ACMTF-R: supervised multi-omics data integration uncovering shared and distinct outcome-associated variation

G.R. van der Ploeg[1*], F.T.G. White[1], R.R. Jakobsen[2], J.A. Westerhuis[1], A. Heintz-Buschart[1], A.K. Smilde[1]

[1] *Biosystems Data Analysis, Swammerdam Institute for Life Sciences, University of Amsterdam, Amsterdam, The Netherlands*

[2] *Department of Food Science, University of Copenhagen, Rolighedsvej 26, DK – 1958 Frederiksberg C, Denmark*

*Author for correspondence: Geert Roelof van der Ploeg g.r.ploeg@uva.nl

Geert Roelof van der Ploeg:	https://orcid.org/0009-0007-5204-3386
Fred White:	https://orcid.org/0000-0003-0434-6028
Rasmus Riemer Jakobsen	https://orcid.org/0000-0001-7416-1358
Johan Westerhuis:	https://orcid.org/0000-0002-6747-9779
Anna Heintz-Buschart:	https://orcid.org/0000-0002-9780-1933
Age Smilde:	https://orcid.org/0000-0002-3052-4644

Abstract

The rapid growth of high-dimensional biological data has necessitated advanced data fusion techniques to integrate and interpret complex multi-omics and longitudinal datasets. Advanced Coupled Matrix and Tensor Factorization (ACMTF) has emerged as a powerful framework for uncovering common, local, and distinct sources of variation across datasets. However, ACMTF lacks the ability to model variation linked to a dependent variable, limiting its applicability to studies investigating biological phenotypes. N-way Partial Least Squares (NPLS) is a supervised method that identifies variation in relation to a dependent variable but lacks the ability to identify common, local and distinct sources of variation across multiple datasets. To bridge the gap between data exploration and prediction, we introduce ACMTF-Regression (ACMTF-R), an extension of ACMTF

that incorporates a regression term, allowing for the simultaneous decomposition of multi-way data while explicitly capturing variation associated with an outcome variable.

We present a detailed mathematical formulation of ACMTF-R, including its optimisation algorithm and implementation. Through extensive simulations, we systematically evaluate its ability to recover a small $\boldsymbol{y}$-related component shared between multiple blocks, its robustness to noise, and the impact of the tuning parameter (π) which controls the balance between data exploration and outcome prediction. Our results demonstrate that ACMTF-R can robustly identify the $\boldsymbol{y}$-related component, correctly identifying outcome-associated shared and distinct variation, distinguishing it from existing approaches such as N-way Partial Least Squares and ACMTF.

To validate its applicability in a real-world setting, we apply ACMTF-R to a multi-omics dataset integrating human milk microbiome, human milk metabolome, and infant faecal microbiome data, investigating how maternal pre-pregnancy BMI affects microbial and metabolic signatures. ACMTF-R successfully identifies novel mother-infant relationships associated with maternal pre-pregnancy BMI, underscoring its utility in multi-omics research. Our findings establish ACMTF-R as a versatile tool for multi-way data fusion, offering new insights into complex biological systems by integrating common, local, and distinct variation in the context of a dependent variable.

Key Words

Coupled matrix and tensor factorization, multi-omics data integration, tensor decomposition

Background

The advent of high-throughput omics technologies has led to a rapid growth of complex, high-dimensional biological datasets, including genomics [1–3], transcriptomics [4], proteomics [5], metabolomics [4], and microbiome [6]. When measured on the same samples, these multi-omics datasets give researchers the opportunity to obtain a systems-level understanding of biological processes [5]. However, their integration and analysis pose significant challenges due to data heterogeneity [5, 7], high dimensionality [4, 6], and complex interactions between different omics datasets [8, 9]. Advanced data analytical methods are required to extract meaningful biological insights by distinguishing variation that is shared across datasets from variation that is dataset-specific [10–13].

A critical need in the analysis of multi-omics data is identifying common, local and distinct sources of variation (**Figure 1**; [8–10, 14]). Common variation is shared across all datasets, local variation is shared between a subset of datasets, and distinct variation is unique to a dataset. Identifying the

Common, Local and Distinct (CLD) structure of a multi-omics dataset elucidates the complex biological processes that occur within the system.

Advanced Coupled Matrix and Tensor Factorization (ACMTF) has emerged as a powerful multi-way data integration approach capable of uncovering common, local, and distinct sources of variation across datasets [10, 15–18]. However, ACMTF is unsupervised and therefore does not allow the identification of shared and distinct variation that relates to a dependent variable. N-way Partial Least Squares (N-PLS) [19] is a supervised method that identifies variation in relation to a dependent variable, but is limited to analysing one dataset at a time and thus cannot simultaneously identify common, local, and distinct variation across multiple datasets.

To bridge the gap between data exploration and prediction, we introduce Advanced Coupled Matrix and Tensor Factorization Regression (ACMTF-R). ACMTF-R extends ACMTF by incorporating a regression term into the objective function, enabling the extraction of common, local, and distinct variation while simultaneously predicting an outcome. By introducing a tuning parameter (π), which has been used successfully in other approaches such as Principal Covariates Regression (PCovR) [20, 21] and Multiway Covariates Regression (MCovR) [22], the solution can be steered towards explaining the data or towards predicting $\boldsymbol{y}$.

In this study, we first provide a detailed overview of the ACMTF framework, and the methodological development represented by ACMTF-R. We then present a simulation-based approach to systematically evaluate the ability of ACMTF-R to capture a small, hidden, $\boldsymbol{y}$-related component under different noise conditions and tuning parameter values. Finally, we apply ACMTF-R to a real-world dataset, integrating human milk (HM) microbiome, HM metabolome, and infant gut microbiome data to study how maternal pre-pregnancy body mass index (ppBMI) is associated with these data. Our findings highlight the potential of ACMTF-R as a versatile tool for multi-omics data integration, facilitating the discovery of biologically meaningful common, local and distinct variation, while accommodating both exploratory and predictive research objectives.

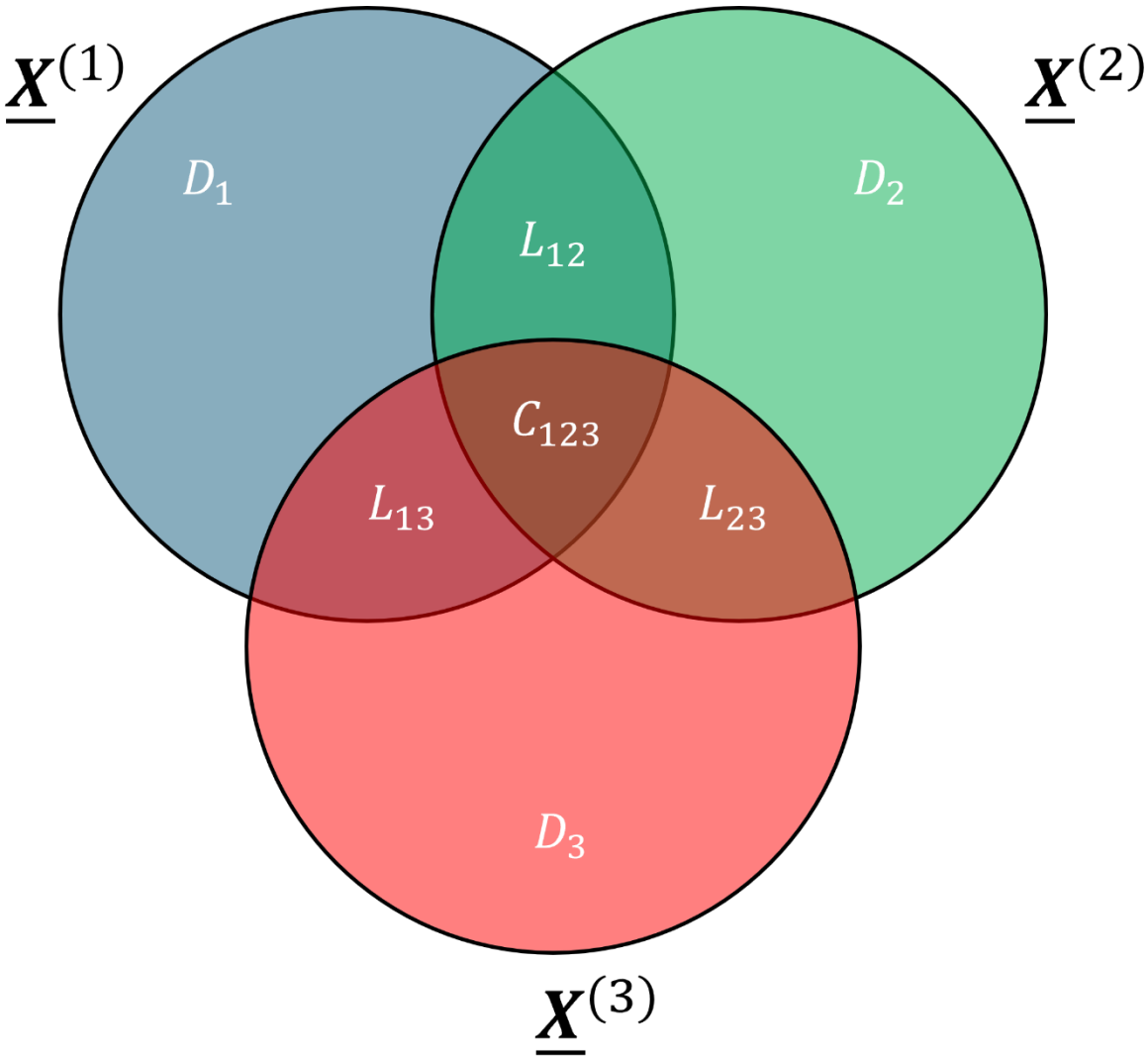


Figure 1: common, local and distinct variation. Common variation C_{123} is shared between all datasets $\underline{\boldsymbol{X}}^{(1)}$,$\underline{\boldsymbol{X}}^{(2)}$ and $\underline{\boldsymbol{X}}^{(3)}$. Local variation is shared between some of the datasets. Hence L_{12} is shared between $\underline{\boldsymbol{X}}^{(1)}$ and $\underline{\boldsymbol{X}}^{(2)}$, but not $\underline{\boldsymbol{X}}^{(3)}$. Similarly, L_{13} is shared between $\underline{\boldsymbol{X}}^{(1)}$ and $\underline{\boldsymbol{X}}^{(3)}$, but not $\underline{\boldsymbol{X}}^{(2)}$, and L_{23} is shared between $\underline{\boldsymbol{X}}^{(2)}$ and $\underline{\boldsymbol{X}}^{(3)}$, but not $\underline{\boldsymbol{X}}^{(1)}$. Distinct variation is unique to one dataset: D_1 for block $\underline{\boldsymbol{X}}^{(1)}$, D_2 for block $\underline{\boldsymbol{X}}^{(2)}$, and D_3 for block $\underline{\boldsymbol{X}}^{(3)}$. Together, they make up the Common, Local and Distinct (CLD) structure of the data.

Methods

Notation and definitions

We briefly define the mathematical notation that will be used throughout this paper. The notation proposed by Kiers [23] is followed with some minor extensions for multi-omics. Scalars are denoted by lower-case letters (e.g., a, b, c). Column vectors are denoted by boldface lowercase letters (e.g., $\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}$). Matrices are denoted by boldface capital letters (e.g., $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}$). Three-way arrays are denoted by underlined boldface capital letters (e.g., $\underline{\boldsymbol{X}}$, $\underline{\boldsymbol{Y}}$, $\underline{\boldsymbol{Z}}$). While $\underline{\boldsymbol{X}}$ can more generally be used to indicate four-way or higher-order arrays, this paper does not discuss such data.

The characters I, J, and K are reserved to indicate the first, second and third mode of an array. A two-way array $\boldsymbol{X}$ will be assumed to be of size $I \times J$, while a three-way array $\underline{\boldsymbol{X}}$ will be assumed to be of size $I \times J \times K$. Similarly, lowercase characters i, j, and k will be used as indices for the first, second, and third modes, respectively. Columns of a matrix are vectors with a subscript of their index (e.g., $\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_J$ for some matrix $\boldsymbol{X}$). Similarly, the ith-row and jth-column entry is denoted as a scalar with subscripts of their indices (e.g., x_{ij}). For three-way arrays, the entries per mode are given as subscripts (e.g., x_{ijk}).

The index $p = 1, \dots, P$ is used to indicate the block number, which is denoted by a superscript (e.g., $\underline{\boldsymbol{X}}^{(1)}, \underline{\boldsymbol{X}}^{(2)}, \dots, \underline{\boldsymbol{X}}^{(P)}$) and the size is indicated likewise (e.g., the size of $\underline{\boldsymbol{X}}^{(p)}$ is $I \times J^{(p)} \times K^{(p)}$). In ACMTF and ACMTF-R $\underline{\boldsymbol{X}}^{(1)}, \underline{\boldsymbol{X}}^{(2)}, \dots, \underline{\boldsymbol{X}}^{(P)}$ can be any mixture of two-way or three-way arrays, but this study focuses only on the case where all blocks are three-way arrays with a shared subject mode. Hence the superscript (p) is not used to describe the size of the first mode I.

Mathematically it is convenient to matricise a three-way array $\underline{\boldsymbol{X}}$, as this transformation facilitates the description of multi-way models using matrix notation [23–25]. There are three different variants of matricisation, depending on which mode of the three-way array is preserved as rows in the resulting matrix. Hence, first mode matricisation yields a matrix $\boldsymbol{X}_a$ of size $I \times JK$. Similarly, second mode matricization produces a matrix $\boldsymbol{X}_b$ of size $J \times IK$, while third mode matricization generates a matrix $\boldsymbol{X}_c$ of size $K \times IJ$.

The vectorisation of an $I \times J$ matrix $\boldsymbol{X}$ is denoted $\text{vec}(\boldsymbol{X})$, such that $\text{vec}(\boldsymbol{X}) = \left[\boldsymbol{x}_1^T\ \boldsymbol{x}_2^T \dots \boldsymbol{x}_J^T\right]^T$. The Hadamard product [26] of two equally sized arrays $\boldsymbol{X}$ and $\boldsymbol{Y}$ is denoted $\boldsymbol{X} * \boldsymbol{Y}$, such that $(\boldsymbol{X} * \boldsymbol{Y})_{ij} = x_{ij} y_{ij}$. The Kronecker product, denoted $\otimes$, of two matrices $\boldsymbol{X}$ with size $I \times J$ and $\boldsymbol{Y}$ with size $K \times L$

produces a block matrix containing all pairwise products of their entries following $(\boldsymbol{X} \otimes \boldsymbol{Y})_{ik,jl} = x_{ij} y_{kl}$. Thus, we have

$$\boldsymbol{X} \otimes \boldsymbol{Y} = \begin{bmatrix} x_{11}\boldsymbol{Y} & \cdots & x_{1J}\boldsymbol{Y} \\ \vdots & \ddots & \vdots \\ x_{I1}\boldsymbol{Y} & \cdots & x_{IJ}\boldsymbol{Y} \end{bmatrix} \tag{1}$$

where the outcome of $\boldsymbol{X} \otimes \boldsymbol{Y}$ is size $IK \times JL$.

The Khatri-Rao product, denoted $\odot$, of two matrices $\boldsymbol{X}$ of size $I \times R$ and $\boldsymbol{Y}$ of size $J \times R$ is defined as the column-wise Kronecker product [27, 28]

$$\boldsymbol{X} \odot \boldsymbol{Y} = \begin{bmatrix} | & | & \cdots & | \\ \boldsymbol{x}_1 \otimes \boldsymbol{y}_1 & \boldsymbol{x}_2 \otimes \boldsymbol{y}_2 & \cdots & \boldsymbol{x}_R \otimes \boldsymbol{y}_R \\ | & | & \cdots & | \end{bmatrix} = \begin{bmatrix} | & | & \cdots & | \\ \mathrm{vec}(\boldsymbol{y}_1 \boldsymbol{x}_1^T) & \mathrm{vec}(\boldsymbol{y}_2 \boldsymbol{x}_2^T) & \cdots & \mathrm{vec}(\boldsymbol{y}_R \boldsymbol{x}_R^T) \\ | & | & \cdots & | \end{bmatrix}$$

where the outcome of $\boldsymbol{X} \odot \boldsymbol{Y}$ has size $IJ \times R$.

The Frobenius (or Euclidean) norm of a three-way array $\underline{\boldsymbol{X}}$ is defined as

$$\|\underline{\boldsymbol{X}}\| = \sqrt{\sum_{i=1}^{I} \sum_{j=1}^{J} \sum_{k=1}^{K} x_{ijk}^2} \tag{2}$$

and the notation $\| \cdot \|$ will also be used to refer to the Frobenius norm for matrices and vectors, respectively [24]. The Moore-Penrose inverse [29] of $\boldsymbol{X}$ is denoted $\boldsymbol{X}^+$.

In this paper we follow the canonical polyadic form of defining multi-way arrays as the sum of rank-one arrays with normalized factor matrices and a scalar λ to capture the magnitude per term [15, 30–32]. The notation $[\![\boldsymbol{\lambda}_p ; \boldsymbol{A}, \boldsymbol{B}^{(p)}, \boldsymbol{C}^{(p)}]\!]$ is used to describe the model of the three-way array $\underline{\boldsymbol{X}}^{(p)}$, defined elementwise as

$$x_{ijk}^{(p)} = \sum_{f=1}^{F} \lambda_{pf} a_{if} b_{jf}^{(p)} c_{kf}^{(p)} + e_{ijk}^{(p)} \tag{3}$$

where columns of the loading matrices $\boldsymbol{A}, \boldsymbol{B}^{(p)}, \boldsymbol{C}^{(p)}$ are norm 1 and the column vector $\boldsymbol{\lambda}_p$ is used to modify the size of each identified component $f = 1, \ldots, F$ to norm λ_{pf}. For a two-way array, equation (3) reduces to $[\![\boldsymbol{A}, \boldsymbol{B}]\!] = \boldsymbol{A}\boldsymbol{B}^T$, assuming that $\boldsymbol{\lambda} = 1$ with size $1 \times F$ and that the singular values are multiplied into $\boldsymbol{A}$ to stay in line with common practice.

Advanced Coupled Matrix and Tensor Factorization (ACMTF)

Advanced Coupled Matrix and Tensor Factorization is a joint factorisation approach that is applicable to mixtures of matrices and tensors [10, 15, 16, 33]. The method seeks a decomposition of the datasets $\underline{\boldsymbol{X}}^{(1)}, \underline{\boldsymbol{X}}^{(2)}, \dots, \underline{\boldsymbol{X}}^{(P)}$ as $\hat{\underline{\boldsymbol{X}}}^{(p)} = [\![\boldsymbol{\lambda}_p; \boldsymbol{A}, \boldsymbol{B}^{(p)}, \boldsymbol{C}^{(p)}]\!]$ such that loading matrices corresponding to shared modes are equal between the blocks. In this paper, we will assume that only the first mode is shared between all blocks. In that case, the matrix containing the first mode loadings $\boldsymbol{A}$ will be the same for every block p and the superscript can be left out.

