Tilburg University

Model selection techniques for sparse weight-based principal component analysis

de Schipper, Niek; Van Deun, Katrijn

Published in:
Journal of Chemometrics

DOI:
10.1002/cem.3289

Publication date:
2021

Document Version
Publisher's PDF, also known as Version of record

Citation for published version (APA):
de Schipper, N., & Van Deun, K. (2021). Model selection techniques for sparse weight-based principal component analysis. *Journal of Chemometrics*, *35*(2), [e3289]. https://doi.org/10.1002/cem.3289

Received: 20 December 2019 | Revised: 13 July 2020 | Accepted: 15 July 2020

DOI: 10.1002/cem.3289

SPECIAL ISSUE - RESEARCH ARTICLE

Journal of CHEMOMETRICS WILEY

Model selection techniques for sparse weight-based principal component analysis

Niek C. de Schipper | **Katrijn Van Deun**

Department of Methodology and Statistics, Tilburg University, Tilburg, Netherlands

Correspondence
Niek de Schipper, Department of Methodology and Statistics, Tilburg University, Tilburg, Netherlands.
Email: n.c.deschipper@uvt.nl

Funding information
Nederlandse Organisatie voor Wetenschappelijk Onderzoek, Grant/Award Number: NWO-VIDI 452.16.012

Abstract

Many studies make use of multiple types of data that are collected for the same set of samples, resulting in so-called multiblock data (e.g., multiomics studies). A popular analysis framework is sparse principal component analysis (PCA) of the concatenated data. The sparseness in the component weights of these models is usually induced by penalties. A crucial factor in the use of such penalized methods is a proper tuning of the regularization parameters used to give more or less weight to the penalties. In this paper, we examine several model selection procedures to tune these regularization parameters for sparse PCA. The model selection procedures include cross-validation, Bayesian information criterion (BIC), index of sparseness, and the convex hull procedure. Furthermore, to account for the multiblock structure, we present a sparse PCA algorithm with a group least absolute shrinkage and selection operator (LASSO) penalty added to it, to either select or cancel out blocks of data in an automated way. Also, the tuning of the group LASSO parameter is studied for the proposed model selection procedures. We conclude that when the component weights are to be interpreted, cross-validation with the one standard error rule is preferred; alternatively, if the interest lies in obtaining component scores using a very limited set of variables, the convex hull, BIC, and index of sparseness are all suitable.

KEYWORDS

model selection, multiblock data, sparse PCA

1 | INTRODUCTION

Many studies make use of multiple types of data that are collected for the same set of samples, resulting in so-called multiblock data.[1] Examples include multiomics studies in which the same set of samples is profiled using different molecular assays such as mRNA expression, DNA methylation, DNA copy number, and somatic mutation data; see Wang et al.[2] for a multiomics study of breast cancer and Reinke et al.[3] for a joint analysis of six different data blocks collected from 22 individuals from an asthma cohort. Another example is multimodal studies that use different magnetic resonance imaging (MRI) modalities (e.g., anatomical, diffusion, and resting state functional magnetic resonance), for example, to study the same group of Alzheimer patients.[4] Each of the data blocks gives a partial view of the complex system under study. A full understanding of how the system works requires to understand both the drivers of the system that operate independently and those that operate only by concerted action. At the level of the data, this means that insight is needed in the relations between variables both within and between the data blocks: components of the system that work independently will show up as variation that is determined by the variables of a single block only whereas those components that work by concerted action will show up as variation that is determined jointly by variables linked throughout the blocks. A particular challenge in studying the jointly and individually determined variation is the need to automatically select variables that are of interest; not only to ease interpretation but also because data are often collected in an untargeted way and one of the primary aims of the data analysis is to hint at variables that may be key players in the process under study.[5] This is of particular relevance when using high-throughput approaches resulting in thousands of measured variables.

Following the strong rise of multiblock data in many disciplines, several integrative methods for exploratory data analysis have been put forward including clustering and dimension reduction techniques and combinations thereof; see, for example, the review by Ment et al.[6] Among the dimension reduction techniques, a number of methods that model joint and individual variation, also called common and distinctive latent variables or components, have been proposed.[7,8] Some of these methods perform variable selection[9-11] by adding a least absolute shrinkage and selection operator (LASSO) penalty to the objective function.[12] This penalty has the property to shrink the estimates to zero, some exactly with the implication that that variable does not contribute (e.g., a zero regression weight means that the predictor does not contribute to the prediction, and a zero component weight does mean that the variable does not contribute to the component). The use of such penalties that introduce zeros in the estimates is the current state of the art in variable selection. The main reasons for the popularity of penalties over subset selection methods such as best subset selection are better stability of the penalized regression model[12] and their computational efficiency (e.g., compared with calculating the solutions for all possible subsets of variables[13]). A popular framework for the analysis of multiblock data is sparse principal component analysis (PCA) (in the multiblock case also known as sparse simultaneous component analysis [SCA][14]); this framework will be the focus of the current paper.

A crucial factor in the use of penalized methods is the tuning of the regularization parameters used to give more or less weight to the penalties. In practice, the amount of sparseness in the data and the number of common and distinct components are not known beforehand. Hence, to make good use of penalized PCA approaches, model selection tools are needed to determine the strength of the LASSO and group LASSO penalties. In the context of sparse PCA, a few methods have been put forward to address this issue: these include popular solutions such as cross-validation (CV)[15] and the Bayesian information criterion (BIC)[16,17] but also less known alternatives such as the index of sparseness (IS)[18,19] and the convex hull (CHull) procedure.[20,21] A comparison of these methods in the context of sparse PCA misses.

In this paper, we will discuss and evaluate several existing model selection procedures to select proper values of the tuning parameters used in sparse PCA. Furthermore, we will extend sparse PCA with a group LASSO penalty[22] to model the common and distinct variation, by selecting at the level of the data blocks. The main focus of the paper will be on comparing several model selection procedures with respect to finding those values of the tuning parameters that yield the correct structure of the data, that is, selecting the right set of variables both in the single block setting and in the multiblock setting with common and distinct variations. The following model selection procedures, and adaptations thereof, will be discussed: CV with the eigenvector method,[15] BIC,[16,17] CHull,[21] and the IS.[18,19] We will examine these model selection procedures because they are readily available from the existing literature and can be used to estimate metaparameters for the weight-based PCA model with little to no modification of the original propositions. For sparse PCA, we will examine these model selection procedures in a simulation study with a single block of data (the most common case where all variables are assumed to represent one unit of interest). For sparse SCA, we will examine the procedures by making use of multiblock data (several sets of variables are available for the same cases with variables within one set representing a unit of interest). In the multiblock case, we will assess whether the model selection procedures produce a final model that correctly identifies the joint and individual structure of the components. In order to inform the analysis of the block structure of the variables, we implemented the group LASSO penalty in a blockwise fashion, to either select or cancel out blocks of data in an automated way.

The remainder of the paper is structured as follows: first, we will introduce sparse PCA with the LASSO penalty and its extension to the multiblock setting including a group LASSO penalty. Second, we will discuss several existing or adapted model selection procedures for tuning the LASSO and group LASSO penalty in sparse PCA. Third, we will examine these model selection procedures in the case of single and multiblock data in a simulation study. Lastly, we conclude with a discussion.

2 | SPARSE PCA FOR SINGLE AND MULTI-BLOCK DATA

In this section, we will introduce the notation and give a brief introduction to sparse PCA. Then we will discuss the extension to the multiblock setting and introduce the group LASSO penalty to account for common and distinct variation.

We will make use of the standardized notation proposed by Kiers[23]: bold lowercase and uppercases will denote vectors and matrices, respectively; the superscript "T" denotes the transpose of a vector or matrix, and a running index will range from 1 to its uppercase letter (e.g., there is a total of I cases where i runs from $i = 1, \ldots, I$).

Given is a data matrix $\mathbf{X}$ that contains the scores for I observations on J variables; we follow the convention to present the J variable scores of observation i in row i and thus $\mathbf{X}$ has size $I \times J$. PCA decomposes the data into Q components, as follows:

$$\begin{array}{l}\mathbf{X} = \mathbf{XWP}^T + \mathbf{E} \\ \text{subject to } \mathbf{P}^T\mathbf{P} = \mathbf{I},\end{array} , \tag{1}$$

where $\mathbf{W}$ is a $J \times Q$ component weight matrix, $\mathbf{P}$ is a $J \times Q$ loading matrix, and $\mathbf{E}$ is a $I \times J$ residual matrix. Often, the model is presented using the notation $\mathbf{T}$ for the component score matrix that results from the linear combinations shown explicitly in $\mathbf{XW}$. In this type of representation of the PCA model, interpretation is usually based on the loadings. Yet an attractive property of the PCA formulation in (1) is that it explicitly shows how the variables contribute to the construction of the components: the meaning of the components scores $t_{iq} = \sum_j x_{ij} w_{jq}$ can be derived by inspecting what variables are weighted together to form the components; see de Schipper and Van Deun11 for a further discussion of weights versus loadings. Automatic selection of variables that contribute to the component scores can be obtained by penalizing $\mathbf{W}$ in the least squares problem that is typically solved to obtain suitable estimates for the component weights and loadings. This leads to the following penalized least squares problem: minimize with respect to $\mathbf{W}$ and $\mathbf{P}$

$$\begin{array}{l}L(\mathbf{W},\mathbf{P}) = \|\mathbf{X} - \mathbf{XWP}^T\|_2^2 + \lambda_L \|\mathbf{W}\|_1 + \lambda_R \|\mathbf{W}\|_2^2 \\ \text{subject to } \mathbf{P}^T\mathbf{P} = \mathbf{I}\end{array} , \tag{2}$$

with $\|\mathbf{W}\|_1 = \sum_{j,r} |w_{jq}|$ the LASSO penalty (tuned by $\lambda_L \geq 0$) and $\|\mathbf{W}\|_2^2 = \sum_{j,r} w_{jq}^2$ the ridge penalty, also known as Tikhonov regularization (tuned by $\lambda^R \geq 0$). The objective function in Equation (2) has been popularized by Zou et al.24 As pointed out there, the inclusion of a ridge penalty is needed in the high dimensional setting, and this has $J > I$; the combination of LASSO and ridge is known as the elastic net.

