Received: 12 April 2022 | Revised: 17 June 2022 | Accepted: 24 June 2022

DOI: 10.1002/cem.3432

RESEARCH ARTICLE

Journal of CHEMOMETRICS WILEY

The canonical partial least squares approach to analysing multiway datasets—N-CPLS

Kristian Hovde Liland[1] | **Ulf Geir Indahl**[1] | **Joakim Skogholt**[1] | **Puneet Mishra**[2]

[1]Faculty of Science and Technology, Norwegian University of Life Sciences, Ås, Norway

[2]Food and Biobased Reseach, Wageningen University & Research, Wageningen, The Netherlands

Correspondence
Kristian Hovde Liland, Faculty of Science and Technology, Norwegian University of Life Sciences, Ås, Norway.
Email: kristian.liland@nmbu.no

Abstract

Multiway datasets arise in various situations, typically from specialised measurement technologies, as a result of measuring data over varying conditions in multiple dimensions or simply as sets of possibly multichannel images. When such measurements are intended for predicting some external properties, the amount of methods available is limited. The multilinear partial least squares (PLS) is among the few available options. In the present work, we generalise the canonical partial least squares framework to handle multiway data. We demonstrate the resulting multiway data analysis method to be capable of building parsimonious models, encompassing continuous and categorical responses—both single and multiple—in a unifying framework. This also enables inclusion of additional responses/information that can contribute to more parsimonious models. Finally, we achieve a considerable advantage in computational speed without sacrificing numerical precision by deflating the responses and orthogonalising scores rather than the more costly deflations of the predictor data.

KEYWORDS

canonical correlation, multiway, partial least squares, tensor multiplication

1 | INTRODUCTION

In scientific research applications, data may be generated in different forms such as univariate and multivariate signals, images and higher-order tensors. A wide range of statistical and machine learning methods are available, and the choice of approach and method depends on the need for model interpretation or the goal of achieving good predictions without requiring particular interpretations and insights into the underlying process. The two most widely used approaches for such tasks in chemometrics are principal component analysis (PCA)[1] and partial least squares (PLS) analysis.[2]

For regression and classification modelling purposes within the domain of chemometrics, several efforts have been made for improving on the traditional PLS modelling approach (e.g. earlier studies[3–5]), but most of the efforts were concentrated on two-way data and bilinear modelling.

A multiway extension of the PLS method,[6,7] known as the *multilinear PLS*[8] often referred to as the N-PLS, was published more than 25 years ago. N-PLS enables predictive subspace analysis for data with three or more modes, that is, data where the typical *samples* × *variables* matrix is extended to a *samples* × $variables_1$ × $variables_2$ × ...× $variables_d$ array. Some typical applications include data where the samples are obtained as fluorescence measurements with excitation–emission spectra,[9,10] chromatography with mass spectrometry[11] or any type of measurements combining two or more variable sets. Although available in some software toolboxes,[12] the fundamental ideas of the N-PLS have not caused a similar focus on methodological development as for traditional PLS modelling.

The purpose of the present paper is to demonstrate an extension of the PLS "toolbox" for multiway data analysis by using the ideas of the canonical partial least squares (CPLS)[13] in combination with the more recent computational improvements where predictor matrix deflations are replaced by score vector orthogonalisations. For several practical cases, we demonstrate that by including these ingredients in a multiway data analysis analysis approach, more parsimonious models (fewer components needed) can be obtained at considerably lower computational costs. In Section 2, we briefly review a special case of the tensor dot product and transposition rules, and in Section 3 we present an extension of the (two-way) CPLS into a multiway analogue of the N-PLS called N-CPLS. We briefly discuss its implications for multiway modelling and how it affects predictions and the structure in the resulting loading weights. For three real data sets, we demonstrate the building of parsimonious N-CPLS models and compare them with corresponding models obtained by the N-PLS method available in N-way toolbox.[12]

2 | TENSOR OPERATOR DEFINITIONS AND NOTATION

2.1 | The tensor dot product

The general definition of the tensor dot product is given in Arfken and Weber,[14] and we consider here only the special case needed for the N-CPLS algorithm. Let $\underline{\mathbf{A}}$ and $\underline{\mathbf{B}}$ be two tensors of dimensions $(a_1 \times ...a_n \times i_1 \times ... \times i_k)$ and $(i_1 \times ... \times i_k \times b_1 \times ... \times b_m)$, respectively. When forming the tensor dot product between $\underline{\mathbf{A}}$ and $\underline{\mathbf{B}}$, we consider the last k dimensions of $\underline{\mathbf{A}}$ and the first k dimensions of $\underline{\mathbf{B}}$ to be the inner dimensions of the two tensors, while the remaining dimensions are considered outer dimensions. Next, we unfold $\underline{\mathbf{A}}$ into a $(a_1 \cdot ... \cdot a_n) \times (i_1 \cdot ... \cdot i_k)$ matrix, and we unfold $\underline{\mathbf{B}}$ into a $(i_1 \cdot ... \cdot i_k) \times (b_1 \cdot ... \cdot b_m)$ matrix. The tensor dot product of $\underline{\mathbf{A}}$ with $\underline{\mathbf{B}}$ with k inner dimensions, denoted $\underline{\mathbf{A}} \textcircled{k} \underline{\mathbf{B}}$, is defined as the matrix product of the two unfolded matrices refolded back to a tensor of dimension $a_1 \times ... \times a_n \times b_1 \times ... \times b_m$.

Note that we above require each inner dimension of $\underline{\mathbf{A}}$ to be equal to each corresponding inner dimension of $\underline{\mathbf{B}}$, and in this case, it can be shown that the tensor dot product does not depend on the ordering of the dimensions that are collapsed in the product (as long as the matricization of $\underline{\mathbf{A}}$ and $\underline{\mathbf{B}}$ is done in a consistent manner).

To illustrate the concept, suppose we have a tensor $\underline{\mathbf{A}}$ of dimension $(3 \times 2 \times 5 \times 4 \times 6)$ and a tensor $\underline{\mathbf{B}}$ of dimension $(5 \times 4 \times 6 \times 7)$. Then the tensor dot product $\underline{\mathbf{A}}\ \underline{\mathbf{B}}$ is well-defined, and the resulting tensor is of dimension $(3 \times 2 \times 7)$. An illustration of the calculation of a tensor dot product is shown in Figure 1.

2.2 | Tensor transpose

The tensor transpose is defined as the interchangement of the specified inner and outer tensor dimensions. For a tensor $\underline{\mathbf{B}}$ of dimension $(2 \times 4 \times 3 \times 5 \times 6)$ with the $k = 2$ right facing dimensions being considered as the inner dimensions, the resulting transpose $\underline{\mathbf{B}}^{t_k}$ has dimensions $(5 \times 6 \times 2 \times 4 \times 3)$. Considering only the last dimension as inner, the tensor transpose would be $\underline{\mathbf{B}}^{t_1}$ with dimensions $(6 \times 2 \times 4 \times 3 \times 5)$. To assure notational consistency for the tensor transposition, the inner part of the tensor is given by the k right-hand side dimensions as for the first tensor (from the left) in the tensor product above.