The loss function for ACMTF is defined as

$$
\begin{aligned}
f\left(\alpha, \beta, \varepsilon, \boldsymbol{\Lambda}, \boldsymbol{A}, \boldsymbol{B}^{(1)}, \boldsymbol{C}^{(1)}, \dots, \boldsymbol{B}^{(P)}, \boldsymbol{C}^{(P)}\right) = & \sum_{p=1}^{P} \left\| \underline{\boldsymbol{X}}^{(p)} - \hat{\underline{\boldsymbol{X}}}^{(p)} \right\|^2 \\
& + \beta \sum_{p=1}^{P} \sum_{f=1}^{F} \sqrt{\lambda_{pf}^2 + \epsilon} \\
& + \alpha \sum_{p=1}^{P} \sum_{f=1}^{F} \left(\|\boldsymbol{a}_f\| - 1 \right)^2 \\
& + \alpha \sum_{p=1}^{P} \sum_{f=1}^{F} \left(\left\| \boldsymbol{b}_f^{(p)} \right\| - 1 \right)^2 \\
& + \alpha \sum_{p=1}^{P} \sum_{f=1}^{F} \left(\left\| \boldsymbol{c}_f^{(p)} \right\| - 1 \right)^2
\end{aligned}
\tag{4}
$$

where the $P \times F$ matrix $\boldsymbol{\Lambda} = [\boldsymbol{\lambda}_1\ \boldsymbol{\lambda}_2 \dots \boldsymbol{\lambda}_P]^T$ scales the factors per component and per data block, $\alpha > 0$ is a penalty setting for norm-1 components, $\beta > 0$ is a sparsity penalty setting for $\boldsymbol{\Lambda}$, F is the number of components, and p is the block index. Here the 1-norm loss term for the elements in $\boldsymbol{\Lambda}$ is replaced with a differentiable approximation $\beta \|\lambda_{pf}\|_1 \approx \beta \sqrt{\lambda_{pf}^2 + \epsilon}$ for a sufficiently small $\epsilon > 0$ [33, 34]. The originally suggested default settings are $\alpha = 1$, $\beta = 10^{-3}$ and $\epsilon = 10^{-8}$ [10]. The gradient of the loss function is reported in **Supplementary Methods**.

The ACMTF framework of putting the norm of a component into $\boldsymbol{\Lambda}$ and constraining the loading vectors to become norm one, allows the $\boldsymbol{\Lambda}$ matrix to encode the Common, Local, and Distinct (CLD) structure of the data (**Figure 1**). For example, the CLD-structure of a three-block case $\underline{\boldsymbol{X}}^{(1)}, \underline{\boldsymbol{X}}^{(2)}, \underline{\boldsymbol{X}}^{(3)}$ can be encoded in $\boldsymbol{\Lambda}$ through

$$\boldsymbol{\Lambda} = \begin{array}{c} \\ \underline{\boldsymbol{X}}^{(1)} \\ \underline{\boldsymbol{X}}^{(2)} \\ \underline{\boldsymbol{X}}^{(3)} \end{array} \begin{array}{c} \begin{array}{ccccccc} C_{123} & L_{12} & L_{13} & L_{23} & D_1 & D_2 & D_3 \end{array} \\ \begin{bmatrix} \lambda_{11} & \lambda_{12} & \lambda_{13} & 0 & \lambda_{15} & 0 & 0 \\ \lambda_{21} & \lambda_{22} & 0 & \lambda_{24} & 0 & \lambda_{26} & 0 \\ \lambda_{31} & 0 & \lambda_{33} & \lambda_{34} & 0 & 0 & \lambda_{37} \end{bmatrix} \end{array}$$

where $\lambda_{pf} > 0$ corresponds to data block p contributing to component f, 0 corresponds to a data block not contributing to a component, C_{123} indicates a common component shared between all blocks, L_{12}, L_{13} and L_{23} indicate local components shared between some blocks, and D_1, D_2 and D_3 indicate distinct components unique to one block. Sparsity is then achieved by applying an L_1 penalty on the $\boldsymbol{\Lambda}$-matrix**.** While this example uses 7 components to showcase the entire CLD-structure, multiple components may be needed to fully describe a specific type of variation in real data (e.g., two components for C_{123}) and some types of variation may be absent altogether.

Advanced Coupled Matrix and Tensor Factorization Regression (ACMTF-R)

We present Advanced Coupled Matrix and Tensor Factorization Regression (ACMTF-R) to describe common, local, and distinct variation of interest in the data blocks in the context of a dependent variable $\boldsymbol{y}$ by adding a regression term and a tuning parameter π to the loss function. The introduction of a tuning parameter has been used successfully in other approaches such as Principal Covariates Regression (PCovR) [20, 21] and Multiway Covariates Regression (MCovR) [22] to steer the solution towards explaining the data blocks or towards predicting $\boldsymbol{y}$. This gives the user more information about how the addition of $\boldsymbol{y}$ affects the model of the data. We only define ACMTF-R in the case where the first mode is shared across all data blocks and $\boldsymbol{y}$. Hence $\boldsymbol{y}$ is a column vector of size $I \times 1$. The loss function of ACMTF-R is then defined as

$$\begin{aligned} f\left(\alpha, \beta, \varepsilon, \pi, \boldsymbol{\Lambda}, \boldsymbol{A}, \boldsymbol{B}^{(1)}, \boldsymbol{C}^{(1)}, \dots, \boldsymbol{B}^{(P)}, \boldsymbol{C}^{(P)}\right) = {} & \pi \sum_{p=1}^{P} \left\| \underline{\boldsymbol{X}}^{(p)} - \underline{\widehat{\boldsymbol{X}}}^{(p)} \right\|^2 \\ & + (1-\pi) \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\rho}\|^2 \\ & + \beta \sum_{p=1}^{P} \sum_{f=1}^{F} \sqrt{\lambda_{pf}^2 + \epsilon} \\ & + \alpha \sum_{p=1}^{P} \sum_{f=1}^{F} \left(\|\boldsymbol{a}_f\| - 1\right)^2 \\ & + \alpha \sum_{p=1}^{P} \sum_{f=1}^{F} \left(\left\|\boldsymbol{b}_f^{(p)}\right\| - 1\right)^2 \\ & + \alpha \sum_{p=1}^{P} \sum_{f=1}^{F} \left(\left\|\boldsymbol{c}_f^{(p)}\right\| - 1\right)^2 \end{aligned} \tag{6}$$

where π is a tuning parameter used to focus the model on explaining the data blocks $\underline{\boldsymbol{X}}^{(1)}$, $\underline{\boldsymbol{X}}^{(2)}$, …, $\underline{\boldsymbol{X}}^{(P)}$ versus predicting $\boldsymbol{y}$, and $\boldsymbol{\rho}$ is the $F \times 1$ vector of regression coefficients of the factor matrix of the first mode $\boldsymbol{A}$ onto $\boldsymbol{y}$. Here $\boldsymbol{\rho}$ is given when $\boldsymbol{A}$ is defined, through

$$\boldsymbol{\rho} = \boldsymbol{A}^{+}\boldsymbol{y}. \tag{7}$$

The gradient of the loss function is reported in **Supplementary Methods**.

Prediction of $\boldsymbol{y}$ *using a new sample*

A fitted ACMTF-R model can be used to predict y_{new} for a new sample using a similar procedure as Latent Root Regression [35, 36]. First, this requires a brief description of obtaining the subject mode loadings for a new sample in CANDECOMP/PARAFAC (CP) [30, 32, 37, 38]. Subsequently, this result is extended to ACMTF-R, after which the prediction step is described.

In the CP case, finding the subject mode loadings $\boldsymbol{a}_{new}$ with size $F \times 1$ for a sample $\boldsymbol{X}_{new}$ with size $J \times K$ requires solving the problem

$$\arg\min_{\boldsymbol{a}_{new}} \|\boldsymbol{X}_{new} - \boldsymbol{B}\mathrm{diag}(\boldsymbol{a}_{new})\boldsymbol{C}^{T}\|_F^2 = \arg\min_{\boldsymbol{a}} \|\mathrm{vec}(\boldsymbol{X}_{new}) - (\boldsymbol{C}\odot\boldsymbol{B})\boldsymbol{a}_{new}\|_F^2 \tag{9}$$

where $\boldsymbol{B}$ and $\boldsymbol{C}$ are the feature and time mode loadings of a previously fitted model [32]. This problem statement comes down to the least squares solution

$$\boldsymbol{a}_{new} = (\boldsymbol{C}\odot\boldsymbol{B})^{+}\mathrm{vec}(\boldsymbol{X}_{new}) = \boldsymbol{Z}^{+}\mathrm{vec}(\boldsymbol{X}_{new}) \tag{10}$$

where the block matrix $\boldsymbol{Z} = \boldsymbol{C}\odot\boldsymbol{B}$ with size $JK \times F$ is used for convenience and where $\boldsymbol{Z}^{+}$ is the Moore-Penrose inverse of $\boldsymbol{Z}$.

In the case of ACMTF-R, the shared subject mode loadings across the blocks need to be identified. This is done by finding the block matrix $\boldsymbol{Z}^{(p)}$ with size $K^{(p)}J^{(p)} \times F$ through

$$\boldsymbol{Z}^{(p)} = \boldsymbol{\lambda}_p^{T}\odot\boldsymbol{C}^{(p)}\odot\boldsymbol{B}^{(p)} \tag{11}$$

which can be concatenated across all blocks into one matrix for all data blocks $1, \ldots, P$ simultaneously as

$$\mathbf{Z} = \begin{pmatrix} \boldsymbol{Z}^{(1)} \\ \boldsymbol{Z}^{(2)} \\ \vdots \\ \boldsymbol{Z}^{(P)} \end{pmatrix} \tag{12}$$

where $\boldsymbol{Z}$ is of size $\left(K^{(1)}J^{(1)} + K^{(2)}J^{(2)} + \cdots + K^{(P)}J^{(P)}\right) \times F$. The shared subject mode loadings $\boldsymbol{a}_{new}$ with size $F \times 1$ for a new sample $\boldsymbol{X}_{new}$ can then be found by reusing Equation 10.

Finally, the prediction of y_{new} is done using the ACMTF-R regression coefficients $\boldsymbol{\rho}$ through

$$y_{new} = \boldsymbol{\rho}^{\boldsymbol{T}} \boldsymbol{a}_{new}. \tag{12}$$

When multiple new samples are obtained, this procedure is performed per sample. The procedure is performed by the npred() function in the supplied R package.

Implementation and stopping criteria

The decomposition of ACMTF and ACMTF-R is achieved through an all-at-once optimisation algorithm, originally developed in [15] for MATLAB, now available in the `CMTFtoolbox` package for R on CRAN. The all-at-once optimisation is achieved by defining the loss and gradient function and subsequently using the nonlinear conjugate gradient (NCG) method with Hestenes-Stiefel updates and the Moré-Thuente line search as originally suggested [15, 18]. For the provided R package, this was implemented using the `mize` package (v0.2.4, [39]). The provided package also supports the Limited-memory Broyden-Fletcher-Goldfarb-Shanno algorithm (L-BFGS) to speed up the line search as an experimental feature, but this setting is not used for any results in this paper [40–42].

ACMTF-R uses the same default values as ACMTF for the minimum value of the loss function (default 10^{-10}) and the minimum change in loss function between two function evaluations (default 10^{-10}) as stopping criteria [33]. The all-at-once optimization implementation through the `mize` package also defines three other stopping criteria: the maximum number of iterations to calculate (default 10 000), the maximum number of function evaluations that are allowed (default 10 000), the minimum l2-norm of the gradient vector (default 10^{-10}), and the absolute value of the size of the parameter update (default 10^{-10}). Fitting an ACMTF-R model for any of the datasets in this paper takes less than a minute of computational time on an average computer (Microsoft Windows 11 Home v10.0.22631 with an 12th Gen Intel® Core™ i5-12400F running 2500 MHz with 6 cores or 12 logical processors and 16 Gb RAM). Multi-core parallelisation has been implemented as part of the provided R package to allow many randomly initialised models to be fitted simultaneously. This is needed to efficiently find the appropriate number of components through cross-validation, as well as to find the global minimum for a given number of components.

Simulation approach

We demonstrate the effect of the ACMTF-R model parameters using various simulations (**Figure 2**). We follow the guidelines provided in previous work [10]. We generate factor matrices $\boldsymbol{A} \in \mathbb{R}^{I \times F}$, $\boldsymbol{B}^{(1)} \in \mathbb{R}^{J^{(1)} \times F}, \boldsymbol{C}^{(1)} \in \mathbb{R}^{K^{(1)} \times F}, \boldsymbol{B}^{(2)} \in \mathbb{R}^{J^{(2)} \times F}, \boldsymbol{C}^{(2)} \in \mathbb{R}^{K^{(2)} \times F}, \boldsymbol{B}^{(3)} \in \mathbb{R}^{J^{(3)} \times F}$ and $\boldsymbol{C}^{(3)} \in \mathbb{R}^{K^{(3)} \times F}$ with random entries drawn from the standard normal distribution $N \sim (0,1)$. The columns of the factor matrices are normalized to unit norm. We set $I = 100$, $J^{(1)} = J^{(2)} = J^{(3)} = 200$, and $K^{(1)} = K^{(2)} = K^{(3)} = 5$ to resemble a typical pre-processed multi-omics dataset (for example, a mixture of microbiome and metabolomics data). The factor matrices are used to create three third-order tensors $\underline{\boldsymbol{X}}^{(1)} \in \mathbb{R}^{100 \times 200 \times 5}, \underline{\boldsymbol{X}}^{(2)} \in \mathbb{R}^{100 \times 200 \times 5}, \underline{\boldsymbol{X}}^{(3)} \in \mathbb{R}^{100 \times 200 \times 5}$ through

$$\underline{\boldsymbol{X}}^{(1)} = [\![\boldsymbol{\lambda}_1; \boldsymbol{A}, \boldsymbol{B}^{(1)}, \boldsymbol{C}^{(1)}]\!],$$

$$\underline{\boldsymbol{X}}^{(2)} = [\![\boldsymbol{\lambda}_2; \boldsymbol{A}, \boldsymbol{B}^{(2)}, \boldsymbol{C}^{(2)}]\!],$$

and

$$\underline{\boldsymbol{X}}^{(3)} = [\![\boldsymbol{\lambda}_3; \boldsymbol{A}, \boldsymbol{B}^{(3)}, \boldsymbol{C}^{(3)}]\!].$$

where we encode the required CLD-structure into $\boldsymbol{\Lambda}$ consisting of eight components total: two common components shared across all blocks, three local components, and a distinct component for each block. This corresponds to the $\boldsymbol{\Lambda}$-matrix

$$\boldsymbol{\Lambda} = \begin{bmatrix} 1 & 1 & 1 & \delta & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & \delta & 1 & 0 & 0 & 1 \end{bmatrix}$$

where all components are norm 1, except for the second local component whose size δ varies per simulation.

Next, randomly distributed noise is added to the third-order tensors through

$$\underline{\boldsymbol{X}}_{noisy}^{(p)} = \underline{\boldsymbol{X}}^{(p)} + \eta_X \frac{\|\underline{\boldsymbol{X}}^{(p)}\|}{\|\underline{\boldsymbol{N}}^{(p)}\|} \underline{\boldsymbol{N}}^{(p)} \tag{13}$$

where η_X indicates the noise level, which is the same across all blocks.

We generate $\boldsymbol{y}$ equal to the subject mode loadings of the second local component

$$\boldsymbol{y} = \boldsymbol{a}_{L12}$$

and some noise is added through

$$\boldsymbol{y}_{noisy} = \boldsymbol{y} + \eta_y \frac{\|\boldsymbol{y}\|}{\|\boldsymbol{n}_y\|} \boldsymbol{n}_y \tag{14}$$

where η_y indicates the noise level on $\boldsymbol{y}$. The outcome $\boldsymbol{y}_{noisy}$ is then normalized to norm 1.

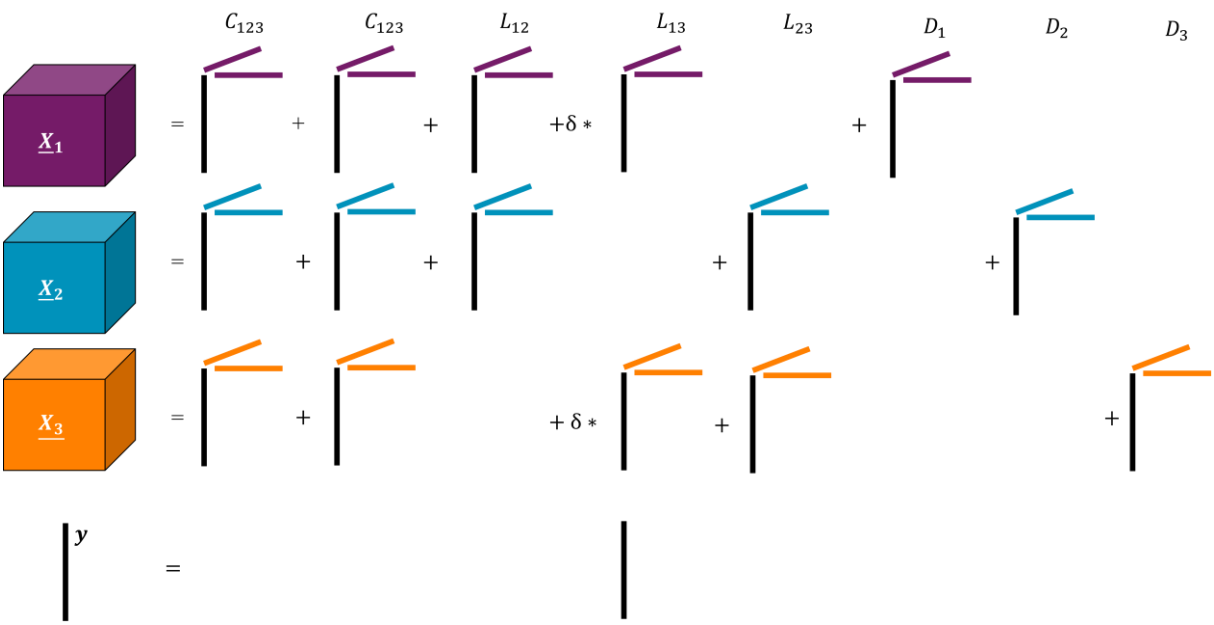


Figure 2: overview of the simulation setup of the study. All simulations contain a CLD-structure of 2 global components shared between blocks $\underline{\boldsymbol{X}}^{(1)}, \underline{\boldsymbol{X}}^{(2)}, \underline{\boldsymbol{X}}^{(3)}$, 3 local components, and a distinct component unique to each block. This yields 8 components total for ACMTF-R to identify. All components are norm 1, except for the second local component whose norm δ varies per simulation. $\boldsymbol{y}$ is equal to the subject mode loadings of the second local component. Noise and terms corresponding to components that have no contribution to a block have been removed for visual clarity. Subject mode loadings (black lines) are equal between the data blocks per component but are different between components.

