which we refer the reader for more details on the study design and experimental techniques. Data were downloaded from the MetaboLights metabolomics public data repository ([www.ebi.ac.uk/metabolights](http://www.ebi.ac.uk/metabolights), with accession number MTBLS112).

To show the potential of the GPCA for the exploration/analysis of biological data, we start by first exploring with a conventional PCA. The scores for the first four components and associated loadings are given in Figure 12. A separation between water mock control- and DON-treated samples appears along the first two components but the loadings are of difficult interpretation. In contrast to the security network example, where some sort of sparsity was inherent to the data and could be observed in the PCA loadings (see Figure 4), here, as typically happens with complex biological data, all variables have (almost) similar loadings resulting in poorly interpretable components. This problem can be alleviated by using the GPCA approach.

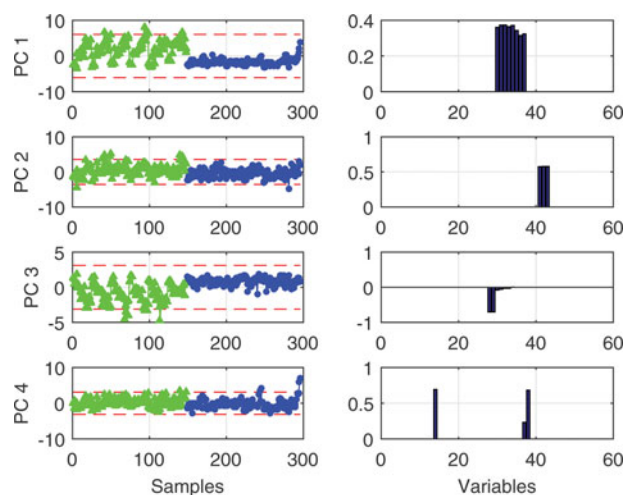
Scores and loadings for the first four GPCA components are given in Figure 13(a). Details on the definition of the MEDA map are in the supplementary material. In the first and in the third components the DON and the mock water control treatments are clearly visible together with effects due to the different harvesting time after treatment with DON. It should be

noted that the score plots are similar (although not identical) to those obtained by the standard PCA but the loading structure is much more simple with a large amount of zero loadings. This greatly facilitates interpretation: the nonzero loadings correspond mainly to amino acids (isoleucine, valine, threonine, tryptophan, tyrosine, phenylalanine, lysine, proline, methionine, glutamate, aspartate), ampholytic amino acids (phosphoric acid), or derivatives (putrescine), indicating that dysregulation of metabolic pathways in which these compounds are involved may be affected by the DON treatment. It is interesting to note that in the fourth component (and, at a less extent, in the second one) a time effect appears also for sample treated with water mock control, which cannot be attributed to DON. The nonzero loadings show contribution of amino-acids and other compounds and quinic acid in particular which could point to response to some sort of abiotic stress. In Figure 13(b), we represent the GPCA model obtained from the covariance matrix, instead of from MEDA. Due to the higher noise level in the covariance matrix, we can see that some of the groups are combined in the first PC, hardening interpretation.

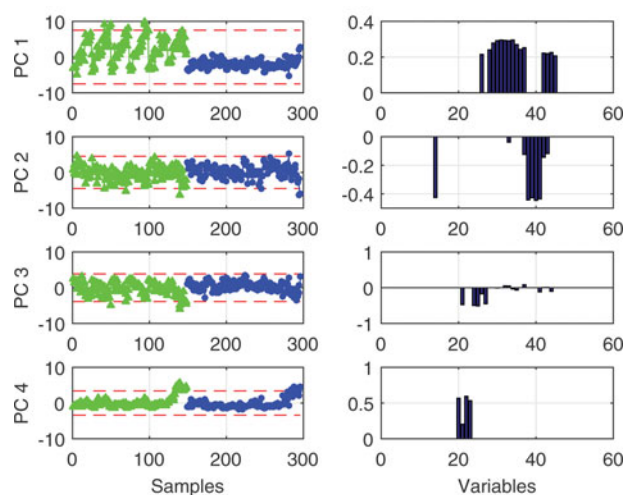
Detailed analysis of this dataset using SPCA and SSPCA and comparison with GPCA is provided in the supplementary material. Results confirm the findings in the previous example.