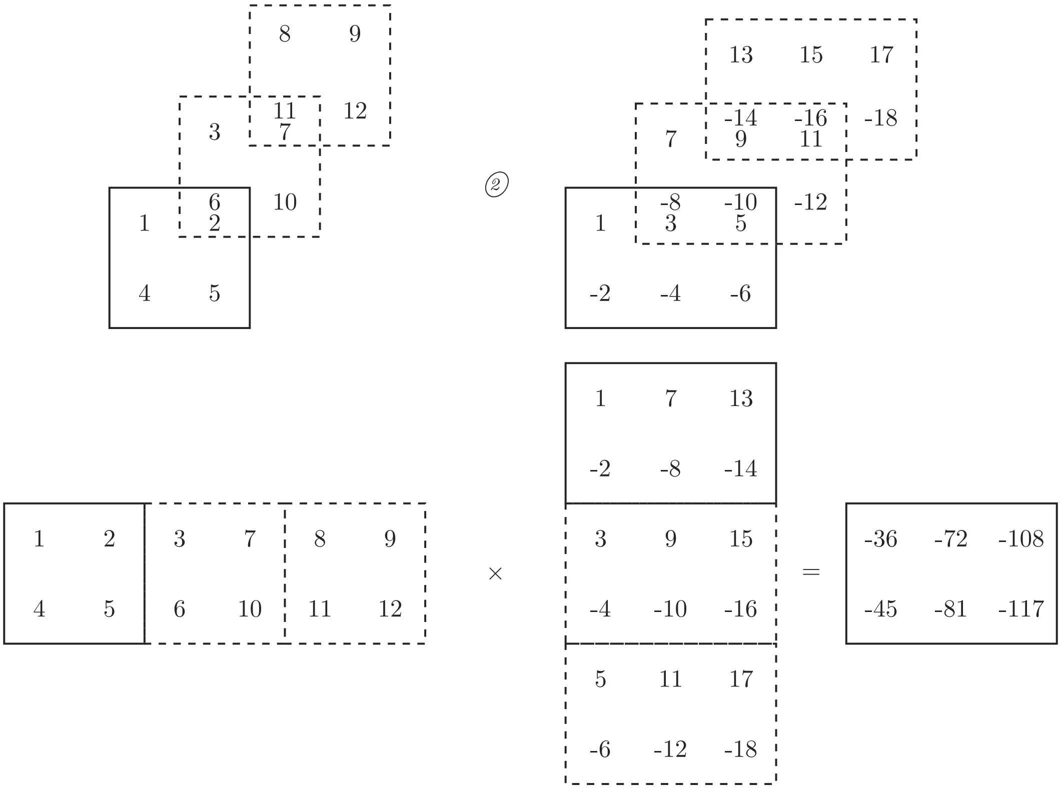

FIGURE 1 An illustration of the tensor dot product $\underline{\mathbf{A}}\,\underline{\mathbf{B}}$, where $\underline{\mathbf{A}}$ is of dimension $2 \times 2 \times 3$ and $\underline{\mathbf{B}}$ is of dimension $2 \times 3 \times 3$. To calculate the tensor dot product with two inner dimensions, $\underline{\mathbf{A}}$ is unfolded to a 2×6 matrix and $\underline{\mathbf{B}}$ to a 6×3 matrix and the matrices are subsequently multiplied. For this particular tensor dot product, the result is a 2×3 matrix, but if the result was a higher-order tensor, the resulting matrix should be refolded back into a tensor of appropriate dimension. In the illustration, the unfolded matrices are filled out along the first inner dimension and then the second, but the result would be the same if the unfolded matrix had first been filled out along the second inner dimension and then the first as long as the same filling is used for both matrices

The general tensor dot products and tensor transpose collapse to ordinary matrix product and matrix transpose in the special cases where the tensors are just matrices or vectors.

3 | METHOD

The algorithm suggested in the present work builds on the NIPALS PLS algorithm as well as the idea of utilising canonical correlation analysis (CCA) as outlined in Indahl et al.[13]

3.1 | The canonical PLS (CPLS)

In the CPLS method, a set of loading weights is calculated according to the computational elements described below, where $\mathbf{Y}$ is a matrix of 1 or more columns representing reference measurements of each sample, and the matrix $\mathbf{Y}_{\mathbf{additional}}$ represents one or more additional variable(s) containing sample-related information or measurements that may be available for the original measurements but may not be available for future measurements. (The inclusion of additional information can improve the model if it includes additional structure in the data that is not available in other responses.) The vector $\mathbf{c}$ represents the dominant vector of left canonical weights resulting from the indicated CCA.

When the matrix $\mathbf{Y}_{\mathbf{additional}}$ is available, it will contribute to a larger space for the calculation of loading weights based on information that is possibly only available for training samples (not used in prediction), while the canonical correlation tunes the prediction towards $\mathbf{Y}$ (single or multiple responses). In the case of only one column in $\mathbf{Y}$ and no available matrix $\mathbf{Y}_{\mathbf{additional}}$, the CPLS reduces to the ordinary single response PLS (PLS1).

$$\begin{aligned}
\mathbf{W}_0 &= \mathbf{X}^t[\mathbf{Y}\mathbf{Y}_{\mathbf{additional}}] && \text{- candidate loading weights}\\
\mathbf{Z}_0 &= \mathbf{X}\mathbf{W}_0 && \text{- candidate scores}\\
\mathbf{c} &\Leftarrow canoncorr(\mathbf{Z}_0, \mathbf{Y}) && \text{- canonical correlation analysis (the vector of dominant left canonical weights)}\\
\mathbf{w} &= \mathbf{W}_0\mathbf{c} && \text{- transform } \mathbf{c} \text{ by } \mathbf{W}_0 \text{ to obtain the loading weights } \mathbf{w}\\
\mathbf{w} &= \mathbf{w}/\|\mathbf{w}\| && \text{- normalise } \mathbf{w}
\end{aligned}$$

The CPLS differs from the multiresponse PLS (PLS2) in that (i) the covariance maximisation criterion is replaced by a correlation maximisation criterion, and (ii) in the calculation of the loading weights, the $\mathbf{X}$ matrix is replaced by a candidate score matrix calculated from $\mathbf{X}$ and the candidate weights. The candidate scores will typically have much fewer columns than $\mathbf{X}$, and as the efficiency of the calculation of the CCA depends on the number of columns in the two matrices, this will contribute to the computational efficiency of the algorithm. As the responses are used both in the calculation of the candidate weights and in the subsequent canonical correlation problems, CPLS is more focused on contributing to predicting $\mathbf{Y}$ from $\mathbf{X}$ for each component when compared to PLS2.[13] Further, the use of canonical correlation, in addition to maximising correlation for continuous responses, also maximises the Rayleigh quotient associated with Fisher's canonical discriminant analysis (FCDA) for dummy-coded discrete responses. This means that the component extraction, when compared with PLS2, is more focused on approximating the directions in $\mathbf{Y}$ for both regression and classification problems. We will call the method N-CPLS in all cases throughout the work, though it covers both the regression situation (as N-CPLSR) and the discriminant analysis situations (as N-CPLS-DA).

3.2 | The N-CPLS algorithm

Based on the computational elements of the CPLS, we introduce the multiway N-CPLS algorithm. In the following, we will take as a general assumption that the tensor $\underline{\mathbf{X}}$ has $d+1$ dimensions, that is, one sample dimension of length N and d variable dimensions and that we have a response matrix $\mathbf{Y}$ consistent with the sample dimension and having M columns. We assume both $\underline{\mathbf{X}}$ and $\mathbf{Y}$ to be centred along the sample dimension. In the algorithmic description of N-CPLS, tensor notation is used wherever the highest possible dimension would be a tensor, but our formulations works for two-, three-, and multi-dimensional inputs, single and multiple continuous and categorical responses, as well as with or without additional sample information. Correspondingly, we use the general tensor dot product and tensor transpose notation wherever a tensor may be involved.

