LIMPCA: AN R PACKAGE FOR THE LINEAR MODELING OF HIGH-DIMENSIONAL DESIGNED DATA BASED ON ASCA/APCA FAMILY OF METHODS

Michel Thiel, Nadia Benaiche, Manon Martin, Sébastien Franceschini, Robin Van Oirbeek, Bernadette Govaerts

REPRINT | 2023 / 19

UCLouvain

ISBA | LIDAM
Louvain Institute of Data Analysis and Modeling in economics and statistics

ISBA

Voie du Roman Pays 20 - L1.04.01

B-1348 Louvain-la-Neuve

Email : lidam-library@uclouvain.be

https://uclouvain.be/en/research-institutes/lidam/isba/publication.html

Title page

Title: limpca: an R package for the linear modeling of high-dimensional designed data based on ASCA/APCA family of methods

Authors list:
Michel Thiel[1], Nadia Benaiche[1], Manon Martin[2], Sébastien Franceschini[3], Robin Van Oirbeek[1], Bernadette Govaerts[1]

Affiliations:
[1] Institute of Statistics, Biostatistics and Actuarial Sciences (ISBA), Louvain Institute of Data Analysis and Modeling in economics and statistics (LIDAM), UCLouvain, Voie du Roman Pays 20, bte L1.04.01, B-1348 Louvain-la-Neuve, Belgium
[2] Louvain Institute of Biomolecular Science and Technology (LIBST), UCLouvain, Croix du sud 4-5/L7.07.02, B-1348 Louvain-la-Neuve, Belgium
[3] Gembloux Agro-Bio Tech, Département GxABT, Modelisation and developpement TERRA Research Centre, ULiège, Passage des déportés 2, Bât. ABT01 G18, B-5030 Gembloux, Belgium

Corresponding author:
Michel Thiel[1], Phone: +3214602575, email: michel.thiel@uclouvain.be

This article presents the R package `limpca` which implements ASCA+ and APCA+ methods. Those two methods extend the use of ASCA and APCA to unbalanced designs with several principles from the theory of General Linear Models. The package structure is presented and applied on metabolomics data to highlight biomarkers corresponding to effects of interest in an experimental design.

Summary

Many modern analytical methods are used to analyze samples issued from an experimental design; for example in medical, biological, chemical or agronomic fields. Those methods generate most of the time, highly multivariate data like spectra or images, where the number of variables (descriptor responses) tends to be much larger than the number of experimental units. Therefore, multivariate statistical tools are necessary to identify variables that are consistently affected by experimental factors.

In this context, two recent methods combining ANOVA and PCA, namely ASCA (ANOVA-Simultaneous Component Analysis) and APCA (ANOVA-Principal Component Analysis), were developed. They provide powerful visualization tools for multivariate structures in the space of each effect of the statistical model linked to the experimental design. Their main limitation is that they produce biased estimators of the factor effects when the design of experiment is unbalanced. This article presents the R package `limpca` (for linear models with principal component effects analysis) which implements ASCA+ and APCA+, an enhanced version of ASCA and APCA methods based on several principles from the theory of General Linear Models (GLM). In this paper the methodology is reviewed, the package structure and functions are presented, and a metabolomics data set is used to clearly demonstrate the potential of ASCA+ and APCA+ methods to highlight true biomarkers corresponding to effects of interest in unbalanced designs.

Keywords: General Linear Model, Design of Experiment, ASCA, APCA, ANOVA

1. Introduction and objectives

Many experiments in life sciences such as in the omics field produce high dimensional data. Omics sciences aim to understand biological systems by focussing mainly on genes (genomics), messenger RNAs (transcriptomics), proteins (proteomics) and metabolites (metabolomics). The acquisition of the related data can be conducted by various technologies: Nuclear Magnetic Resonance spectroscopy (NMR), Liquid Chromatography combined with Mass Spectrometry (LC-MS), RNAseq (RNA sequencing), microarrays or qPCR (real-time Polymerase Chain Reaction) [1]. Since experiments in these fields are often relying on multifactorial experimental designs, adapted methods are needed to interpret these data.

A common method to analyze the effect of several factors on a single variable is ANOVA (ANalysis Of VAriance) which decomposes the variance of the data into the variance linked to each model effect [2]. However, in most of the omics experiments, multiple variables are measured to describe the observed phenomenon and a more sophisticated statistical approach is required. The multivariate extension of ANOVA is called MANOVA (multivariate ANOVA), but this method is not suitable when the number of variables exceeds the number of experiments such as often encountered in omics [3]. Another method wildely used to get a simplified view of multivariate data sets is Principal Component Analysis (PCA), but it has the drawback to be an unsupervised method which can not take into account the structure of an experimental design [4].

Two methods developed in the early 2000's, ASCA [5] and APCA [6], combine ANOVA decomposition and PCA to visualize and interpret modeling results in a reduced space. However these methods are only correct in the case of a balanced design, which represents a major limitation in real case studies.

In this regard, Thiel et al. [7] exposed two new approaches, ASCA+ and APCA+, in which the General Linear Model (GLM) is used instead of ANOVA. These methods give the same results for the balanced cases and correct the bias for the unbalanced cases by using Ordinary Least Squares (OLS) estimators rather than simple differences of means to estimate the parameters. The GLM approach is applicable to all ANOVA models with fixed categorical effects and can also be extended to models including quantitative factors or random effects.

This article presents the implementation of the R package `limpca`, standing for *linear models with principal component effects analysis*, which allows the statistical analysis of high-dimensional designed data using the ASCA+ and APCA+ methods [8]. The package offers a comprehensive workflow first to explore the data set of interest, then to fit a chosen model to the data and finally to analyze the model results through a wide variety of statistics, tables and graphics. The package is now available on GitHub with two detailed vignettes and a user's manual, and is aimed to be published on Bioconductor. `limpca` presents different advantages compared to other software in this field: (1) it is able to treat any balanced or unbalanced experimental design for fixed categorical factors, (2) it offers optimized methods to calculate effect importance and test their significance, (3) it allows the user to represent data and ASCA/APCA results with various and rich `ggplot2`-based graphical outputs that are highly customizable and (4) the package is open to future extensions to more sophisticated statistical models.

Several implementations of ASCA and ASCA-related methods exist in scientific software. Most of these are described in the two recent papers by Bertinetto et al. [9] and Jarmund et al. [10]. They are either available as R packages or Matlab code and other implemented in dedicated software like `MetaboAnalyst` [11]. In R, one can cite `lmdme` [12], `MetStaT` [13], `ALASCA` [10], `gASCA` [14] and `multiblock` [15]. Two of these are not updated like `lmdme` (since 2014), or have been archived like `MetStaT`. Two others are focused on specific models like `ALASCA` which treats longitudinal and cross-sectional multivariate data with RM-ASCA+ [16, 9], and gASCA devoted to the analysis of count data [14]. `Multiblock` package, which offers a large collection of methods for multiblock data analysis, includes an implementation of ASCA for a large panel of models but its implementation remains, at this stage, quite basic and documentation limited [15]. Compared to these software, `limpca` supports all fixed effects linear models and offers the widest set of modeling outputs like: general measures of effect importance, bootstrap significance tests, combined effects and many `ggplot2` graphical outputs. It is also implemented with an aim of generalization to other models.

This paper is organized as follows. The following sections will first describe the methodology behind ASCA+ and APCA+ methods using the GLM approach. Thereafter, the `limpca` package structure and its

main functions will be presented, including for data exploration, modeling or visualization. Finally, the use of the different functions will be illustrated on a metabolomic data set obtained by ^{1}H-NMR spectroscopy.

2. Methodology

2.1. Overview

This section presents the main concepts and methods implemented in `limpca` and, more precisely, ASCA+ and APCA+, the General Linear Model (GLM) extension of ASCA-like methodologies as presented in the article of [7]. The main motivation is that GLM provides a general solution to analyze designed data with fixed effects and supports as well unbalanced designs which is not the case of the classical ANOVA decomposition [7].