The decomposition in (1) can be extended to the case of multiblock data by taking $\mathbf{X} = [\mathbf{X}^1, \ldots, \mathbf{X}^K]$; this is concatenating the K data blocks composed of different sets of variables for the same observation units. The decomposition of $\mathbf{X}$ has the same block structured[25] decomposition with $\mathbf{W} = [\mathbf{W}_1^T, \ldots, \mathbf{W}_K^T]^T$ and $\mathbf{P} = [\mathbf{P}_1^T, \ldots, \mathbf{P}_K^T]^T$. This multiblock formulation of PCA is known as SCA. Also in the multiblock case, $\mathbf{W}$ can be penalized to obtain sparse weights, and we will call this variant sparse SCA. When analyzing multiblock data with sparse SCA, we can search for blockwise structures in the component weights that tell us whether a component is uniquely determined by variables from one single data block (distinctive component), or whether it is a component that is determined by variables from multiple data blocks (common component). In other words, a distinctive component is a linear combination of variables of a particular data block only, whereas a common component is a linear combination of variables of multiple data blocks. An example of common and distinctive components in the situation with two data blocks is given below. The first two components are distinctive components, and the third component is a common component:

$$\mathbf{T} = [\mathbf{X}_1 \;\; \mathbf{X}_2] \begin{bmatrix} \mathbf{W}_1 \\ \mathbf{W}_2 \end{bmatrix} = [\mathbf{X}_1 \;\; \mathbf{X}_2] \begin{bmatrix} 0 & w_{1_1 2} & w_{1_1 3} \\ 0 & w_{2_1 2} & w_{2_1 3} \\ 0 & w_{3_1 2} & w_{3_1 3} \\ w_{1_2 1} & 0 & w_{1_2 3} \\ w_{2_2 1} & 0 & w_{2_2 3} \\ w_{3_2 1} & 0 & w_{2_2 3} \end{bmatrix}.$$

In total, there are $\binom{(2^K-1)+Q-1}{Q}$ possible combinations of common and distinctive components. There are 2^K-1 states for each component (minus one to exclude components with only zero weights), and each of these specific states can be assigned to each of the components where the ordering does not matter. Therefore, counting all possible common and distinct configurations for Q components takes on the form of unordered sampling with replacement.

In the work of de Schipper and Van Deun,[11] the challenge of finding the right sparse block structure for the component weight matrix was handled by an exhaustive approach, examining all possible common and distinctive structures. If the number of components and blocks is not too large, calculating all possible models is feasible. However, if the number of blocks and components is large, it is not and can be expected to yield highly variable results (as is the case with the best subset selection method for variable selection). Another option to perform selection at the level of the blocks is to add a group LASSO penalty to the PCA objective; see Jenatton et al.,[26] Deun et al.,[14] and Erichson et al.[27] for similar proposals. Let $\mathbf{w}_q^{(k)}$ denote the component weights of the variables of block k in component q. To have selection at both the level of the blocks and within blocks, the following penalized least squares criterion can be used:

$$\begin{aligned} L(\mathbf{W},\mathbf{P}) &= \|\mathbf{X}-\mathbf{XWP}^T\|_2^2+\lambda_L\|\mathbf{W}\|_1+\lambda_R\|\mathbf{W}\|_2^2+\lambda_G\sum_{q,k}\left(\sqrt{J_k}\|\mathbf{w}_q^{(k)}\|_2\right) \\ &\text{subject to } \mathbf{P}^T\mathbf{P}=\mathbf{I}. \end{aligned} \tag{3}$$

Hence, the group LASSO is tuned by $\lambda_G \geq 0$ with sufficiently large values resulting in components that are based on a linear combination of variables of just one or a few data blocks. To find estimates that minimize Equation (3) under the constraint of orthonormal component loading vectors, we rely on an alternating procedure that yields a non-increasing sequence of loss values thus converging—in practice—to a fixed point. The details of this numerical routine are discussed in Appendix (A0.1). Importantly, the numerical procedure only optimizes with respect to the component weights and loadings and thus needs fixed values for the number of components and the tuning parameters of the penalties. How to obtain suitable values for λ_L, λ_R, and λ_G is the main topic of this paper.

3 | MODEL SELECTION PROCEDURES FOR SPARSE PCA

In this paper, we will discuss several model selection techniques for the selection of the penalty tuning parameters. These methods are CV with the eigenvector method,[15] the BIC,[16,17] CHull[21], and the IS.[18,19] These model selection techniques have been previously proposed in the context of PCA, some also in the context of sparse PCA as defined here; this is with penalties on the weights. The application of these methods to sparse SCA with a group LASSO penalty is novel. A thorough comparison of these methods—for both sparse PCA and SCA—is lacking.

3.1 | CV with the eigenvector method

In the context of PCA, CV can be applied in several ways; a discussion and comparison with respect to selecting the number of components for the $\mathbf{X}=\mathbf{TP}^T$ model can be found in Bro et al.[15] In that comparison, the best performing method was CV with the eigenvector method; de Schipper and Van Deun[11] discussed the method in the context of sparse SCA to determine the value of the LASSO and ridge tuning parameters. Let $(-j)$ denote that (the coefficients of) variable j are removed. Following Bro et al. and de Schipper and Van Deun, given a number of components Q, to determine the value of a tuning parameter λ, the method then works as follows[1]:

1. Divide the sample into K folds each of size I_k.
2. Leave out the kth fold and calculate $\hat{\mathbf{W}}$ and $\hat{\mathbf{P}}$ on the remainder given a set of tuning parameters λ.
3. For the left-out samples in the kth fold $i=1,\ldots,I_k$, for variables $j=1,\ldots,J$,
   (a.) Estimate the score as $\mathbf{t}_i^{(-j)}=\mathbf{x}_i^{(-j)T}\hat{\mathbf{W}}^{(-j)}$.

[1] Note that here K is used to denote the number of folds used in the CV procedure and *not* the number of data blocks

(b.) Estimate x_{ij} as $\hat{x}_{ij} = \mathbf{t}_i^{(-j)} \hat{\mathbf{p}}_j^T$, where $\hat{\mathbf{p}}_j$ is the jth row of $\hat{\mathbf{P}}$.

(c.) Find the prediction error of the element x_{ij} by taking $e_{ij} = x_{ij} - \hat{x}_{ij}$.

4. Calculate the mean squared error of the kth fold, $\widehat{MSE}(\lambda)_k = \frac{1}{I_k J}\sum_i^{I_k}\sum_j^J e_{ij}^2$.
5. Repeat 2 and 3 for each fold and calculate the overall mean squared error,

$$\widehat{MSE}(\lambda) = \frac{1}{\sum I_k}\sum_{k=1}^{K} I_k \widehat{MSE}(\lambda)_k. \tag{4}$$

The standard error of Equation (4) is obtained by taking the sample standard deviation of $\widehat{MSE}(\lambda)_1$,..., $\widehat{MSE}(\lambda)_K$ divided by $\sqrt{K}$ (see, e.g., Gordon et al.[28]). Typically, the data are split into $K = 10$ folds of (approximately) equal size, which we will also do in the current paper. The attractive features of CV with the eigenvector method are that it is relatively fast to perform and that the estimated data $\hat{x}_{ij}$ are obtained independent of the data used to construct the model. For more detailed information, we refer the reader to Bro et al.[15]

The model with the lowest MSE is chosen as the best model. CV tends to select models that are too complex; therefore, the one standard error rule was developed.[29] The one standard error rule selects the set of tuning parameters that lead to the least complex model, still within one standard error of the best model. In this paper, we will examine the models chosen according to the best (this has the lowest MSE) and the one standard error rule.

3.2 | The BIC

Let RV be the residual variance resulting from the PCA decomposition with Q components,

$$RV = \|\mathbf{X} - \mathbf{X}\hat{\mathbf{W}}\hat{\mathbf{P}}^T\|_2^2. \tag{5}$$

Likewise, let $\widetilde{RV}$ denotes the residual variance for a given a model with a specific λ and Q. Following Guo et al.[16] and Croux et al.,[17] the BIC for a set of tuning parameters λ and given the number of components Q is then given by

$$BIC(\lambda) = \frac{\widetilde{RV}}{RV} + df(\lambda)\frac{\log(I)}{I}, \tag{6}$$

with $df(\lambda)$ the number of nonzero weights in $\hat{\mathbf{W}}$. The optimal set of λ values is then defined as the set of λ's that results in the model with the lowest BIC.

3.3 | CHull: A convex hull-based model selection method

CHull,[21] also known as L-curve (see, e.g., Hansen and O'Leary[30]), is a generic model selection procedure that aims at striking an optimal balance between the goodness-of-fit/misfit and model complexity. As stated by the authors: "The CHull procedure consists of (1) determining the convex hull of the fit-measure-by-complexity-measure plot of the models under consideration and (2) identifying the model on the boundary of the convex hull for which it is true that increasing the complexity (i.e., adding more parameters) has only a small effect on the fit measure, whereas lowering complexity (e.g., dropping parameters from the model) changes the goodness of fit (or, respectively, the misfit) substantially."[21, p. 2] In this application of CHull, we will use the variance accounted for (VAF) as a goodness-of-fit measure:

$$VAF_\lambda = \frac{\|\mathbf{X}\hat{\mathbf{W}}\hat{\mathbf{P}}^T\|_2^2}{\|\mathbf{X}\|_2^2}. \tag{7}$$

This is the goodness-of-fit measure that the authors originally used in their application of CHull as a method to determine the number of components in PCA. In our example, we will also make use of the $\widehat{MSE}(\lambda)$ as a goodness-of-fit measure; that is, the MSE values we obtain from the CV procedure as described before; see Equation (4). Our motivation is that VAF_λ is subject to overfitting and gives a downward biased estimate of the error. For the complexity measure, we use the number of nonzero weights in $\hat{\mathbf{W}}$. The models are selected using the multichull package.[31]

3.4 | Index of sparseness

According to Gajjar et al.[18] and Trendafilov et al.,[19] the IS given by

$$IS(\lambda) = VAF_{pca} \times VAF_\lambda \times \frac{df(\lambda)}{JQ}, \tag{8}$$

where $df(\lambda)$ is defined as previously; the VAF_{pca} is given by Equation (7) with $\hat{\mathbf{W}}$ and $\hat{\mathbf{P}}$ resulting from the PCA decomposition with Q components and all $\lambda = 0$; and VAF_λ is also given by Equation (7) but with $\hat{\mathbf{W}}$ and $\hat{\mathbf{P}}$ resulting from PCA with Q components and a set of regularization parameters $\lambda \geq 0$. The IS increases with goodness of fit and the sparseness of the solution. The (combination of) value(s) of the tuning parameter(s) λ that result(s) in the model with the largest IS is picked as the optimal value(s).