The N-CPLS algorithm resembles the CPLS algorithm, and the main difference is that some matrix/vector products are replaced with tensor dot products, and the N-CPLS algorithm is modified to include an optional multilinearity step. Like the CPLS, the weights are obtained by solving a canonical correlation problem involving matrices. The dimensions of the matrices used in the canonical correlation problem in N-CPLS are the same as for CPLS, improving the computational efficiency of the algorithm compared to using $\underline{\mathbf{X}}$.

In the description of N-CPLS algorithm below, we loop over components $1,\ldots,a,\ldots,A$ but avoid using indices a for simplicity and compactness. We use j for the variable dimensions, for example, the loading weight vector for a given component along dimension j becomes $\mathbf{w}^{[j]}$.

```
for (a = 1, ..., A)                                                            - loop over A components
    $\underline{\mathbf{W}}_0 \;= \underline{\mathbf{X}}^{t_d} \textcircled{1} [\mathbf{Y}\ \mathbf{Y}_{\mathbf{additional}}]$                                     - form candidate loading weights
    $\mathbf{Z}_0 \;= \underline{\mathbf{X}} \textcircled{d} \underline{\mathbf{W}}_0$                                                - form candidate scores
    $\mathbf{Z}_0 \;:= \mathbf{Z}_0 - \mathbf{T}_A \mathbf{T}_A^t \mathbf{Z}_0$                                          - orthogonalise on accumulated score vectors †
    $\mathbf{C} \;\Leftarrow canoncorr(\mathbf{Z}_0, \mathbf{Y})$                                         - form left canonical weights of canonical analysis
    $\underline{\mathbf{W}} \;= \underline{\mathbf{W}}_0 \textcircled{1} \mathbf{C}$                                                 - form loading weights
    ______________________________________                        – start of multilinear loading weights option –
    switch$(d = dim(\underline{\mathbf{W}}))$                                               - processing alternatives according to the W-dimensionality
        $d = 1 : \mathbf{w}^{[1]} = \mathbf{w}/\|\mathbf{w}\|$                                        - $\underline{\mathbf{W}} = \mathbf{w}$ (vector): only normalise
        $d = 2 : \mathbf{w}^{[1]}, \mathbf{w}^{[2]} \Leftarrow svd(\mathbf{W})$                                  - $\underline{\mathbf{W}} = \mathbf{W}$ (matrix): use its dominant pair of left-/right singular vectors
        $d \geq 3 : \mathbf{w}^{[1]}, ..., \mathbf{w}^{[d]} \Leftarrow parafac(\underline{\mathbf{W}})$                        - $\underline{\mathbf{W}}$ (tensor): use its one-component PARAFAC model vectors
    end
    for $(j = 1, ..., d)$                                                          - loop over dimensions (optional) Δ
        $\mathbf{w}^{[j]} := \mathbf{w}^{[j]} - \mathbf{W}_A^{[j]} \mathbf{W}_A^{[j]\,t} \mathbf{w}^{[j]}$                               - orthogonalise on accumulated loading weight vectors
        $\mathbf{w}^{[j]} := \mathbf{w}^{[j]}/\|\mathbf{w}^{[j]}\|$               $\rightarrow \mathbf{W}_A^{[j]}$               - normalise loading weights and accumulate
    end
    $\underline{\mathbf{W}}^O \;= outer(\mathbf{w}^{[1]}, ..., \mathbf{w}^{[d]})$       $\rightarrow \underline{\mathbf{W}}_A^O$               - form loading weight tensor from vector outer product and accumulate
    ______________________________________                        – end of multilinear loading weights option –
    $\mathbf{t} \;= \underline{\mathbf{X}} \textcircled{d} \underline{\mathbf{W}}^O$                                                 - form score vector (for unfolded analyses set $\underline{\mathbf{W}}^O := \underline{\mathbf{W}}$, see below)
    $\mathbf{t} \;:= \mathbf{t} - \mathbf{T}_A \mathbf{T}_A^t \mathbf{t}$                                             - orthogonalise on accumulated score vectors
    $\mathbf{t} \;:= \mathbf{t}/\|\mathbf{t}\|$                            $\rightarrow \mathbf{T}_A$                   - normalise score vector and accumulate
    $\underline{\mathbf{P}} \;= \underline{\mathbf{X}}^{t_d} \textcircled{1} \mathbf{t}$                              $\rightarrow \underline{\mathbf{P}}_A$                   - form loading tensor and accumulate
    $\mathbf{q} \;= \mathbf{Y}^t \mathbf{t}$                                  $\rightarrow \mathbf{Q}_A$                   - form response loading vector and accumulate
    $\mathbf{Y} \;:= \mathbf{Y} - \mathbf{t}\mathbf{q}^t$                                                - deflate $\mathbf{Y}$
end                                                                            - end component loop
$\underline{\mathbf{R}} = \underline{\mathbf{W}}_A \textcircled{1} (\underline{\mathbf{P}}_A^{t_1} \textcircled{d} \underline{\mathbf{W}}_A)^{-1}$                                         - projections for score prediction
$\underline{\mathbf{B}} = cumsum(\underline{\mathbf{R}} \odot_1 \mathbf{Q}_A^t)$                                         - regression coefficients *
```

† Orthogonalisation of $\mathbf{Z}_0$ on previous scores is only needed when $\mathbf{Y}_{\mathbf{additional}}$ is used and can otherwise be skipped.

Δ Orthogonalisation of $\mathbf{w}_j$ on previous loading weights is optional and useful for scatter plot interpretations, but it also represents a restriction in the extraction of components. See *Unfolded analysis* below.

* The regression coefficients are formed by using element-wise multiplication along the component direction of the tensor and matrix involved, denoted $\odot_1$ and cumulative sum over the same dimension. This way, multiplying the centred input data with the regression coefficients (and adding the mean response values) will result in predictions for all components and all responses. In other words $\hat{\mathbf{Y}} = \underline{\mathbf{X}} @ \underline{\mathbf{B}} + \mathbf{1}_{N \times A} \odot_0 \overline{\mathbf{Y}}$ has dimensions (samples (N) × components (A) × responses (M)).

4 | DIMENSION-WISE LOADING WEIGHT CALCULATIONS

Due to the CPLS approach, the dimension corresponding to the number of response(s) in $\mathbf{Y}$ is collapsed by the canonical weights $\mathbf{c}$. The resulting $\underline{\mathbf{W}}$ (vector/matrix/tensor) is the only part where our algorithm requires different handling depending on dimensionality. Similar to N-PLS, for component number a, if the d-dimensional $\underline{\mathbf{W}}$ is

- a vector ($\underline{\mathbf{W}} = \mathbf{w}$): normalise it,
- a matrix ($\underline{\mathbf{W}} = \mathbf{W}$): take the first left- and right singular vectors of $\mathbf{W}$ as the required loading weights pair, $\mathbf{w}_a^{[1]}, \mathbf{w}_a^{[2]}$,

- a tensor of three or more dimensions ($\underline{\mathbf{W}} = \underline{\mathbf{W}}$): compute a PARAFAC[15] model with one component and take the resulting loadings as the required loading weights tuple, $\mathbf{w}_a^{[1]}, ..., \mathbf{w}_a^{[d]}$.

In the algorithm, the choice based on dimensions, d, is referred to as an optional multilinearity switch. In the matrix and tensor cases, after possible orhtogonalisation on previous components and renormalisation, the tensor outer product of $\mathbf{w}_a^{[1]}, ..., \mathbf{w}_a^{[d]}$ is used when producing score vectors.