As illustrated in Figure 1, in a first step, the original response data matrix $\mathbf{Y}$ termed "*outcomes*" in `limpca`, is decomposed into a series of effect matrices $\hat{\mathbf{M}}$ according to a general linear model (GLM) based on the experimental design. In a second step, the importance of the model effects can be evaluated by estimating the percentage of variance explained by each effect, as well as by testing their significance. In a third step, PCA is applied on the decomposed effect matrices to visualize the model results through informative loading and score plots.

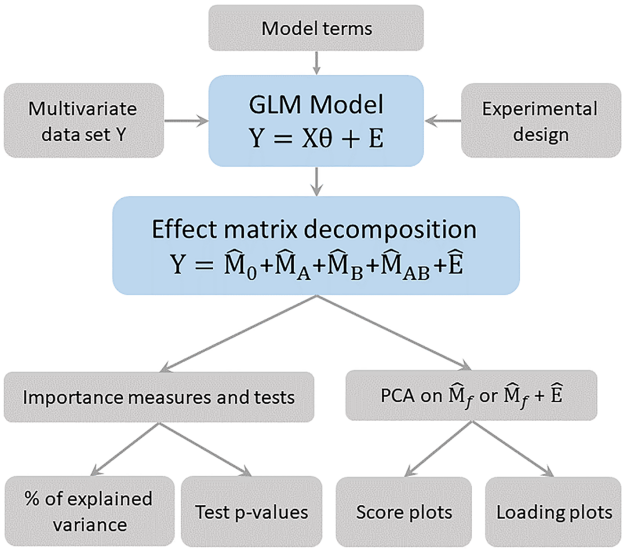

Figure 1: Overview of ASCA+ and APCA+ approaches based on [17].

2.2. GLM decomposition

Any fixed effects ANOVA model can be rewritten in the form of a GLM in matrix notation as $\mathbf{Y} = \mathbf{X}\Theta + \mathbf{E}$. $\mathbf{Y}$ is the response matrix (outcomes) of dimension $n \times m$, $\mathbf{X}$ is the model matrix of dimension $n \times p$, $\mathbf{\Theta}$ is the parameter matrix of dimension $p \times m$ and $\mathbf{E}$ is the error matrix of dimension $n \times m$, where m is the number of response variables, n is the number of observations or trials in the experiment and p is the number of parameters of the linear model for one response.

The OLS method provides a solution to obtain unbiased estimators of the model parameters: $\widehat{\mathbf{\Theta}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}$.

The effect matrices for the different terms of the model are then obtained straightforwardly by building for each effect f a new model matrix $\mathbf{X}_f$ obtained by retaining from $\mathbf{X}$ only the p_f columns linked to the

effect of interest and multiplying it by the corresponding sub-block of the model parameters matrix $\widehat{\mathbf{\Theta}}_f$: $\widehat{\mathbf{M}}_f = \mathbf{X}_f\widehat{\mathbf{\Theta}}_f$. In other terms, if $\mathbf{L}$ is defined as the $p_f \times p$ contrast matrix used to select the lines of $\widehat{\mathbf{\Theta}}_f$ linked to effect f, then $\widehat{\mathbf{M}}_f = \mathbf{X}\mathbf{L}'\mathbf{L}\widehat{\mathbf{\Theta}}$. The ASCA$^+$ decomposition of the response matrix into effect matrices is then $\mathbf{Y} = \hat{\mathbf{M}}_0 + \sum_{f=1}^{F}\hat{\mathbf{M}}_f + \hat{\mathbf{E}}$.

This procedure has the big advantage to provide a general solution to build the ASCA effect matrices for any fixed effects ANOVA model including crossed and/or nested effects. The only need is a routine to build the model matrix $\mathbf{X}$ with the appropriate sum or deviation coding with respect to the model of interest [18]. It also offers the possibility to extend the analysis to covariance or mixed effect models. Currently the package takes into account all linear models with a quantitative normal response and fixed categorical factors (principal effects, interactions and hierarchical effects) but developments for quantitative factors, random factors, longitudinal data or non normal responses are ongoing.

As an illustrative example, the two-way crossed ANOVA model is often used to describe the method:

$$\mathbf{Y} = \hat{\mathbf{M}}_0 + \hat{\mathbf{M}}_A + \hat{\mathbf{M}}_B + \hat{\mathbf{M}}_{AB} + \hat{\mathbf{E}} \tag{1}$$

where $\mathbf{Y}$ is a matrix of dimension $(n \times m)$ which is decomposed into five matrices having the same dimensions: $\hat{\mathbf{M}}_0$ the intercept matrix; $\hat{\mathbf{M}}_A$ and $\hat{\mathbf{M}}_B$ the main effect matrices; $\hat{\mathbf{M}}_{AB}$ the interaction effect matrix and $\hat{\mathbf{E}}$ the error matrix or matrix of residuals. Note that these matrices are closely linked to the choice of the factor coding in the model matrix and sum-coding is recommended in ASCA in order to recover classical ANOVA results. The application of ASCA+ and APCA+ to this model is extensively developed in [7].

2.3. Importance measures

In line with the principle of type III Sum of Squares, the following method is proposed in [7] to calculate the percentage of variation explained by an effect f of the ANOVA model: $\%Var_f = \dfrac{\|\widehat{\mathbf{E}}_{/f}\|^2 - \|\widehat{\mathbf{E}}_{full}\|^2}{\|\mathbf{Y} - \widehat{\mathbf{M}}_0\|^2} \times 100$, where $\widehat{\mathbf{E}}_{/f}$ is the error matrix of the model estimated without taking into account the effect f and $\widehat{\mathbf{E}}_{full}$ is the error matrix of the complete model. The percentage of variance explained by the error term is the following: $\%Var_e = \dfrac{\|\widehat{\mathbf{E}}_{full}\|^2}{\|\mathbf{Y} - \widehat{\mathbf{M}}_0\|^2} \times 100$.

This approach gives an exact interpretation of the difference in terms of Sum of Squares but will not provide percentages of variances summing up to 100% when the design is not balanced. Note that, in practice, each sub-model does not need to be estimated to calculate the $\%Var_f$. Indeed, the numerator of $\%Var_f$ can be calculated as follows: $tr(\hat{\mathbf{\Theta}}'\mathbf{L}'(\mathbf{L}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{L}')^{-1}\mathbf{L}\hat{\mathbf{\Theta}})$.

2.4. Significance tests

In a multiple effect model, it is interesting to determine which ones have a significant effect on the responses. In classical ANOVA or GLM with a single response, this can be done by F or t tests. In ASCA and APCA, it is common to use permutation tests to generalize these tests to multiple responses. In `limpca`, a parametric bootstrap test is implemented because it has the advantage to be independent from the model structure (compared to permutation tests) and can be adapted to further extensions. To test the significance of an effect f, the procedure involves the following steps [8]:

1. Define the model under H_0, obtained by excluding the effect f from the model matrix $\mathbf{X}$: $\mathbf{Y} = \mathbf{X}_{/f}\mathbf{\Theta}_{/f} + \mathbf{E}_{/f}$.
2. Fit the reduced H_0 and full H_1 models to the response matrix $\mathbf{Y}$.

3. Calculate the test statistic:

$$F_{obs} = \frac{(\|\widehat{\mathbf{E}}_{/f}\|^2 - \|\widehat{\mathbf{E}}_{full}\|^2)/(p - p_{/f})}{\|\widehat{\mathbf{E}}_{full}\|^2/(n-p)}$$

where $p_{/f}$ is the number of parameters of the partial model for one response.