Simulation 1: ACMTF-R can detect a small, hidden component by leveraging $\boldsymbol{y}$

In the first simulation, the ability of ACMTF-R to detect a small, hidden component by leveraging $\boldsymbol{y}$ was examined. This was done using the three-block simulation with $\delta = 0.03$, 40% noise on $\underline{\boldsymbol{X}}^{(1)}, \underline{\boldsymbol{X}}^{(2)}, \underline{\boldsymbol{X}}^{(3)}$ ($\eta_X = 0.40$), and 5% noise on $\boldsymbol{y}$ ($\eta_y = 0.05$). The data were then block scaled to Frobenius norm 1. One hundred randomly initialised eight-component ACMTF models were compared with one hundred randomly initialised eight-component ACMTF-R models using different values of $\pi \in \{0.1, 0.2, \ldots, 0.9\}$, keeping all other parameter and convergence settings at their default values. Subsequently, factor recovery was assessed using Factor Match Score and Tucker Congruence Coefficient. The line search algorithm in most ACMTF-R models was terminated due to the absolute change for the size of the parameter update stopping criterion, while ACMTF was terminated only due to the minimum value of the loss function being reached (**Supplementary Figure 1**).

Simulation 2: the benefit of supervision using $\boldsymbol{y}$ *versus the size of the hidden component*

In the second simulation, the improvement of ACMTF-R compared to ACMTF was assessed by changing the size of the small, hidden component. This was done using the three-block simulation with

$$\delta \in \{0.01, 0.02, 0.03, 0.04, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5\}.$$

The noise on $\underline{\boldsymbol{X}}^{(1)}, \underline{\boldsymbol{X}}^{(2)}, \underline{\boldsymbol{X}}^{(3)}$ was set at 40% ($\eta_X = 0.40$), and the noise on $\boldsymbol{y}$ was 5% ($\eta_y = 0.05$). The data were then block scaled to Frobenius norm 1. One hundred randomly initialised eight-component ACMTF models were compared with one hundred randomly initialised eight-component ACMTF-R models using different values of $\pi \in \{0.1, 0.2, \ldots, 0.9\}$ for each case, keeping all other parameter and convergence settings at their default values. Subsequently, factor recovery was assessed using Factor Match Score and Tucker Congruence Coefficient. The line search algorithm in most ACMTF-R models was terminated due to the absolute change for the size of the parameter update stopping criterion, while most ACMTF models were terminated only due to the minimum value of the loss function being reached (**Supplementary Figure 2**).

Simulation 3: the benefit of supervision using $\boldsymbol{y}$ versus noise on X

In the third simulation, the improvement of ACMTF-R compared to ACMTF was assessed by changing the amount of noise on $\underline{\boldsymbol{X}}^{(1)}, \underline{\boldsymbol{X}}^{(2)}, \underline{\boldsymbol{X}}^{(3)}$. This was done using the three-block simulation with $\delta = 0.03$ and a variable amount of noise on $\underline{\boldsymbol{X}}^{(1)}, \underline{\boldsymbol{X}}^{(2)}, \underline{\boldsymbol{X}}^{(3)}$,

$$\eta_X \in \{0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1, 2, 3, 4, 5, 10, 50, 100\},$$

and the noise on $\boldsymbol{y}$ was 5% ($\eta_y = 0.05$). The data were then block scaled to Frobenius norm 1. One hundred randomly initialised eight-component ACMTF models were compared with one hundred randomly initialised eight-component ACMTF-R models using different values of $\pi \in \{0.1, 0.2, \ldots, 0.9\}$ for each case, keeping all other parameter and convergence settings at their default values. Subsequently, factor recovery was assessed using Factor Match Score and Tucker Congruence Coefficient. For most noise levels ($\eta_X < 2$), the line search algorithm in most ACMTF-R models was terminated due to the absolute change for the size of the parameter update stopping criterion, while most ACMTF models were terminated only due to the minimum value of the loss function being reached (**Supplementary Figure 3**). At higher noise levels ($\eta_X > 2$), most ACMTF models were terminated due to the relative change in the loss function being reached.

Simulation 4: the benefit of supervision using $\boldsymbol{y}$ versus noise on $\boldsymbol{y}$

In the fourth and final simulation, the improvement of ACMTF-R compared to ACMTF was assessed by changing the amount of noise on $\boldsymbol{y}$. This is done using the three-block simulation with $\delta = 0.03$, with 40% noise on $\underline{\boldsymbol{X}}^{(1)}, \underline{\boldsymbol{X}}^{(2)}, \underline{\boldsymbol{X}}^{(3)}$ ($\eta_X = 0.40$), and a variable amount of noise on $\boldsymbol{y}$:

$$\eta_y \in \{0.0, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1, 2, 3, 4, 5, 10, 50, 100\}.$$

The data were then block scaled to Frobenius norm 1. One hundred randomly initialised eight-component ACMTF models were compared with one hundred randomly initialised eight-component ACMTF-R models using different values of $\pi \in \{0.1, 0.2, \ldots, 0.9\}$ for each case, keeping all other parameter and convergence settings at their default values. Subsequently, factor recovery was assessed using Factor Match Score and Tucker Congruence Coefficient. The line search algorithm in most ACMTF-R models was terminated due to the absolute change for the size of the parameter update stopping criterion, while ACMTF was terminated only due to the minimum value of the loss function being reached (**Supplementary Figure 4**).

Simulation evaluation – Tucker Congruence Coefficient

In this paper, the recovered factors are compared against the input factors of the simulation using the absolute Tucker Congruence Coefficient (TCC) [43, 44] through

$$|\phi(\boldsymbol{x},\boldsymbol{y})| = \left| \frac{\sum_i x_i\, y_i}{\sqrt{\sum_i x_i^2 \sum_i y_i^2}} \right| \tag{15}$$

where $\boldsymbol{x}$ and $\boldsymbol{y}$ are the recovered and true factors, respectively. The TCC can be interpreted as the cosine of the angle between $\boldsymbol{x}$ and $\boldsymbol{y}$. A TCC value in the range of 0.85-0.94 suggests that the recovered and input factors are fairly similar, whereas a value of 0.95-1 means that they are essentially equal [45]. Although TCC is insensitive to scalar multiplication of the vectors (that is, $\phi(\boldsymbol{x},\boldsymbol{y}) = \phi(a\boldsymbol{x}, b\boldsymbol{y})$ for any non-zero scalars a and b), it is sensitive to sign flips between $\boldsymbol{x}$ and $\boldsymbol{y}$ [45]. For this reason, we use the absolute value instead.

Simulation evaluation – Factor Match Score

The Factor Match Score (FMS) is an adaptation of the Tucker Congruence Coefficient and indicates the similarity of two multi-way models across all components. For model evaluation, FMS was used to compare the decomposition against the input factors. This depends on the CLD-type of the relevant factor. For the hidden local component L_{13} in the simulations, FMS_{L13} is defined as

$$FMS_{L13} = \frac{\left|\boldsymbol{a}_{L13}^T \tilde{\boldsymbol{a}}_{L13}\right|}{\|\boldsymbol{a}_{L13}\|\|\tilde{\boldsymbol{a}}_{L13}\|} \frac{\left|\left[\boldsymbol{b}_{L13}^{(1)}\right]^T \tilde{\boldsymbol{b}}_{L13}^{(1)}\right|}{\left\|\boldsymbol{b}_{L13}^{(1)}\right\| \left\|\tilde{\boldsymbol{b}}_{L13}^{(1)}\right\|} \frac{\left|\left[\boldsymbol{c}_{L13}^{(1)}\right]^T \tilde{\boldsymbol{c}}_{L13}^{(1)}\right|}{\left\|\boldsymbol{c}_{L13}^{(1)}\right\| \left\|\tilde{\boldsymbol{c}}_{L13}^{(1)}\right\|} \frac{\left|\left[\boldsymbol{b}_{L13}^{(3)}\right]^T \tilde{\boldsymbol{b}}_{L13}^{(3)}\right|}{\left\|\boldsymbol{b}_{L13}^{(3)}\right\| \left\|\tilde{\boldsymbol{b}}_{L13}^{(3)}\right\|} \frac{\left|\left[\boldsymbol{c}_{L13}^{(3)}\right]^T \tilde{\boldsymbol{c}}_{L13}^{(3)}\right|}{\left\|\boldsymbol{c}_{L13}^{(3)}\right\| \left\|\tilde{\boldsymbol{c}}_{L13}^{(3)}\right\|} \tag{16}$$

where $\boldsymbol{a}_{L13}$, $\boldsymbol{b}_{L13}$, and $\boldsymbol{c}_{L13}$ contain the true subject, feature and time loadings of component L_{13}, and $\tilde{\boldsymbol{a}}_{L13}$, $\tilde{\boldsymbol{b}}_{L13}$, and $\tilde{\boldsymbol{c}}_{L13}$ are the best matching subject, feature and time loadings in the fitted model. The FMS of the other components are defined equivalently. In this paper, the best matching component is found using the Hungarian algorithm [46, 47] for all pairwise combinations of components such that the FMS is maximised and the expected CLD structure is recovered (**Supplementary Methods**). The FMS is a value in the range [0,1], where $FMS = 1$ corresponds to the model correctly recovering the input loadings.

The recovery of all input loadings can be quantified through $FMS_{overall}$, defined as

$$FMS_{overall} = \frac{FMS_{C123} + FMS_{C123} + FMS_{L12} + FMS_{L13} + FMS_{L23} + FMS_{D1} + FMS_{D2} + FMS_{D3}}{8} \tag{17}$$

where $FMS_{overall} = 1$ corresponds to all input components being recovered correctly. Recovery of the CLD structure is assessed using the Lambda Similarity Index, which is defined and reported in the **Supplementary Materials**.

Example dataset: Jakobsen2025

We include an application study of ACMTF-R on longitudinally measured human milk (HM) microbiome, HM metabolomics and infant faecal microbiome data. Full details on the preprocessing of the data is outlined in the original paper [48].

Briefly, the zOTU data and taxonomic information of the infant faecal microbiome and mother milk microbiome were processed in R (v4.4.1, [49]). For the infant faecal microbiome data, features were kept if they had ≤75% sparsity across the dataset. This step resulted in 93 out of 565 features being selected. For the mother milk microbiome data, features were kept if they had ≤85% sparsity across the dataset. This step resulted in 115 out of 707 features being selected. We performed a centred log-ratio transformation with a pseudo-count of 1 to correct for compositionality [50, 51]. Subsequently the data was converted to a three-way tensor, keeping missing samples as a row of NAs. This resulted in three-way arrays of size 160 subjects × 93 microbial taxa × 3 time points and 169 subjects × 115 microbial taxa × 4 time points containing microbial abundances for the infant faecal and mother milk microbiomes, respectively.

The human milk (HM) metabolomics data were processed using R (v4.4.1, [49]). Values below the detection limit were imputed with a random value between 0 and the detection limit per metabolite to preserve their distribution. Next, the dataset was (natural) log transformed to stabilise the variance. The dataset was then converted to a three-way tensor of size 165 subjects × 70 metabolites × 4 time points.

Since ACMTF-R assumes that the subject mode is shared between all data blocks, we took the intersection of the subject identifiers across all three datasets and removed the other subjects. This homogenised the subject mode size I across all data blocks to 158 subjects. The data were then centred across the subject mode such that variation per time point is only due to between-subject variation, and scaled within the feature mode to make all features equally important for the modelling procedure [52, 53]. As a result, the models will focus on inter-subject differences and the features involved, while the time mode loadings indicate when these differences are largest.

After centring and scaling, the datasets were block scaled to Frobenius norm 1 to equalise the variance across the blocks. The response variable $\boldsymbol{y}$ of size 158×1 contained the maternal pre-pregnancy BMI for all mother-infant dyads, which was centred and subsequently scaled to norm 1 prior to modelling. ACMTF-R models were then created using the appropriate number of components and regressing on maternal pre-pregnancy BMI.

Selecting the appropriate number of ACMTF components

This paper presents a two-armed approach for selecting the appropriate number of components for ACMTF: a K-fold cross-validation scheme, and a random initialisation scheme. The full procedure is available as the ACMTF_modelSelection() function in the supplied R package.

In the K-fold cross-validation scheme, one fold of the subjects is withheld for all blocks as test set data, while randomly initialised ACMTF models are fitted on the training data using f components, keeping all other model parameters (α, β, ϵ) fixed. To avoid data leakage, the means, standard deviations, and norms that are removed from the training data are also removed from the test set data. Model stability is then reported by pairwise comparing the models with the lowest loss per fold and calculating the FMS_{CV}, defined as

$$FMS_{CV} = \frac{1}{F}\sum_{f=1}^{F} \frac{\left|\boldsymbol{b}_f^T \tilde{\boldsymbol{b}}_f\right|}{\|\boldsymbol{b}_f\|\|\tilde{\boldsymbol{b}}_f\|} \frac{\left|\boldsymbol{c}_f^T \tilde{\boldsymbol{c}}_f\right|}{\|\boldsymbol{c}_f\|\|\tilde{\boldsymbol{c}}_f\|} \tag{18}$$

where $\boldsymbol{b}_f$, and $\boldsymbol{c}_f$ contain the feature and time loadings for component f in one model, $\tilde{\boldsymbol{b}}_f$, and $\tilde{\boldsymbol{c}}_f$ are the best matching subject, feature and time loadings for component f in a second model, and the subject mode term is eliminated from the calculated due to not being comparable between CV folds [15, 54]. The matching of components is achieved using the Hungarian algorithm [46, 47] for all pairwise combinations of components such that the FMS is maximised. Hence, the distribution of FMS values across all pairwise comparisons of CV (or jack-knife) folds give an indication of the stability of the decomposition for a given number of components F. Previous work has established that 95% of folds should have $FMS_{CV} \geq 0.95$ for three-way data blocks and $FMS_{CV} \geq 0.9$ for two-way data blocks in the ACMTF case for the decomposition to be considered replicable [54]. The appropriate number of components is expected to have a stable FMS_{CV} near one across all comparisons.

In the second arm of the approach, randomly initialised ACMTF models are fitted to the full data using f components. Model sensitivity to the starting values and to local minima is then assessed by pairwise comparing the models and calculating the FMS_{random} through

$$FMS_{random} = \frac{1}{F}\sum_{f=1}^{F} \frac{\left|\boldsymbol{a}_f^T \tilde{\boldsymbol{a}}_f\right|}{\|\boldsymbol{a}_f\|\|\tilde{\boldsymbol{a}}_f\|} \frac{\left|\boldsymbol{b}_f^T \tilde{\boldsymbol{b}}_f\right|}{\|\boldsymbol{b}_f\|\|\tilde{\boldsymbol{b}}_f\|} \frac{\left|\boldsymbol{c}_f^T \tilde{\boldsymbol{c}}_f\right|}{\|\boldsymbol{c}_f\|\|\tilde{\boldsymbol{c}}_f\|} \tag{19}$$

where $\boldsymbol{a}_f$, $\boldsymbol{b}_f$, and $\boldsymbol{c}_f$ contain the subject, feature and time loadings for component f in one model, and $\tilde{\boldsymbol{a}}_f$, $\tilde{\boldsymbol{b}}_f$, and $\tilde{\boldsymbol{c}}_f$ are the best matching subject, feature and time loadings for component f in a second model [15, 54]. The matching of components is again achieved using the Hungarian algorithm [46, 47] for all pairwise combinations of components such that the FMS is maximised. The distribution of FMS values across all pairwise comparisons of randomly initialised models for a given number of components F gives an indication of the stability of the decomposition. The appropriate number of components is expected to have a stable FMS_{random} near one across all randomly initialised models.

Additionally, it may be possible for ACMTF and ACMTF-R to split up common or local components into distinct components with highly similar subject mode loadings when too many components are chosen. This phenomenon can be detected by pairwise comparing all columns in the subject mode per model and calculating the maximum absolute Tucker Congruence Coefficient φ [43–45]. We define this as the Degeneracy Score, which is computed through

$$Degeneracy\ Score = \max_{p \neq q} \left|\varphi(\boldsymbol{a}_p, \boldsymbol{a}_q)\right| \tag{20}$$

where p and q indicate component numbers of the subject mode within the same model, and $p \neq q$. The Degeneracy Score is in [0, 1], will be close to one when a factor is split into two highly collinear factors, and will have intermediate values otherwise. This procedure is only performed for the randomly initialised ACMTF models since the complete subject mode is needed. The appropriate number of components is expected to have a low Degeneracy Score across all randomly initialised models.

Selecting the appropriate number of ACMTF-R components

This paper presents a similar two-armed approach for selecting the appropriate number of components for ACMTF-R, reporting a few additional metrics related to the prediction and fit of $\boldsymbol{y}$.

The full procedure is available as the ACMTFR_modelSelection() function in the supplied R package.

In the cross-validation arm, the root mean squared error of cross-validation (RMSECV) is calculated by using the models fitted on the training set to predict $\boldsymbol{y}$ in the test set. The appropriate number of components is expected to minimise the RMSECV while having a stable FMS_{CV} near one across all comparisons.

In the random initialisation arm, the root mean squared error (RMSE) of $\boldsymbol{y}$ and the variance explained in $\boldsymbol{y}$ are reported using the ACMTF-R model with the lowest loss. The appropriate number of components is expected to minimise the RMSE while having a low Degeneracy Score and a stable FMS_{random} near one across all comparisons.

Results

Simulation 1: ACMTF-R can detect a small, hidden component by leveraging $\boldsymbol{y}$

We evaluated how the tuning parameter π influences the recovery of a local component that is weakly present in some data blocks and is strongly reflected in $\boldsymbol{y}$. This was done by applying ACMTF-R to a simulated three-tensor dataset with a known structure containing 2 common, 3 local, and 3 distinct components and 40% noise. All the components were norm 1, except for the second local component (L_{13}), which was present across $\underline{\boldsymbol{X}}^{(1)}$ and $\underline{\boldsymbol{X}}^{(3)}$ with norm $\delta = 0.03$. Since this component was equal to $\boldsymbol{y}$ with 5% noise, we expected that lowering π would enhance its detection. By systematically varying the tuning parameter π from 1 (fully focused on explaining the data blocks) to 0 (fully focused on predicting $\boldsymbol{y}$), we examined whether outcome information could reveal a latent structure that would otherwise remain hidden (**Figure 3**). One hundred randomly initialised eight-component ACMTF-R and ACMTF models were created for each setting of π.

As assessed by the Factor Match Score, the lowest-loss ACMTF-R models using $0.6 \leq \pi \leq 0.9$ showed an improvement in factor recovery for all components compared to the lowest-loss ACMTF model (**Figure 3A, Supplementary Figure 5**). This was also the case for the input factor recovery of the hidden component L_{13} (**Figure 3B**, **Supplementary Figure 6**). Meanwhile, the fraction of models that correctly identified the input CLD structure was comparable to ACMTF (**Supplementary Figure 7-8**).