4.1 | Analysis with unfolded feature dimensions

For input data of three or more dimensions, the N-CPLS algorithm above is multilinear in the sense that the loading weight tensor $\underline{\mathbf{W}}^O$ is composed by an outer product of loading weight vectors for each variable dimension, that is, a set of $\mathbf{w}^{[j]}$. This enables subsequent plotting of the loading weights for each of the variable dimension, but it also represents a restriction on the amount of component variation that can be captured, due to $\underline{\mathbf{W}}^O$ having fewer estimable weights than $\underline{\mathbf{W}}$. Defining N_j as the number of variables in dimension j, the number of weights are $\sum_{j=1}^{d} N_j$ versus $\prod_{j=1}^{d} N_j$, respectively.

An alternative is to bypass the entire switching part of the algorithm and rather consider $\underline{\mathbf{W}}^O = \underline{\mathbf{W}}$. This is equivalent to first unfolding the multiway measurements of each sample in $\underline{\mathbf{X}}$ and using CPLS directly, that is, stopping after step (C) in Figure 2 and sacrificing the set of $\mathbf{w}^{[j]}$'s otherwise available for plotting. This type of approach provides a less restricted solution as the loading weights now become linear combinations of all combinations of variables along the different dimensions, rather than being restricted to just one variable dimension at a time. Alternatively, one might consider postprocessing of the component-wise loading weight tensors, $\underline{\mathbf{W}}$, to obtain pseudo $\mathbf{w}^{[j]}$'s. However, this will not yield an exact representation of the component and seems less valuable for interpretations.

4.2 | N-PLS

The original N-PLS and its main implementation differ from N-CPLS in some key areas. Four of these choices have a direct impact on the modelling done in this work. (1): N-PLS uses an iterative procedure for component estimation,

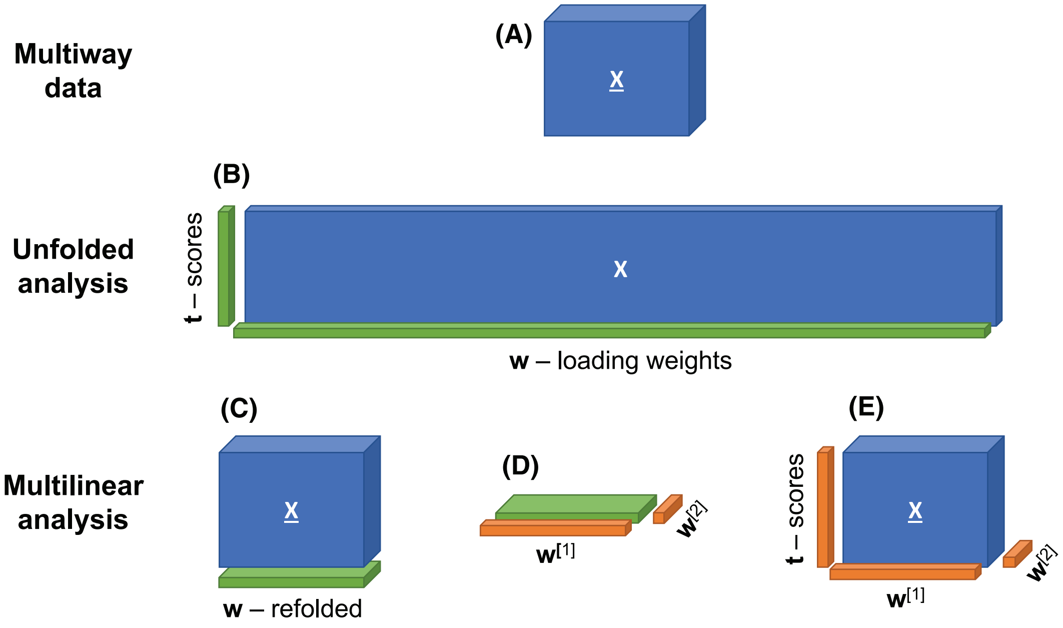

FIGURE 2 Illustration of calculation of loading weights. Using multiway input data (A) in unfolded format B, unfolded loading weights are produced with canonical partial least squares (CPLS). To perform multilinear analysis, the loading weights are folded to the shape of the variable dimensions (C) and multilinear loading weights are estimated (D) corresponding to the dimensions of the original data (E). Scores are computed by multiplying together data and loading weights in (B) or (E)

repeatedly going back and forth between loading weights, scores, Y-loadings, and Y-scores until convergence, where N-CPLS extracts each component in a faster, CCA-based calculation. For single response modelling without use of additional sample information through $\mathbf{Y}_{\mathbf{additional}}$ or explicit loading weight orthogonalisation, these approaches are equivalent, while all other cases lead to differing maximisation criteria. (2): In N-PLS, data are not automatically centred, while centring is done by default in N-CPLS. For predictive modelling, it is usually most efficient to centre the data, while for multilinear models like PARAFAC it is more important to enable recovery of true chemical components from raw data. For N-PLS, the choice has been made to follow the PARAFAC tradition, while our experience is that lack of centring can be at the expense of explained response variance. Thus, we have centred all data used with N-PLS in this work. (3): In N-PLS, the deflation of $\underline{\mathbf{X}}$ is based on loading weights, that is, $\mathbf{TG}\underline{\mathbf{W}}^{Ot}$ where $\mathbf{G}$ is a Tucker3 core array (see Smilde et al.[16]), whereas in N-CPLS the deflation follows the more common loadings-based deflation scheme, equivalent to using $\mathbf{T}\underline{\mathbf{P}}^{t}$. As a consequence, N-PLS essentially deflates in the row-/sample direction of the data, while N-CPLS essentially deflates the column-/variable direction. (4): N-PLS imposes no orthogonality on scores or loading weights, while N-CPLS components are orthogonal for scores and loading weight tensors. This, however, does not affect predictions, unless also the dimension-wise loading weights are orthogonalised.

N-PLS has two additional properties that have not been included in the N-CPLS: (i) the possibility of having multiway responses and (ii) a strategy for handling of missing data, both in $\underline{\mathbf{X}}$ and $\underline{\mathbf{Y}}$. Allowing for multiway $\underline{\mathbf{Y}}$ in N-CPLS would require a less obvious approach to the canonical correlation step, while the handling of missing data requires a compensation/bookkeeping strategy in the tensor dot product computations. Extensions to cope with such data is feasible also for the N-CPLS approach but have not been prioritised for this work mainly addressing applications with complete spectral and imaging data.

In the below examples, we have used the implementation of N-PLS from the N-way MATLAB toolbox.[12]

5 | DATA SETS

Descriptions of the data sets considered are specified below. For each data set, every second sample, counting from the first, was used for training and the remainder for testing. Different data splits were also tested, for example, in a ratio of 2:1 for training and testing, respectively. Due to some heterogeneity in the samples, variations in the resulting plots were observed, but the main patterns and order of methods with regard to performance were retained for the various splits considered.

5.1 | Sugar, fluorescence

The sugar data[17] are an example of data typically modelled by multilinear methods because of its trilinear structure. Fluorescence has been measured on 268 samples as excitation-emission spectra with seven excitation wavelengths from 230 to 340 nm and 571 emission wavelengths from 275 to 560 nm. Two quality parameters: *colour* (derived from absorbance and used as an indicator for miscolouring of the sugar) and *ash content* (determined by conductivity and is used as an indicator of inorganic impurities) were measured. In addition, the time of measurements is included in three forms: *year*, *month* and *time of day* (morning, afternoon, night). See the data set webpage (http://www.models.life.ku.dk/Sugar_Process) for more details. This dataset is included as an example where multilinear modelling makes particular sense. It is also rich in responses, enabling modelling alternatives with both multiple primary responses and with utilisation of additional responses.