4. Calculate the predicted values $\hat{\mathbf{Y}}_{/f} = \mathbf{X}_{/f}\widehat{\mathbf{\Theta}}_{/f}$ and the residuals $\hat{\mathbf{E}}_{/f} = \mathbf{Y} - \hat{\mathbf{Y}}_{/f}$ of the model under H_0 to use them in the bootstrap loop.
5. Repeat for a large number B of iterations ($b = 1, \ldots, B$):
   (a) Sample $\hat{\mathbf{E}}_b$ with replacement among the rows of $\hat{\mathbf{E}}_{/f}$
   (b) Generate a bootstrap response matrix $\mathbf{Y}_b = \hat{\mathbf{Y}}_{/f} + \hat{\mathbf{E}}_b$
   (c) Estimate the full and restricted models with this new response matrix $\mathbf{Y}_b$
   (d) Calculate the corresponding test statistics F_b by using the optimized formula:

$$F_b = \frac{tr(\hat{\mathbf{\Theta}}_b'\mathbf{L}'(\mathbf{L}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{L}')^{-1}\mathbf{L}\hat{\mathbf{\Theta}}_b)/(p - p_{/f})}{\|\widehat{\mathbf{E}}_{full}\|^2/(n-p)} \quad (2)$$

where $\mathbf{L}$ is, as above, the contrast matrix used to select in $\mathbf{X}$ and $\hat{\mathbf{\Theta}}_b$ the columns/lines linked to an effect f.

6. Calculate the bootstrap p-value for the effect of interest:

$$p - value_f^{boot} = \frac{\sum_{b=1}^{B} I(F_b \geq F_{obs}) + 1}{B+1}$$

In `limpca`, the computation of the F_f statistics has been optimized by using the F_b formula (2), which does not require the re-estimation of partial models, and by using parallel computation in R. In `limpca`, it is also possible to apply such tests on other restricted models than the removal of a single effect (referred below as tests on "combined" effects).

2.5. Multivariate dimension reduction

Following the matrix decomposition, PCA is applied to each of the decomposed matrices in order to interpret the effects of the model terms on the responses. The major difference between ASCA and APCA is the matrix used for this step. In ASCA, PCA is applied to the pure effect matrices $\hat{\mathbf{M}}_f$ directly obtained from the ANOVA/GLM decomposition. In APCA, PCA is applied to the residual-augmented effect matrices $\tilde{\mathbf{M}}_f = \hat{\mathbf{M}}_f + \hat{\mathbf{E}}$.

Due to this difference, ASCA score plots only present mean scores for each factor, while APCA score plots also include individual variations between observations not explained by the model. This last approach provides a better view of the effects significance.

Another variant of ASCA, called ASCA-E, allows to represent the variability between the observations in the score plots. Loadings matrices $\mathbf{P}_f$ are first calculated on the effect matrix $\hat{\mathbf{M}}_f$ only, similarly to ASCA, but the score matrices are calculated using the residual augmented effect matrices as follows [19]:

$$\mathbf{T}_f^E = (\hat{\mathbf{M}}_f + \hat{\mathbf{E}})\mathbf{P}_f = \hat{\mathbf{M}}_f\mathbf{P}_f + \hat{\mathbf{E}}\mathbf{P}_f \quad (3)$$

The projection $\mathbf{T}_f^E$ describes the variation among factor levels in terms of the loadings $\mathbf{P}_f$ of the PCA model for $\hat{\mathbf{M}}_f$. Contrarily to the variation in APCA, the variation in $\mathbf{T}_f^E$ is not maximized.

In terms of interpretation, the score plots obtained in APCA/ASCA-E can help to see if the groups of observations for the different levels of the factors from the design do cluster together, whereas ASCA shows only means of the levels since the residuals are left out. Based on the loading plots, ASCA/ASCA-E

highlights the most influential variables for the factors of the design, while APCA loadings will take into account the residuals. Let's also emphasize that it is always informative to plot the scores of the residuals matrix $\hat{\mathbf{E}}$ in order to spot potential outliers or discover some hidden factors which were not taken into account in the design. Finally, note that when a multifactorial model includes interaction terms, it may be interesting to apply the matrix decomposition to a combination of effect matrices. This is possible in `limpca` and will be illustrated in the case study.

3. `limpca` package organisation and functions

The package is organized in two sets of functions. The first set is dedicated to the exploration and visualization of the data set (outcomes and design) before modeling. The second set allows the modeling through the ASCA+ approach, as well as the visualization of the results. All these functions enable to build a comprehensive workflow for ASCA+/APCA+ data analysis. Figure 2 presents these two sets of functions with their respective input and output objects applied successively.

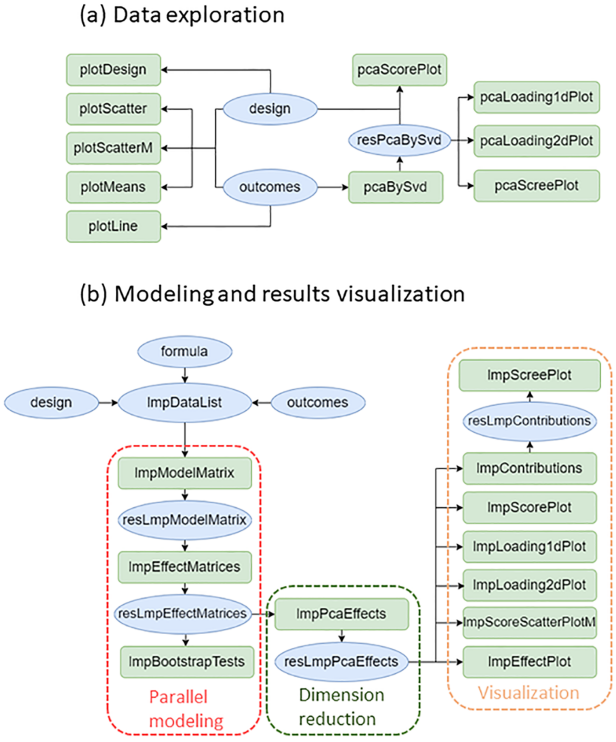

Figure 2: Overview of the main `limpca` objects and functions [8]: (a) Data exploration, (b) Modeling and results visualization.

The package can currently be downloaded from GitHub with the following command: `remotes::install_github("ManonMartin/limpca", dependencies = TRUE)` and is aimed to be published in Bioconductor.

3.1. Data exploration

This first part of the package aims to explore the data included in the outcomes and design R objects (Figure 2(a)). The `outcomes` object should be a $n \times m$ numerical matrix of n observations and m response variables and the `design` object must be a $n \times k$ R data frame of n observations and k factors. A first series of graphical functions helps to represent the design and the responses and another set of functions allows to perform a PCA on the response matrix and observe its results according to the levels of the design factors.

plotDesign
Provides a graphical representation of the experimental design. It allows to visualize factor levels and check the design balancing.

plotScatter
Produces a plot describing the relationship between two columns of the outcomes matrix $\mathbf{Y}$. It allows to choose colors and symbols for the levels of the design factors. Ellipses, polygons or segments can be added to group different sets of points on the graph.

plotScatterM
Produces a scatter plot matrix between the selected columns of the outcomes matrix $\mathbf{Y}$ choosing specific colors and symbols for up to 4 factors from the design on the upper and lower diagonals.

plotMeans
Draws, for a given response variable, a plot of the response means by levels of up to three categorical factors from the design. When the design is balanced, it allows to visualize main effects or interactions for the response of interest. For unbalanced designs, this plot must be used with caution.

plotLine
Generates the response profile of one or more observations i.e. plots of one or more rows of the outcomes matrix on the y-axis against the m response variables on the x-axis. Depending on the response type (spectra, gene expression...), point, line or segment plots can be used.

pcaBySvd
Operates a Principal Component Analysis on the $\mathbf{Y}$ outcome/response matrix by a singular value decomposition. Outputs are represented with the functions `pcaScorePlot`, `pcaLoadingPlot` and `pcaScreePlot`.

pcaScreePlot
Returns a bar plot of the percentage of variance explained by each Principal Component (PC) calculated by `pcaBySvd`.

pcaScorePlot
Produces score plots from `pcaBySvd` output with the same graphical options as `plotScatter`. This is a wrapper of `plotScatter`.

pcaLoading1dPlot
Plots loading vectors from `pcaBySvd` output with different available line types. This is a wrapper of `plotLine`.

pcaLoading2dPlot
Produces 2D loading plots from `pcaBySvd` output with the same graphical options as `plotScatter`. This is a wrapper of `plotScatter`.