Next, the ACMTF-R models with the lowest loss for each π setting were selected for comparison with the lowest-loss ACMTF model (**Figure 3C-H**). The absolute Tucker Congruence Coefficient (ϕ) was used to compare the best matching recovered factor loadings with the input loadings of L_{13} per mode (**Supplementary Methods**). This analysis revealed that the recovered shared subject mode (**Figure 3C**) and time mode loadings of $\underline{\boldsymbol{X}}^{(1)}$ (**Figure 3E**) and $\underline{\boldsymbol{X}}^{(3)}$ (**Figure 3G**) were almost identical to the input loadings in the ACMTF-R models for $0.6 \leq \pi \leq 0.9$, while ACMTF failed to identify them. While the recovered feature mode loadings in $\underline{\boldsymbol{X}}^{(1)}$ (**Figure 3D**) and $\underline{\boldsymbol{X}}^{(3)}$ (**Figure 3F**) were not exactly equal to the input factor of the hidden component, their order largely matched the input (Kendall's $\tau = 0.35 - 0.45$ for $0.4 \leq \pi \leq 0.9$; **Supplementary Table 1**).

Taken together, ACMTF-R recovered the hidden component L_{13} across intermediate values of π by leveraging $\boldsymbol{y}$, while ACMTF did not.

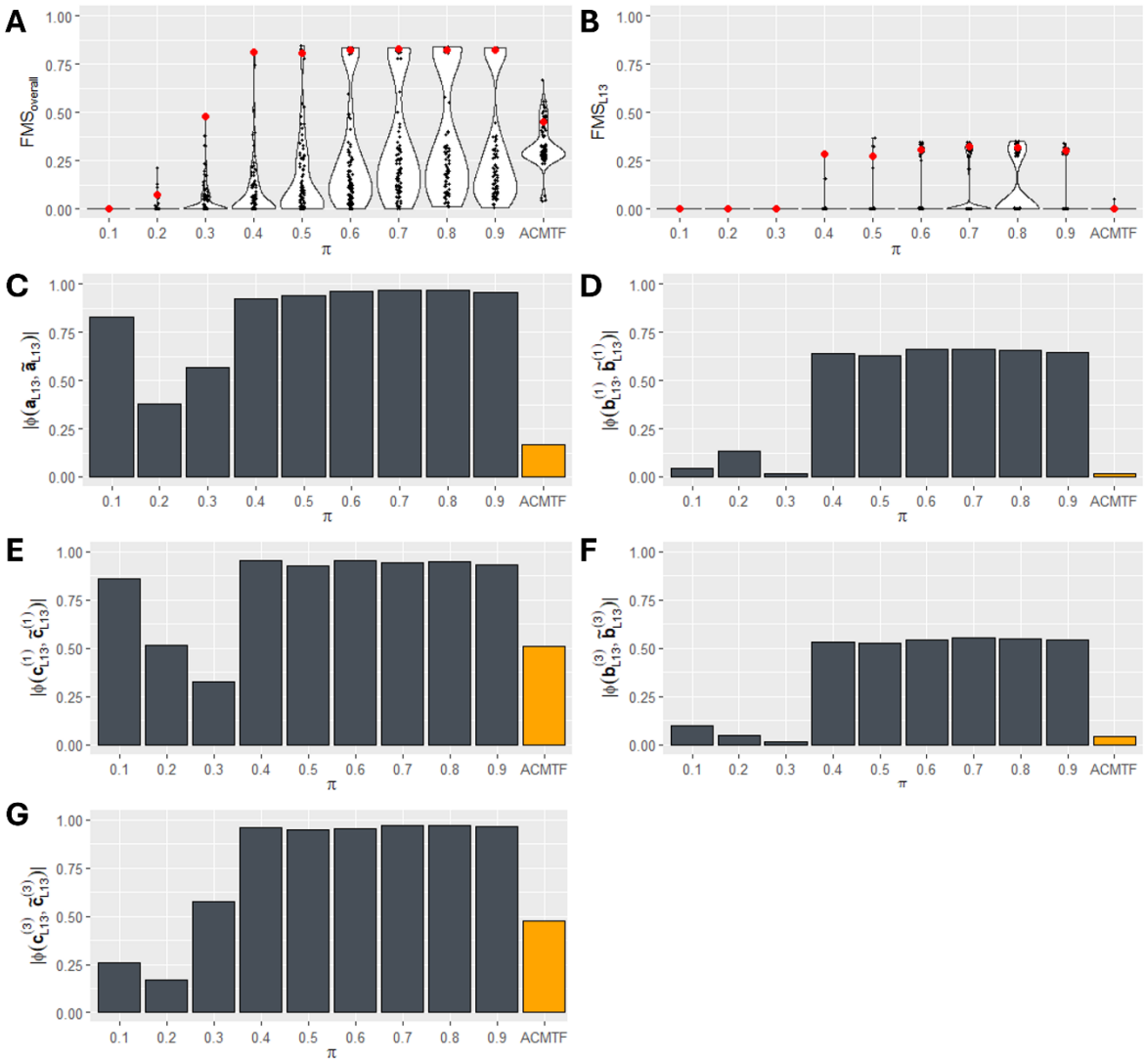


Figure 3: recovery of a small, hidden component L_{13} using ACMTF and ACMTF-R. One hundred randomly initialised ACMTF-R models were fitted per setting of tuning parameter π. These were compared with one hundred randomly initialised ACMTF models. (A) Overall factor recovery, measured using $FMS_{overall}$, was higher in ACMTF-R using $0.6 \leq \pi \leq 0.9$ compared to ACMTF. (B) Likewise, the FMS_{L13} maximum was higher for ACMTF-R than ACMTF. Red data points in (A,B) indicate the model with the lowest loss, which were compared in (C-H). The absolute Tucker Congruence Coefficient (ϕ) between the recovered loadings and the input (C) subject mode loadings, (D) feature mode loadings of $\underline{\boldsymbol{X}}^{(1)}$, (E) time mode loadings of $\underline{\boldsymbol{X}}^{(1)}$, (F) feature mode loadings of $\underline{\boldsymbol{X}}^{(3)}$, and (G) time mode loadings of $\underline{\boldsymbol{X}}^{(3)}$ for hidden component L_{13}. For all modes, factor recovery using ACMTF-R with intermediate values of π was higher compared to ACMTF.

Simulation 2: the benefit of supervision using $\boldsymbol{y}$ depends on the size of the hidden component

We next evaluated how the size of the hidden component L_{13} affected the improvement of ACMTF-R compared to ACMTF. This was done using the same three-tensor setup as in the first simulation, except that the size of the second local component (L_{13}) was varied between $\delta = 0.01$ to $\delta = 0.5$. We expected the benefit of supervision to decrease as the size of the hidden component δ increased. As before, the tuning parameter π was varied from 0.1 to 0.9 and one hundred randomly initialised eight-component ACMTF-R and ACMTF models were created for each setting (**Figure 4**).

Input factor recovery across all components improved in ACMTF-R compared to ACMTF at $0.4 \leq \pi \leq 0.9$ when the hidden component was small ($0.01 \leq \delta \leq 0.05$), while it deteriorated at lower values of π regardless of hidden component size due to overfitting (**Figure 4A**). Meanwhile, factor recovery of the hidden component L_{13} improved in ACMTF-R at $0.4 \leq \pi \leq 0.9$ only at intermediate component sizes ($0.03 \leq \delta \leq 0.05$) (**Figure 4B**). At $\delta > 0.05$, ACMTF and ACMTF-R were both able to identify L_{13}, while at $\delta < 0.03$ neither method could recover it (**Supplementary Figure 9**).

The recovered shared subject mode loadings of the hidden component showed a consistent recovery at small component sizes (0.01-0.05), regardless of π (**Figure 4C**). Meanwhile, the recovery of the feature and time mode loadings improved at similar component sizes only for intermediate values of π (0.4-0.9) (**Figure 4D-G**). Taken together, the simulation revealed that a hidden component that is roughly 1/20th the size of the other components can be successfully recovered by supervising with ACMTF-R using intermediate values of π.

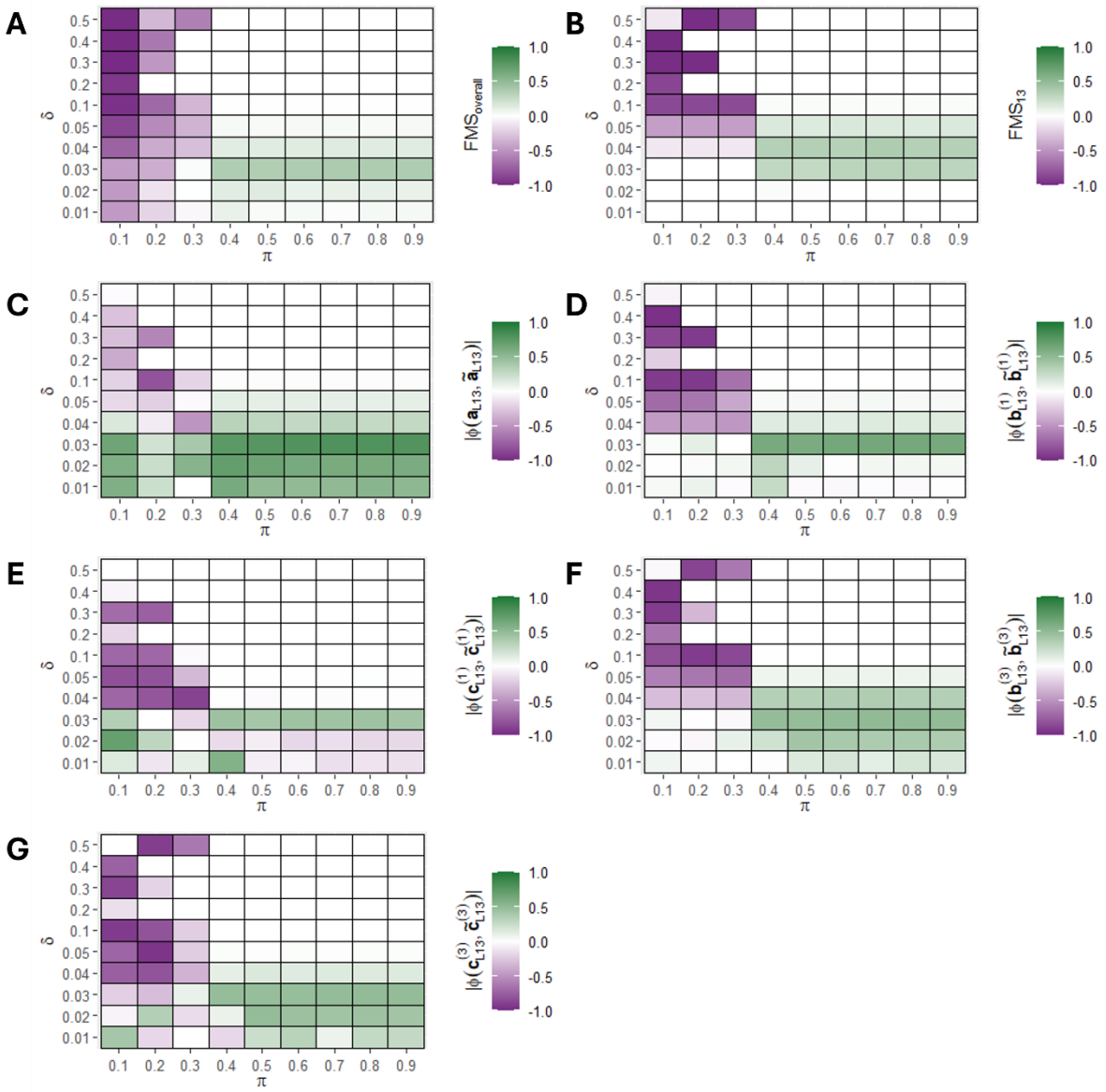


Figure 4. Improvement in factor recovery for varying sizes of the hidden component L_{13} using ACMTF-R. One hundred randomly initialised eight-component ACMTF-R and ACMTF models were fitted for each combination of hidden component size (δ) and tuning (π), after which the lowest-loss models were compared. The improvement in factor recovery is computed by subtracting ACMTF performance from ACMTF-R performance in all panels. Improvement per setting for (A) overall factor recovery, (B) factor recovery of the hidden component, (C) shared subject mode recovery of the hidden component, (D) feature mode recovery of $\underline{\boldsymbol{X}}^{(1)}$ for the hidden component, (E) time mode recovery of $\underline{\boldsymbol{X}}^{(1)}$ for the hidden component, (F) feature mode recovery of $\underline{\boldsymbol{X}}^{(3)}$ for the hidden component, (G) time mode recovery of $\underline{\boldsymbol{X}}^{(3)}$ for the hidden component. Improvement compared to ACMTF is shown in green, while deterioration is shown in purple. A version of this figure without subtraction of ACMTF performance is shown in **Supplementary Figure 9**.

523 *Simulation 3: the benefit of supervision using* $\mathbf{y}$ *is consistent for biologically realistic noise levels*

524 We evaluated how the amount of noise on the independent data blocks $\underline{\boldsymbol{X}}^{(1)}$, $\underline{\boldsymbol{X}}^{(2)}$, and $\underline{\boldsymbol{X}}^{(3)}$ affected
525 the improvement in recovering the hidden component L_{13} using ACMTF-R compared to ACMTF.
526 This was done using the same three-tensor setup as in the first simulation, except that the amount
527 of noise on the independent data blocks was varied from $\eta_X = 0.0$ to $\eta_X = 100$. We expected the
528 benefit of supervision to increase as the amount of noise increased. As before, the tuning parameter
529 π was varied from 0.1 to 0.9 and one hundred randomly initialised eight-component ACMTF-R and
530 ACMTF models were created for each setting (**Figure 5**).

531 Input factor recovery across all components improved in ACMTF-R compared to ACMTF at most
532 π values (0.3-0.9) at 0-50% noise (**Figure 5A**). However, at 10-30% noise, the improvement was low
533 due to both ACMTF and ACMTF-R recovering the hidden component (**Supplementary Figure 10**).
534 Likewise, factor recovery of the hidden component L_{13} improved only at intermediate noise levels
535 (30-50%) and at intermediate π values (0.6-0.9) (**Figure 5B**). When no noise was present, the
536 improvement in factor recovery of the hidden component was zero because both methods were
537 unable to identify it (**Supplementary Figure 10**).

538 The recovered shared subject mode loadings of the hidden component showed a consistent recovery
539 at all noise levels and values of π (**Figure 5C**). The recovery of the feature mode loadings in block
540 $\underline{\boldsymbol{X}}^{(1)}$ improved only at intermediate noise levels (20-50%) and intermediate π values (0.6-0.9), while
541 the recovery of the feature mode loadings in block $\underline{\boldsymbol{X}}^{(3)}$ were largely independent of the noise level
542 (**Figure 5D,F**). Surprisingly, the recovery of the time mode loadings in block $\underline{\boldsymbol{X}}^{(1)}$ deteriorated at
543 high noise levels (50+%), while the recovery of the time mode loadings in block $\underline{\boldsymbol{X}}^{(3)}$ improved at
544 those levels (**Figure 5E,G**). Taken together, this simulation revealed that factor recovery by ACMTF-
545 R improves at biologically realistic noise levels (>30%).

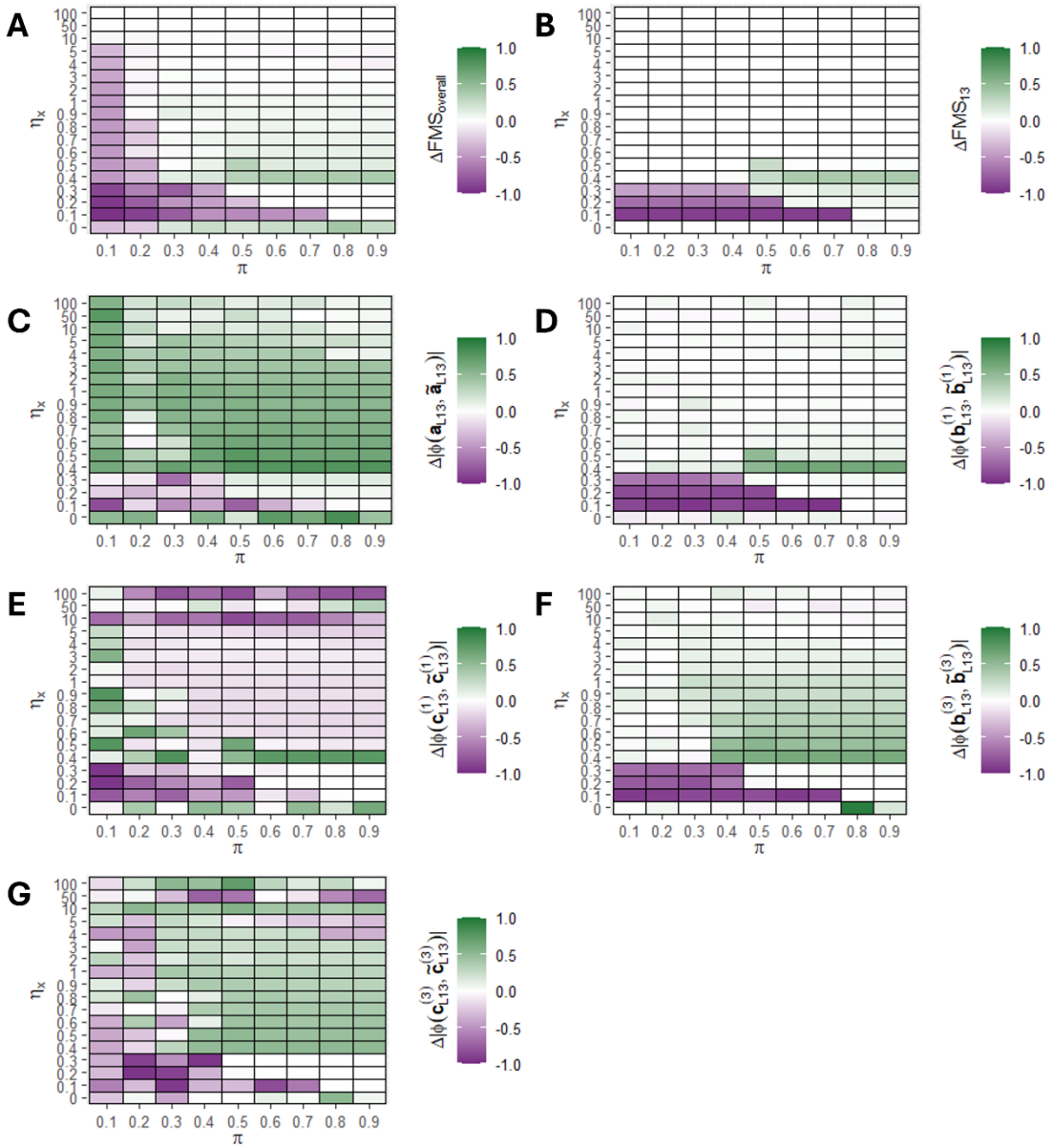


Figure 5. The improvement in factor recovery by ACMTF-R under different amounts of noise on X. One hundred randomly initialised eight-component ACMTF-R and ACMTF models were fitted for each combination of noise on X (η_X) and tuning parameter (π), after which the lowest-loss models were compared. The improvement in factor recovery is computed by subtracting ACMTF performance from ACMTF-R performance in all panels. Improvement per setting for (A) overall factor recovery, (B) factor recovery of the hidden component, (C) shared subject mode recovery of the hidden component, (D) feature mode recovery of $\underline{\boldsymbol{X}}^{(1)}$ for the hidden component, (E) time mode recovery of $\underline{\boldsymbol{X}}^{(1)}$ for the hidden component, (F) feature mode recovery of $\underline{\boldsymbol{X}}^{(3)}$ for the hidden component, (G) time mode recovery of $\underline{\boldsymbol{X}}^{(3)}$ for the hidden component. Improvement compared to ACMTF is shown in green, while deterioration is shown in purple. A version of this figure without subtraction of ACMTF performance is shown in **Supplementary Figure 10**.

