Package 'missPLS'

April 13, 2026

Type Package

Title Methods and Reproducible Workflows for Partial Least Squares with Missing Data

Version 0.2.0

Date 2026-04-07

Depends R (>= 4.1.0)

Imports mice, plsRglm, stats, utils, VIM

Suggests bcv, knitr, mlbench, plsdof, rmarkdown, testthat (>= 3.0.0)

Author Titin Agustin Nengsih [aut],
Frederic Bertrand [aut, cre],
Myriam Maumy-Bertrand [aut]

Maintainer Frederic Bertrand <frederic.bertrand@lecnam.net>

Description Methods-first tooling for reproducing and extending the partial least squares regression studies on incomplete data described in Nengsih et al. (2019) <doi:10.1515/sagmb-2018-0059>. The package provides simulation helpers, missingness generators, imputation wrappers, component-selection utilities, real-data diagnostics, and reproducible study orchestration for Nonlinear Iterative Partial Least Squares (NIPALS)-Partial Least Squares (PLS) workflows.

License GPL-3

Encoding UTF-8

LazyData true

VignetteBuilder knitr

Config/testthat/edition 3

RoxygenNote 7.3.3

URL https://fbertran.github.io/missPLS/, https://github.com/fbertran/missPLS

BugReports https://github.com/fbertran/missPLS/issues

NeedsCompilation no

Repository CRAN

Date/Publication 2026-04-13 15:00:02 UTC

Contents

`add_missingness` *Add missing values to a predictor matrix*

Description

Create MCAR or MAR missingness on the predictor matrix `x`. Missingness is generated column-wise so that each predictor receives approximately the same missing-data proportion, matching the simulation strategy used in the original work.

Usage

```
add_missingness(
  x,
  y,
  mechanism = c("MCAR", "MAR"),
  missing_prop,
  seed = NULL,
  mar_y_bias = 0.8
)
```

Arguments

`x`	Predictor matrix or data frame.
`y`	Numeric response vector.
`mechanism`	Missingness mechanism: `"MCAR"` or `"MAR"`.
`missing_prop`	Missing-data proportion as a fraction (`0.05`) or a percentage (`5`).
`seed`	Optional random seed. If supplied, it is used only for this call.
`mar_y_bias`	Proportion of missing values assigned to the upper half of the observed `y` values under the MAR mechanism.

Value

A list with components `x_incomplete`, `missing_mask`, `missing_prop`, `mechanism`, and `seed`.

Examples

```
sim <- simulate_pls_data(n = 20, p = 10, true_ncomp = 2, seed = 1)
miss <- add_missingness(sim$x, sim$y, mechanism = "MCAR", missing_prop = 10, seed = 2)
mean(is.na(miss$x_incomplete))
```

`bromhexine` *Bromhexine dataset*

Description

Bromhexine in pharmaceutical syrup used in the article and thesis.

Usage

```
bromhexine
```

Format

A `misspls_dataset` list with components:

name Dataset name.

x A numeric `23 x 64` predictor matrix.

y A numeric response vector of length `23`.

data A data frame with response `y` and predictors `x1` to `x64`.

source A short source reference.

preprocessing Dataset preprocessing notes.

notes Additional study notes.

Source

Goicoechea and Olivieri (1999a), calibration and test files bundled in `extra_docs/pls_data`.

diagnose_real_data *Diagnose a real dataset*

Description

Compute correlation summaries and VIF-style diagnostics for a packaged real dataset.

Usage

```
diagnose_real_data(dataset, cor_threshold = 0.7)
```

Arguments

dataset	A packaged dataset name or missplsdataset object.
cor_threshold	Absolute-correlation threshold used when reporting predictor pairs and predictor-response associations.

Value

A list with correlation and VIF summaries.

Examples

```
diag_bromhexine <- diagnose_real_data("bromhexine")
names(diag_bromhexine)
```

impute_pls_data *Impute a predictor matrix*

Description

Apply one of the imputation strategies used in the article and thesis.