3.2. Modeling and results visualization

The second part of the package contains the functions that apply the modeling step as well as the visualization of the results (Figure 2(b)). The functions, framed by the red dashed line, perform the modeling and test the significance of the model effects, while those framed by the orange dotted line allow the visualization of the PCA dimension reduction results.

The analysis starts from the `lmpDataList` object which is a list of three elements. The first two elements are the $n \times m$ outcomes matrix and the $n \times k$ design data frame, both discussed in Section 3.1. The third

element is the R formula of the model to be estimated. For example, for a two-way crossed ANOVA 2 model with factors `A` and `B`, it will take the form: `~A+B+A:B`.

From the `lmpDataList`, a series of functions can be applied consecutively, each one corresponding to a step in the ASCA+ methodology. The GLM is first estimated, effects importance evaluated. Finally, the result of this modeling, `resLmpPcaEffects`, serves as an input argument to all visualization and interpretation functions.

lmpModelMatrix
Creates the model matrix $\mathbf{X}$ from the design matrix and the model formula.

lmpEffectMatrices
Estimates the model by OLS based on the outcomes and model matrices provided in the outputs of `lmpModelMatrix` function and calculates the estimated effect matrices $\hat{\mathbf{M}}_0, \hat{\mathbf{M}}_1, ...\hat{\mathbf{M}}_f$ and residual matrix $\hat{\mathbf{E}}$. It calculates also the type III percentage of variance explained by each effect.

lmpBootstrapTests
Tests the significance of one or a combination of the model effects using bootstrap. This function is based on the outputs of `lmpEffectMatrices` function.

lmpPcaEffects
Performs a PCA on each of the effect matrices from the outputs of `lmpEffectMatrices`. It has an option to choose the method applied: ASCA, APCA or ASCA-E. Combined effects (i.e. linear combinations of original effect matrices) can also be created and decomposed by PCA.

lmpContributions
Reports the contribution of each effect to the total variance, but also the contribution of each PC to the total variance per effect. Moreover, these contributions are summarized in a barplot.

lmpScreePlot
Provides a barplot of the percentage of variance associated to the PCs of the effect matrices ordered by importance based on the outputs of `lmpContributions`.

lmpScorePlot
Draws the score plots of each effect matrix provided in `lmpPcaEffects` outputs. As a wrapper of the `plotScatter` function, it allows to visualize the scores of the effect matrices for two components at a time with all the available options in `plotScatter`.

lmpLoading1dPlot
Plots the loading vectors for each effect matrix from the `lmpPcaEffects` outputs with line plots. This is a wrapper of `plotLine`.

lmpLoading2dPlot
Draws a 2D loading plot of each effect matrix provided in `lmpPcaEffects` outputs. As a wrapper of the `plotScatter` function, it allows the visualization of effect loading matrices for two components at a time with all options available in `plotScatter`.

lmpScoreScatterPlotM
Plots the scores of all model effects simultaneously in a scatterplot matrix. By default, the first PC only is kept for each model effect and, as a wrapper of `plotScatterM`, the choice of symbols and colors to distinguish factor levels allows an enriched visualization of the factors' effect on the responses.

lmpEffectPlot
Plots the ASCA scores by effect levels for a given model effect and for one PC at a time. This graph is especially appealing to interpret interactions or combined effects. This is a wrapper of `plotMeans`.

4. Case study

The `limpca` package includes two data sets with their dedicated vignettes: `UCH` and `Trout`. In this article, due to edition limitations, only a basic analysis of the `UCH` data set is provided. But the reader can find extended results for both datsets in the vignettes of the github associated to the package: `https://manonmartin.github.io/limpca/`.

4.1. Data set

The data set used to illustrate the methods in the next section is a subset of the Urine-Citrate-Hippurate data set described in [20] and already used by [17] to compare ASCA+/APCA+, PARAFASCA, AComDim and AMOPLS methodologies.

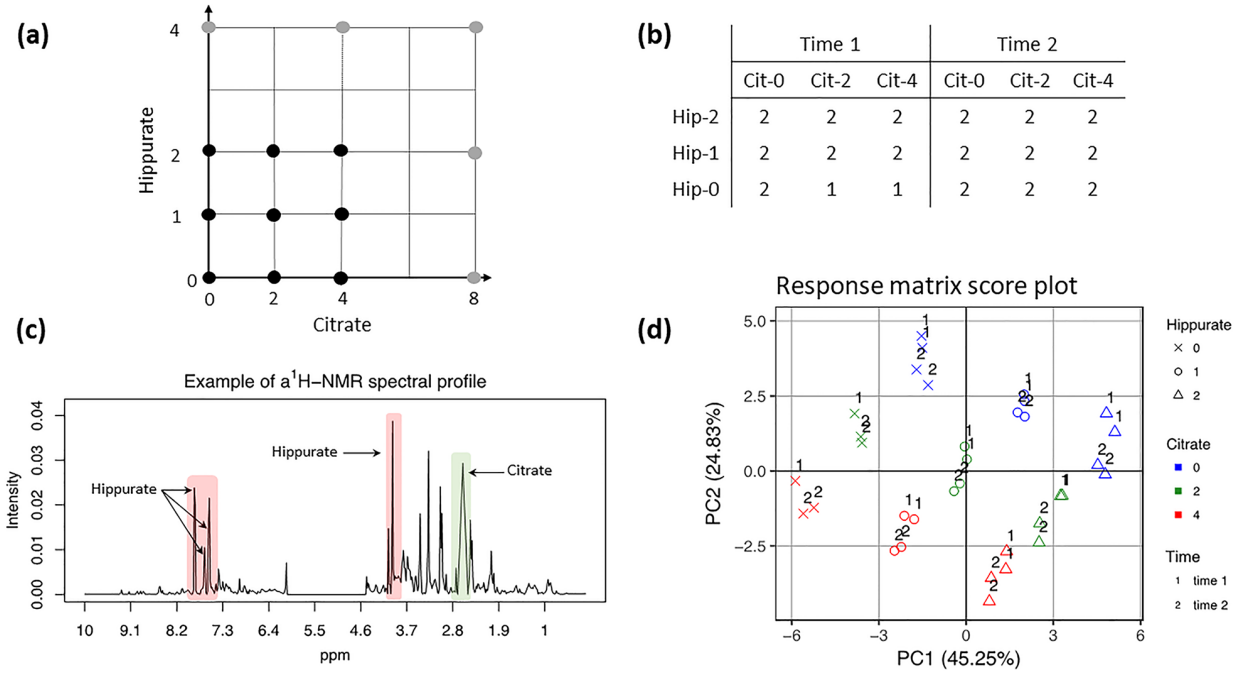

Figure 3: Presentation of the UCH data set (a) Experimental design (b) Sample sizes for each of the 9 combinations of Citrate, Hippurate and Time levels (c) Example of a ^{1}H-NMR spectral profile showing the Hippurate and Citrate peaks (d) Score plot for the first two PCs of the UCH data set response matrix (from [17]).

This data set contains metabolomic spectral profiles obtained by ^{1}H-NMR spectroscopy of samples coming from a pool of rat urine samples. These samples were prepared by spiking the pool with known concentrations of chemicals (Hippurate and Citrate) in order to obtain data based on an experimental design and study the quality of resulting spectra. In the subset of this data set used here and in [17], and further referred to as the `UCH` data set, each of these factors has three levels: no added chemical (a concentration of 0), a medium concentration (2 mM for the Citrate factor (Qc/4) and 1 mM for the Hippurate factor (Qh/4)) and a high concentration (4 mM for the Citrate factor (Qc/2) and 2 mM for the Hippurate factor (Qh/2)).