5.2 | Milk, NIR

The milk data are an example of one-dimensional spectra acquired for each of 296 samples in two different spectral modes. The wavelength range 1550 to 1950 nm was used with for both the transmission and reflection mode, and the resulting matrices have size 201 × 2. Reference measurements were acquired for fat and protein content as a percentage of the sample weight for each sample. See the original publication[18] for details. This dataset is included as an example where orthogonalisation of loading weights and use of additional responses in modelling has a profound effect on model parsimony (the number of required extracted components).

5.3 | ORL faces, images

The ORL faces data set (which originated from a database formerly known as 'The ORL Database of Faces'; see https://cam-orl.co.uk/facedatabase.html) is chosen as a demonstration of data that are multiway in dimensionality but do not have a multilinear structure. It also demonstrates the use of N-CPLS for classification. The data set consists of 400 grey scale images of size 92 × 112 pixels acquired from 40 subjects represented by 10 frontal face photographs each under varying lighting conditions and time points with different facial expressions and accessories (glasses, facial hair, etc.). See also Liland and Indahl[19] for an unfolded analysis using power partial least squares discriminant analysis. All classes were represented with the same frequency in both the training and test set. The task is to identify the subjects of the unseen (new) images from a classification model obtained by dummy regression with one-hot encoded responses and class assignments chosen according to the index of the largest among the predicted responses.

6 | RESULTS

6.1 | Unfolded versus multilinear modelling —Sugar data

We first demonstrate how alternative restrictions in the N-CPLS affect modelling with the sugar data. In the analyses, every second sample (2, 4, ..., 268) is used as test samples, and curves of R^2 show test set performance. Figures 3 and 4

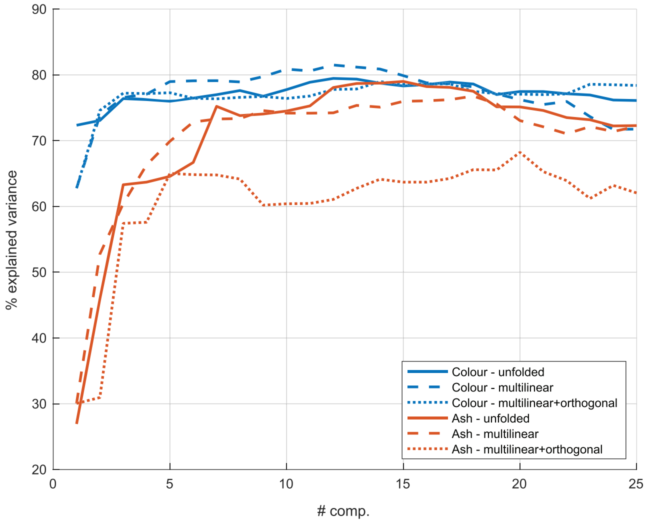

FIGURE 3 Explained variance of test set predictions for N-CPLS on *sugar* data with unfolded analysis (solid line) and trilinear analysis without (dashed line) and with (dotted line) $\mathbf{w}^{[j]}$ orthogonalisation on previous components. Two responses, *colour* (blue) and *ash* (red), are modelled simultaneously

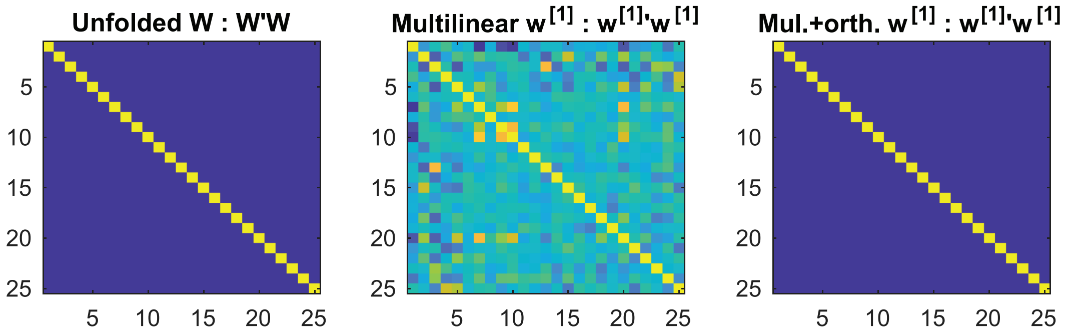

FIGURE 4 Patterns of variations in $\mathbf{W}^t\mathbf{W}$ for multiway canonical partial least squares (CPLS) with unfolded analysis and in $\mathbf{w}^{[1]t}\mathbf{w}^{[1]}$ for trilinear analysis without and with $\mathbf{w}^{[j]}$ orthogonalisation on previous components (*sugar* data). Dark colour (blue) indicates zero, while light colour (yellow) indicates one as inner product

show the prediction results for the unfolded analysis (least restricted), the (tri-/) multilinear analysis (more restricted), and the multilinear analysis with orthogonalisation of $\mathbf{w}^{[j]}$ on previous components (most restricted). The models predict *colour* and *ash*. For the sugar data, it is evident that the orthogonalisation on previous components is the most severe modelling restriction as the explained variance on the test set never reaches the same level as the other models. Between the unfolded and nonorthogonalised multilinear analysis there is considerably less difference, each explaining one of the responses best by a small margin.

Inspection of inner products is associated with $\mathbf{W}^t\mathbf{W}$ from the unfolded modelling and $\mathbf{w}^{[1]t}\mathbf{w}^{[1]}$ from the multilinear modelling. We note that the least restricted and the most restricted models both result in orthogonal loading weights but obviously not for the medium restriction (multilinear without orthogonality). Regarding the modelling with respect to the unfolded data, it should be noted that orthogonality is obtained in the unfolded, combined variable space (and not for each dimension/mode). The explicitly orthogonalised loading weights for the most restrictive model is of course orthogonal both in the separate modes and in the unfolded space since outer products of orthonormal vectors are themselves orthonormal.

Figure 5 shows the loading weights corresponding to the first two components for the multilinear/trilinear model with loading weight orthogonalisation based on the *sugar* data. On the left-hand side are dimension-wise loading weights, and on the right-hand side are their outer product for the first two components (sign flipped to match raw spectra) often called excitation-emission landscapes. As mentioned above, the one-to-one correspondence between dimension-wise loading weights and the loading weights used for further computation of score vectors is only available for the multilinear approach, and the orthogonalisation makes components less ambiguous with respect to interpretations. If variable/wavelength contributions along each dimension/mode is a concern, modelling based on unfolded data should be avoided.

6.2 | N-PLS versus N-CPLS model building —Sugar and milk data

Here, we focus on modelling of a single response and concentrate on prediction of the chemical property *ash* as our primary response $\mathbf{Y}$. In addition we use two nonchemical easily available properties as additional responses ($\mathbf{Y}_{\mathbf{additional}}$): *colour* (numeric) and *year* of measurements (dummy-coded by one-hot encoding). The year variable is included as it may correlate with other variables that can affect the measurements, such as change in equipment, raw materials, etc.