**Figure 12.** Scores and loadings for the first four components of a conventional PCA model for the wheat dataset. Different symbols relate to the DON (triangles) and water mock control (dots) treatments. Scores are shown on the left, between 99% control limits, and loadings are shown on the right.



(a) GPCA from MEDA



(b) GPCA from covariance

**Figure 13.** Scores and loadings for the first four components of the GPCA model for the wheat dataset: (a) groups obtained from MEDA and (b) from the covariance matrix. Different symbols relate to the DON (triangles) and water mock control (dots) treatments. Scores are shown on the left, between 99% control limits, and loadings are shown on the right.

## 6. Conclusion

In this article, we propose a new framework for matrix factorization in the context of exploratory data analysis (EDA). This framework imposes sparsity on a matrix factorization based on principal component analysis (PCA). Rather than using regularization terms, with the corresponding calibration problem in the EDA context, the structure to impose sparsity is defined in terms of correlated groups of variables found in correlation matrices or maps. With this idea, the framework is based on three new contributions: an algorithm to identify the groups of variables in correlation maps, a visualization for the resulting groups, and a matrix factorization based on PCA. Together with a method to compute correlation maps with minimum noise level, referred to as missing-data for exploratory data analysis (MEDA), these three contributions constitute a complete analysis framework. This approach has a number of advantages in the context of

EDA. First, choices during data processing can be easily made through interactive data visualization and analysis. This simplifies the application of the approach in real problems. In particular, regularization terms are avoided, and datasets for which the approach is not suitable, for instance because sparsity hypotheses are not adequate, can be identified. Second, the resulting PCs are fully interpretable in terms of scores and loadings. Finally, the fitting algorithm is straightforward, since it can be defined as a set of nested PCA iterations and a suitable deflation procedure. Using two real examples, it was shown that this approach is seamlessly applicable to certain datasets, for which the proposed factorization improves their understanding to a large extent.

## Supplementary Materials

Supplementary materials contain a description of the Group Identification Algorithm, additional information on the experiments of the paper and the code for reproducibility of results.

## Acknowledgments

José Camacho designed and programmed the GIA and GPCA algorithms, and performed and discussed the analysis of the Network Security dataset. Rafael A. Rodríguez-Gómez designed and programmed the Treemap Visualization. Edoardo Saccenti performed and discussed the analysis of the Metabolomic dataset. The authors thank Alejandro Pérez-Villegas for his help in the development of some of the figures. Anonymous reviewers are acknowledged for their useful comments. This work is partly supported by the Spanish Ministry of Economy and Competitiveness and FEDER funds through project TIN2014-60346-R and by the European Commission funded FP7 project INFECT (contract no. 305340).

## ORCID

José Camacho <http://orcid.org/0000-0001-9804-8122>  
 Rafael A. Rodríguez-Gómez <http://orcid.org/0000-0001-7159-8706>  
 Edoardo Saccenti <http://orcid.org/0000-0001-8284-4829>