558 *Simulation 4: the benefit of supervision using $\boldsymbol{y}$ is largely independent of the amount of noise on $\boldsymbol{y}$*

559 We evaluated how the amount of noise on $\boldsymbol{y}$ affects the improvement in recovering the hidden
560 component L_{13} using ACMTF-R compared to ACMTF. This was done using the same three-tensor
561 setup as in the first simulation, except that the amount of noise on $\boldsymbol{y}$ was varied from $\eta_y = 0.0$ to
562 $\eta_y = 100$. We expected the benefit of supervision to decrease as the amount of noise on $\boldsymbol{y}$ increased.
563 As before, the tuning parameter π was varied from 0.1 to 0.9 and one hundred randomly initialised
564 eight-component ACMTF-R and ACMTF models were created for each setting (**Figure 6**).

565 Input factor recovery across all components improved in ACMTF-R compared to ACMTF at
566 intermediate π values (0.4-0.9) for most noise levels (0-100%) (**Figure 6A**). Likewise, factor recovery
567 of the hidden component L_{13} improved at intermediate π values (0.5-0.9) for most noise levels (0-
568 100%) (**Figure 6B**). At lower π values, factor recovery in ACMTF-R deteriorated due to the strong
569 focus on predicting $\boldsymbol{y}$, due to ACMTF-R not finding the hidden component (**Supplementary Figure**
570 **11**). At very high noise levels (100+%), neither ACMTF nor ACMTF-R were able to recover the
571 hidden component.

572 The recovered shared subject mode loadings of the hidden component showed a consistent recovery
573 at most noise levels (0-100%) and intermediate values of π (0.4-0.9) (**Figure 6C**). Similarly, recovery
574 of the feature and time mode loadings was consistent at most noise levels (0-100%) and intermediate
575 values of π (0.4-0.9) (**Figure 6D-G**). Taken together, the simulation revealed that recovery of the
576 hidden component by ACMTF-R is insensitive to biologically realistic noise levels on $\boldsymbol{y}$.

577

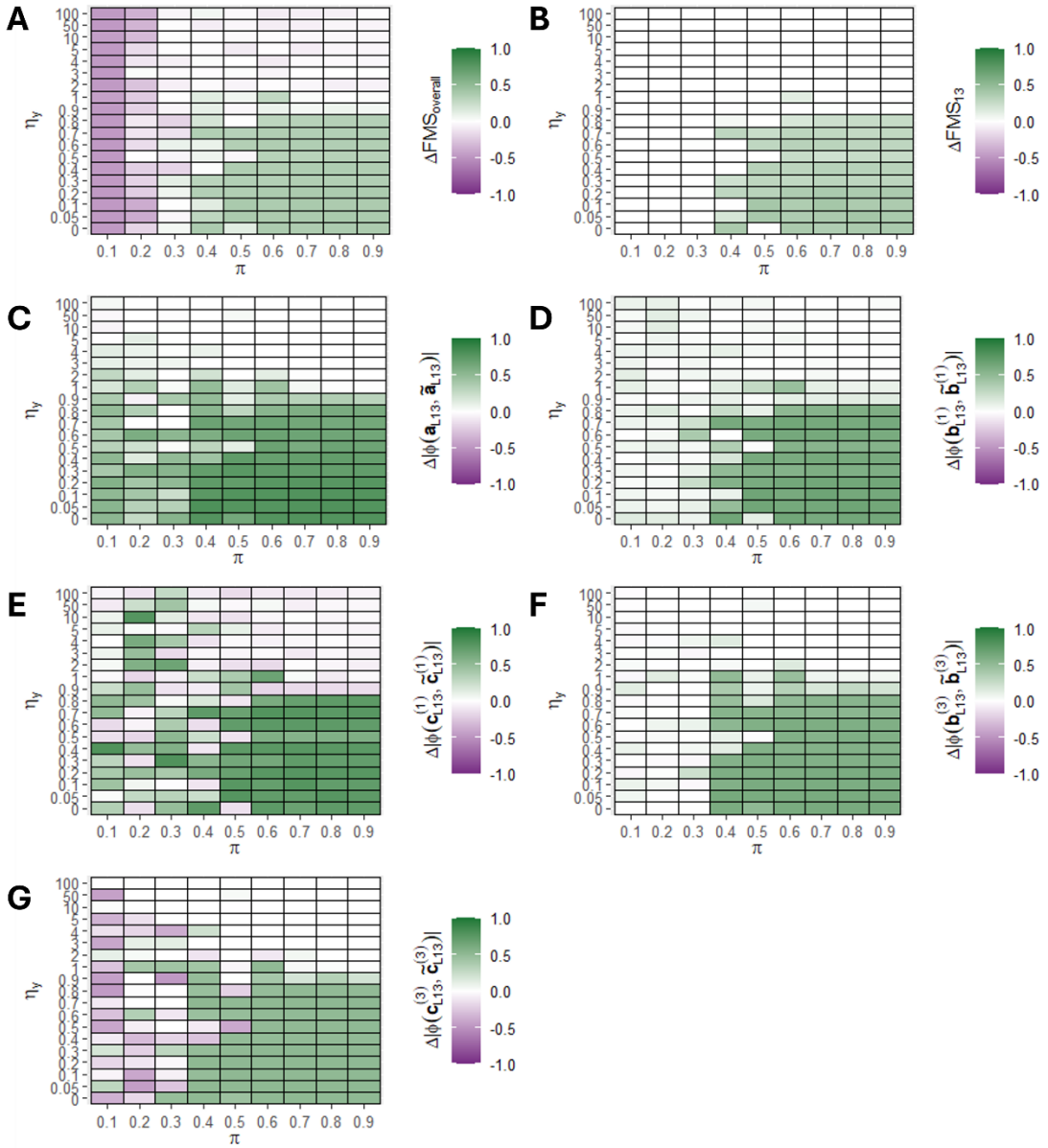


Figure 6. The improvement in factor recovery by ACMTF-R under different amounts of noise on $\boldsymbol{y}$. One hundred randomly initialised eight-component ACMTF-R and ACMTF models were fitted for each combination noise on $\boldsymbol{y}$ (η_y) and tuning parameter (π), after which the lowest-loss models were compared. The improvement in factor recovery is computed by subtracting ACMTF performance from ACMTF-R performance in all panels. Improvement per setting for (A) overall factor recovery, (B) factor recovery of the hidden component, (C) shared subject mode recovery of the hidden component, (D) feature mode recovery of $\underline{\boldsymbol{X}}^{(1)}$ for the hidden component, (E) time mode recovery of $\underline{\boldsymbol{X}}^{(1)}$ for the hidden component, (F) feature mode recovery of $\underline{\boldsymbol{X}}^{(3)}$ for the hidden component, (G) time mode recovery of $\underline{\boldsymbol{X}}^{(3)}$ for the hidden component. Improvement compared to ACMTF is shown in green, while deterioration is shown in purple. A version of this figure without subtraction of ACMTF performance is shown in **Supplementary Figure 11**.

590 *Application: ACMTF-R enhances multi-omics integration compared to ACMTF and NPLS*

591 To demonstrate the advantage of ACMTF-R for integrating multi-omics data, we analysed a
592 longitudinal dataset comprising human milk (HM) metabolomics, HM microbiome and infant faecal
593 microbiome data, using pre-pregnancy BMI (ppBMI) as the outcome ($\boldsymbol{y}$). To contextualise
594 performance, we evaluated the results against ACMTF, which focuses only on the CLD structure,
595 and N-way Partial Least Squares (NPLS) models per block [48], which focus only on prediction. We
596 expected ACMTF-R to identify more strongly maternal ppBMI associated components compared to
597 ACMTF.

598 A combined random initialisation and cross-validation procedure was used to select the number of
599 components for ACMTF and ACMTF-R (**Methods**). The following tuning parameter values were
600 investigated: $\pi = 0.75$, $\pi = 0.80$, $\pi = 0.85$, and $\pi = 0.90$. For each π value, 10 randomly initialised
601 models were fitted and a 10-fold cross-validation was performed across models with one to five
602 components (**Supplementary Figure 12**). This analysis revealed that one component was optimal for
603 all ACMTF-R settings. For ACMTF, a similar procedure indicated that two components were
604 appropriate, each explaining a different source of variation (**Supplementary Figure 13**). For NPLS,
605 cross-validation indicated that one component minimised RMSECV for the infant faecal and HM
606 microbiome data, while two components were required for the HM metabolomics data
607 (**Supplementary Figure 14**).

608 After selecting the number of components, 100 randomly initialised one-component ACMTF-R
609 models were fitted for each π value. The model with the lowest overall loss in each case was selected,
610 and the variance explained was computed (**Table 1**). To contextualise performance, the ACMTF
611 model explained 14.34%, 9.24%, and 7.03% of the variation in the infant faecal microbiome, HM
612 microbiome, and HM metabolomics data blocks, respectively, while explaining 9.28% of the
613 variation in $\boldsymbol{y}$. Additionally, the NPLS models explained 5.78%, 5.41%, and 8.25% of the variation
614 in the infant faecal microbiome, HM microbiome, and HM metabolomics datasets, respectively,
615 while explaining 11.97%, 11.22%, and 35.95% of the variation in $\boldsymbol{y}$.

616 The ACMTF-R model with $\pi = 0.90$ explained 2.08-8.31% of the variation in the independent data
617 blocks, and 24.08% of the variation in $\boldsymbol{y}$. At $\pi = 0.85$, this dropped to between 1.78-5.38% of the
618 variation in the independent data blocks, and 62.06% in $\boldsymbol{y}$. At $\pi = 0.80$ and $\pi = 0.75$, the models

619 were most heavily tuned towards predicting $\boldsymbol{y}$, yielding 1-2% of explained variation in the
620 independent data blocks and capturing 90+% of the variation in $\boldsymbol{y}$ – a clear indication of overfitting.

621 **Table 1. Variance explained per dataset for the fitted NPLS, ACMTF and ACMTF-R models**. F: the
622 number of components.

Dataset	Model	F	Variation explained (%)			
			Infant faecal microbiome	Mother milk microbiome	Mother milk metabolome	ppBMI ($\boldsymbol{y}$)
Infant faecal microbiome	NPLS	1	5.78	-	-	11.97
HM microbiome	NPLS	1	-	5.41	-	11.22
HM metabolome	NPLS	2	-	-	8.25	35.95
All	ACMTF-R ($\pi = 0.75$)	1	1.52	1.15	1.24	96.55
	ACMTF-R ($\pi = 0.80$)	1	2.45	1.91	1.41	89.97
	ACMTF-R ($\pi = 0.85$)	1	5.38	4.35	1.78	62.06
	ACMTF-R ($\pi = 0.90$)	1	8.31	6.69	2.08	24.08
	ACMTF	2	14.34	9.24	7.03	9.28

623

624 MLR was performed on the subject mode loadings to test for associations with subject metadata,
625 including maternal pre-pregnancy BMI, infant growth at six months, birth mode, maternal secretor
626 (Se), and maternal Lewis (Le) blood group status (**Table 2**). Maternal Se and Le status determine the
627 presence of specific fucosylated human milk oligosaccharides, which may influence the composition
628 of the infant faecal microbiome.

629 The subject mode loadings of the NPLS models all contained at least one component that was
630 associated with maternal pre-pregnancy BMI. While both components of the ACMTF model were
631 associated with both ppBMI and maternal secretor status, the first component was most strongly
632 associated with maternal secretor status and the second was most strongly associated with ppBMI.
633 This was not the case for the ACMTF-R models with $\pi = 0.90$ and $\pi = 0.85$, which both contained
634 subject mode loadings associated exclusively with ppBMI after correction. The subject mode
635 loadings of the ACMTF-R models with lower values of π were strongly associated with ppBMI due
636 to overfitting.

Due to the extraction of a component exclusively related to maternal ppBMI, as well as the trade-off between explaining the independent data and predicting $\boldsymbol{y}$, the ACMTF-R model using $\pi = 0.90$ was selected for further interpretation.

Table 2. MLR results for the subject mode loadings of the created multi-way models. ppBMI: maternal pre-pregnancy body mass index. *Se*: maternal secretor status. *Le*: maternal Lewis status. WHZ: infant weight-of-height z-score at 6 months. Subject mode loadings were normalised to norm 1 prior to MLR fitting for comparability of the results between the models. Values indicate MLR coefficients times one thousand. Stars indicate the original p-values from the MLR model (*: $p < 0.05$, **: $p < 0.01$, ***: $p < 0.001$), while hats indicate Benjamini-Hochberg corrected p-values (^: $q < 0.05$, ^^: $q < 0.01$, ^^^: $q < 0.001$). Coefficients with a significant p-value after correction are made bold. The correlation matrix of the metadata is reported in **Supplementary Table 2**.

Dataset	Model	F	ppBMI	Birth mode	*Se*	*Le*	WHZ
Infant faecal microbiome	NPLS	1	**5.9*****, ^^^	26.2	18.7	-67.4	1.5
HM microbiome	NPLS	1	**5.3*****, ^^^	22.5	10.1	-25.4	-0.6
HM metabolome	NPLS	1	**8.7*****, ^^^	11.1	19.7	-78.3	**-12.6***, ^
		2	-0.2	-12.8	**-111*****, ^^^	43.0	7.0
All	ACMTF-R ($\pi = 0.75$)	1	**15.0*****, ^^^	6.7	3.0*	-18.5	-0.9
	ACMTF-R ($\pi = 0.80$)	1	**-14.7*****, ^^^	-10.8	-5.3*	31.7	1.4
	ACMTF-R ($\pi = 0.85$)	1	**-12.5*****, ^^^	-18.1	-12.3*	59.6	2.4
	ACMTF-R ($\pi = 0.90$)	1	**-8.3*****, ^^^	-21.5	-21.8	79.4	2.8
	ACMTF	1	**-2.9***, ^	-2.0	**-74.8*****, ^^^	65.5	3.2
		2	**-4.4*****, ^^	-40.1	**-41.3***, ^	72.3	1.0

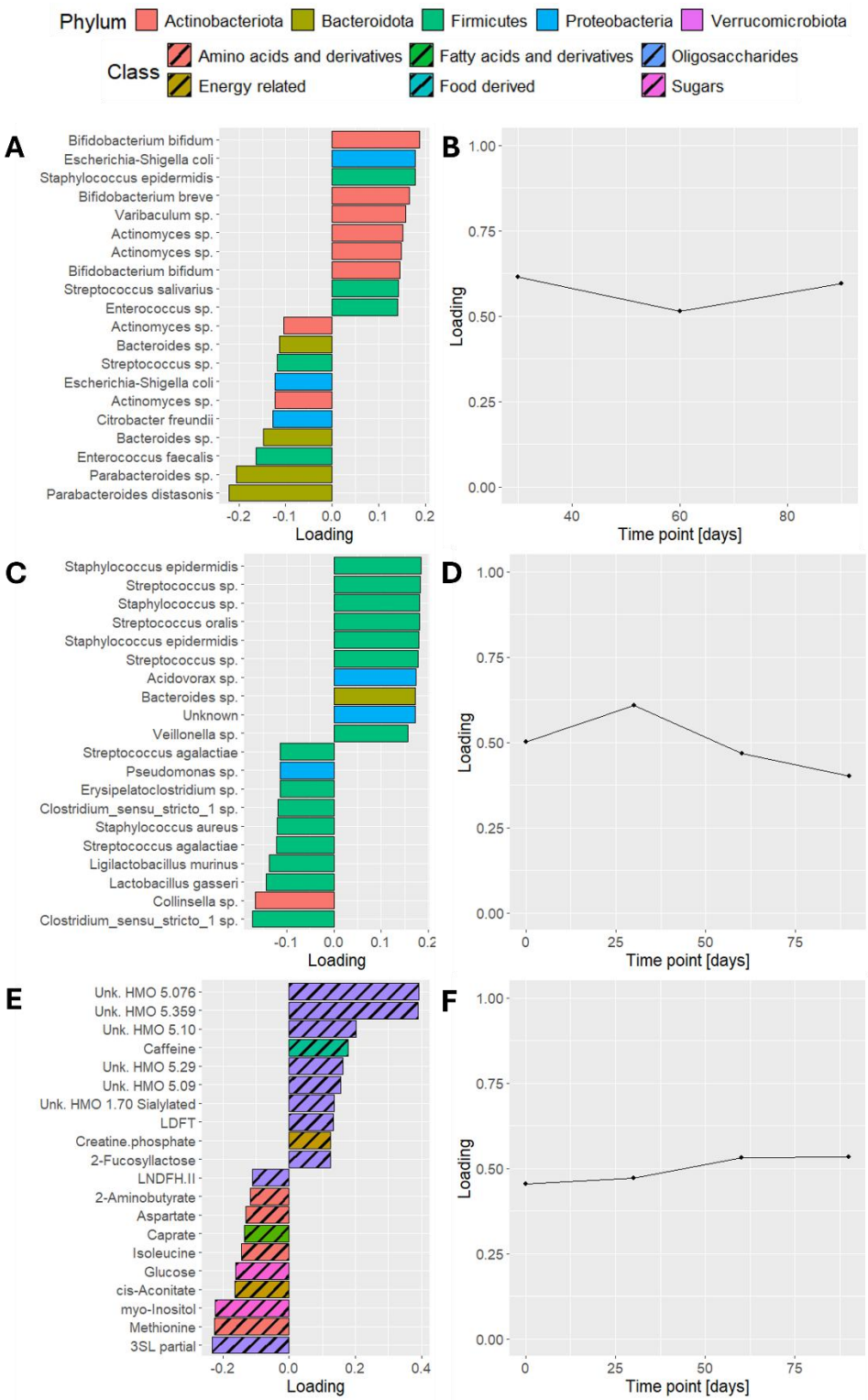
The $\boldsymbol{\Lambda}$-matrix of the ACMTF-R model using $\pi = 0.90$ was examined to interpret the common, local, and distinct (CLD) structure of the component. In this model, λ values of the only available component were equal to 0.34, 0.29, and 0.15, for the infant faecal microbiome, HM microbiome, and HM metabolomics, respectively. Based on the relative magnitudes of the λ values, the component appears to be primarily local to the infant faecal and HM microbiome blocks, with a

654 minor contribution of the HM metabolomics block. This suggests that ppBMI-associated variation
655 was mainly present in the microbiomes of mother and infant.