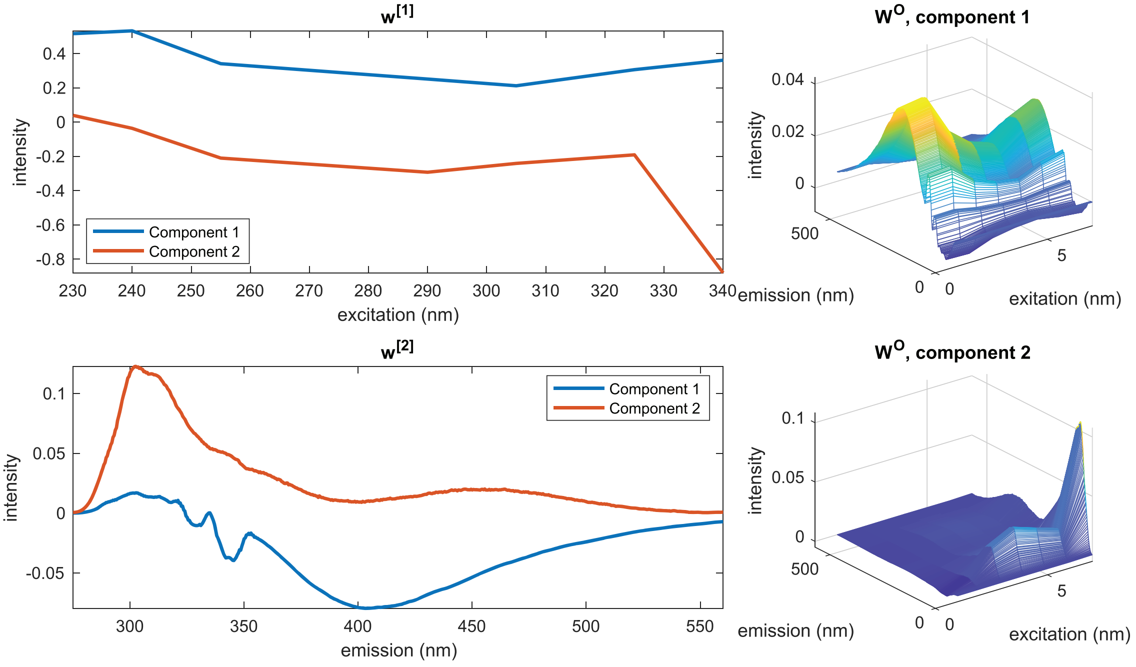

FIGURE 5 Loading weights for *sugar* data using trilinear N-CPLS with orthogonal dimension-wise loading weights, $\mathbf{w}^{[1]}$ and $\mathbf{w}^{[2]}$ (left), and their outer products, $\mathbf{W}^{O}$, for the first two components (right)

Figure 6 shows the explained variance of the test set for the N-PLS,[8] the multilinear N-CPLS (equal to N-PLS in the single response case), the multilinear N-PLS with dimension-wise loading weight orthogonalisation and the unfolded N-CPLS with and without use of the additional response information. With regard to the maximum predictive performances, the four models are not very different, though the multilinear approach with orthogonalisation achieves the largest overall explained variance. However, the explained response variances for the parsimonious model alternatives are very different: The single component models range from explaining 48.0% (unfolded N-CPLS) via 62.3% (N-PLS and multilinear N-CPLS) to 79.1% of the variance in the primary response (obtained by N-CPLS with additional response information). Due to extra loading weight orthogonalisation, N-PLS and multilinear N-CPLS with orthogonalisation diverge slightly from the second component and more from the fourth component, where the orthogonalised version of multilinear N-CPLS is clearly dominant until 16 components are included. However, from a parsimony perspective, the first few components are usually most relevant. It must be assumed that *colour* and the *year* of measurements has a profound effect on the samples or instrument, since the N-CPLS model based on utilising the additional response information from *colour* and *year* is capable of accounting for much more ash information in a single component. In the comparisons we have omitted multilinear N-CPLS with additional response information as the combination of the two strategies does not seem to work well for the datasets we have investigated.

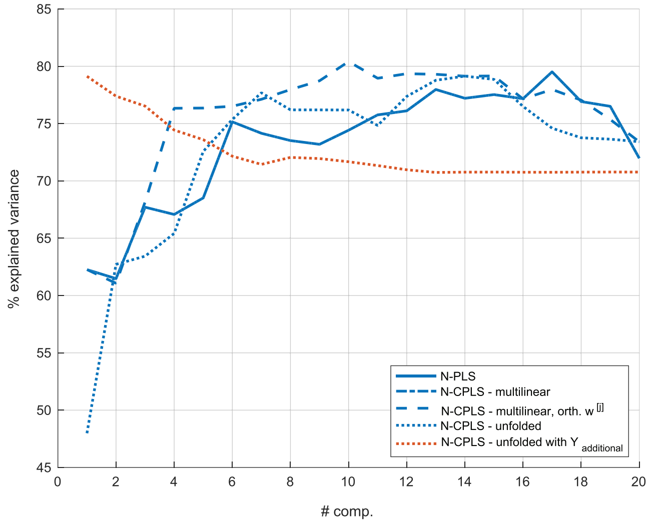

FIGURE 6 Explained variance of test set predictions of *ash* in *sugar* for N-PLS, multilinear N-PLS (identical to N-PLS), multilinear N-PLS with orthogonalisation of $w^{[j]}$, unfolded N-CPLS and unfolded N-CPLS with additional responses *colour* and *year* (dummy-coded)

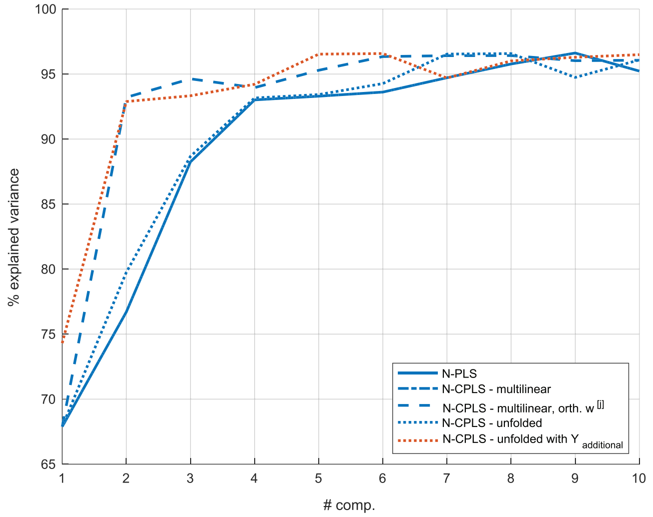

FIGURE 7 Explained variance of test set predictions of *fats* in *milk* for N-PLS, multilinear N-CPLS (identical to N-PLS), multilinear N-PLS with orthogonalisation of $w^{[j]}$, unfolded N-CPLS and unfolded N-CPLS with additional responses *protein* and *total solids*

Predictive analyses with the milk data also show considerable differences in the first few components of the N-CPLS models compared with the N-PLS model. Here we used *fats* as the response (**Y**) and *protein* and *total solids* as additional response information ($\mathbf{Y}_{\mathbf{additional}}$) for the N-CPLS. The maximum explained variances were reached after 5, 6, 7, 9, and 9 components, for the modelling alternatives displayed in Figure 7. It should be noted that the best two-component model alternatives of multilinear N-CPLS with orthogonalisation and the N-CPLS using additional response information are quite close to each model's maximum while the same is true for the four component models of N-PLS, multilinear N-CPLS, and unfolded N-CPLS.