## References

- Alcala, C. E., and Joe Qin, S. (2011), “Analysis and Generalization of Fault Diagnosis Methods for Process Monitoring,” *Journal of Process Control*, 21, 322–330. [2]
- Arteaga, F., and Ferrer, A. (2002), “Dealing With Missing Data in mspc: Several Methods, Different Interpretations, Some Examples,” *Journal of Chemometrics*, 16, 408–418. [1,4]
- (2005), “Framework for Regression-Based Missing Data Imputation Methods in On-Line mspc,” *Journal of Chemometrics*, 19, 439–447. [1]
- Bach, F. (2007), “Consistency of the Group Lasso and Multiple Kernel Learning,” *Journal of Machine Learning Research*, 9, 1179–1225. [3]
- Browne, M. W. (2001), “An Overview of Analytic Rotation in Exploratory Factor Analysis,” *Multivariate Behavioral Research*, 36, 111–150. [2]
- Camacho, J. (2010), “Missing-Data Theory in the Context of Exploratory Data Analysis,” *Chemometrics and Intelligent Laboratory Systems*, 103, 8–18. [1,2,6]
- (2011), “Observation-Based Missing Data Methods for Exploratory Data Analysis to Unveil the Connection Between Observations and Variables in Latent Subspace Models,” *Journal of Chemometrics*, 25, 592–600. [2]
- Camacho, J., Maciá-Fernández, G., Díaz-Verdejo, J., and García-Teodoro, P. (2014), “Tackling the Big Data 4 vs for Anomaly Detection,” *Proceedings—IEEE INFOCOM*, 500–505. [6]
- Camacho, J., Padilla, P., Daz-Verdejo, J., Smith, K., and Lovett, D. (2011), “Least-Squares Approximation of a Space Distribution for a Given