This leads to an experimental design with 9 possible combinations of concentrations as illustrated in Figure 3(a). Four samples of each of the nine urine combinations were prepared, then frozen and analyzed (by sets of 18) over two days. Each day, the analysis of the two samples with the same Hippurate and Citrate proportions was done under two conditions: just after defreezing and one hour later in order to evaluate the stability of samples over time. The UCH data selected for the paper originally contained 36 spectra: 9 combinations of concentrations $\times$ 2 time points $\times$ 2 replicates. However, two outliers were removed, resulting in 34 observations and a slightly unbalanced experimental design (see Figure 3(b)).

The resulting spectral matrix $\mathbf{Y}$ has a dimension of (34×600) since the resolution of the ^{1}H-NMR spectra equals 600. In these spectra, 4 Hippurate peaks are located around 3.96, 7.54, 7.62 and 7.82 ppm (part per million) and the aggregated Citrate peak is located around 2.6ppm (see Figure 3(c)). Figure 3(d) presents the score plot of the first two PCs for the response matrix. It shows that the information on the Citrate and Hippurate factor levels from the experimental design can be recovered, but only to some extent and mixed in each principal component.

4.2. `limpca` results

4.2.1. Data importation and preparation

Before any analysis, the data need to be loaded in R and organized in the form of a list containing a `design` matrix, an `outcomes` matrix and a model `formula`. The `UCH` object can be loaded with the `data(UCH)` command and contains 3 elements: a matrix of outcomes with 34 observations and 600 response variables representing the spectra coming from the ^{1}H-NMR spectroscopy, the design matrix with 34 observations and 5 explanatory variables and the R formula for the GLM model used. In the design matrix, Hippurate, Citrate and Time are the only variables used as factors in the model. The two days (Time) are considered as single replicates and included in the residuals. The Dilution is a constant factor. The chosen model is the full three-way ANOVA containing the main effects of the three factors and all interactions between them. The R formula can be written as follows:

```
outcomes ~ Hippurate + Citrate + Time + Hippurate:Citrate + Time:Hippurate
         + Time:Citrate + Hippurate:Citrate:Time
```

4.2.2. Data visualization

This section shows different functions offered by `limpca` to visualize the data before the modeling step. The design matrix can be visualized with the function `plotDesign` as illustrated in Figure 4(a) with the Citrate factor along the x axis, Hippurate along the y axis, Day in rows and Time in columns. The profiles of ^{1}H-NMR spectra can be visualized with `plotLine` as shown in Figure 4(b) for two chosen spectra. Values of two responses (ppm descriptors) for all samples can be visualized using the function `plotScatter`. Figure 4(c) shows the spectral intensities at 2.61 vs 3.98 ppm which are the peaks corresponding to the Citrate and Hippurate respectively. This plot shows 9 groups of points corresponding to the different combinations of Citrate and Hippurate concentrations in the experimental design.

Another way to visualize the values of several variables simultaneously and adding the information from up to 4 factors of the design is to use the function `plotScatterM` working in a similar way as function `lmpScoreScatterPlotM` and demonstrated in Figure 7. Finally, the function `plotMeans` allows to visualize, for a given response variable, a plot of the means by factors' level of up to three categorical factors of the design. Figure 4(d) shows the mean response at 3.98 ppm, which is one of the peak of Hippurate, as a function of Hippurate, Time and Citrate levels. It can be observed that, as expected, a linear evolution in spectral intensity occurs when the level of Hippurate increases.

Function `pcaBySvd` is useful to compute a PCA decomposition of the outcomes matrix. Figure 5 (a) shows the percentages of variance explained by the first 6 components obtained with the function `pcaScreePlot`. It can be observed that the first three PCs explain more than 86% of the total variability. Figures 5(b) and (c) show the scores and the loadings of the first two PCs plotted with the functions `pcaScorePlot` and

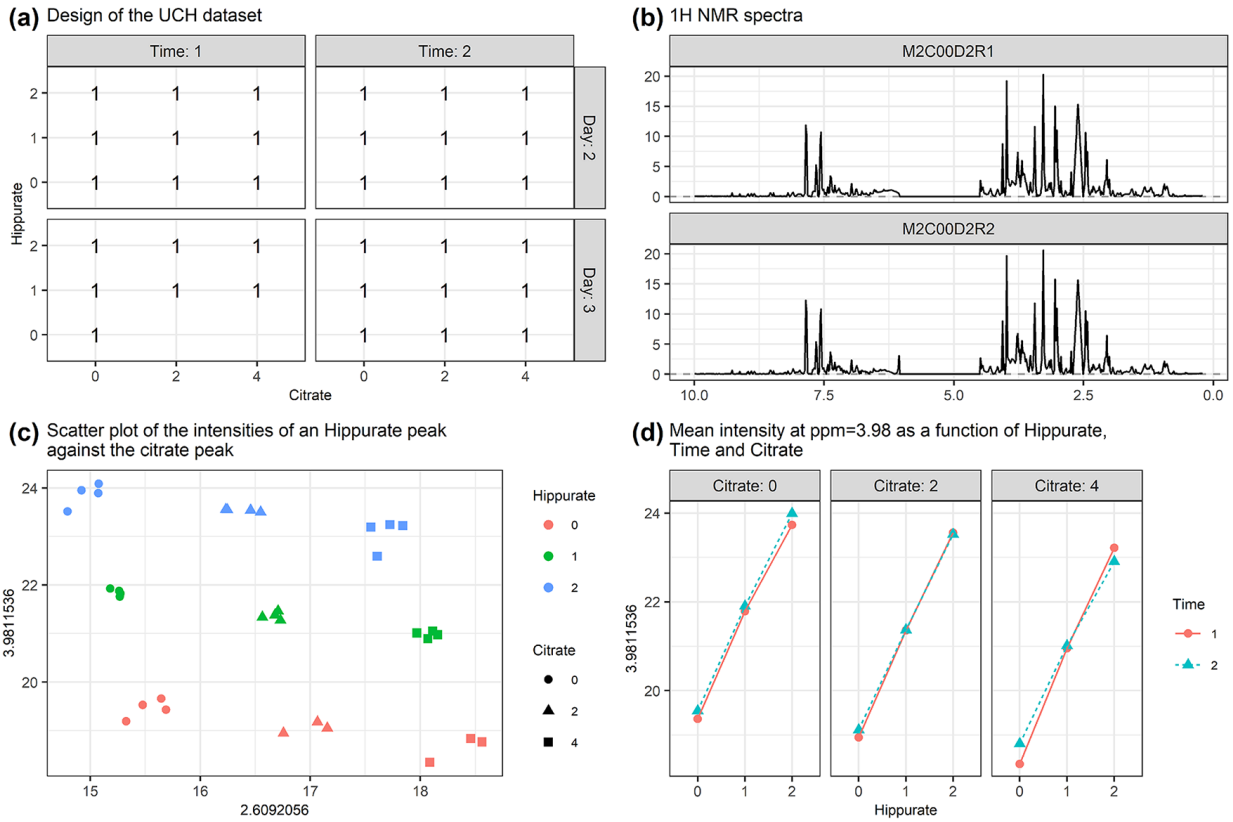

Figure 4: (a) Design of the UCH data set, (b) ^{1}H-NMR spectra, (c) Scatter plot of two responses and (d) typical `plotMeans` output.

`pcaLoading1dPlot`. The score plot allows to see the 9 groups of points associated with the different combinations of Citrate and Hippurate concentrations. The loading plot shows also the variables corresponding to Citrate and Hippurate peaks as explained in Figure 3(c). Note that the PCs do not clearly separate the 2 products since the shape of the design is not parallel to the score plot axes and components are then mixtures of the spectral profiles of Citrate and Hippurate.

