# Package ‘NPLStoolbox’

May 7, 2026

**Title** N-Way Partial Least Squares Modelling of Multi-Way Data

**Version** 1.1.0

**Description** Creation and selection of N-way Partial Least Squares (NPLS) models. Selection of the optimal number of components can be done using ncrossreg(). NPLS was originally described by Rasmus Bro, see <doi:10.1002/%28SICI%291099-128X%28199601%2910%3A1%3C47%3A%3AAID-CEM400%3E3.0.CO%3B2-C>.

**License** MIT + file LICENSE

**Encoding** UTF-8

**RoxygenNote** 7.3.2

**Imports** dplyr, parafac4microbiome, pracma, rTensor, stats

**Depends** R (>= 2.10)

**LazyData** true

**Suggests** ggplot2, ggpubr, knitr, rmarkdown, testthat (>= 3.0.0), tidyr

**Config/testthat/edition** 3

**URL** `https://github.com/GRvanderPloeg/NPLStoolbox`,
`https://grvanderploeg.com/NPLStoolbox/`

**BugReports** `https://github.com/GRvanderPloeg/NPLStoolbox/issues`

**VignetteBuilder** knitr

**NeedsCompilation** no

**Author** Geert Roelof van der Ploeg [aut, cre] (ORCID: <`https://orcid.org/0009-0007-5204-3386`>),
Johan Westerhuis [ctb] (ORCID: <`https://orcid.org/0000-0002-6747-9779`>),
Anna Heintz-Buschart [ctb] (ORCID: <`https://orcid.org/0000-0002-9780-1933`>),
Age Smilde [ctb] (ORCID: <`https://orcid.org/0000-0002-3052-4644`>),
University of Amsterdam [cph, fnd]

**Maintainer** Geert Roelof van der Ploeg <`g.r.ploeg@uva.nl`>

**Repository** CRAN

**Date/Publication** 2025-07-31 08:30:02 UTC

# Contents

---

## `Cornejo2025` *Cornejo2025 longitudinal dataset measured in transgender persons*

---

### Description

The Cornejo2025 longitudinal dataset as three-dimensional arrays, with subjects in mode 1, features in mode 2 and time in mode 3.

### Usage

```
Cornejo2025
```

### Format

**'Cornejo2025':**

A list object with seven elements:

**Tongue_microbiome** List object of the tongue longitudinal microbiota data.
**Salivary_microbiome** List object of the saliva longitudinal microbiota data.
**Salivary_cytokines** List object of the longitudinal salivary cytokine data.
**Salivary_biochemistry** List object of the longitudinal salivary biochemistry data.
**Circulatory_hormones** List object of the longitudinal circulatory hormone data.
**Clinical_measurements** List object of the longitudinal clinical outcome data.
**Subject_metadata** Matrix with subject metadata.

### Source

TBD

---

## `Jakobsen2025` *Jakobsen2025 longitudinal dataset measured in mother-infant dyads*

---

### Description

The Jakobsen2025 longitudinal dataset as three-dimensional arrays, with subjects in mode 1, features in mode 2 and time in mode 3.

### Usage

```
Jakobsen2025
```

### Format

**'Jakobsen2025':**
A list object with seven elements:

**faeces** List object of the longitudinal infant faecal microbiota data.
**milkMicrobiome** List object of the longitudinal HM microbiota data.
**milkMetabolomics** List object of the longitudinal salivary cytokine data.

### Source

TBD

---

## `ncrossreg` *Cross-validation of NPLS by classical K-fold CV.*

---

### Description

This function runs ACMTF-R with cross-validation. A deterministic K–fold partition is used: the subjects are split in order into `cvFolds` groups. For each fold the training set consists of the other folds and the test set is the current fold.

### Usage

```
ncrossreg(X, y, maxNumComponents = 5, maxIter = 120, cvFolds = dim(X)[1])
```

### Arguments

| | |
|---|---|
| `X` | Centered tensor of independent data |
| `y` | Centered dependent variable |
| `maxNumComponents` | Maximum number of components to investigate (default 5). |
| `maxIter` | Maximum number of iterations (default 100). |

| | |
|---|---|
| `cvFolds` | Number of folds to use in the cross-validation. For example, if `cvFolds` is 5, then the subjects are deterministically partitioned into 5 groups (each CV iteration uses 4/5 for training and 1/5 for testing). Default: equal to the number of subjects (i.e. jack-knifing). |

## Value

A list with two elements: - **varExp**: a tibble with the variance–explained (for X and Y) per number of components. - **RMSE**: a tibble with the RMSE (computed over the unified CV prediction vector) per number of components.

## Examples

```
set.seed(123)
X <- array(rnorm(25 * 5 * 4), dim = c(25, 5, 4))
y <- rnorm(25)  # Random response variable
result = ncrossreg(X, y, cvFolds=2, maxNumComponents=2)
```

---

# `npred` *Predict Y for new data by projecting the data onto the latent space defined by an NPLS model.*

## Description

Predict Y for new data by projecting the data onto the latent space defined by an NPLS model.

## Usage

```
npred(model, newX)
```

## Arguments

| | |
|---|---|
| `model` | NPLS model |
| `newX` | New data organized in a matrix of (Inew x J x K) with Inew new subjects |

## Value

Ypred: vector of the predicted value(s) of Y for the new data

## Examples

```
Y = as.numeric(as.factor(Cornejo2025$Tongue$mode1$GenderID))
Ycnt = Y - mean(Y)
model = triPLS1(Cornejo2025$Tongue$data, Ycnt, numComponents=1)
npred(model, Cornejo2025$Tongue$data[1,,])
```

---

`triPLS1` *Tri-PLS1: three-way PLS regressed onto a y vector*

---

### Description

Tri-PLS1: three-way PLS regressed onto a y vector

### Usage

```
triPLS1(X, y, numComponents, tol = 1e-10, maxIter = 100)
```

### Arguments

| | |
|---|---|
| `X` | Centered tensor of independent data |
| `y` | Centered dependent variable |
| `numComponents` | Number of components to fit |
| `tol` | Relative change in loss for the model to converge (default 1e-10). |
| `maxIter` | Maximum number of iterations (default 100). |

### Value

Model

### Examples

```
set.seed(123)
X <- array(rnorm(100 * 5 * 4), dim = c(100, 5, 4))  # Random tensor (100 samples, 5 vars, 4 vars)
y <- rnorm(100)  # Random response variable
model <- triPLS1(X, y, numComponents = 2)
```

# Index