4 | SIMULATION STUDIES

The model selection techniques are assessed under different conditions by means of a simulation study. First, we will discuss the case of a single block of data with an unstructured sparsity pattern, and then we will discuss the case of multiblock data with structured sparsity resulting in common and distinct variation.

4.1 | Single-block data

In the simulation study, we kept the number of variables fixed to $J = 50$ and the number of components to $Q = 3$. The study included the following design factors:

- The number of observation units I: 25 , 50, and 100.
- The level of sparseness (percentage of the—in total $JQ = 150$ weights—that are equal to zero): 30% and 80%.
- The noise level: 5% and 20%.

The design is fully crossed, resulting in $3 \times 2 \times 2 = 12$ design cells. For each design cell, 50 data sets were simulated. The generation of the data is detailed in Appendix (A1.0.2). The resulting data were analyzed using an implementation of Algorithm (1) (see the Appendix) in the R software for statistical computing.[32] Algorithm (1) is freely available in R[32] and downloadable from github.com/trbKnl. Each data set was analyzed using a 50×10 grid of LASSO and ridge penalty tuning parameters. For the ridge, a sequence of 10 values equally spaced on the interval ln0 to ln500 was used and for the LASSO 50 equally spaced values on the same interval. Note that the values were back-transformed to the range 0–500. For each obtained (sparse) PCA model, the model selection statistics were calculated, and a best model was obtained for each of the six model selection methods. The chosen models according to the model selection criterion were then evaluated by looking at the following performance measures:

- The similarity between the true model matrix $\mathbf{W}$ and the estimated $\hat{\mathbf{W}}$. We use Tucker congruence between the vectorized version of $\mathbf{W}$ and $\hat{\mathbf{W}}$ to measure the similarity. The Tucker congruence (also known as cosine similarity) is defined as the cosine of the angle between two vectors. If the two vectors share no similarity, they are orthogonal, and the Tucker congruence will be 0. If the vectors are linearly dependent, that is, perfect similarity, the angle between these two vectors is 0 and the Tucker congruence will be 1.

- The percentage of correctly identified zero weights, calculated as the percentage of zero weights in the true matrix that are recovered as a zero weight in the estimated matrix.
- The percentage of correctly identified nonzero weights, calculated as the percentage of nonzero weights in the true matrix that are recovered as a nonzero weight in the estimated matrix.

4.1.1 | Results

The results of the simulation study for the single block data are summarized in Figures 1 and 2. Figure 1 shows the Tucker congruence coefficient for the different model selection methods. Usually, a threshold of 0.85 is recommended.[33] In the condition where the sparsity is 80%, only 10-fold CV, 10-fold CV with the one standard error rule, and CHull with the MSE often attain Tucker congruence values above the threshold value of acceptable similarity. Interestingly, CHull with MSE performs well, whereas this is not the case for the CHull badness-of-fit measure previously used in the literature. In the conditions were the sparsity is 30%, only 10-fold CV and 10-fold CV with the one standard error rule attain Tucker congruence values above 0.85. This means that the BIC, IS, and the CHull with VAF procedures result in models where the estimated component weights are too dissimilar from the true component weights. When the true underlying models are very sparse (the conditions with 80% of sparsity), the procedures in general perform better.

Because the Tucker congruence coefficient is relatively insensitive to whether the correct status of the weights (i.e., zero or nonzero status) is estimated back, we also inspect whether the model selection procedures result in models that select the right subset of variables. The results are summarized in Figure 2. Three patterns can be discerned. First, CV finds almost 100% of the nonzero weights yet recovers very few of the zero weights; this confirms that CV is known to yield too complex models. Second, the IS, BIC, and CHull with VAF show the opposite behavior and favor very sparse models, which results in good recovery of the zero weights at the expense of recovering very few of the nonzero weights. Third, CV with the one standard error rule yields a high percentage of recovery for both the zero and nonzero weights.

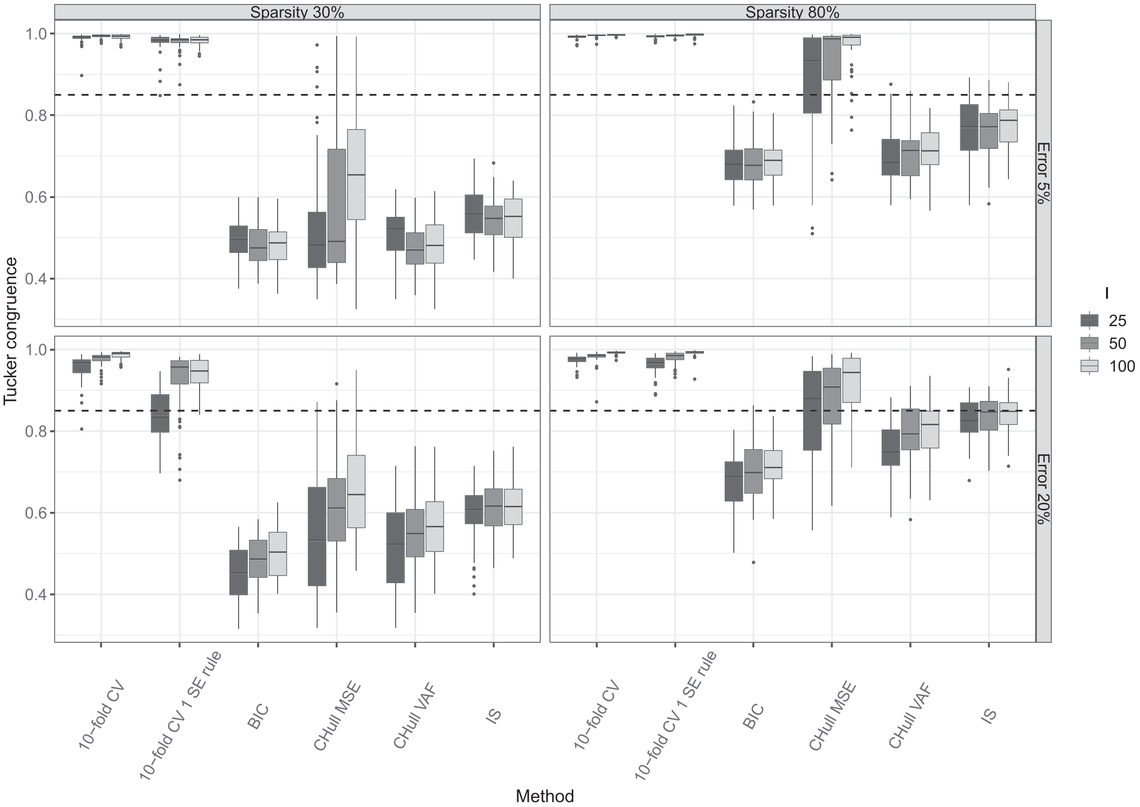

FIGURE 1 The Tucker congruence coefficient between $\mathbf{W}$ and $\hat{\mathbf{W}}$ for the various model selection procedures. The dashed line indicates a threshold value of 0.85 used as a cut-off for fair similarity. In each condition, 50 replicate data sets were used. The boxplots display the median and upper and lower quartiles