Usage

```
impute_pls_data(
  x,
  method = c("mice", "knn", "svd"),
  seed = NULL,
  m,
  k = 15L,
  svd_rank = 10L,
  svd_maxiter = 1000L
)
```

Arguments

x	Incomplete predictor matrix or data frame.
method	Imputation method: "mice", "knn", or "svd".
seed	Optional random seed forwarded to stochastic imputers when supported.
m	Number of imputations for method = "mice". By default this is set to the missing-data percentage rounded to the nearest integer.
k	Number of neighbours for method = "knn".
svd_rank	Target rank for method = "svd".
svd_maxiter	Maximum number of iterations for the fallback SVD routine.

Value

A misspls_imputation object.

Examples

```
sim <- simulate_pls_data(n = 20, p = 10, true_ncomp = 2, seed = 1)
miss <- add_missingness(sim$x, sim$y, mechanism = "MCAR", missing_prop = 10, seed = 2)
imp <- impute_pls_data(miss$x_incomplete, method = "knn", seed = 3)
length(imp$datasets)
```

---

octane *Octane dataset*

Description

Octane in gasoline from NIR data used in the article and thesis.

Usage

```
octane
```

Format

A misspls_dataset list with components:

name Dataset name.

x A numeric 68 x 493 predictor matrix.

y A numeric response vector of length 68.

data A data frame with response y and predictors x1 to x493.

source A short source reference.

preprocessing Dataset preprocessing notes.

notes Additional study notes.

Source

Goicoechea and Olivieri (2003), calibration and test files bundled in extra_docs/pls_data.

ozone_complete *Complete-case ozone dataset*

Description

Los Angeles ozone pollution complete-case dataset used in the article and thesis.

Usage

```
ozone_complete
```

Format

A `misspls_dataset` list with components:

name Dataset name.

x A numeric `203 x 12` predictor matrix.

y A numeric response vector of length `203`.

data A data frame with response `y` and predictors `x1` to `x12`.

source A short source reference.

preprocessing Dataset preprocessing notes.

notes Additional study notes.

Source

`mlbench::Ozone`, restricted to the `203` complete observations used in the published analysis.

run_real_data_study *Run a real-data study*

Description

Run a real-data study

Usage

```
run_real_data_study(
  dataset,
  seed = NULL,
  missing_props = seq(5, 50, 5),
  mechanisms = c("MCAR", "MAR"),
  reps = 1L,
  baseline_reps = 100L,
  max_ncomp = 12L,
```

```
  criteria = c("Q2-LOO", "Q2-10fold", "AIC", "AIC-DoF", "BIC", "BIC-DoF"),
  incomplete_methods = c("nipals_standard", "nipals_adaptative"),
  imputation_methods = c("mice", "knn", "svd"),
  folds = 10L,
  mar_y_bias = 0.8
)
```

Arguments

`dataset`	A packaged dataset name or `misspls_dataset` object.
`seed`	Optional base random seed.
`missing_props`	Missing-data proportions as fractions or percentages.
`mechanisms`	Missing-data mechanisms.
`reps`	Number of replicate missingness draws for each mechanism and proportion.
`baseline_reps`	Number of repeated complete-data `Q2-10fold` selections used to determine `t**`.
`max_ncomp`	Maximum number of extracted components.
`criteria`	Criteria evaluated on incomplete and imputed data.
`incomplete_methods`	Incomplete-data NIPALS workflows.
`imputation_methods`	Imputation methods.
`folds`	Number of folds used by `"Q2-10fold"`.
`mar_y_bias`	MAR bias parameter passed to `add_missingness()`.

Value

A data frame with one row per study run.

`run_simulation_study` *Run a simulation study*

Description

Run the simulation workflows used in the article and thesis.