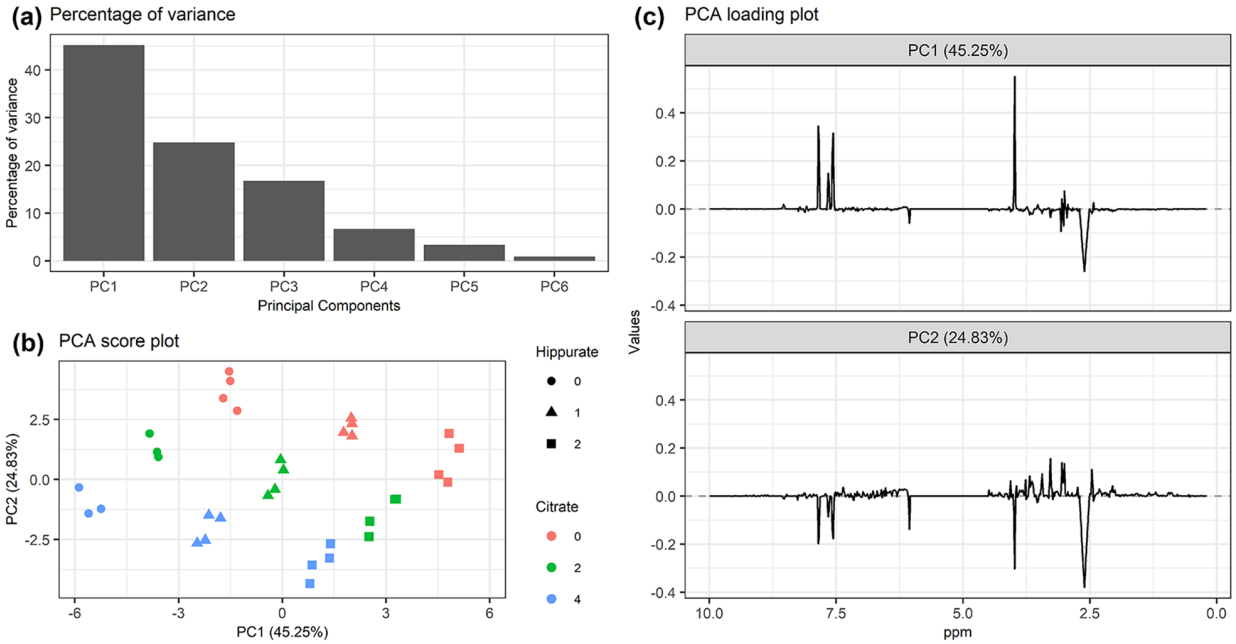

Figure 5: PCA percentages of variance, score and loading plots.

4.2.3. Modeling

The modeling step consists of decomposing the outcome matrix $\mathbf{Y}$ into effect matrices and performing PCA on the effect matrices or their combination with the residual matrix. The function `lmpModelMatrix` creates a model matrix from the design and the model formula using sum coding for the factor coding. As explained above, the model to be estimated is a full three way ANOVA model on factors Hippurate, Citrate and Time. Thereafter, the function `lmpEffectMatrices` estimates the GLM and derives the effect matrix for each model effect. This function computes also the percentage of variance linked to each effect using the type III Sum of Squares.

Bootstrap can then be used to determine which effects are significant using the function `lmpBootstrapTests`. Table I presents the percentages of variance and the corresponding bootstraps p-values, obtained from 1000 resampled outcomes samples. It indicates that the main effects of Hippurate, Citrate and Time as well as the interaction between Hippurate and Time are significant. These effects explain, respectively, 39.31%, 29.91%, 16.24% and 6.23% (i.e. in total 91.69%) of the total variance of the responses. A barplot of the variation percentages can be obtained using the function `lmpEffectMatrices` but is not shown here.

4.2.4. Modeling visualization

This section presents the different functions available to visualize the results of the modeling step. The `lmpPcaEffects` function has an argument to define which method to use among ASCA, ASCA-E or APCA

Table I: Percentages of explained variance and bootstrap test p-values for each model effect.

	% of variance	Observed p-value
Hippurate	39.31	< 0.001
Citrate	29.91	< 0.001
Time	16.24	< 0.001
Hippurate:Citrate	1.54	0.147
Hippurate:Time	6.23	< 0.001
Citrate:Time	0.54	0.418
Hippurate:Citrate:Time	1.68	0.089
Residuals	4.3	

to decompose by PCA each of the effect matrices obtained in the second step of the analysis.

Through the `lmpContributions` function, the results shown in Table II are generated. This table indicates, for each effect, the percentage of variance explained by each PC, reported to the effect contribution to the total variance. This gives a view of the global importance of each PC regardless of the effects. For this case study, the analysis will therefore be focused on the first PCs of the three main effects, the interaction effect between Hippurate and Time and the residuals. The other PCs account for less than 1% of the total variance and are then not of interest.

Table II: PCs contributions by effect.

	PC1	PC2	PC3	PC4	PC5	Contrib (%)
Hippurate	**38.41**	0.9	0	0	0	39.31
Citrate	**29.37**	0.53	0	0	0	29.91
Time	**16.24**	0	0	0	0	16.24
Hippurate:Citrate	0.68	0.59	0.23	0.04	0	1.54
Hippurate:Time	**5.85**	0.38	0	0	0	6.23
Citrate:Time	0.49	0.05	0	0	0	0.54
Hippurate:Citrate:Time	0.8	0.46	0.38	0.05	0	1.68
Residuals	**2.09**	0.73	0.44	0.25	0.19	4.3

Figure 6 shows the scores and the loadings for the significant effect matrices obtained with the functions `lmpScorePlot` and `lmpLoading1dPlot` using the ASCA-E method. Residuals of the model are also displayed.

For the main effects, all score plots show a clear separation between the different levels of the considered effects on the first PC. The distinct separation of the groups suggests a strong effect of those factors, confirmed by their significance.

For the loadings, note that ASCA and ASCA-E loading plots are the same while the ones for APCA contain the residuals of the model. Figure 6 (b) and (d) of Hippurate and Citrate loading plots separate and clearly retrieve the peaks of the Hippurate and Citrate molecules in contrast to the results of the classical PCA where peaks of Hippurate and Citrate were superimposed. Figure 6 (f), representing the loadings of the Time effect, shows a high peak on the left of the (removed) water area of the spectra. This is probably due to a shift of most spectral peaks with Time which is visible all over the spectral profile.

In Figure 6 (g), the interaction score plot is less straightforward to interpret. This is common in ASCA results and will be rediscussed below. In contrast, the corresponding loading plot in Figure 6 (h) is clearly showing a peak shift around 3 ppm. Figure 6 (i), representing the Residual score plot, is also interesting as in the chosen linear model, one has made the hypothesis of a negligible and random Day effect. However, the plot shows a quite clear Day effect indicating that this factor could have been included in the model.

In order to get the full picture, the `lmpScoreScatterPlotM` function can be used to produce a score plot matrix of all the effects simultaneously. Figure 7 shows the scores of the first PC for the factors Hippurate,