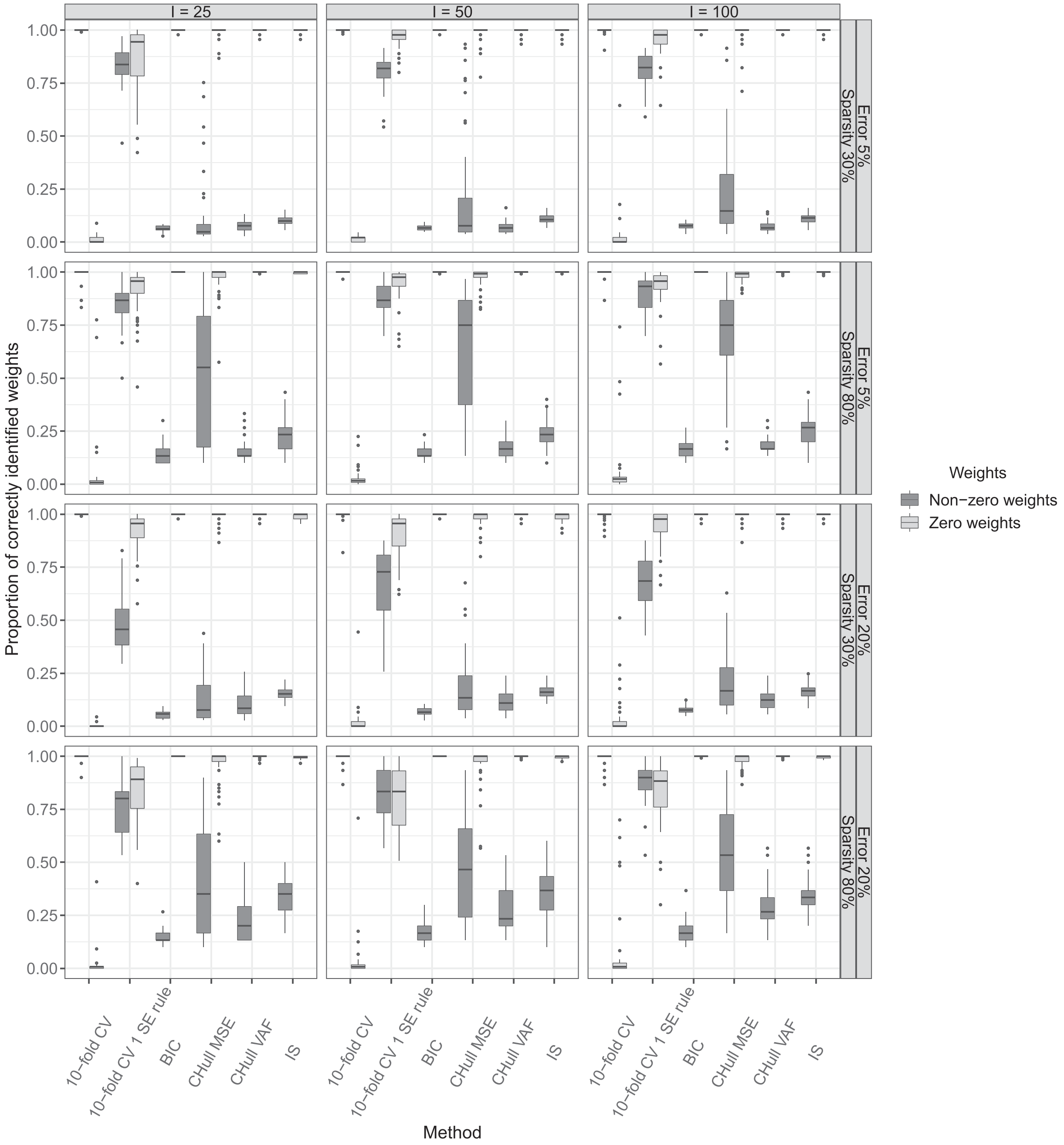

FIGURE 2 The percentage of correctly classified weights in $\hat{\mathbf{W}}$, for the various model selection procedures. For good recovery, both the percentage of correctly classified nonzero and zero weights should be high. The boxplots display the median and upper and lower quartiles

It may seem surprising that most of these model selection techniques perform badly while having showed good performance in the literature with sparse loadings (e.g., Gu et al.[34]). This can be explained by the fact that—for the reconstruction of the data—the component scores and the loadings matter while the component weights play an indirect role. The component weights enter in the construction of the scores: $\hat{\mathbf{T}} = \mathbf{X}\hat{\mathbf{W}}$. As long as the scores are recovered well, the data are reconstructed well. This is the case for the data here: Tucker congruence between $\hat{\mathbf{T}}$ and $\mathbf{T}$ is larger than 0.85 for the bulk of the selected models with each of the model selection procedures; see Figure 3. This in fact means that the component scores themselves can be retrieved rather well without the need of having to estimate that many nonzero weights. Hence, model selection procedures that balance fit with the number of nonzero coefficients result in very sparse models. This implies that few weights actually need to be estimated in order for the model to attain a good fit.

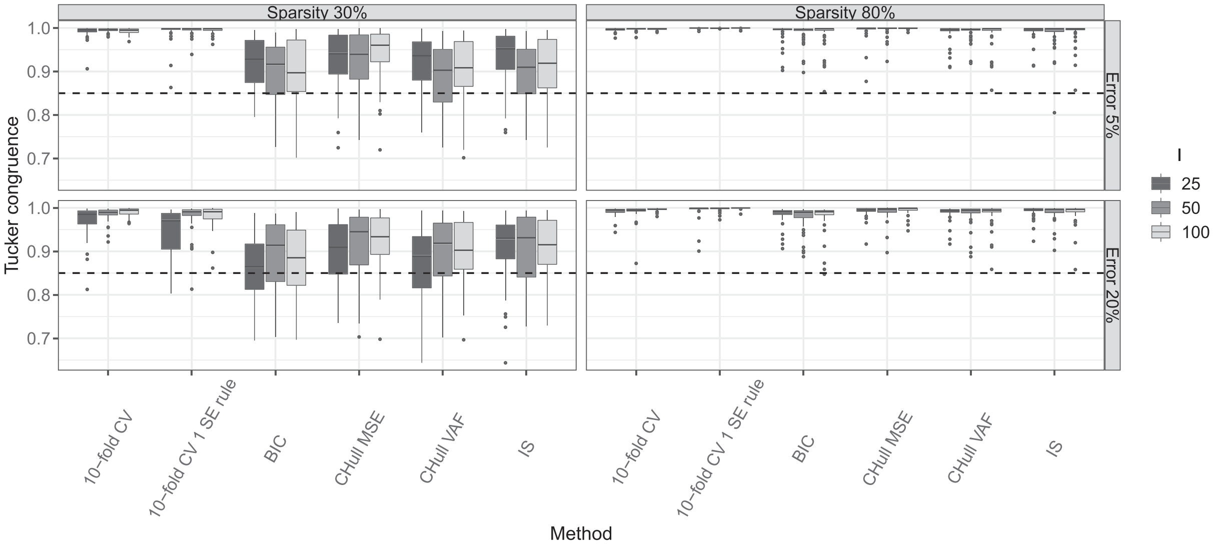

FIGURE 3 The Tucker congruence coefficient between **T** and $\hat{\mathbf{T}}$ for the various model selection procedures. The dashed line indicates a threshold value of 0.85 used as a cut-off for fair similarity. In each condition, 50 replicate data sets were used. The boxplots display the median and upper and lower quartiles

4.2 | Multiblock data

In this simulation study, we assess the performance of the model selection criteria for the case of multiblock data that have structured sparsity; that is, we assume the component weights to have a common and distinctive structure. Here, particular interest will be in evaluating whether the model selection methods recover the common and distinctive structure.

4.2.1 | Simulation study design

The data that will be analyzed in this simulation study consist of two data blocks ($\mathbf{X} = [\mathbf{X}_1 \ \mathbf{X}_2]$) each with 25 variables. The structure imposed on the data is $Q = 3$ components with two distinctive components and one common component. The study includes the following design factors:

- The number of samples I: 25 and 100.
- The level of sparseness in the nonzero blocks in the columns of **W**: 30% and 80%.
- The noise level: 5% and 20%.

The design was fully crossed, resulting in $2 \times 2 \times 2 = 8$ design cells. For each design cell, 50 data sets were generated. The details of the data generation scheme used can be found in Appendix A1.0.2. The data were analyzed with Algorithm 1 for a grid of LASSO, group LASSO, and ridge tuning parameters. The sequences of the ridge, LASSO, and group LASSO parameters are given by a sequence from 0 to 500 of length 10 on the natural log scale for each of the three tuning parameters. The chosen model according to the model selection criteria is then evaluated by looking at the following performance measures:

- Tucker congruence coefficient.
- Whether the two distinctive components are estimated back (i.e., all weights in the zero segments are estimated as zero),
- Whether the common component is estimated back (i.e., at least one nonzero weight in each data block).

4.2.2 | Results

The results of the multiblock simulation study are summarized in Figure 4 and Tables 1 and 2. In Figure 4, the Tucker congruence coefficients are displayed; these mainly show low congruence; this is below the threshold of 0.85, except for the two CV procedures. Also here, as was the case in the single block simulation, the low Tucker congruence coefficients are caused by most model selection procedures having put too many weights to zero, compared with the actual number of zero weights. Compared with the first simulation, Tucker congruence is a bit higher because the distinctive components induce higher levels of overall sparsity, meaning that the true model is more sparse and thus supportive of selection methods that favor higher levels of sparsity.

We now turn to the question whether the model selection methods recover the common and distinct components. Table 1 summarizes whether the common component is identified for the different model selection procedures, that is, whether at least one nonzero component weight within each block was retained. Table 2 summarizes whether the distinctive components are identified by the different model selection procedures (i.e., whether all weights of the block not making up the component are set to zero). Together, these tables show the same patterns previously observed for the single block simulation study: CV favors complex models, which results in defining most components as common and not finding the distinctive components; the BIC, CHull-VAF, CHull-MSE, and the IS estimate models that are too sparse and hence declare most components to be distinctive at the expense of the common components; again, only 10-fold CV with the one standard error rule accurately estimates the sparsity, both within and between blocks.

To decide on which method is best on the basis of combining the identification rates for the common and distinct components, we used sum of ranking differences (SRD) scores and summarized these in a plot. SRD scores are a consensus decision making tool for situations with multiple optimality criteria[35] (for further reading, also see Héberger[36]). Here, the scores are based on rankings of the model selection procedures on the basis of the identification rates for common (see Table 1) and distinctive (see Table 2) components. For further details on how to obtain the SRD score, we refer to Lourenco and Lebensztajn.[35] The SRD scores are summarized in Figure 5 with lower scores indicating better overall performance of the method. The gray solid curve denotes the cumulative distribution of SRD scores on the basis of a random ranking of the methods on the different optimality criteria (we relied on an approximate distribution). In the plot, the score corresponding to the 0.05 smallest SRD scores for the randomly ranked methods is indicated: this is

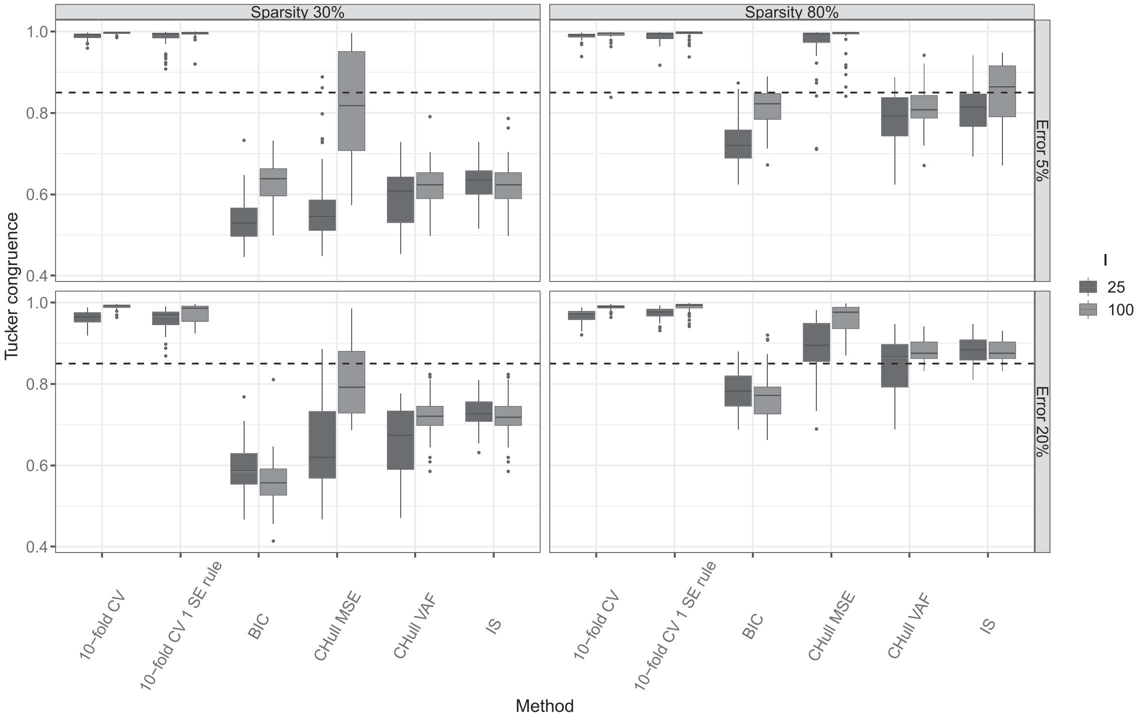

FIGURE 4 The Tucker congruence coefficient between $\mathbf{W}$ and $\hat{\mathbf{W}}$ for the various model selection procedures in the case of multiblock data. The dashed line indicates a threshold value of 0.85, which indicates fair similarity. In each condition, 50 replicated data sets were used. The boxplots display the median and upper and lower quartiles