656 In the infant faecal microbiome, genera associated with higher ppBMI included *Bifidobacterium*
657 *bifidum*, *Escherichia coli, Staphylococcus epidermidis,* and *B. breve* (**Figure 7A**). Genera enriched
658 in infants of low-ppBMI mothers included *Parabacteroides, Enterococcus* and *Streptococcus*. In the
659 HM microbiome, higher ppBMI was associated with increased abundance of *Staphylococcus* and
660 *Streptococcus* spp., while genera associated with lower ppBMI included various taxa such as
661 *Clostridium sensu stricto*, *Collinsella*, and *Lactobacillus* (**Figure 7C**). In the HM metabolome, higher
662 ppBMI was associated with increased levels of many unknown HMOs, caffeine, and LDFT (**Figure**
663 **7E**). Metabolites with increased levels in low-ppBMI mothers included several amino acids and
664 related compounds such as 2-aminobutyrates, aspartate, isoleucine, and methionine.

665 Due to the pre-processing of the data, the largest absolute time loading in the ACMTF-R model
666 indicates the time points at which inter-subject differences related to ppBMI are greatest (**Figure**
667 **7B,D,F**). Overall, the time loadings were relatively flat, suggesting that ppBMI-associated variation
668 remained stable throughout the sampling period. In the HM microbiome, however, these
669 differences peaked at day 30 and declined thereafter.

670 Taken together, ACMTF-R was able to identify variation that was strongly and exclusively
671 associated with maternal ppBMI. Additionally, it revealed that this variation was primarily local to
672 the infant faecal and HM microbiome, with a limited contribution from HM metabolome. This
673 highlights the value of incorporating supervision to isolate outcome-relevant signals in complex
674 multi-omics data.



675

676 **Figure 7. Feature and time mode loadings of the selected ACMTF-R model.** Feature mode loadings
677 of the selected ACMTF-R model using $\pi = 0.90$ describing (A) the infant faecal microbiome, (C)
678 the HM microbiome, and (E) the HM metabolomics datasets. Time loadings of the same model are
679 shown in the same order in (B,D,F). Signs of the feature mode loadings were adjusted post-hoc such
680 that positive values consistently corresponded to higher ppBMI and negative values to lower ppBMI.

Discussion

In this study, we presented ACMTF-Regression (ACMTF-R) as an extension of Advanced Coupled Matrix and Tensor Factorization (ACMTF) that incorporates a regression term to model common, local and distinct (CLD) variation related to a dependent variable. Through simulations, we demonstrated its ability to capture a hidden component, its robustness to noise, and its flexibility in tuning between data exploration and predicting. Our application demonstrated how ACMTF-R finds novel relationships for the effect of maternal ppBMI on the HM metabolome, HM microbiome and infant faecal microbiome.

ACMTF-R extends the capabilities of ACMTF by allowing explicit modelling of outcome-associated variation, bridging the gap between exploratory data integration and supervised learning [10]. Compared to NPLS, which is designed for predictive modelling but lacks the ability to separate common, local, and distinct variation across multiple blocks, ACMTF-R provides a more nuanced view of multi-way data structures [19]. Unlike classical ACMTF, which only uncovers underlying data patterns, ACMTF-R allows for targeted analysis of biological variation of interest [33]. The ability to tune between the two makes it particularly useful for cases where a balance between exploration and prediction is necessary.

Our simulations revealed that ACMTF-R reliably recovers the underlying CLD-structure of multi-way datasets across a range of settings. We found that the tuning parameter (π) governs the trade-off between reconstructing the input data and predicting the outcome variable. While $\pi = 1$ corresponds to regular ACMTF, decreasing π enhances the identification of factors associated with the dependent variable, even revealing weak structures that would otherwise remain undetected. However, tuning too far towards $\boldsymbol{y}$ ($\pi < 0.3$) leads to overfitting and an inaccurate reconstruction. The observation that intermediate values of π (0.6-0.9) lead to recovery of the hidden local component is largely in agreement with PCovR and MCovR [20, 22].

Additionally, the simulations revealed that the hidden component needed to be sufficiently small ($\delta < 0.05$) for ACMTF to fail in its recovery. However, this finding is largely dependent on the relative sizes of the other components, which we have kept at Frobenius norm 1 in our simulations. Further research is needed to investigate the recovery of a hidden component using ACMTF and ACMTF-R in other scenarios.

710 Finally, the simulations showed that recovery of the hidden component using ACMTF-R was most
711 accurate when a biologically realistic amount of noise was present in the data ($\eta_X \leq 0.5$, $\eta_y < 1$).
712 The observation that neither ACMTF nor ACMTF-R can identify the hidden component in the no-
713 noise case is surprising. This may occur when the noise has a smoothing effect on the loss space,
714 resulting in the global minimum being easier to identify, or reduces the collinearity of the loadings
715 in a statistical phenomenon comparable to ridge regression [55]. While the noise level for a real
716 dataset is often not known, it is reassuring that ACMTF-R can reliably capture the $\boldsymbol{y}$-associated
717 subject mode loadings for most tested noise levels (**Supplementary Figure 10-11**).

718 Application to a longitudinal multi-omics dataset integrating human milk microbiome, human milk
719 metabolome, and infant gut microbiome data further highlighted the utility of ACMTF-R by
720 recovering an exclusively ppBMI-associated component [48]. In the infant faecal microbiome,
721 elevated abundances of *Enterococcus* sp., *E. coli*, *Clostridium* sp., and *Staphylococcus epidermidis*
722 in infants born to high-ppBMI mothers are consistent with prior findings linking maternal obesity
723 to pro-inflammatory microbial profiles [56, 57]. In contrast, the increased abundance of
724 *Bifidobacterium breve* and *Bifidobacterium bifidum* in the same group contradicts earlier studies,
725 which generally associate these taxa with leaner maternal phenotypes [56, 57].

726 Interestingly, the genera enriched in infants from low-ppBMI mothers represent a potential novel
727 finding, as comparable reports are currently lacking and warrant further investigation. In the HM
728 microbiome, enrichment of *Streptococcus* and *Staphylococcus* spp. in high-ppBMI mothers is in
729 agreement with previous research [58–60]. However, comparable data for taxa enriched in low-
730 ppBMI mothers were not available for direct comparison.

731 In the HM metabolomics data, elevated levels of lactate and multiple unidentified HMOs were
732 observed in high-ppBMI mothers. While the relationship between ppBMI and HMO concentrations
733 has been investigated in several studies, the results remain inconsistent [61, 62]. One study did
734 report higher levels of a lactate derivative in the milk of obese mothers [61], suggesting a possible
735 link to altered maternal energy metabolism. Conversely, the observation of increased
736 concentrations of several amino acids in low-ppBMI mothers is supported by previous findings [61],
737 as well as by recent evidence showing enrichment of amino acid biosynthesis pathways in this group
738 [63]. These results show that ACMTF-R provides additional biological insights compared to ACMTF
739 and N-PLS, effectively disentangling common, local and distinct variation related to maternal
740 ppBMI.

While ACMTF-R demonstrates strong performance in identifying biologically meaningful patterns, several challenges remain. The choice of the tuning parameter π is dataset-dependent, requiring careful selection through cross-validation or heuristic approaches. Additionally, in multi-omics applications, different data blocks may contain varying amounts of biologically relevant information. Allowing block-specific π values could improve model flexibility, enabling certain datasets to contribute more strongly to the dependent variable while ensuring others remain focused on data structure. Future research should explore whether adaptive weighting strategies or block-specific hyperparameter tuning can further improve performance.

For ACMTF and ACMTF-R, the single guiding metric for selecting the number of components was often the stability of the decomposition through the factor match score (FMS) [15, 54]. This paper has expanded upon this approach by formalising FMS for a cross-validation and random initialisation scheme, as well as developing the degeneracy score diagnostic to identify overfactorisation. However, this approach is computationally intensive for the simulations. Future work should investigate if this approach could be accelerated through the Limited-memory Broyden-Fletcher-Goldfarb-Shanno [40–42] and AO-ADMM algorithms [64–66].

While Factor Match Score (FMS) has been a useful diagnostic for model selection, it suffers from the phenomenon of all blocks contributing to all components in ACMTF outlined above. Due to the way the structural model of ACMTF is implemented, F trilinear components are fitted per data block, even when the rank of that block is lower than F [33] This causes the creation of superfluous feature and time mode loading vectors that contribute to the model due to having non-zero λ values. As a result, the FMS is negatively impacted by considering these components in the calculation. This problem may be avoided by more strongly penalising the $\mathbf{\Lambda}$-matrix through replacement of the L1 penalty term by a smooth L0 penalty, as presented in Relaxed ACMTF (RACMTF) [67, 68]. Future research should investigate if this approach yields more stable ACMTF and ACMTF-R models for multi-omics data.

The multi-omics application presented in this study highlights a critical shortcoming of ACMTF in the identification of common, local, and distinct (CLD) sources of variation by revealing that every data block contributes to all identified components. However, the relative magnitude of the contributing blocks captured by λ still offer a useful interpretation. For example, a $\mathbf{\Lambda}$-matrix threshold could be applied post-hoc to identify CLD-structures [33]. Another approach could be to pre-define the expected number of shared and distinct components, as suggested by Common and

Discriminative subspace Non-negative Tensor Factorization (CDNTF) [69]. Future research should investigate if either approach or a combination of both might yield more interpretable ACMTF models for multi-omics data.

Finally, treating $\boldsymbol{y}$ as an additional data block for ACMTF could offer a different approach to incorporating outcome variation into ACMTF-R. By embedding $\boldsymbol{y}$ within the multi-way decomposition framework, rather than optimizing it separately, the model might be able to better capture latent structures linking $\boldsymbol{y}$ to the input datasets. However, this approach would lose the predictive aspect of ACMTF-R. Future work should focus on the comparability between ACMTF-R and adding $\boldsymbol{y}$ as a block to ACMTF.

Conclusions

Our results demonstrate that ACMTF-R is a powerful framework for multi-way data integration that extends ACMTF by incorporating outcome-associated variation. Through simulations, we showed that ACMTF-R can recover a small hidden component, is robust to noise, and provides flexible tuning between data exploration and outcome prediction. Its application to multi-omics data highlights how ACMTF-R finds novel relationships for the effect of maternal ppBMI on the HM metabolome, HM microbiome and infant faecal microbiome.

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Availability of the data and materials

The package used during the current study is available in the GitHub repository https://github.com/GRvanderPloeg/CMTFtoolbox. The datasets generated and/or analysed during the current study are available in the GitHub repository https://github.com/GRvanderPloeg/CMTF_toolbox_project.

Competing interests

The authors declare that they have no competing interests.

Funding

GRvdP was funded by a grant from the University of Amsterdam, Research Priority Area on Personal Microbiome Health.

Fred White was funded by a grant from the University of Amsterdam, Data Science Centre.

Authors' contributions

G.R. van der Ploeg: conceptualization, visualization, formal analysis, methodology, programming, writing – original draft; F.T.G. White: conceptualization, methodology, programming, writing – original draft; R.R. Jakobsen: programming, writing – review & editing; J.A. Westerhuis: conceptualization, methodology, supervision, formal analysis, writing – original draft; A. Heintz-Buschart: conceptualization, formal analysis, supervision, project administration, writing – original draft; A.K. Smilde: conceptualization, methodology, supervision, funding acquisition, writing – review & editing.

Acknowledgements

We want to thank Daniël Rademaker (RU/UvA) for their useful discussions and suggestions.

References

1. Capobianco E. Ten challenges for systems medicine. Front Genet. 2012;3:193.

2. Davis-Turak J, Courtney SM, Hazard ES, Glen WB, Da Silveira WA, Wesselman T, et al. Genomics pipelines and data integration: challenges and opportunities in the research setting. Expert Rev Mol Diagn. 2017;17:225–37. https://doi.org/10.1080/14737159.2017.1282822.

3. Zhao Z, Jin VX, Huang Y, Guda C, Ruan J. Frontiers in integrative genomics and translational bioinformatics. BioMed Res Int. 2015;2015.

4. Bordbar A. Interpreting the deluge of omics data: new approaches offer new possibilities. Blood Transfus. 2017;15:189.

5. Chen C, McGarvey PB, Huang H, Wu CH. Protein Bioinformatics Infrastructure for the Integration and Analysis of Multiple High-Throughput "omics" Data. Adv Bioinforma. 2010;2010:1–19. https://doi.org/10.1155/2010/423589.

6. Misra BB, Langefeld C, Olivier M, Cox LA. Integrated omics: tools, advances and future approaches. J Mol Endocrinol. 2019;62:R21–45.

7. Vitorino R. Transforming Clinical Research: The Power of High-Throughput Omics Integration. Proteomes. 2024;12:25. https://doi.org/10.3390/proteomes12030025.

8. van der Kloet FM, Sebastián-León P, Conesa A, Smilde AK, Westerhuis JA. Separating common from distinctive variation. BMC Bioinformatics. 2016;17:S195. https://doi.org/10.1186/s12859-016-1037-2.

9. Måge I, Smilde AK, van der Kloet FM. Performance of methods that separate common and distinct variation in multiple data blocks. J Chemom. 2019;33:e3085. https://doi.org/10.1002/cem.3085.

10. Acar E, Papalexakis EE, Gürdeniz G, Rasmussen MA, Lawaetz AJ, Nilsson M, et al. Structure-revealing data fusion. BMC Bioinformatics. 2014;15:239. https://doi.org/10.1186/1471-2105-15-239.

11. Tan CS, Salim A, Ploner A, Lehtiö J, Chia KS, Pawitan Y. Correlating gene and protein expression data using Correlated Factor Analysis. BMC Bioinformatics. 2009;10:272. https://doi.org/10.1186/1471-2105-10-272.

12. Xiao X, Moreno-Moral A, Rotival M, Bottolo L, Petretto E. Multi-tissue analysis of co-expression networks by higher-order generalized singular value decomposition identifies functionally coherent transcriptional modules. PLoS Genet. 2014;10:e1004006.

13. Berger JA, Hautaniemi S, Mitra SK, Astola J. Jointly analyzing gene expression and copy number data in breast cancer using data reduction models. IEEE/ACM Trans Comput Biol Bioinform. 2006;3:2–16.

14. Song Y, Westerhuis JA, Smilde AK. Separating common (global and local) and distinct variation in multiple mixed types data sets. J Chemom. 2020;34:e3197. https://doi.org/10.1002/cem.3197.

15. Acar E, Kolda TG, Dunlavy DM. All-at-once Optimization for Coupled Matrix and Tensor Factorizations. 2011.

16. Acar E, Bro R, Smilde A. Data Fusion in Metabolomics Using Coupled Matrix and Tensor Factorizations. Proc IEEE. 2015;103:1602. https://doi.org/10.1109/JPROC.2015.2438719.

17. Acar E, Bro R, Smilde AK. Data Fusion in Metabolomics Using Coupled Matrix and Tensor Factorizations. Proc IEEE. 2015;103:1602–20. https://doi.org/10.1109/JPROC.2015.2438719.

18. Acar E, Dunlavy DM, Kolda TG, Mørup M. Scalable tensor factorizations for incomplete data. Chemom Intell Lab Syst. 2011;106:41–56. https://doi.org/10.1016/j.chemolab.2010.08.004.

19. Bro R. Multiway calibration. Multilinear PLS. J Chemom. 1996;10:47–61. https://doi.org/10.1002/(SICI)1099-128X(199601)10:1<47::AID-CEM400>3.0.CO;2-C.

20. Vervloet M, Kiers HAL, Noortgate WV den, Ceulemans E. PCovR: An R Package for Principal Covariates Regression. J Stat Softw. 2015;65:1–14. https://doi.org/10.18637/jss.v065.i08.

21. De Jong S, Kiers HA. Principal covariates regression: part I. Theory. Chemom Intell Lab Syst. 1992;14:155–64.

22. Smilde AK, Kiers HAL. Multiway covariates regression models. J Chemom. 1999;13:31–48. https://doi.org/10.1002/(SICI)1099-128X(199901/02)13:1<31::AID-CEM528>3.0.CO;2-P.

23. Kiers HAL. Towards a standardized notation and terminology in multiway analysis. J Chemom. 2000;14:105–22. https://doi.org/10.1002/1099-128X(200005/06)14:3<105::AID-CEM582>3.0.CO;2-I.

24. Kolda TG. Multilinear operators for higher-order decompositions. Sandia National Laboratories (SNL), Albuquerque, NM, and Livermore, CA …; 2006.

25. De Lathauwer L, De Moor B, Vandewalle J. A Multilinear Singular Value Decomposition. SIAM J Matrix Anal Appl. 2000;21:1253–78. https://doi.org/10.1137/S0895479896305696.

26. Styan GP. Hadamard products and multivariate statistical analysis. Linear Algebra Its Appl. 1973;6:217–40.

27. McDonald RP. A simple comprehensive model for the analysis of covariance structures: Some remarks on applications. Br J Math Stat Psychol. 1980;33:161–83. https://doi.org/10.1111/j.2044-8317.1980.tb00606.x.

28. Rao CR. GENERALIZED INVERSE OF A MATRIX AND ITS APPLICATIONS. In: Le Cam LM, Neyman J, Scott EL, editors. Theory of Statistics. University of California Press; 1972. p. 601–20. https://doi.org/10.1525/9780520325883-032.

29. Ben-Israel A, Greville TN. Generalized inverses: theory and applications. Springer Science & Business Media; 2006.

30. Carroll JD, Chang J-J. Analysis of individual differences in multidimensional scaling via an N-way generalization of “Eckart-Young” decomposition. Psychometrika. 1970;35:283–319.

31. Harshman RA. FOUNDATIONS OF THE PARAFAC PROCEDURE: MODELS AND CONDITIONS FOR AN “EXPLANATORY” MULTIMODAL FACTOR ANALYSIS. :84.

32. Bro R. MULTI-WAY ANALYSIS IN THE FOOD INDUSTRY.

33. Acar E, Papalexakis EE, Gürdeniz G, Rasmussen MA, Lawaetz AJ, Nilsson M, et al. Structure-revealing data fusion. BMC Bioinformatics. 2014;15:239. https://doi.org/10.1186/1471-2105-15-239.

34. Lee S-I, Lee H, Abbeel P, Ng AY. Efficient l1 regularized logistic regression. In: Aaai. 2006. p. 401–8.

35. Bertrand D, Qannari EM, Vigneau E. Latent root regression analysis: an alternative method to PLS. Chemom Intell Lab Syst. 2001;58:227–34. https://doi.org/10.1016/S0169-7439(01)00161-7.

36. Webster JT, Gunst RF, Mason RL. Latent Root Regression Analysis. Technometrics. 1974;16:513–22. https://doi.org/10.1080/00401706.1974.10489232.

37. Harshman RA. Foundations of the PARAFAC procedure: Models and conditions for an" explanatory" multimodal factor analysis. 1970.