- Covariance and Latent Sub-Space,” *Chemometrics and Intelligent Laboratory Systems*, 105, 171–180. [5]
- Camacho, J., Pérez-Villegas, A., García-Teodoro, P., and Maciá-Fernández, G. (2016), “PCA-Based Multivariate Statistical Network Monitoring for Anomaly Detection,” *Computers and Security*, 59, 118–137. [6]
- Camacho, J., Pérez-Villegas, A., Rodríguez-Gómez, R. A., and nas, E. J.-M. (2015), “Multivariate Exploratory Data Analysis (meda) Toolbox for Matlab,” *Chemometrics and Intelligent Laboratory Systems*, 143, 49–57. [4,5,6]
- Cliff, N. (1966), “Orthogonal Rotation to Congruence,” *Psychometrika*, 31, 33–42. [2]
- Costello, A., and Osborne, J. (2005), “Best Practices in Exploratory Factor Analysis: Four Recommendations for Getting the Most From Your Analysis,” *Practical Assessment, Research & Evaluation*, 10, 1–9. [1,2]
- Crawford, C. B., and Ferguson, G. A. (1970), “A General Rotation Criterion and Its Use in Orthogonal Rotation,” *Psychometrika*, 35, 321–332. [2]
- de Juan, A., and Tauler, R. (2006), “Multivariate Curve Resolution (MCR) From 2000: Progress in Concepts and Applications,” *Critical Reviews in Analytical Chemistry*, 36, 163–176. [2]
- Fabrigar, L., Wegener, D., MacCallum, R., and Strahan, E. (1999), “Evaluating the Use of Exploratory Factor Analysis in Psychological Research,” *Psychological Methods*, 4, 272–299. [1,2]
- Han, J., and Kamber, M. (2006), *Data Mining: Concepts and Techniques*, Morgan Kaufmann Publishers, San Francisco, CA: Elsevier. [1]
- Jackson, J. (2003), *A User’s Guide to Principal Components*, England: Wiley-Interscience. [1,2]
- Jacob, L., Obozinski, G., and Vert, J.-P. (2009), “Group Lasso With Overlaps and Graph Lasso,” in *Proceedings of the 26th International Conference on Machine Learning, Montreal, Canada*. [3,5]
- Jenatton, R., Audibert, J.-Y., and Bach, F. (2009), “Structured Variable Selection With Sparsity-Inducing Norms,” *Journal of Machine Learning Research*, 12, 2777–2824. [3]
- Jenatton, R., Obozinski, G., and Bach, F. (2009), “Structured Sparse Principal Component Analysis,” in *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics (AISTATS)* (Vol. 9), pp. 366–373. [3,5,6]
- Jolliffe, I. (1995), “Rotation of Principal Components: Choice of Normalization Constraints,” *Journal of Applied Statistics*, 22, 29–35. [2]
- (2002), *Principal Component Analysis*, EEUU: Springer Verlag Inc. [1,2]
- Jolliffe, I., Trendafilov, N., and Uddin, M. (2003), “A Modified Principal Component Technique Based on the LASSO,” *Journal of Computational and Graphical Statistics*, 12, 531–547. [1,2,6]
- Kaiser, H. F. (1958), “The Varimax Criterion for Analytic Rotation in Factor Analysis,” *Psychometrika*, 23, 187–200. [2]
- Lin, C.-J. (2007), “Projected Gradient Methods for Nonnegative Matrix Factorization,” *Neural Computation*, 19, 2756–2779. [2]
- Mackey, L. (2008), “Deflation Methods for Sparse PCA,” in *NIPS*, pp. 1–8. [5]
- Nelson, P., Taylor, P., and MacGregor, J. (1996), “Missing Data Methods in PCA and PLS: Score Calculations With Incomplete Observations,” *Chemometrics and Intelligent Laboratory Systems*, 35, 45–65. [1,4]
- Nomikos, P., and MacGregor, J. (1995), “Multivariate SPC Charts for Monitoring Batch Processes,” *Technometrics*, 37, 41–59. [7]
- Rasmussen, M. A., and Bro, R. (2012), “A Tutorial on the Lasso Approach to Sparse Modeling,” *Chemometrics and Intelligent Laboratory Systems*, 119, 21–31. [3]
- Saccanti, E., and Camacho, J. (2015), “On the Use of the Observation-Wise k-Fold Operation in PCA Cross-Validation,” *Journal of Chemometrics*, 29, 467–478. [4,5]
- Saccanti, E., Smilde, A. K., Westerhuis, J. A., and Hendriks, M. M. W. B. (2011a), “Tracy-Widom Statistic for the Largest Eigenvalue of Autoscaled Real Matrices,” *Journal of Chemometrics*, 25, 644–652. [1]
- Saccanti, E., Westerhuis, J. A., Smilde, A. K., van der Werf, M. J., Hageman, J. A., and Hendriks, M. M. W. B. (2011b), “Simplivariate Models: Uncovering the Underlying Biology in Functional Genomics Data,” *PLoS ONE*, 6, e20747. [1]
- Sjöstrand, K., and Clemmensen, L. (2012), “Spasm: A Matlab Toolbox for Sparse Statistical Modeling,” *Journal of Statistical Software*. Available <http://www2.imm.dtu.dk/projects/spasm/references/spasm.pdf>. [6]
- Tibshirani, R. (1994), “Regression Selection and Shrinkage via the Lasso,” *Journal of the Royal Statistical Society, Series B*, 58, 267–288. [2]
- Timmerman, M. E., Kiers, H. A., and Smilde, A. K. (2007), “Estimating Confidence Intervals for Principal Component Loadings: A Comparison Between the Bootstrap and Asymptotic Results,” *British Journal of Mathematical and Statistical Psychology*, 60, 295–314. [2]
- Warth, B., Parich, A., Bueschl, C., Schoefbeck, D., Neumann, N. K. N., Kluger, B., Schuster, K., Krska, R., Adam, G., Lemmens, M. et al., (2014), “GC-MS Based Targeted Metabolic Profiling Identifies Changes in the Wheat Metabolome Following Deoxynivalenol Treatment,” *Metabolomics*, 11, 722–738. [10]
- Wold, S. (1978), “Cross-Validatory Estimation of the Number of Components in Factor and Principal Components,” *Technometrics*, 20, 397–405. [3]
- Zhang, P. (1993), “Model Selection via Multifold Crossvalidation,” *The Annals of Statistics*, 21, 299–313. [1]
- Zou, H., and Hastie, T. (2005), “Regularization and Variable Selection via The Elastic-Net,” *Journal of the Royal Statistical Society, Series B*, 67, 301–320. [3,6,8]
- Zou, H., Hastie, T., and Tibshirani, R. (2006), “Sparse Principal Component Analysis,” *Journal of Computational and Graphical Statistics*, 15, 265–286. [1,2,3,6]