6.3 | Classification—ORL faces

The third example illustrates both the possibility of using the N-CPLS for classification and a situation where requiring multilinear components makes little sense. We dummy-code the subjects of the ORL face database and use this as a response matrix (**Y**) of size (400×40). As mentioned in Section 3, the canonical correlation employed in the N-CPLS modelling enables discriminant analysis with a more aggressive criterion than what is used in the N-PLS modelling. However, the practical effect of this maximisation depends on the application. Here, we do not have any readily available additional responses at hand, but there are several candidates that easily can be designed (such as coding for the presence of glasses, moustaches, etc.). For simplicity, we have chosen to use the first 10 left singular vectors of the unfolded, centred training data, that is, principal components, as additional responses ($\mathbf{Y}_{\mathbf{additional}}$).

Figure 8 shows that the differences between the unfolded N-CPLS versions and N-PLS are not very large, while the multilinear N-CPLS is not able to achieve the same maximum accuracy in classification. The orthogonalised version ends up at a maximum accuracy of around 58% (not visible in the plot), almost 35% lower than the unfolded versions. In the figure, the simplest models achieving accuracies $\geq 80\%$ and $\geq 90\%$ are highlighted. We observe that the simplest models with accuracy $\geq 90\%$ has 17 (N-CPLS with $\mathbf{Y}_{\mathbf{additional}}$), 21 (N-PLS), 22 (unfolded N-CPLS) and 33 (multilinear N-CPLS) components. For this dataset, it is evident that forcing multilinearity for N-CPLS has an adverse effect on accuracy, while N-PLS, with its less aggressive maximisation criterion, fares surprisingly well. One might claim that the inclusion of additional sample information leads to the most parsimonious models here, but the margins are too small to conclude that the differences are significant. The most significant benefit from using N-CPLS here is the reduction in computational time spent in the model building (see the subsequent section).

The upper parts of Figure 9 shows two faces of different subjects, facial expressions and accessories. The lower parts shows the effect of restricting components to be multilinear (restricting the loading weights of the first component to be

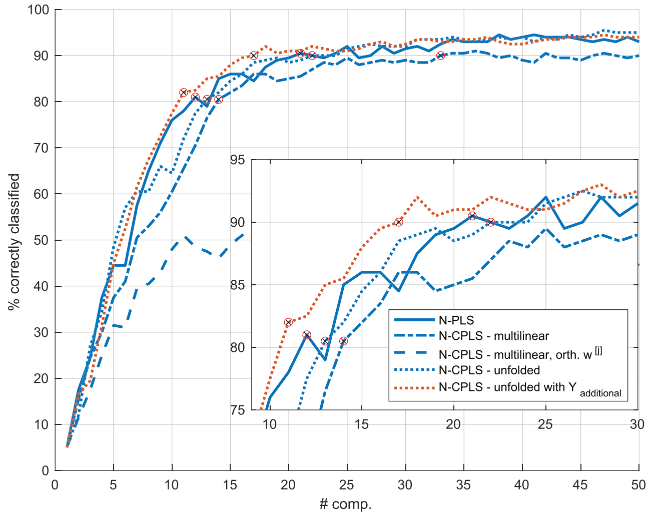

FIGURE 8 Percentage correctly classified test set *faces* for N-PLS, multilinear N-CPLS, multilinear N-PLS with orthogonalisation of $w^{[j]}$ (continues between 45% and 58% behind sub-figure), unfolded N-CPLS and unfolded N-CPLS with additional artificially created sample information. Minimal models that achieve $\geq 80\%$ and $\geq 90\%$ correctly classified *faces* are marked with circles/crosses. The smaller subfigure is a magnified view of the top-left part of the larger subfigure

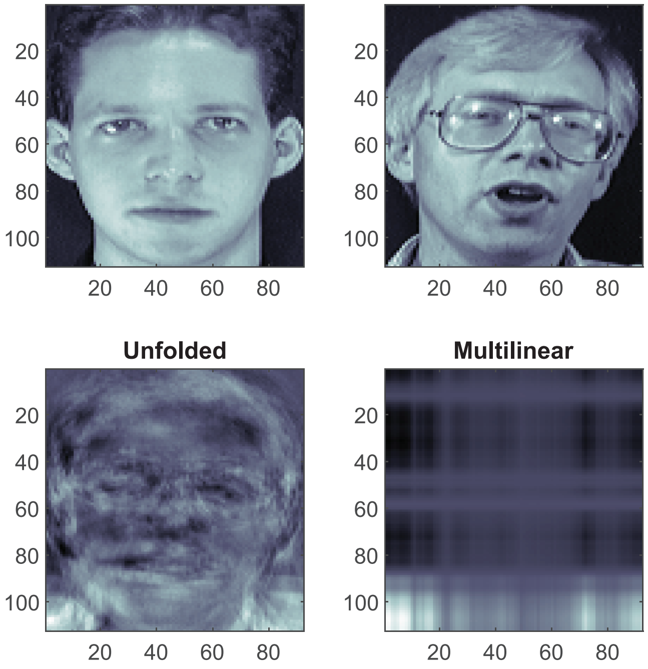

FIGURE 9 Two example photos from the *ORL faces* (top). Loading weights from an unfolded N-CPLS analysis and a multilinear N-CPLS analysis (bottom)

TABLE 1 Time used for fitting models, predicting new samples and summarising as explained variance or accuracy

Dataset	Samples	Features	Responses	Additional	Components	Method	Time (s)
Sugar	134 × 2	571× 7	1	2	20	N-PLS	4.34
						N-CPLS mul.lin.	0.04
						N-CPLS mul.lin.+orth.	0.10
						N-CPLS unfolded	0.05
						N-CPLS ($\mathbf{Y}_{\mathbf{add.}}$)	0.05
Milk	148 × 2	201× 2	1	2	10	N-PLS	0.20
						N-CPLS mul.lin.	0.01
						N-CPLS mul.lin.+orth.	0.01
						N-CPLS unfolded	0.01
						N-CPLS ($\mathbf{Y}_{\mathbf{add.}}$)	0.01
ORL	200 × 2	92× 112	40	10	50	N-PLS	241.85
						N-CPLS mul.lin.	0.71
						N-CPLS mul.lin.+orth.	0.92
						N-CPLS unfolded	0.84
						N-CPLS ($\mathbf{Y}_{\mathbf{add.}}$)	0.86

Note: Reported times are medians of 100, 100 and 10 repetitions, respectively, for the three datasets. Above, "mul.lin." is short for multilinear and $\mathbf{Y}_{\mathbf{add.}}$ is short for $\mathbf{Y}_{\mathbf{additional}}$. All datasets were split in two for training and testing, that is, "Samples" refer to number of samples in each of the sets.

an outer product of per-dimension loading weights, $\mathbf{w}^{[j]}$) compared with just unfolding. The former does not contribute to an efficient solution, since the stack of images does not have any obvious underlying trilinear form.

6.4 | Computational efficiency

Table 1 reports the computational times used for building the various models considered above. The reported times are median times from repeated modelling, prediction and calculations of explained variances or classification accuracy for each method within each dataset. We note that time is only of practical concern for larger datasets, especially in the presence of many response variables. For one-time modelling, time is not a deal-breaker for any of the combinations of dataset and method. However, >4 min waiting for a single run of N-PLS for the ORL data will quickly become a nuisance if 10-fold cross-validation is used, increasing computation time to around 40 min. It should be noted that none of the tested datasets are especially large, meaning that one can assume much longer computation times for high resolution data with many samples and responses when applying N-PLS, while N-CPLS apparently scales very well within the observed cases.