38. Bro R. PARAFAC. Tutorial and applications. Chemom Intell Lab Syst. 1997;:23.

39. Melville J. mize: Unconstrained Numerical Optimization Algorithms. 2019.

40. Liu DC, Nocedal J. On the limited memory BFGS method for large scale optimization. Math Program. 1989;45:503–28. https://doi.org/10.1007/BF01589116.

41. Malouf R. A comparison of algorithms for maximum entropy parameter estimation. In: proceedings of the 6th conference on Natural language learning - Volume 20. USA: Association for Computational Linguistics; 2002. p. 1–7. https://doi.org/10.3115/1118853.1118871.

42. Andrew G, Gao J. Scalable training of L1-regularized log-linear models. In: Proceedings of the 24th international conference on Machine learning. New York, NY, USA: Association for Computing Machinery; 2007. p. 33–40. https://doi.org/10.1145/1273496.1273501.

43. Burt C. The factorial study of temperamental traits. Br J Psychol. 1948.

903 44. Tucker LR. A method for synthesis of factor analysis studies. Educational Testing Service Princeton,
904 NJ; 1951.

905 45. Lorenzo-Seva U, ten Berge JMF. Tucker's congruence coefficient as a meaningful index of factor
906 similarity. Methodol Eur J Res Methods Behav Soc Sci. 2006;2:57–64. https://doi.org/10.1027/1614-
907 2241.2.2.57.

908 46. Kuhn HW. The Hungarian method for the assignment problem. Nav Res Logist Q. 1955;2:83–97.
909 https://doi.org/10.1002/nav.3800020109.

910 47. Kuhn HW. Variants of the hungarian method for assignment problems. Nav Res Logist Q.
911 1956;3:253–8. https://doi.org/10.1002/nav.3800030404.

912 48. Jakobsen R, Ploeg GR, Sundekilde U, Astono J, Poulsen K, Fuglsang J, et al. Supervised Modelling of
913 Longitudinal Human Milk and Infant Gut Microbiome Reveal Maternal Pre-Pregnancy BMI and Early
914 Life Growth Interactions. 2025. https://doi.org/10.21203/rs.3.rs-6244750/v1.

915 49. R Core Team R. R: A language and environment for statistical computing. 2013.

916 50. Gloor GB, Macklaim JM, Pawlowsky-Glahn V, Egozcue JJ. Microbiome Datasets Are Compositional:
917 And This Is Not Optional. Front Microbiol. 2017;8:2224. https://doi.org/10.3389/fmicb.2017.02224.

918 51. Aitchison J. The Statistical Analysis of Compositional Data. J R Stat Soc Ser B Methodol.
919 1982;44:139–60. https://doi.org/10.1111/j.2517-6161.1982.tb01195.x.

920 52. Bro R, Smilde AK. Centering and scaling in component analysis. J Chemom. 2003;17:16–33.
921 https://doi.org/10.1002/cem.773.

922 53. van der Ploeg GR, Westerhuis JA, Heintz-Buschart A, Smilde AK. parafac4microbiome: Exploratory
923 analysis of longitudinal microbiome data using Parallel Factor Analysis. 2024.
924 https://doi.org/10.1101/2024.05.02.592191.

925 54. Li L, Yan S, Horner D, Rasmussen MA, Smilde AK, Acar E. Revealing static and dynamic biomarkers
926 from postprandial metabolomics data through coupled matrix and tensor factorizations. 2024.

927 55. Schott JR. Matrix analysis for statistics. John Wiley & Sons; 2016.

928 56. Singh SB, Madan J, Coker M, Hoen A, Baker ER, Karagas MR, et al. Associations of maternal pre-
929 pregnancy BMI and gestational weight gain with the infant gut microbiome differ according to delivery
930 mode. Int J Obes 2005. 2019;44:23.

931 57. Collado MC, Isolauri E, Laitinen K, Salminen S. Effect of mother's weight on infant's microbiota
932 acquisition, composition, and activity during early infancy: a prospective follow-up study initiated in
933 early pregnancy. Am J Clin Nutr. 2010;92:1023–30.

934 58. Lundgren SN, Madan JC, Karagas MR, Morrison HG, Hoen AG, Christensen BC. Microbial
935 communities in human milk relate to measures of maternal weight. Front Microbiol. 2019;10:2886.

936 59. Cabrera-Rubio R, Collado MC, Laitinen K, Salminen S, Isolauri E, Mira A. The human milk
937 microbiome changes over lactation and is shaped by maternal weight and mode of delivery1234. Am
938 J Clin Nutr. 2012;96:544–51. https://doi.org/10.3945/ajcn.112.037382.

60. Cortés-Macías E, Selma-Royo M, Rio-Aige K, Bäuerl C, Rodríguez-Lagunas MJ, Martínez-Costa C, et al. Distinct breast milk microbiota, cytokine, and adipokine profiles are associated with infant growth at 12 months: an in vitro host–microbe interaction mechanistic approach. Food Funct. 2023;14:148–59.

61. Isganaitis E, Venditti S, Matthews TJ, Lerin C, Demerath EW, Fields DA. Maternal obesity and the human milk metabolome: associations with infant body composition and postnatal weight gain. Am J Clin Nutr. 2019;110:111–20.

62. Bardanzellu F, Puddu M, Peroni DG, Fanos V. The Human Breast Milk Metabolome in Overweight and Obese Mothers. Front Immunol. 2020;11:1533. https://doi.org/10.3389/fimmu.2020.01533.

63. Sivalogan K, Liang D, Accardi C, Diaz-Artiga A, Hu X, Mollinedo E, et al. Human Milk Composition Is Associated with Maternal Body Mass Index in a Cross-Sectional, Untargeted Metabolomics Analysis of Human Milk from Guatemalan Mothers. Curr Dev Nutr. 2024;8:102144. https://doi.org/10.1016/j.cdnut.2024.102144.

64. Schenker C, Cohen JE, Acar E. A Flexible Optimization Framework for Regularized Matrix-Tensor Factorizations With Linear Couplings. IEEE J Sel Top Signal Process. 2021;15:506–21. https://doi.org/10.1109/jstsp.2020.3045848.

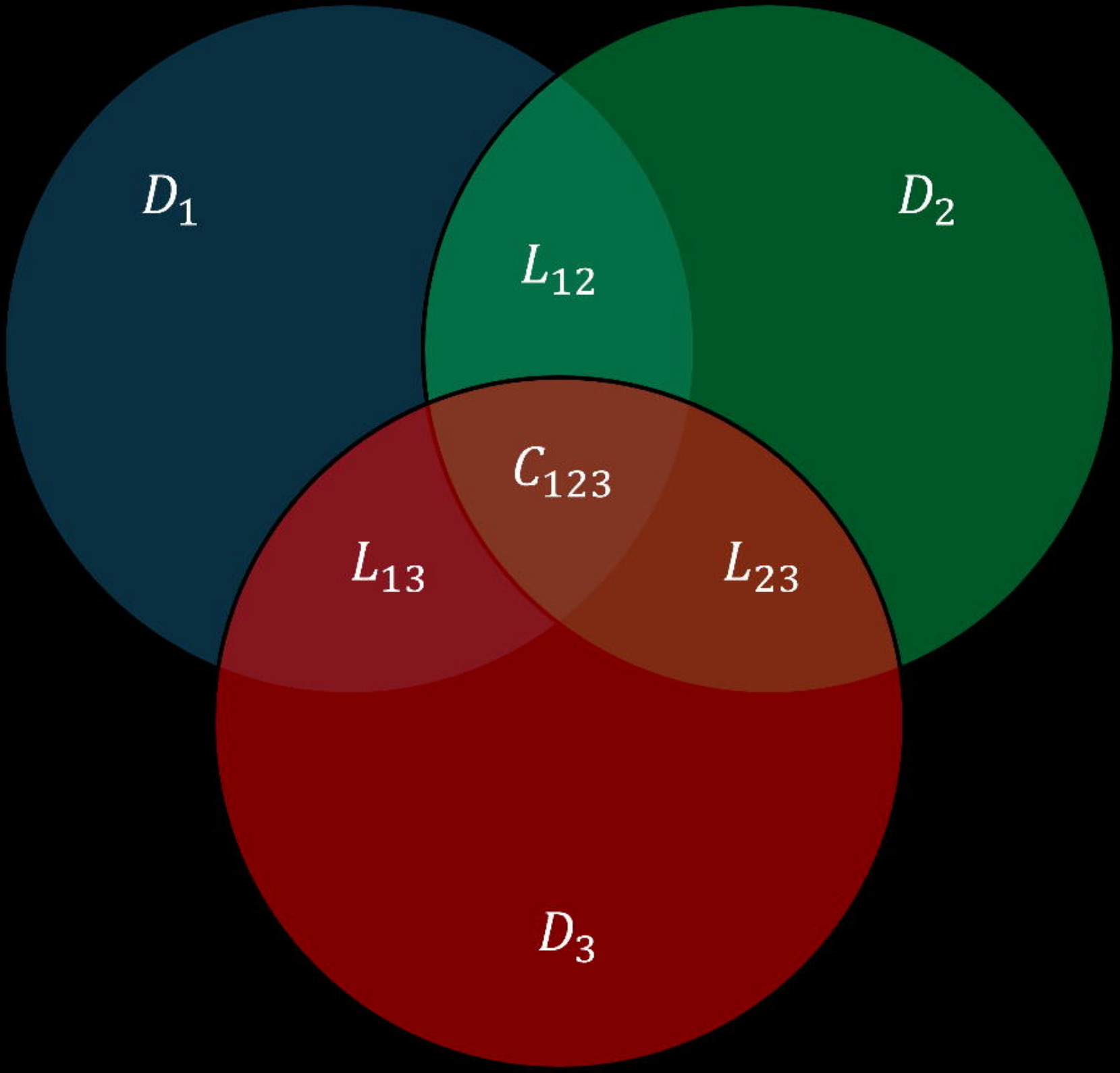
65. Schenker C, Cohen JE, Acar E. An Optimization Framework for Regularized Linearly Coupled Matrix-Tensor Factorization. In: 2020 28th European Signal Processing Conference (EUSIPCO). Amsterdam, Netherlands: IEEE; 2021. p. 985–9. https://doi.org/10.23919/eusipco47968.2020.9287459.

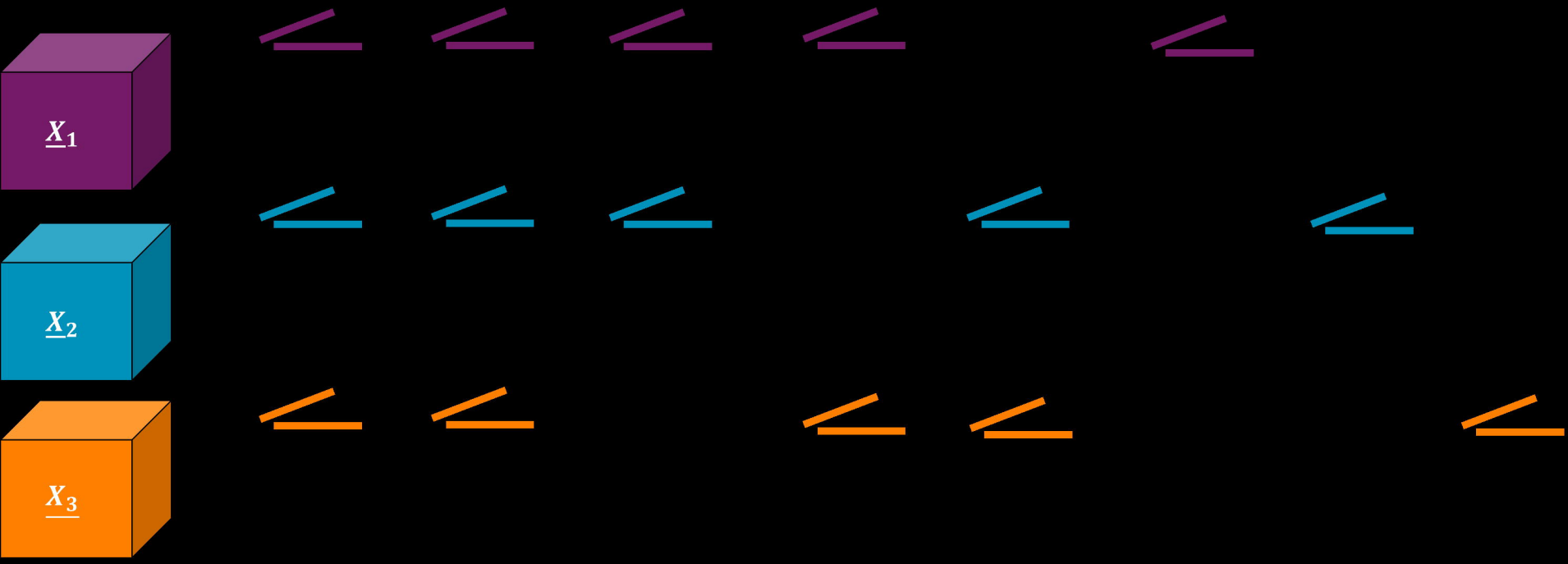
66. Roald M. MatCoupLy: Learning coupled matrix factorizations with Python. SoftwareX. 2023;21:101292. https://doi.org/10.1016/j.softx.2022.101292.

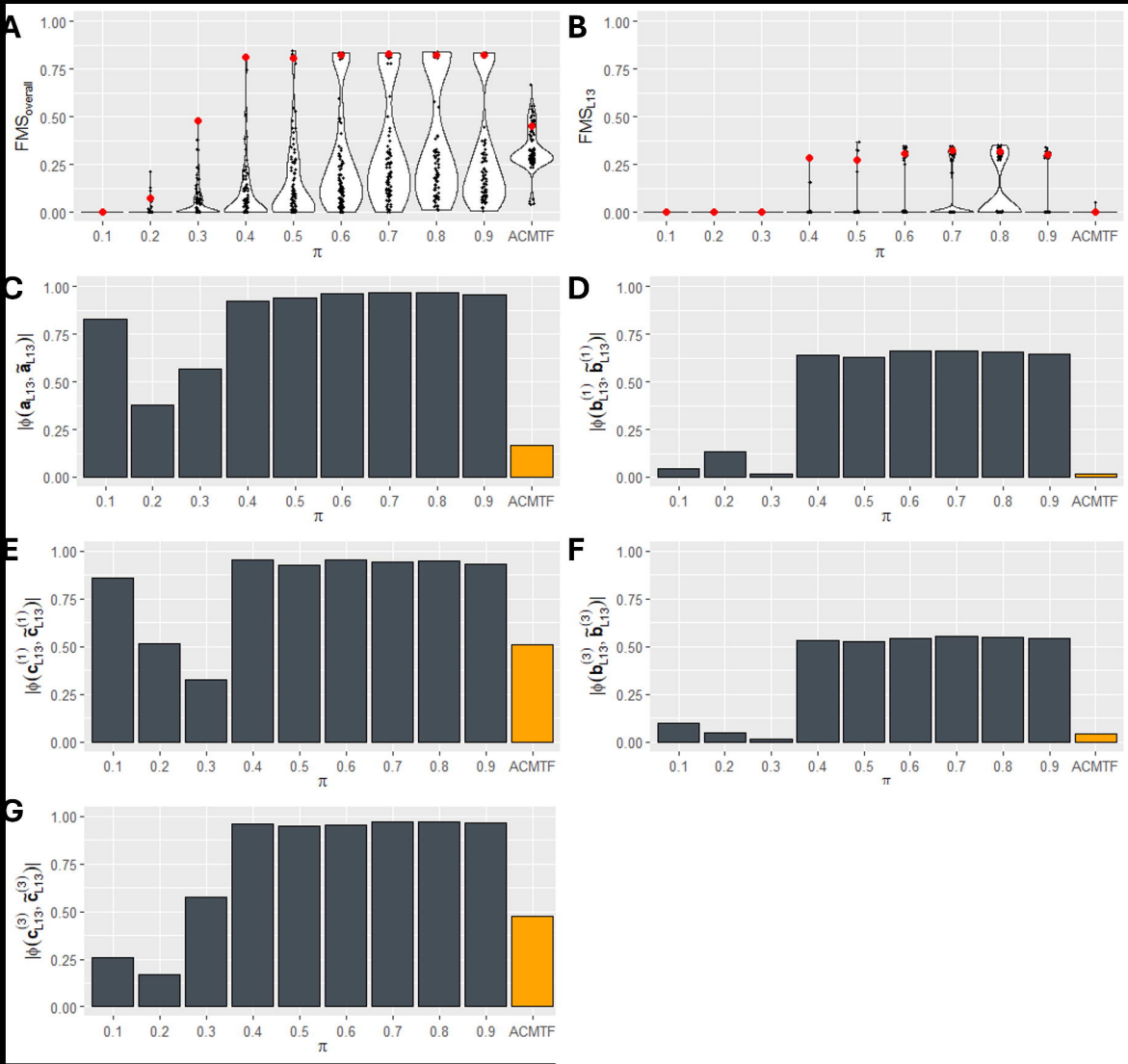
67. Mohimani H, Babaie-Zadeh M, Jutten C. A fast approach for overcomplete sparse decomposition based on smoothed $\backslashell^{0} $ norm. IEEE Trans Signal Process. 2008;57:289–301.