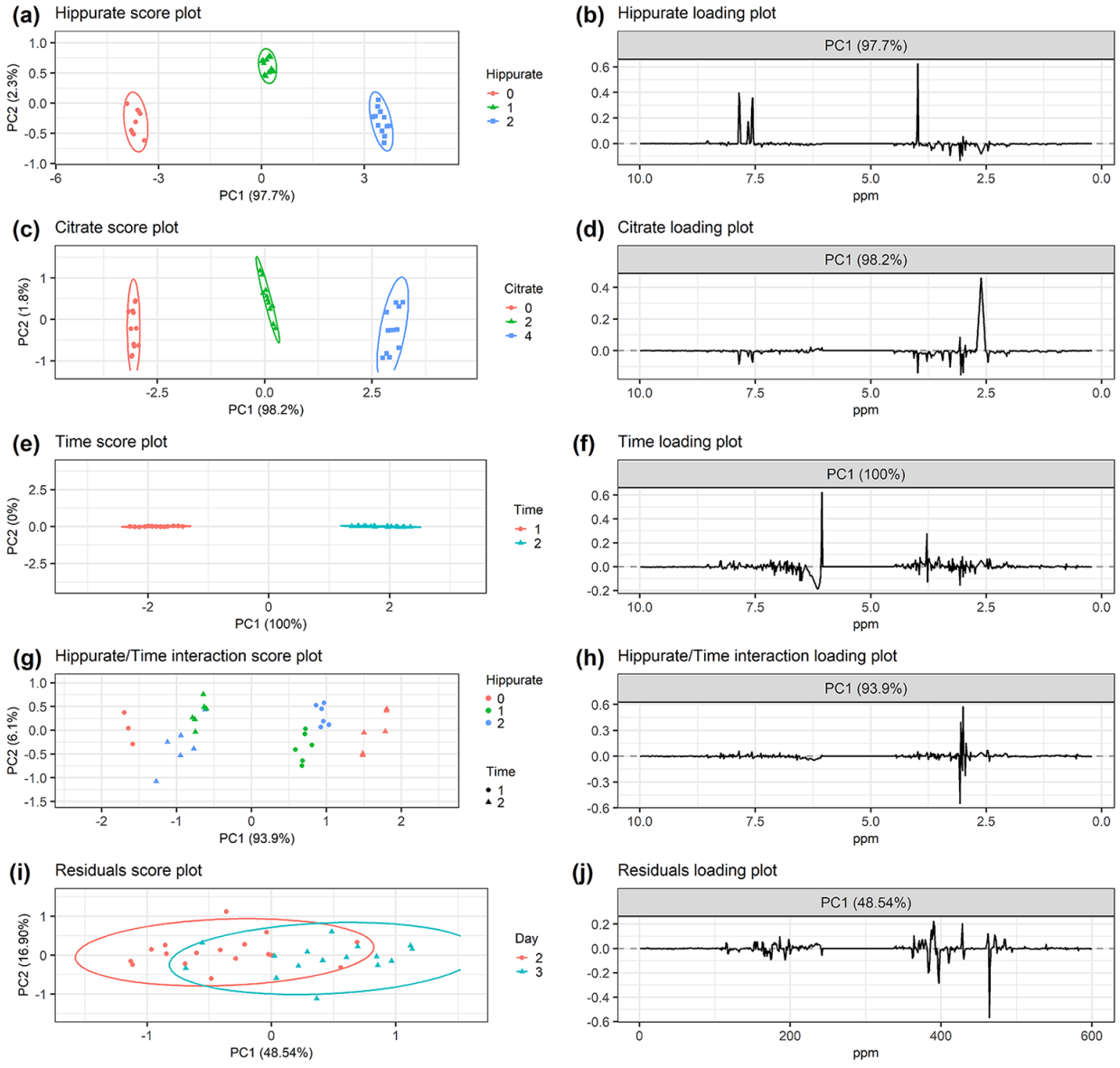

Figure 6: Score and loading plots in ASCA-E of the significant effects as well as residuals.

Citrate, Time and the Hippurate:Time interaction obtained with ASCA-E. When using different colors and symbols for factor levels, such graph can be very informative.

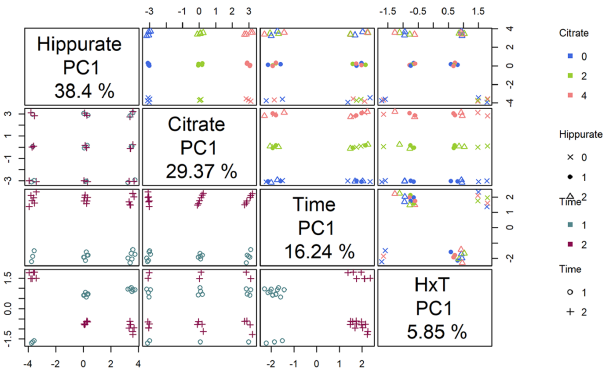

Figure 7: Output of the function `lmpScoreScatterPlotM`.

The interpretation of the interaction effects being always quite complicated in ASCA models, `limpca` offers two interesting functionalities. First, effect matrices can be combined before being decomposed by PCA. Second, the function `lmpEffectPlot` allows to represent the ASCA scores of an effect according to the levels of the same effect or another one. Figure 8 illustrates this on the Hippurate:Time interaction. In the first line, the scores of the Hippurate:Time interaction are displayed first for the 2 first PCs. They are then represented in a line plot with the function `lmpEffectPlot` for the two components separately. This allows to better visualize the interaction than with classical score plots. The first line gives the impression of a quite important interaction effect.

The second line shows the same graphs but on the basis of the PCA decomposition of the combination (sum) of the Hippurate, Time and Hippurate:Time effect matrices and shows that, even if significant, the interaction is quite small compared to the main Hippurate and Time effects. The Hippurate effect is mainly visible on the first PC and the Time effect on the second one.

In summary, all the graphs given above illustrate how ASCA methodology allows to analyze very deeply the informations contained in multivariate responses issued form multi-factorial designed experiments and discover information that is difficult to extract from a simple PCA.

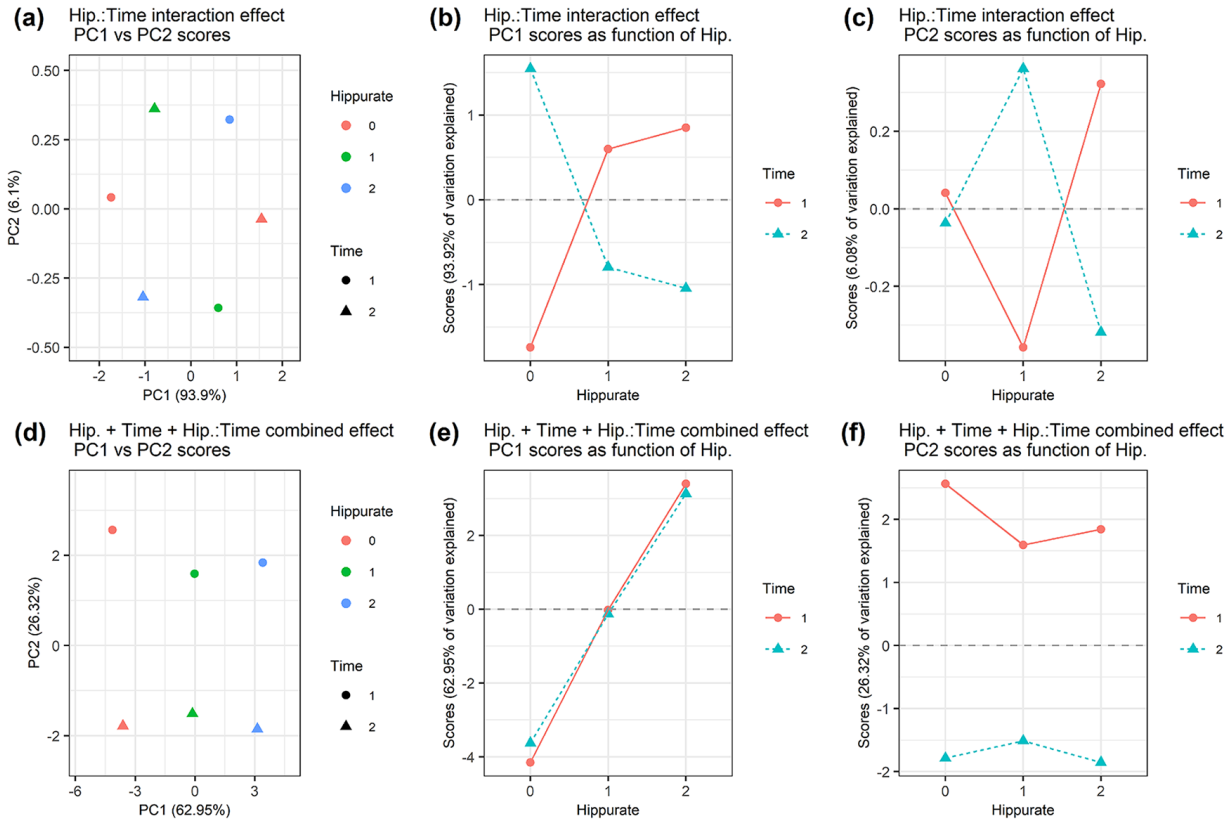

Figure 8: Outputs of `lmpEffectPlot` function.