TABLE 1 Common components identified in percentages

	Error 5%				Error 20%			
	I = 25		*I* = 100		*I* = 25		*I* = 100	
	Sparsity 30%	Sparsity 80%	Sparsity 30 %	Sparsity 80%	Sparsity 30%	Sparsity 80%	Sparsity 30%	Sparsity 80%
10-fold CV	100	98	100	96	100	100	100	100
10-fold CV 1std error	88	78	100	82	96	82	88	82
BIC	44	12	82	36	58	30	48	20
CHull-MSE	46	86	88	92	66	62	94	70
CHull-VAF	66	44	82	38	82	62	96	68
IS	76	52	82	56	96	76	96	68

Note: The percentages of times the common component were identified (there is at least one nonzero weight in each data block). The percentages are based upon 50 replicate data sets.

Abbreviations: BIC, Bayesian information criterion; CHull, convex hull; CV, cross-validation; IS, index of sparseness; MSE, mean squared error; VAF, variance accounted for.

TABLE 2 Distinctive components identified in percentages

	Error 5%				Error 20%			
	I = 25		*I* = 100		*I* = 25		*I* = 100	
	Sparsity 30%	Sparsity 80%	Sparsity 30 %	Sparsity 80%	Sparsity 30%	Sparsity 80%	Sparsity 30%	Sparsity 80%
10-fold CV	8	2	8	4	0	2	6	6
10-fold CV 1std error	82	88	98	92	94	70	90	72
BIC	98	100	96	100	98	100	98	100
CHull-MSE	92	70	78	84	84	76	80	82
CHull-VAF	82	94	94	100	68	68	90	100
IS	78	92	92	100	46	48	92	100

Note: The percentages of times the two distinctive components were found (there are no nonzero weights estimated in the zero data block). The percentages are based upon 50 replicate data sets. Abbreviations: BIC, Bayesian information criterion; CHull, convex hull; CV, cross-validation; IS, index of sparseness; MSE, mean squared error; VAF, variance accounted for.

our chosen cut-off for significance with methods having higher scores being considered to not perform consistently better on each of the optimality criteria than based on a random ranking. It can be observed that only 10-fold CV with the one standard error rule falls (just) below the cut-off, indicating that it is all-round better than the other methods. The other model selection procedures do not consistently perform better.

For the interested reader, we will provide an example of the analysis of multiblock data making use of Equation (3) in the next section.

5 | EMPIRICAL EXAMPLE: HERRING DATA

We will now provide an illustrative example where we analyze a data set on salted herring samples using sparse weight-based SCA. The data set on salted herring consists of two blocks, each containing a specific set of variables on 21 herring samples with the samples corresponding to different ripening conditions; see Bro et al.[37] and Nielsen et al.[38] for more information about the data. The first block contains chemical and physical measurements, whereas the second block consists of sensory variables. For an overview of the variable names, see Table 3.

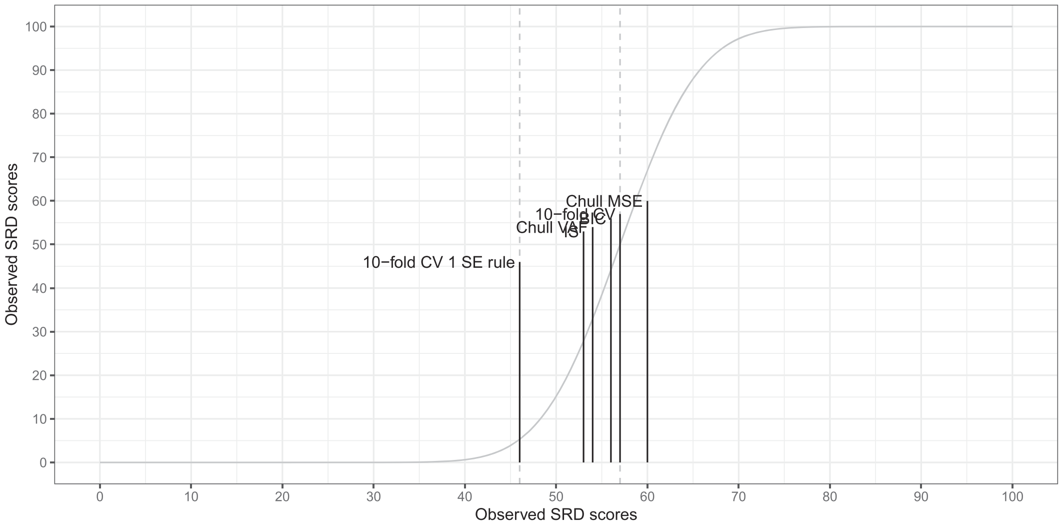

FIGURE 5 The sum of ranking differences scores of the various model selection procedures. The solid gray line indicates an approximation of the cumulative distribution function of random sum of ranking differences (SRD) scores. The vertical dashed lines indicate the 0.05 and 0.95 cut-off points. Model selection procedures having an SRD score to the right of the 0.05 cut are not statistically different, at the 0.05 level of significance, from random rankings

The analysis of multiblock data follows three steps:

- Preprocessing of the data.
- Tuning the model; selecting the metaparameters.
- Analyzing the final model; interpreting the component weights.

We will discuss each of these steps here below.

5.1 | Preprocessing of the data

Preprocessing has a large impact on the final results of the analysis and should be done according to the needs of the researchers; see Deun et al.[25] for an overview. For the herring data here, we first centered and scaled (to unit variance) the variables as we are not interested in scale differences. As the two blocks have the same number of variables, no further block scaling is needed.

5.2 | Tuning the model

Multiple metaparameters need to be tuned in order to get to a satisfactory final model. For the sparse PCA method that we use here, these are the number of components and the regularization parameters λ_L, λ_G, and λ_R. Also for the selection of the number of components, CV has been recommended.[15] Hence, two strategies can be considered, namely, tuning all parameters together or following a sequential strategy. Because of the computational burden of the simultaneous strategy, we opt for the sequential approach: first, we select the number of components, and then, given the selected number of components, we tune the LASSO and group LASSO parameters. To determine the number of components, we used 10-fold CV with the one standard error rule on each block. This resulted twice in three components; hence, we analyzed the concatenated data with the maximum number of components possible, that is, six distinctive components.

Also, the regularization parameters were tuned using 10-fold CV with the one standard error rule. More specifically, we tuned the LASSO, ridge, and group LASSO regularization parameters on a three-dimensional grid with 25, equally spaced values between 0 and 500 on the log scale; for the data here, this covers solutions ranging from no sparseness at all to all coefficients being zero. We chose a log scale because it tends to do well in practice and has been recommended elsewhere; see Friedman et al.[39, p. 10] Note that the upper bound depends on the scale of the data.

TABLE 3 MM sparse SCA: Estimated component weights for the herring data

Components	1	2	3	4	5	6
pHB	0	−0.148	−0.481	0	0	0.162
ProteinM	−0.081	0.228	0	0	0	0
ProteinB	0.148	0	0	0.176	0	0
Water	0	−0.790	0	0	0	0
AshM	0	−0.126	0.595	−0.030	0	0
Fat	0	0.394	0	0	0	0
TCAIndexM	0.320	0	0	0	0	0.484
TCAIndexB	0	0.549	0	0	0	0
TCAM	0.102	0	0	0	0	0.502
TCAB	0.464	0	0	0	0	0
Ripened	0.275	0	0	0	0	0
Rawness	0	0	−0.126	−0.491	0	−0.196
Malt	0.402	0	0	−0.001	0	0
Stockfish smell	0	0.002	0.674	0	0	0
Sweetness	0.289	0	0	0	0	−0.530
Salty	0	0	0	0.855	0	−0.109
Spice	0	0	0	0	1.000	0
Softness	0.355	0	0	0	0	0
Toughness	0.349	0	0	0	0	0
Watery	0.360	0	0	0	0	0
%VAF: per component	42.4	20.0	11.4	9.1	5.5	5.2
%VAF: total			93.9			

Note: The first block, corresponding to the first 10 variables (rows), consist of physical and chemical analyses of the herring samples measured either in brine (B) or fish muscle (M). The second block contains sensory data on the herring samples. These are the results are obtained using Algorithm 1 with the chosen tuning parameters.