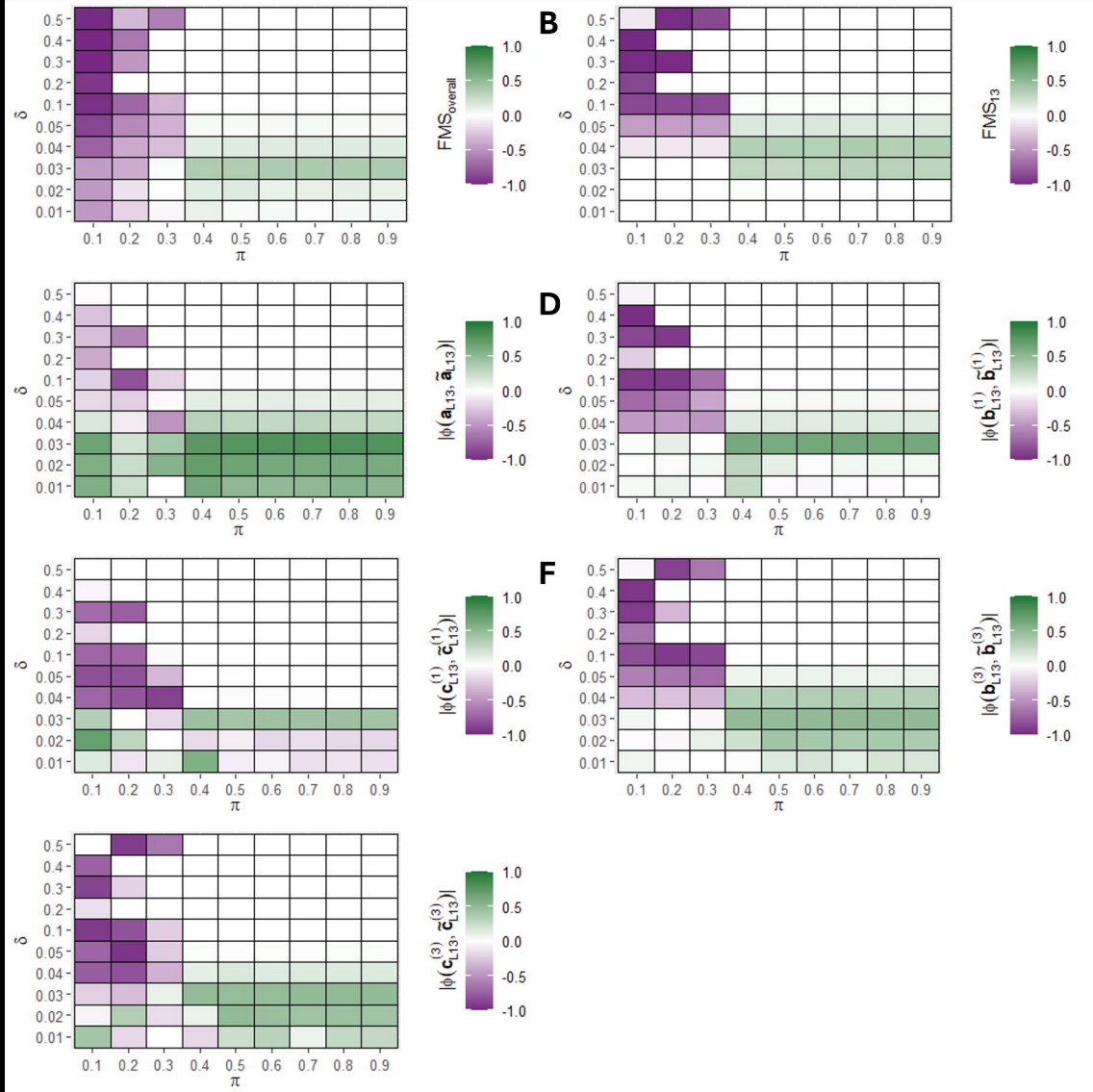
68. Rivet B, Duda M, Guérin-Dugué A, Jutten C, Comon P. Multimodal approach to estimate the ocular movements during EEG recordings: A coupled tensor factorization method. In: 2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). 2015. p. 6983–6. https://doi.org/10.1109/EMBC.2015.7319999.

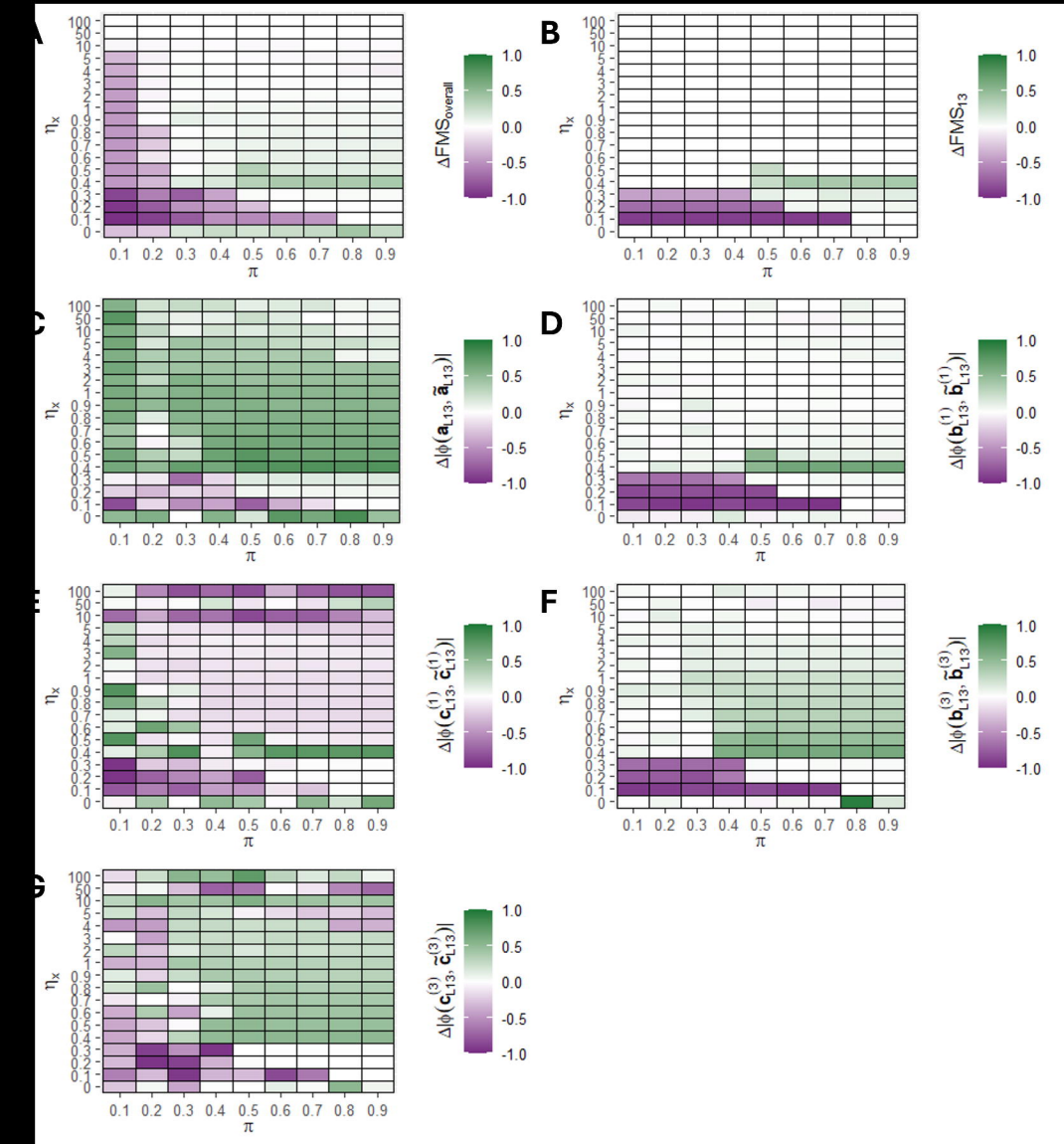
69. Liu W, Chan J, Bailey J, Leckie C, Ramamohanarao K. Mining Labelled Tensors by Discovering both their Common and Discriminative Subspaces. In: Proceedings of the 2013 SIAM International Conference on Data Mining (SDM). Society for Industrial and Applied Mathematics; 2013. p. 614–22. https://doi.org/10.1137/1.9781611972832.68.

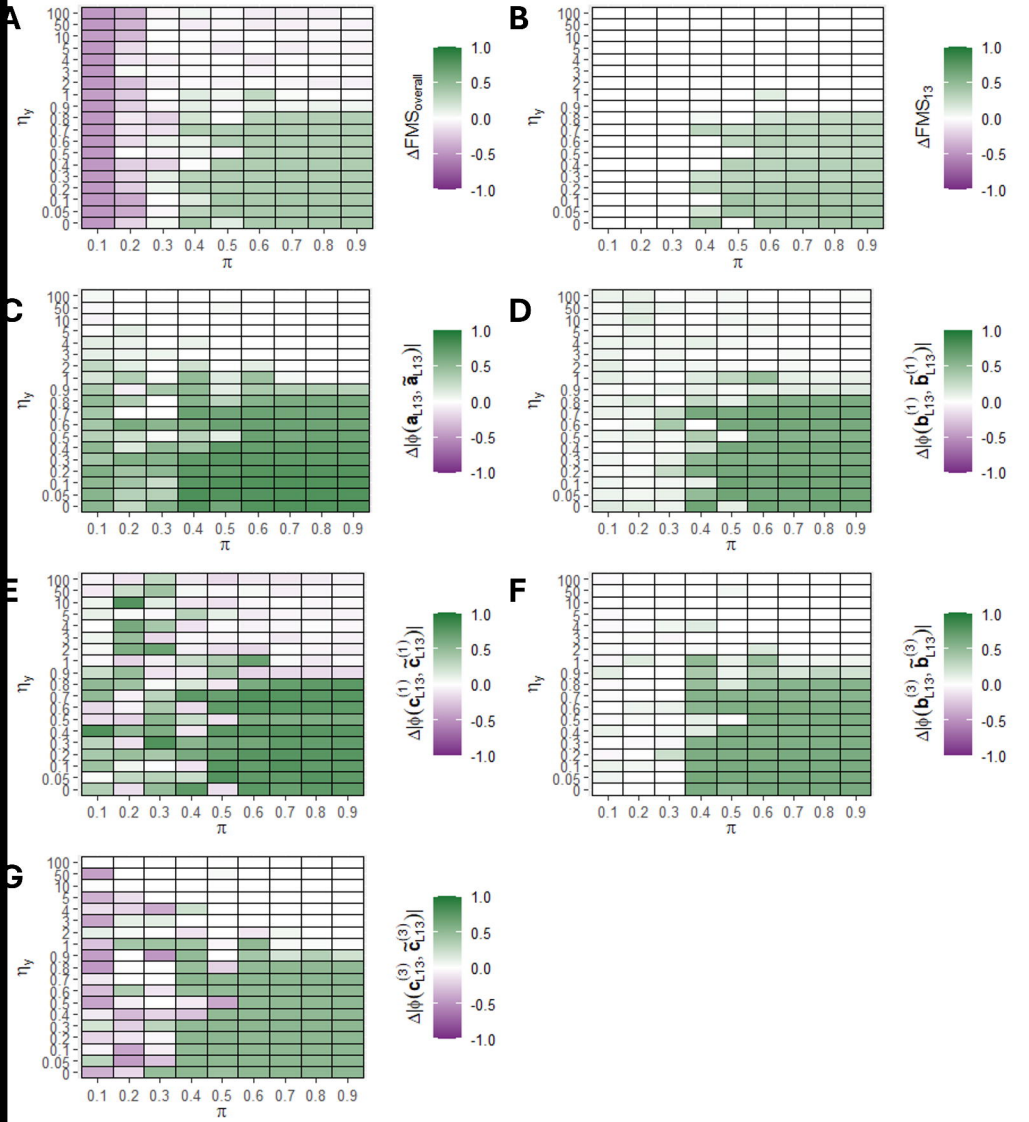

D_1
D_2
L_{12}
C_{123}
L_{13}
L_{23}
D_3

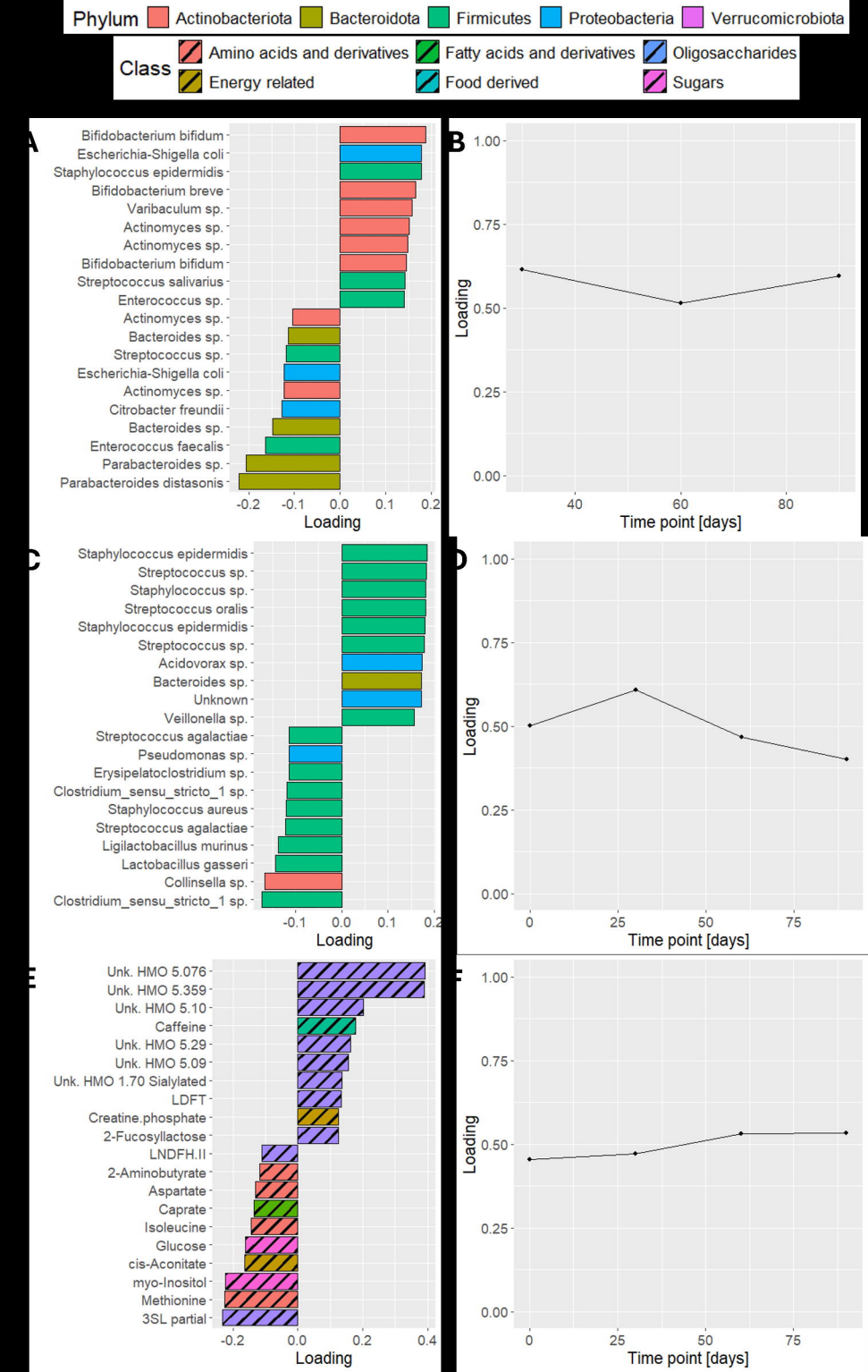

$\underline{X}_1$
$\underline{X}_2$
$\underline{X}_3$


A
B
C
D
E
F
G
FMS_overall
FMS_L13
|ϕ(a_L13, ã_L13)|
|ϕ(b_L13^(1), b̃_L13^(1))|
|ϕ(c_L13^(1), c̃_L13^(1))|
|ϕ(b_L13^(3), b̃_L13^(3))|
|ϕ(c_L13^(3), c̃_L13^(3))|
1.00
0.75
0.50
0.25
0.00
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
ACMTF
π


B
D
F
δ
π
0.01
0.02
0.03
0.04
0.05
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
0.5
0.0
-0.5
-1.0
$FMS_{overall}$
FMS_{13}
$|\phi(\mathbf{a}_{L13}, \tilde{\mathbf{a}}_{L13})|$
$|\phi(\mathbf{b}^{(1)}_{L13}, \tilde{\mathbf{b}}^{(1)}_{L13})|$
$|\phi(\mathbf{c}^{(1)}_{L13}, \tilde{\mathbf{c}}^{(1)}_{L13})|$
$|\phi(\mathbf{b}^{(3)}_{L13}, \tilde{\mathbf{b}}^{(3)}_{L13})|$
$|\phi(\mathbf{c}^{(3)}_{L13}, \tilde{\mathbf{c}}^{(3)}_{L13})|$


A
B
C
D
E
F
G
η_x
π
$\Delta FMS_{overall}$
ΔFMS_{13}
$\Delta|\phi(\mathbf{a}_{L13}, \tilde{\mathbf{a}}_{L13})|$
$\Delta|\phi(\mathbf{b}^{(1)}_{L13}, \tilde{\mathbf{b}}^{(1)}_{L13})|$
$\Delta|\phi(\mathbf{c}^{(1)}_{L13}, \tilde{\mathbf{c}}^{(1)}_{L13})|$
$\Delta|\phi(\mathbf{b}^{(3)}_{L13}, \tilde{\mathbf{b}}^{(3)}_{L13})|$
$\Delta|\phi(\mathbf{c}^{(3)}_{L13}, \tilde{\mathbf{c}}^{(3)}_{L13})|$


A
B
C
D
E
F
G
η_y
π
$\Delta FMS_{overall}$
ΔFMS_{13}
$\Delta|\phi(\mathbf{a}_{L13}, \tilde{\mathbf{a}}_{L13})|$
$\Delta|\phi(\mathbf{b}^{(1)}_{L13}, \tilde{\mathbf{b}}^{(1)}_{L13})|$
$\Delta|\phi(\mathbf{c}^{(1)}_{L13}, \tilde{\mathbf{c}}^{(1)}_{L13})|$
$\Delta|\phi(\mathbf{b}^{(3)}_{L13}, \tilde{\mathbf{b}}^{(3)}_{L13})|$
$\Delta|\phi(\mathbf{c}^{(3)}_{L13}, \tilde{\mathbf{c}}^{(3)}_{L13})|$


Phylum
Actinobacteriota
Bacteroidota
Firmicutes
Proteobacteria
Verrucomicrobiota
Class
Amino acids and derivatives
Fatty acids and derivatives
Oligosaccharides
Energy related
Food derived
Sugars
A
Bifidobacterium bifidum
Escherichia-Shigella coli
Staphylococcus epidermidis
Bifidobacterium breve
Varibaculum sp.
Actinomyces sp.
Actinomyces sp.
Bifidobacterium bifidum
Streptococcus salivarius
Enterococcus sp.
Actinomyces sp.
Bacteroides sp.
Streptococcus sp.
Escherichia-Shigella coli
Actinomyces sp.
Citrobacter freundii
Bacteroides sp.
Enterococcus faecalis
Parabacteroides sp.
Parabacteroides distasonis
-0.2 -0.1 0.0 0.1 0.2
Loading
B
1.00
0.75
0.50
0.25
0.00
Loading
40 60 80
Time point [days]
C
Staphylococcus epidermidis
Streptococcus sp.
Staphylococcus sp.
Streptococcus oralis
Staphylococcus epidermidis
Streptococcus sp.
Acidovorax sp.
Bacteroides sp.
Unknown
Veillonella sp.
Streptococcus agalactiae
Pseudomonas sp.
Erysipelatoclostridium sp.
Clostridium_sensu_stricto_1 sp.
Staphylococcus aureus
Streptococcus agalactiae
Ligilactobacillus murinus
Lactobacillus gasseri
Collinsella sp.
Clostridium_sensu_stricto_1 sp.
-0.1 0.0 0.1 0.2
Loading
D
1.00
0.75
0.50
0.25
0.00
Loading
0 25 50 75
Time point [days]
E
Unk. HMO 5.076
Unk. HMO 5.359
Unk. HMO 5.10
Caffeine
Unk. HMO 5.29
Unk. HMO 5.09
Unk. HMO 1.70 Sialylated
LDFT
Creatine.phosphate
2-Fucosyllactose
LNDFH.II
2-Aminobutyrate
Aspartate
Caprate
Isoleucine
Glucose
cis-Aconitate
myo-Inositol
Methionine
3SL partial
-0.2 0.0 0.2 0.4
Loading
F
1.00
0.75
0.50
0.25
0.00
Loading
0 25 50 75
Time point [days]