The main theoretical reasons for the speed gains in the case of N-CPLS are due to the absence of a predictor matrix deflation step which is required for the N-PLS method and the absence of iterative component estimation. However, in practice, there is also a substantial overhead in poorly designed code. For the ORL problem around 43% of the modelling time is due to use of the pseudo inverse and 42% is due to custom made matrix multiplication code, both of which are included to compensate for missing data, even in the absence of any missing numbers. However, even compensating for these weaknesses in the code, the speed-up by using the N-CPLS instead of N-PLS is around 50-fold for the ORL problem.

7 | DISCUSSION AND CONCLUSIONS

Based on the definitions of tensor dot product and tensor transpose from Section 2, we have shown how to describe predictive multiway modelling in a compact and transparent manner. Likewise, the corresponding software implementation is just slightly longer than a matrix based implementation.

By extending the ideas of multiway PLS modelling with the ideas of the CPLS, the resulting N-CPLS method is demonstrated to yield more parsimonious models, in particular, due to the possibility of including additional responses in the model building process. This also ensures efficient handling of both continuous and categorical responses. The switching option in the N-CPLS algorithm due to the dimensionality (number of dimensions/modes) in the input data leads to an efficient handling of input data represented as vectors, matrices, threeway- or multiway tensors.

We have further shown how a multilinear decomposition can be acquired and how this restricts the search space of the multiway modelling. The choice between multilinear or unfolded model building process is mostly depending on the type of data being analysed, but it is also a matter of choice by the user with regard to possible benefits of inspecting dimension-wise loading weights. The included examples show that the predictive accuracy and parsimony of models when choosing between multilinear and unfolded analysis is highly data dependent and even dependent on which responses are used for the same predictors.

A further restriction can be added in the form of requiring orthogonalisation of the extracted dimension-wise loading weights. One can argue for this requirement in terms of more interpretable loading weight plots. On the other hand, one can argue against it due to the effects this may have on the loading weight outer product tensor and the reduced fit, as demonstrated in our first example. For some applications, it might seem evident which model variation to use, based on known properties of the data. However, as the included examples show, until one has applied and properly validated the methods, it is difficult to predict which version will give the most precise prediction.

The summary of computation times shown in Table 1 leaves no doubt that the proposed N-CPLS is orders of magnitude faster than the N-PLS implementation from Andersson et al.[12] This is both due to the non-iterative canonical correlation approach to loading weight computations and the avoided deflation of the predictors, in addition to some less efficient choices for the coding in the N-PLS implementation. The N-CPLS also has regression coefficients and projection matrices for producing scores as two of its outputs, meaning that predictions for new data can be done through a single tensor dot product and intercept compensation.

As with any multivariate modelling method, there are user choices involved and hyperparameters to tune. The N-CPLS opens up the space of possibilities with regard to model restrictiveness and possible additional sample information to include in the modelling of multiway data. As such, the N-CPLS can be used as a flexible stand-alone tool or be further integrated into multiblock frameworks such as Sequential and Orthogonalised N-PLS[20] and the Swiss Knife PLS[5] where even more possibilities arise.

PEER REVIEW

The peer review history for this article is available at https://publons.com/publon/10.1002/cem.3432.

ORCID

Kristian Hovde Liland https://orcid.org/0000-0001-6468-9423
Ulf Geir Indahl https://orcid.org/0000-0002-3236-463X
Joakim Skogholt https://orcid.org/0000-0001-8511-993X
Puneet Mishra https://orcid.org/0000-0001-8895-798X

REFERENCES

1. Wold S, Esbensen K, Geladi P. Principal component analysis. *Chemom Intell Lab Syst*. 1987;2(1-3):37-52.
2. Wold S, Hellberg S, Lundstedt T, Sjöström M, Wold H. PLS modelling with latent variables in two or more dimensions. PLS model building: Theory and application; 1987.
3. Björck Å, Indahl UG. Fast and stable partial least squares modelling: A benchmark study with theoretical comments. *J Chemom*. 2017;31(8):e2898.
4. Liland KH, Næs T, Indahl UG. ROSA – a fast extension of partial least squares regression for multiblock data analysis. *J Chemom*. 2016;30(11):651-662.
5. Mishra P, Liland KH. Swiss knife partial least squares (SKPLS): One tool for modelling single block, multiblock, multiway, multiway multiblock including multi-responses and meta information under the rosa framework. *Anal Chimica Acta*. 2022;1206:339786.
6. Wold S, Martens H, Wold H. The multivariate calibration problem in chemistry solved by the PLS method. *Matrix pencils*: Springer; 1983:286-293.
7. Wold S, Ruhe A, Wold H, Dunn WJ. The collinearity problem in linear regression. The partial least squares (PLS) approach to generalized inverses. *SIAM J Sci Stat Comput*. 1984;5(3):735-743.
8. Bro R. Multiway calibration. multilinear pls. *J Chemom*. 1996;10(1):47-61.
9. Piccirilli GN, Escandar GM. Second-order advantage with excitation-emission fluorescence spectroscopy and a flow-through optosensing device. simultaneous determination of thiabendazole and fuberidazole in the presence of uncalibrated interferences. *Analyst*. 2010;135(6):1299-1308. doi:10.1039/B923565E
10. Lia F, Formosa JP, Zammit-Mangion M, Farrugia C. The first identification of the uniqueness and authentication of maltese extra virgin olive oil using 3d-fluorescence spectroscopy coupled with multi-way data analysis. *Foods*. 2020;9(4):498. https://www.mdpi.com/2304-8158/9/4/498
11. Durante C, Cocchi M, Grandi M, Marchetti A, Bro R. Application of N-PLS to gas chromatographic and sensory data of traditional balsamic vinegars of modena. *Chemom Intell Lab Syst*. 2006;83(1):54-65.
12. Andersson CA, Bro R. The n-way toolbox for matlab. *Chemom Intell Lab Syst*. 2000;52(1):1-4.
13. Indahl UG, Liland KH, Næs T. Canonical partial least squares—A unified PLS approach to classification and regression problems. *J Chemom*. 2009;23(9):495-504.
14. Arfken GB, Weber H-J. *Mathematical methods for physicists*. Elsevier; 2005.
15. Harshman RA. Foundations of the Parafac procedure: models and conditions for an 'explanation' multi-modal factor analysis. *UCLA Working Pap Phon*. 1970;16:1-84.
16. Smilde AK, Bro R, Geladi P. *Multiway analysis. Applications in the Chemical Sciences*. New York: John Wiley & Sons; 2004.
17. Bro R. Exploratory study of sugar production using fluorescence spectroscopy and multi-way analysis. *Chemom Intell Lab Syst*. 1999;46(2):133-147.
18. Uusitalo S, Diaz-Olivares J, Sumen J, Hietala E, Adriaens I, Saeys W, Utriainen M, Frondelius L, Pastell M, Aernouts B. Evaluation of mems nir spectrometers for on-farm analysis of raw milk composition. *Foods*. 2021;10(11):2686.
19. Liland KH, Indahl UG. Powered partial least squares discriminant analysis. *J Chemom*. 2009;23(1):7-18.
20. Biancolillo A, Næs T, Bro R, Måge I. Extension of SO-PLS to multi-way arrays: SO-N-PLS. *Chemom Intell Lab Syst*. 2017;164:113-126.

How to cite this article: Liland KH, Indahl UG, Skogholt J, Mishra P. The canonical partial least squares approach to analysing multiway datasets—N-CPLS. *Journal of Chemometrics*. 2022;36(7):e3432. doi:10.1002/cem.3432