5.3 | Analysis of the final mode and interpretation of the results

The component weights resulting from the final analysis (i.e., using six components and with the values of the regularization parameters set at those selected under the CV scheme) are summarized in Table 3. Note that in this case there are two distinctive components (Components 2 and 5) and four common components (Components 1, 3, 4, and 6). The component weights directly relate the components to the observed variable as $t_{iq} = \sum_j w_{jq} x_{ij}$. For comparison, we included results from a nonspare PCA of the concatenated data in Table 4 where the weights/loadings [2]are estimated using the singular value decomposition and subsequently rotated to a simple structure using varimax rotation.[40] Strikingly, there is no structuring of the components into common and distinctive components. Furthermore, components in PCA are a linear combination of all variables, and interpreting these is much more difficult than sparse SCA. Take, for example, the fifth component; in the case of PCA with varimax rotation, the component is a linear combination of mainly the variable Spice, but other variables are weighted as well with nonnegligible loadings such as TCAIndexB, Malt, and Stockfish smell. In the case of sparse SCA, the fifth component is just the variable Salty, making it the unequivocal Salty component. Importantly, the gain in interpretation obtained by imposing sparseness comes at barely any cost in terms of the variation accounted for.

As for the meaning of the components, we examine the first and most important component in terms of explained variance (42.4% variance explained) from the sparse SCA. We observe that ProteinB, TCAIndexM, TCAM, and TCAB, from the first block, together with Ripened, Malt, Sweetness, Softness, Toughness, and Watery from the second block, make up the first component. Nielsen et al.[38] note that softness (the most important quality indicator used in the

[2] Note that the loadings and the weights are the same in PCA when $\mathbf{P}^T\mathbf{P} = \mathbf{I}$

TABLE 4 Varimax: Estimated component weights for the herring data

Components	1	2	3	4	5	6
pHB	−0.107	−0.214	−0.470	0.053	−0.180	0.372
ProteinM	−0.116	0.341	−0.113	−0.036	−0.084	−0.107
ProteinB	0.191	−0.075	0.029	0.264	−0.008	0.097
Water	−0.009	−0.483	−0.076	−0.076	0.040	0.079
AshM	−0.054	−0.327	0.597	−0.138	0.105	−0.024
Fat	0.025	0.486	0.047	0.019	0.036	0.062
TCAIndexM	0.082	−0.046	−0.000	0.019	0.019	0.481
TCAIndexB	0.032	0.448	−0.027	−0.245	0.115	0.181
TCAM	0.071	0.042	−0.011	0.032	0.009	0.504
TCAB	0.222	0.120	0.080	0.180	0.002	0.160
Ripened	0.281	0.056	0.244	0.064	−0.135	0.070
Rawness	0.105	0.024	−0.110	−0.510	0.018	−0.266
Malt	0.354	0.027	0.006	−0.098	−0.117	0.108
Stockfish smell	−0.055	0.106	0.526	0.157	−0.161	0.079
Sweetness	0.502	−0.083	−0.176	0.060	−0.095	−0.353
Salty	−0.069	0.053	−0.105	0.709	0.074	−0.254
Spice	0.039	−0.012	−0.044	0.039	0.922	0.046
Softness	0.372	0.008	−0.018	0.004	0.076	0.029
Toughness	0.366	0.036	0.032	0.024	0.060	0.021
Watery	0.353	−0.101	−0.018	−0.023	0.016	0.011
%VAF: per component	31.9	19.9	11.8	12.5	5.0	12.8
%VAF: total			94			

Note: The first block, corresponding to the first 10 variables (rows), consists of physical and chemical analyses of the herring samples measured either in brine (B) or fish muscle (M). The second block contains sensory data on the herring samples. These are the results from the loadings obtained from the singular value decomposition rotated according to the varimax criterion.
Abbreviations: MM, majorization–minimization; SCA, simultaneous component analysis; VAF, variance accounted for.

herring industry) correlates with TCAM/B TCAIndexM and ProteinB, which to them makes sense because: “A correlation between these parameters and softness may be expected as muscle proteins are broken down during the ripening thus explaining the increase in low molecular nitrogenous compounds and at the same time softening of the tissue is encountered. ProteinB is mainly salt soluble muscle protein diffusing into brine from the muscle. The solubilisation of muscle proteins will therefore also probably affect the texture.”[38, p. 23] Furthermore, they note that Softness, Toughness and Watery measure the same characteristics and that TCAIndexM, TCAM and TCAB measure the same characteristics. This corresponds to the reported weights for the first component, except for the small weight of ProteinM. We could view component one as a “quality of herring” component. The first component obtained with PCA followed by varimax rotation is also the most important component (31.9% variance explained), and we may expect this to also represent quality of herring. The weights for this component in Table 4 show a somewhat similar pattern, yet there are some deviations, and the interpretation is much harder because all variables make up the first component. Furthermore, this component explains less variance compared with the first component of sparse SCA.

6 | CONCLUSION

The current paper examined several model selection procedures to select the penalty tuning parameters of sparse weight-based PCA for the unstructured case of a single block of data and of sparse weights SCA for the multiblock case having structured sparsity. Most model selection procedures that have been proposed in the sparse PCA literature did not perform well in terms of finding back the correct component weights. When analyzing single block data, the procedures led to either too complex or too sparse models. When analyzing multiblock data, it led to either identifying most components as common components and not as distinctive or not identifying common components as such. The only model selection procedure that seems to strike a good balance between model complexity and goodness of fit in both the single and multiblock cases was 10-fold CV with the eigenvector method employing the one standard error rule. It has to be noted that we did not tune the number of components together with the tuning parameters; this could be addressed in further research.

As discussed in the paper, although the weights are recovered badly, this barely affects the recovery of the component scores nor the reconstruction of the data and hints at the fact that the estimation of the weights is an ill-conditioned problem. Importantly, this means that if the goal is to obtain good estimates of the component scores, loadings, or data yet with no interest in the estimates of the component weights, proper tuning of the penalties on the weights is not needed. In this situation, an economical decision may be to select a very sparse model (e.g., as resulting from the IS, BIC, or CHull) as good estimates of the component scores can be obtained with few variables. Yet when insight in the processes at play in the data is needed, our advice is to use CV with the one standard error rule.

It has to be noted that a good solution for the component weights is in the eyes of the beholder; a situation where a very sparse solution might be desirable is when the component scores themselves are of interest and when observing new data are expensive. For newly observed cases, only the variables with nonzero component weights have to be observed to compute component scores.

ACKNOWLEDGEMENTS

This research was funded by a personal grant from the Netherlands Organisation for Scientific Research (Nederlandse Organisatie voor Wetenschappelijk Onderzoek NWO-VIDI 452.16.012) awarded to Katrijn Van Deun. The funder did not have any additional role in the study design, data collection and analysis, decision to publish, or preparation of the manuscript.

PEER REVIEW

The peer review history for this article is available at https://publons.com/publon/10.1002/cem.3289.

ORCID

Niek C. de Schipper https://orcid.org/0000-0002-8462-9791

Katrijn Van Deun https://orcid.org/0000-0002-2271-793X

REFERENCES

1. Tenenhaus A, Tenenhaus M. Regularized generalized canonical correlation analysis. *Psychometrika*. 2011;76(2):257-284. http://doi.org/10.1007/s11336-011-9206-8
2. Wang L, Xiao Y, Ping Y, et al. Integrating multi-omics for uncovering the architecture of cross-talking pathways in breast cancer. *PLoS ONE*. 2014;9(8):e104282. http://doi.org/10.1371/journal.pone.0104282
3. Reinke SN, Galindo-Prieto B, Skotare T, et al. Onpls-based multi-block data integration: a multivariate approach to interrogating biological interactions in asthma. *Anal Chem*. 2018;90(22):13400-13408. http://doi.org/10.1021/acs.analchem.8b03205
4. Schouten TM, Koini M, de Vos F, et al. Combining anatomical, diffusion, and resting state functional magnetic resonance imaging for individual classification of mild and moderate Alzheimer's disease. *NeuroImage: Clin*. 2016;11:46-51. http://doi.org/10.1016/j.nicl.2016.01.002
5. Rasmussen MA, Bro R. A tutorial on the lasso approach to sparse modeling. *Chem Intell Lab Syst*. 2012;119:21-31. http://doi.org/10.1016/j.chemolab.2012.10.003
6. Meng C, Zeleznik OA, Thallinger GG, Kuster B, Gholami AM, Culhane AC. Dimension reduction techniques for the integrative analysis of multi-omics data. *Brief Bioinform*. 2016;17(4):628-641. http://doi.org/10.1093/bib/bbv108
7. Smilde AK, Måge I, Naes T, et al. Common and distinct components in data fusion. *J Chemo*. 2017;31(7):e2900. http://doi.org/10.1002/cem.2900
8. Shu H, Wang X, Zhu H. D-CCA: a decomposition-based canonical correlation analysis for high-dimensional datasets. *J Amer Stat Asso*. 2019;115(529):292-306. https://doi.org/10.1080/01621459.2018.1543599
9. Lock EF, Hoadley KA, Marron JS, Nobel AB. Joint and individual variation explained (jive) for integrated analysis of multiple data types. *Annals Appl Stat*. 2013;7(1):523-542. http://doi.org/10.1214/12-AOAS597
10. Gu Z, Deun KV. A variable selection method for simultaneous component based data integration. *Chemometr Intell Lab Syst*. 2016;158:187-199. https://doi.org/10.1016/j.chemolab.2016.07.013
11. de Schipper NC, Van Deun K. Revealing the joint mechanisms in traditional data linked with big data. *Zeitschrift für Psychologie*. 2018;226(4):212-231. http://doi.org/10.1027/2151-2604/a000341
12. Tibshirani R. Regression shrinkage and selection via the lasso. *J R Stat Soc Ser B (Methodological)*. 1996;58(1):267-288. http://doi.org/10.1111/j.2517-6161.1996.tb02080.x