5. Conclusions and perspectives

Experiments in omics fields, which aim to understand biological systems, produce a huge amount of data. Since experiments in these fields are often rely on multifactorial experimental designs, adapted methods are needed to interpret the data. In this context, and in many other fields where multiple responses are measured, the methodologies of ANOVA-simultaneous component analysis (ASCA) and ANOVA-PCA (APCA) have shown to be powerful and very-informative tools to explore deeply the information available in measured data sets. These methods are a combination of a parallel statistical modelling that allows each response of the outcomes matrix to be analyzed separately, followed by a dimension reduction step. Their extensions to ASCA+ and APCA+ were then developed to take into account unbalanced designs. This article presented the implementation of the R package `limpca`, which stands for *linear models with principal component effects analysis*, and allows the statistical analysis of high-dimensional designed data using the ASCA+ and APCA+ methods.

As illustrated by the case study, it enables the analysis of multivariate data obtained from a design of experiment and can handle all experimental designs containing fixed categorical factors. ASCA+ provided a much more subtle analysis than PCA since the decomposition of the effect matrices and the possibility to study each effect separately brought additional results that are highly informative. The case study has also highlighted the performances offered by the `limpca` package, such as data exploration functions, graphical features producing `ggplot` objects, tables and plots of variance percentages, and an optimised parametric bootstrap testing function that provides measures of importance and significance of the model effects.

`limpca` package has also been engineered to be open to further extensions to models including quantitative or random factors that are typically found in longitudinal, hierarchical or repeatability studies. Finally in the near future, the priority for the `limpca` package is to be submitted and published in Bioconductor, which is a free open source software project that supports the analysis of data coming from biological experiments.

Acknowledgements

The authors thank Université catholique de Louvain for its support and first author thanks Johnson & Jonhson, and Robin van Oirbeek (at OpenAnalytics at that time) for their support in the development of the early version of the `limpca` package.

Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Softwares

The R software environment was exclusively used (http://www.R-project.org).

Conflict of Interest

Authors declare that they have no conflict of interest.

Compliance with ethical requirements

This study analyzes collected data which are not involving human participants or animal experiments.

References

[1] M. Vailati-Riboni, V. Palombo, J. J. Loor, What Are Omics Sciences?, Springer International Publishing, Cham, 2017, pp. 1–7. doi:10.1007/978-3-319-43033-1_1.
URL https://doi.org/10.1007/978-3-319-43033-1_1

[2] J. Neter, M. Kutner, C. Nachtsheim, W. Li, Applied linear statistical models, 5th Edition, McGraw-Hill/Irwin, 2004.

[3] L. Stahle, S. Wold, Multivariate analysis of variance (manova), Chemometrics and Intelligent Laboratory Systems 9 (2) (1990) 127–141.

[4] J. J. Jansen, H. C. J. Hoefsloot, J. van der Greef, M. E. Timmerman, J. A. Westerhuis, A. K. Smilde, Asca: analysis of multivariate data obtained from an experimental design, Journal of Chemometrics 19 (9) (2005) 469–481. doi:10.1002/cem.952.
URL http://dx.doi.org/10.1002/cem.952

[5] A. K. Smilde, J. J. Jansen, H. C. J. Hoefsloot, R.-J. A. N. Lamers, J. van der Greef, M. E. Timmerman, Anova-simultaneous component analysis (asca): a new tool for analyzing designed metabolomics data, Bioinformatics 21 (13) (2005) 3043–3048. arXiv:http://bioinformatics.oxfordjournals.org/content/21/13/3043.full.pdf+html, doi:10.1093/bioinformatics/bti476.
URL http://bioinformatics.oxfordjournals.org/content/21/13/3043.abstract

[6] P. Harrington, N. E. Vieira, J. Espinoza, J. K. Nien, R. Romero, A. L. Yergey, Analysis of variance–principal component analysis: A soft tool for proteomic discovery, Analytica chimica acta 544 (1) (2005) 118–127.

[7] M. Thiel, B. Féraud, B. Govaerts, ASCA+ and APCA+: Extensions of ASCA and APCA in the analysis of unbalanced multifactorial designs, Journal of Chemometrics 31 (6) (Jun. 2017). doi:10.1002/cem.2895.
URL http://onlinelibrary.wiley.com.proxy.bib.ucl.ac.be:8888/doi/10.1002/cem.2895/abstract

[8] N. Benaiche, Stabilisation of the r package lmwire - linear models for wide responses, Master's thesis, UCLouvain, Belgium, Louvain-la-Neuve, Belgium (2021).

[9] C. Bertinetto, J. Engel, J. Jansen, Anova simultaneous component analysis: A tutorial review, Analytica Chimica Acta: X 6 (2020) 100061. doi:https://doi.org/10.1016/j.acax.2020.100061.
URL https://www.sciencedirect.com/science/article/pii/S2590134620300232

[10] A. H. Jarmund, T. S. Madssen, G. F. Giskeødegård, Alasca: An r package for longitudinal and cross-sectional analysis of multivariate data by asca-based methods, Frontiers in molecular biosciences 9 (2022) 962431. doi:10.3389/fmolb.2022.962431.
URL https://europepmc.org/articles/PMC9645785

[11] J. Xia, I. V. Sinelnikov, B. Han, D. S. Wishart, MetaboAnalyst 3.0—making metabolomics more meaningful, Nucleic Acids Research 43 (W1) (2015) W251–W257. arXiv:https://academic.oup.com/nar/article-pdf/43/W1/W251/17435854/gkv380.pdf, doi:10.1093/nar/gkv380.
URL https://doi.org/10.1093/nar/gkv380

[12] C. Fresno, M. Balzarini, E. Fernandez, lmdme: Linear models on designed multivariate experiments in r, Journal of statistical software 56 (01 2014). doi:10.18637/jss.v056.i07.

[13] T. Dorscheidt, MetStaT: Statistical metabolomics tools, r package version 1.0 (2019).
URL https://cran.r-project.org/src/contrib/Archive/MetStaT/

[14] P. Franceschi, gASCA: glm based ANOVA – Simultaneous Component Analysis (ASCA), r package version 1.0 (2022).

[15] K. H. Liland, multiblock: Multiblock Data Fusion in Statistics and Machine Learning, https://khliland.github.io/multiblock/, https://github.com/khliland/multiblock/ (2023).

[16] T. S. Madssen, G. F. Giskeødegård, A. K. Smilde, J. A. Westerhuis, Repeated measures asca+ for analysis of longitudinal intervention studies with multivariate outcome data, PLOS Computational Biology 17 (11) (2021) 1–21. doi:10.1371/journal.pcbi.1009585.
URL https://doi.org/10.1371/journal.pcbi.1009585

[17] S. Guisset, M. Martin, B. Govaerts, Comparison of parafasca, acomdim, and amopls approaches in the multivariate glm modelling of multi-factorial designs, Chemometrics and Intelligent Laboratory Systems 184 (2019) 44–63. doi:https://doi.org/10.1016/j.chemolab.2018.11.006.
URL https://www.sciencedirect.com/science/article/pii/S0169743917307748

[18] H. Madsen, P. Thyregod, Introduction to General and Generalized Linear Models, Chapman & Hall/CRC Texts in Statistical Science, Taylor & Francis, 2010.
URL https://books.google.be/books?id=JhuDNgAACAAJ

[19] G. Zwanenburg, H. C. Hoefsloot, J. A. Westerhuis, J. J. Jansen, A. K. Smilde, Anova–principal component analysis and anova–simultaneous component analysis: a comparison, Journal of Chemometrics 25 (10) (2011) 561–567. doi:10.1002/cem.1400.
URL http://dx.doi.org/10.1002/cem.1400

[20] R. Rousseau, Statistical contributrion to the analysis of metabonomic data in 1h-NMR spectroscopy, Ph.D. thesis, Université Catholique de Louvain, Louvain-la-Neuve, Belgium (2011).