Usage

```
run_simulation_study(
 dimensions = list(c(500L, 100L), c(500L, 20L), c(100L, 20L), c(80L, 25L), c(60L, 33L),
   c(40L, 50L), c(20L, 100L)),
 true_ncomp = c(2L, 4L, 6L),
 missing_props = seq(5, 50, 5),
 mechanisms = c("MCAR", "MAR"),
 reps = 1L,
```

```
  seed = NULL,
  max_ncomp = 8L,
  criteria = c("Q2-LOO", "Q2-10fold", "AIC", "AIC-DoF", "BIC", "BIC-DoF"),
  incomplete_methods = c("nipals_standard", "nipals_adaptative"),
  imputation_methods = c("mice", "knn", "svd"),
  folds = 10L,
  mar_y_bias = 0.8
)
```

Arguments

`dimensions`	List of `(n, p)` integer pairs.
`true_ncomp`	Vector of true component counts.
`missing_props`	Missing-data proportions as fractions or percentages.
`mechanisms`	Missing-data mechanisms.
`reps`	Number of replicates.
`seed`	Optional base random seed.
`max_ncomp`	Maximum number of extracted components.
`criteria`	Criteria evaluated on complete and imputed data.
`incomplete_methods`	Incomplete-data NIPALS workflows.
`imputation_methods`	Imputation methods.
`folds`	Number of folds used by `"Q2-10fold"`.
`mar_y_bias`	MAR bias parameter passed to `add_missingness()`.

Value

A data frame with one row per study run.

`select_ncomp` *Select the number of PLS components*

Description

Select the number of components for complete, imputed, or incomplete-data PLS workflows.

Usage

```
select_ncomp(
  x,
  y,
  method = c("complete", "nipals_standard", "nipals_adaptative"),
  criterion = c("Q2-LOO", "Q2-10fold", "AIC", "AIC-DoF", "BIC", "BIC-DoF"),
```

```
  max_ncomp,
  seed = NULL,
  folds = 10L,
  threshold = 0.0975
)
```

Arguments

x	Predictor matrix, dataset object, or misspls_imputation object.
y	Numeric response vector. This may be omitted when x already contains a response.
method	Selection workflow: "complete", "nipals_standard", or "nipals_adaptative".
criterion	Selection criterion: "Q2-LOO", "Q2-10fold", "AIC", "AIC-DoF", "BIC", or "BIC-DoF".
max_ncomp	Maximum number of components to consider.
seed	Optional random seed used by the cross-validation and imputation aggregation steps.
folds	Number of cross-validation folds used by "Q2-10fold".
threshold	Threshold applied to Q2 criteria.

Value

A one-row data frame describing the selected component count.

Examples

```
sim <- simulate_pls_data(n = 25, p = 10, true_ncomp = 2, seed = 1)
select_ncomp(sim$x, sim$y, method = "complete", criterion = "AIC", max_ncomp = 4, seed = 2)
```

---

simulate_pls_data *Simulate PLS data*

Description

Simulate a univariate-response PLS dataset using the Li et al.-style generator available in plsRglm.

Usage

```
simulate_pls_data(n, p, true_ncomp, seed = NULL, model = "li2002")
```

Arguments

n	Number of observations.
p	Number of predictors.
true_ncomp	True number of latent components.
seed	Optional random seed. If supplied, it is used only for this call.
model	Simulation model. Only "li2002" is currently supported.

Value

A list with components `x`, `y`, `data`, `true_ncomp`, `seed`, and `model`.

Examples

```
sim <- simulate_pls_data(n = 20, p = 10, true_ncomp = 2, seed = 42)
str(sim)
```

`summarize_simulation_study`
Summarize simulation or real-data study results

Description

Summarize simulation or real-data study results

Usage

```
summarize_simulation_study(results)
```

Arguments

`results`	A results data frame returned by `run_simulation_study()` or `run_real_data_study()`.

Value

A grouped summary data frame.

Examples

```
sim_results <- run_simulation_study(
  dimensions = list(c(30, 12)),
  true_ncomp = 2,
  missing_props = numeric(0),
  mechanisms = character(0),
  reps = 2,
  seed = 1
)
summarize_simulation_study(sim_results)
```

`tetracycline` *Tetracycline dataset*

Description

Tetracycline in serum used in the article and thesis.

Usage

```
tetracycline
```

Format

A `misspls_dataset` list with components:

name Dataset name.

x A numeric `107 x 101` predictor matrix.

y A numeric response vector of length `107`.

data A data frame with response `y` and predictors `x1` to `x101`.

source A short source reference.

preprocessing Dataset preprocessing notes.

notes Additional study notes.

Source

Goicoechea and Olivieri (1999b), calibration and test files bundled in `extra_docs/pls_data`.

Index