13. Bertsimas D, King A, Mazumder R. Best subset selection via a modern optimization lens. *Annals Stat*. 2016;44(2):813-852. http://doi.org/10.1214/15-AOS1388
14. Deun KV, Wilderjans TF, van den Berg RA, Antoniadis A, Mechelen IV. A flexible framework for sparse simultaneous component based data integration. *BMC Bioinformatics*. 2011;12(1):448. https://doi.org/10.1186/1471-2105-12-448
15. Bro R, Kjeldahl K, Smilde AK, Kiers HAL. Cross-validation of component models: a critical look at current methods. *Anal Bioanal Chem*. 2008;390(5):1241-1251. http://doi.org/10.1007/s00216-007-1790-1
16. Guo J, James G, Levina E, Michailidis G, Zhu J. Principal component analysis with sparse fused loadings. *J Comput Graph Stat*. 2010; 19(4):930-946. http://doi.org/10.1198/jcgs.2010.08127
17. Croux C, Filzmoser P, Fritz H. Robust sparse principal component analysis. *Technometrics*. 2013;55(2):202-214. https://doi.org/10.1080/00401706.2012.727746
18. Gajjar S, Kulahci M, Palazoglu A. Selection of non-zero loadings in sparse principal component analysis. *Chemometr Intell Lab Syst*. 2017;162:160-171. http://doi.org/10.1016/j.chemolab.2017.01.018
19. Trendafilov NT, Fontanella S, Adachi K. Sparse exploratory factor analysis. *Psychometrika*. 2017;82(3):778-794. http://doi.org/10.1007/s11336-017-9575-8
20. Timmerman ME, Kiers HAL, Ceulemans E. Searching components with simple structure in simultaneous component analysis: blockwise simplimax rotation. *Chemometr Intell Lab Syst*. 2016;156:260-272. http://doi.org/10.1016/j.chemolab.2016.05.001
21. Wilderjans TF, Ceulemans E, Meers K. CHull: a generic convex-hull-based model selection method. *Behav Res Methods*. 2012;45(1):1-15. http://doi.org/10.3758/s13428-012-0238-5
22. Yuan M, Lin Y. Model selection and estimation in regression with grouped variables. *J R Stat Soc Ser B (Stat Methodol)*. 2006;68(1):49-67. http://doi.org/10.1111/j.1467-9868.2005.00532.x
23. Kiers HAL. Towards a standardized notation and terminology in multiway analysis. *J Chemometr*. 2000;14(3):105-122. https://onlinelibrary.wiley.com/doi/abs/10.1002/1099-128X%28200005/06%2%914%3A3%3C105%3A%3AAID-CEM582%3E3.0.CO%3B2-I
24. Zou H, Hastie T, Tibshirani R. Sparse principal component analysis. *J Comput Graph Stat*. 2006;15(2):265-286. http://doi.org/10.1198/106186006x113430
25. Deun KV, Smilde AK, van der Werf MJ, Kiers HenkAL, Mechelen IV. A structured overview of simultaneous component based data integration. *BMC Bioinf*. 2009;10(1):246. https://doi.org/10.1186/1471-2105-10-246
26. Jenatton R, Obozinski G, Bach F. Structured sparse principal component analysis. *Proc Thirteenth Int Conf Artif Intell Stat*. 2010;9:366-373. http://proceedings.mlr.press/v9/jenatton10a.html
27. Erichson NB, Zheng P, Manohar K, Brunton SL, Kutz JN, Aravkin AY. Sparse principal component analysis via variable projection. *SIAM J Appl Math*. 2020;80(2):977-1002. https://doi.org/10.1137/18m1211350
28. Gordon AD, Breiman L, Friedman JH, Olshen RA, Stone CJ. Classification and regression trees. *Biometrics*. 1984;40(3):874. http://doi.org/10.2307/2530946
29. Hastie T, Tibshirani R, Friedman J. *The Elements of Statistical Learning*. New York: Springer Series in Statistics; 2009. http://doi.org/10.1007/978-0-387-84858-7
30. Hansen PC, O'Leary DP. The use of the L-curve in the regularization of discrete ill-posed problems. *SIAM J Scient Comput*. 1993;14(6): 1487-1503. http://doi.org/10.1137/0914086
31. Vervloet M, Wilderjans T, Durieux J, Ceulemans E. multichull: a generic convex-hull-based model selection method. https://CRAN.R-project.org/package=multichull, R package version 1.0.0; 2017.
32. R Core Team. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing; 2019. http://www.R-project.org/
33. Lorenzo-Seva U, Ten Berge JMF. Tucker's congruence coefficient as a meaningful index of factor similarity. *Methodology*. 2006;2(2):57-64. http://doi.org/10.1027/1614-2241.2.2.57
34. Gu Z, de Schipper NC, Deun KV. Variable selection in the regularized simultaneous component analysis method for multi-source data integration. *Scien Rep*. 2019;9(1):18608. https://doi.org/10.1038/s41598-019-54673-2
35. Lourenco JM, Lebensztajn L. Post-pareto optimality analysis with sum of ranking differences. *IEEE Trans Mag*. 2018;54(8):1-10. http://doi.org/10.1109/tmag.2018.2836327
36. Héberger K. Sum of ranking differences compares methods or models fairly. *TrAC Trends in Analytical Chemistry*. 2010;29(1):101-109. http://doi.org/10.1016/j.trac.2009.09.009
37. Bro R, Nielsen HH, Stefánsson G, Skåra T. A phenomenological study of ripening of salted herring. Assessing homogeneity of data from different countries and laboratories. *J Chemometr*. 2002;16(2):81-88. http://doi.org/10.1002/cem.691
38. Nielsen HH, Bro R, Stefansson G, Skåra T. *Salting and ripening of herring: collection and analysis of research results and industrial experience within the Nordic countries*. Copenhagen: TemaNord; 1999;578.
39. Friedman J, Hastie T, Tibshirani R. Regularization paths for generalized linear models via coordinate descent. *J Stat Softw*. 2010;33(1): 1–22. http://doi.org/10.18637/jss.v033.i01
40. Kaiser HF. The varimax criterion for analytic rotation in factor analysis. *Psychometrika*. 1958;23(3):187-200. http://doi.org/10.1007/bf02289233
41. Ten Berge JMF. *Least Squares Optimization in Multivariate Analysis*. Leiden: DSWO Press; 2005.
42. Hunter DR, Lange K. A tutorial on MM algorithms. *Amer Stat*. 2004;58(1):30-37. http://doi.org/10.1198/0003130042836

How to cite this article: de Schipper NC, Van Deun K. Model selection techniques for sparse weight-based principal component analysis. *Journal of Chemometrics*. 2020;e3289. https://doi.org/10.1002/cem.3289

APPENDIX A

Description of algorithm

In order to obtain the component weights, we need to optimize the following objective function with respect to $\mathbf{W}_c$ and $\mathbf{P}_c$:

$$\begin{aligned} L(\mathbf{W}_c,\mathbf{P}_c) &= \|\mathbf{X}_c - \mathbf{X}_c\mathbf{W}_c\mathbf{P}_c^T\|_2^2 + \lambda_L\|\mathbf{W}_c\|_1 + \lambda_R\|\mathbf{W}_c\|_2^2 \\ &+ \sum_{q,k}\left(\lambda_G\sqrt{J_k}\|\mathbf{w}_q^{(k)}\|_2 + \lambda_E\|\mathbf{w}_q^{(k)}\|_{1,2}\right), \end{aligned} \tag{A1}$$

where $\mathbf{W}_c = [(\mathbf{W}^{(1)})^T, \ldots, (\mathbf{W}^{(K)})^T]^T$, and $\mathbf{w}_q^{(k)}$ denotes the qth column from the submatrix $\mathbf{W}^{(k)}$. In order to get a minimum for (A1), we alternate between the estimation of $\mathbf{W}_c$ and $\mathbf{P}_c$. Given $\mathbf{W}_c$, we can estimate $\mathbf{P}_c$ by using procrustes rotation.[24,41] Given $\mathbf{P}_c$, we can find estimates for $\mathbf{W}_c$ by using the majorization–minimization algorithm; for a short review, see Hunter and Lange.[42]To majorize (A1), we can majorize all individual terms separately. First, we majorize $\|\mathbf{W}_c\|_1$. For the ease of simplicity, let $j = 1, \ldots, \sum_k J_k$ be index the rows of $\mathbf{W}_c$ and let $q = 1, \ldots, Q$ be index the columns of $\mathbf{W}_c$, and then we can majorize $\|\mathbf{W}_c\|_1$ as follows:

$$\begin{aligned} \lambda_L\|\mathbf{W}_c\|_1 &= \lambda_L\sum_{j,q}|w_{jq}| \leq \lambda_L\sum_{q,j}\left(\frac{1}{2}\frac{w_{jq}^2}{|\tilde{w}_{jq}|} + \frac{1}{2}|\tilde{w}_{jq}|\right) \\ &= \frac{\lambda_L}{2}\text{vec}(\mathbf{W}_c)^T\mathbf{D}_1\text{vec}(\mathbf{W}_c) + c, \end{aligned} \tag{A2}$$

where $\tilde{w}_{jq}$ is the current estimate of w_{jq}, vec() denotes the vectorized version of a matrix, $\mathbf{D}_1$ a diagonal matrix of $|w_{jq}^{-1}|$, and c contains the terms that do not depend on w_{jq} and thus can be neglected in solving the optimization problem with respect to the elements of $\mathbf{W}$. Next, we consider a majorizing function for the QK group LASSO terms,

$$\begin{aligned} \lambda_G\sum_{k,q}\sqrt{J_k}\|\mathbf{w}_q^{(k)}\|_2 &= \lambda_G\sum_{k,q}\sqrt{J_k}\left(\sum_{j=1}^{J_k}\left(w_{jq}^{(k)}\right)^2\right)^{1/2} \\ &\leq \frac{\lambda_G}{2}\sum_{k,q}\frac{\sqrt{J_k}}{2}\left(\sum_{j=1}^{J_k}\left(\tilde{w}_{jq}^{(k)}\right)^2\right)^{1/2} \\ &+ \frac{\lambda_G}{2}\sum_{k,q}\frac{\sqrt{J_k}}{2}\left(\sum_{j=1}^{J_k}\left(\tilde{w}_{jq}^{(k)}\right)^2\right)^{-1/2}\sum_{j=1}^{J_k}\left(w_{jq}^{(k)}\right)^2 \\ &= \frac{\lambda_G}{2}\sum_{k,q}\left(\mathbf{w}_q^{(k)}\right)^T\mathbf{D}_2^{(k,q)}\mathbf{w}_q^{(k)} + c, \end{aligned} \tag{A3}$$

with $\mathbf{D}_2^{(k,q)}$ being a diagonal matrix containing $\frac{\sqrt{J_k}}{2}\left(\sum_{j=1}^{J_k}\left(\tilde{w}_{jq}^{(k)}\right)^2\right)^{-1/2}$ on its diagonal for a given k and q. The sum of quadratic forms in the majorizing function in (A3) can be rewritten into one quadratic form by arranging the terms,

$$\frac{\lambda_G}{2}\sum_{k,q}\left(\mathbf{w}_q^{(k)}\right)^T\mathbf{D}_2^{(k,q)}\mathbf{w}_q^{(k)} = \frac{\lambda_G}{2}\text{vec}(\mathbf{W}_c)^T\mathbf{D}_2\text{vec}(\mathbf{W}_c) + c. \tag{A4}$$

Lastly, we will majorize the QK elitist LASSO penalty terms,

$$\begin{aligned}\lambda_E \sum_{k,q} \|\mathbf{w}_q^{(k)}\|_{1,2} &= \lambda_E \sum_{k,q} \left(\sum_{j=1}^{J_k} |w_{jq}^{(k)}| \right)^2 \\ &\leq \lambda_E \sum_{q,k} \left(\left(\sum_{j=1}^{J_k} \left| \tilde{w}_{jq}^{(k)} \right| \right) \sum_{j=1}^{J_k} \frac{\left(w_{jq}^{(k)} \right)^2}{\left| \tilde{w}_{jq}^{(k)} \right|} \right) \\ &= \lambda_E \sum_{k,q} \left(\mathbf{w}_q^{(k)} \right)^T \mathbf{D}_3^{(k,q)} \mathbf{w}_q^{(k)},\end{aligned} \tag{A5}$$

with $\mathbf{D}_3^{(k,q)}$ being a diagonal matrix containing on its $\left(\sum_{j=1}^{J_k} \left| \tilde{w}_{jq}^{(k)} \right| \right) \left(\left| \tilde{w}_{jq}^{(k)} \right| \right)^{-1}$ diagonal for a given k and q. Equation (A5) can be rewritten (the same was as Equation (A4)) into $\lambda_E \text{vec}(\mathbf{W}_c)^T \mathbf{D}_2 \text{vec}(\mathbf{W}_c)$ by arranging the terms correctly. Combining the above results, we can majorize Equation (A1) as follows:

$$\begin{aligned}L(\mathbf{W}_c, \mathbf{P}_c) = \;& \|\mathbf{X}_c - \mathbf{X}_c \mathbf{W}_c \mathbf{P}_c^T\|_2^2 + \lambda_L \|\mathbf{W}_c\|_1 + \lambda_R \|\mathbf{W}_c\|_2^2 \\ & + \sum_{q,k} \left(\lambda_G \sqrt{J_k} \|\mathbf{w}_q^{(k)}\|_2 + \lambda_E \|\mathbf{w}_q^{(k)}\|_{1,2} \right) \\ \leq \;& \|\text{vec}(\mathbf{X}_c) - (\mathbf{P}_c \otimes \mathbf{X}_c) \text{vec}(\mathbf{W}_c)\|_2^2 + \text{vec}(\mathbf{W}_c)^T \mathbf{D}_{sup} \text{vec}(\mathbf{W}_c) + c \\ = \;& Q(\mathbf{W}_c, \mathbf{P}_c),\end{aligned} \tag{A6}$$

with $\mathbf{D}_{sup} = \frac{\lambda_L}{2} \mathbf{D}_1 + \frac{\lambda_G}{2} \mathbf{D}_2 + \lambda_E \mathbf{D}_3 + \lambda_R \mathbf{I}$. Because $Q(\mathbf{W}_c, \mathbf{P}_c)$ is a quadratic function that can be easily minimized by taking the partial derivatives with respect to the elements to vec($\mathbf{W}_c$) and setting them to zero, also see Deun et al.,14 doing this gives us the following estimates for vec($\mathbf{W}_c$):

$$\text{vec}(\hat{\mathbf{W}}_c) = (\mathbf{D}_{sup} + \mathbf{I} \otimes \mathbf{X}_c^T \mathbf{X}_c)^{-1} \text{vec}(\mathbf{X}_c^T \mathbf{X}_c \mathbf{P}_c), \tag{A7}$$

with $\mathbf{I}$ being a $Q \times Q$ identity matrix. Estimates for vec($\hat{\mathbf{W}}_c$) can be found relatively efficiently by making use of the block diagonality of $(\mathbf{D}_{sup} + \mathbf{I} \otimes \mathbf{X}_c^T \mathbf{X}_c)$, meaning that the weights can be estimated per component separately,

$$\hat{\mathbf{w}}_q = (\mathbf{D}_{sup}^{(q)} + \mathbf{X}_c^T \mathbf{X}_c)^{-1} \mathbf{a}_q, \tag{A8}$$

with $\hat{\mathbf{w}}_q$ denoting the estimates of the qth component, $\mathbf{D}_{sup}^{(q)}$ denoting the part of $\mathbf{D}_{sup}$ corresponding to the qth component, and $\mathbf{a}_q$ denoting the qth column of $\mathbf{A} = \mathbf{X}_c^T \mathbf{X}_c \mathbf{P}_c$. The computation of the inverse in Equation (A8) can be costly if $\sum J_k$ is large. To make the algorithm well suited to handle a large number of variables, we can implement a coordinate descent procedure to solve for $\mathbf{W}_c$ in $Q(\mathbf{W}_c, \mathbf{P}_c)$. For the ease of notation, we will drop subscript c and let $j = 1, \ldots, Q \sum J_k$. Then, the update for an element of vec($\mathbf{W}_c$) is given by

$$\text{vec}(\hat{\mathbf{W}})_j := \frac{(\mathbf{P} \otimes \mathbf{X})_j^T \text{vec}(\mathbf{X}) - (\mathbf{P} \otimes \mathbf{X})_j^T (\mathbf{P} \otimes \mathbf{X})_{-j} \text{vec}(\mathbf{W})_{-j}}{(\mathbf{P} \otimes \mathbf{X})_j^T (\mathbf{P} \otimes \mathbf{X})_j + D_{jj}}, \tag{A9}$$

where subscript j denotes the jth element of a vector or the jth column of a matrix and $-j$ denotes the object minus the jth element or column. Making use of the orthogonality of $\mathbf{P}$ and with $j = 1, \ldots, \sum J_k$, this simplifies to

$$\hat{w}_{jq} := \frac{\mathbf{p}_q^T \mathbf{X}^T \mathbf{x}_j - \mathbf{x}_j^T \mathbf{X}_{-j} \mathbf{w}_{-jq}}{\mathbf{x}_j^T \mathbf{x}_j + D_{jj}^{(q)}}. \tag{A10}$$

With these derivations, the estimation of $\mathbf{W}_c$ can be summarized in Algorithm 1.

Although different regularizers are implemented in Algorithm 1, it is not advised to combine them all together. For example, it is not advised to combine the group LASSO and the elitist LASSO as they have opposing goals. A use case for the elitist LASSO is when common components have to be extracted; this is imposing zeros on each block in such a way that for each block segment also nonzero component weights remain.

Algorithm 1: MM algorithm for sparse SCA

```
sparse SCA ($\mathbf{X}_c, Q, \lambda$);
Input  : $\mathbf{X}_c, Q$, initialize $\widehat{\mathbf{W}}_c$ with the right singular vectors of $\mathbf{X}_c$, or a random initialization
Output: $\widehat{\mathbf{W}}_c$
while $\Delta$loss function value $> \epsilon$ do
    $\widehat{\mathbf{P}}_c \leftarrow$ procrustes rotation($\mathbf{X}_c, \widehat{\mathbf{W}}_c$)
    for $q \leftarrow 1$ to $Q$ do
        if $\sum J_k >> N$ then
            for $j \leftarrow 1$ to $\sum J_k$ do
                $\hat{w}_{jq} \leftarrow$ Equation (A10)
            end
        else
            $\hat{\mathbf{w}}_q \leftarrow$ Equation (A8)
    end
end
return $\widehat{\mathbf{W}}_c$;
```

Data generation

Single block

The data for the simulation study were generated from the following model:

$$\mathbf{X} = \mathbf{XWP}^T, \tag{A11}$$

where $\mathbf{W}$ is $J \times J$, $\mathbf{W}^T\mathbf{W} = \mathbf{I}$, and $\mathbf{W} = \mathbf{P}$. $\mathbf{W}$ is manipulated such that it contains a given level of sparsity. To achieve this, we make use of an iterative procedure that proceeds as follows. First, a random $\mathbf{W}$ matrix is generated with zero weights in the desired places. After this step, orthogonality of the columns is attempted by applying the Gram–Schmidt orthogonalization procedure *only* on the intersection of the nonzero weights between two columns of $\mathbf{W}$. When $\mathbf{W}$ only has sets of columns that contain nonoverlapping sparsity patterns, this immediately results into orthogonal columns, but when the columns in $\mathbf{W}$ have overlapping sparsity patterns, the procedure will not always lead to $\mathbf{W}^T\mathbf{W} = \mathbf{I}$ on the first pass. In such cases, multiple passes are needed in order to achieve orthogonality (additional coefficients might need to be put to zero). Some sparsity patterns are not possible, for example, an initialization where $\mathbf{W}$ does not have full column rank, or an initial set that degenerates to a linearly dependent set after multiple passes. In those cases, the algorithm fails to converge.

After a suitable $\mathbf{W}$ has been obtained, $\mathbf{\Sigma}$ can be constructed by taking $\mathbf{\Sigma}$=$\mathbf{W\Lambda W}^T$. Here, $\mathbf{\Lambda}$ is a diagonal matrix with eigenvalues of the J components underlying the full decomposition. We specify these eigenvalues such that the first Q components account for a set amount of structural variance and the remaining eigenvalues for a set amount of noise variance. The data matrices $\mathbf{X}$ having a desired underlying sparse structure and noise level can then be obtained by sampling from the multivariate normal distribution using $\mathbf{\Sigma}$ and a zero mean vector.

Multiblock

The data generation for the multiblock simulation study is the same as the data generation in the single set simulation study, except that the data have been generated with two distinctive components and one common component. We define a distinctive component as being a linear combination of variables from a particular data block and a common component as a linear combination of all data blocks. In order to achieve the desired common and distinctive structure, full block segments of zeros are inserted in the **W** matrix.