

Package ‘msPCA’

June 12, 2026

Type Package

Title Sparse Principal Component Analysis with Multiple Principal Components

Version 0.4.1

Date 2026-06-12

Description Implements an algorithm for computing multiple sparse principal components of a dataset. The method is based on Cory-Wright and Pauphilet “Sparse PCA with Multiple Principal Components” (2026) <[doi:10.48550/arXiv.2209.14790](https://doi.org/10.48550/arXiv.2209.14790)>. The algorithm uses an iterative deflation heuristic with a truncated power method applied at each iteration to compute sparse principal components with controlled sparsity.

License MIT + file LICENSE

Imports Rcpp (>= 1.0.11)

Suggests datasets, knitr, rmarkdown

LinkingTo Rcpp, RcppEigen

RoxygenNote 7.3.3

Encoding UTF-8

VignetteBuilder knitr

NeedsCompilation yes

Author Ryan Cory-Wright [aut, cph] (ORCID: <<https://orcid.org/0000-0002-4485-0619>>),
Jean Pauphilet [aut, cre, cph] (ORCID: <<https://orcid.org/0000-0001-6352-0984>>)

Maintainer Jean Pauphilet <jpauphilet@london.edu>

Repository CRAN

Date/Publication 2026-06-12 16:30:09 UTC

Contents

feasibility_violation_off	2
-------------------------------------	---

fraction_variance_explained	3
fraction_variance_explained_perPC	3
mspca	4
print_mspca	5
tpm	6
variance_explained_perPC	7
Index	8

feasibility_violation_off

Feasibility Violation

Description

Computes the feasibility violation defined as $\sum_{t>s} u_t^\top u_s$ if orthogonality constraints are enforced (feasibilityConstraintType = 0) and $\sum_{t>s} u_t^\top C u_s$ if zero-correlation constraints are enforced (feasibilityConstraintType = 1).

Usage

```
feasibility_violation_off(C, U, feasibilityConstraintType)
```

Arguments

C A matrix. The correlation or covariance matrix (p x p).

U A matrix. Each column correspond to an p-dimensional PC.

feasibilityConstraintType An integer. Type of feasibility constraints to be enforced. 0: orthogonality constraints; 1: uncorrelatedness constraints.

Value

A float.

Examples

```
TestMat <- cor(datasets::mtcars)
mspcars <- mspca(TestMat, 2, c(4,4))
feasibility_violation_off(TestMat, mspcars$x_best, 0)
```

fraction_variance_explained
Fraction of Variance Explained

Description

Computes the fraction of variance explained (variance explained normalized by the trace of the covariance/correlation matrix) by a set of PCs.

Usage

```
fraction_variance_explained(C, U)
```

Arguments

C A matrix. The correlation or covariance matrix (p x p).
U A matrix. The matrix containing the r PCs (p x r).

Value

A float.

Examples

```
TestMat <- cor(datasets::mtcars)
mspcars <- mspca(TestMat, 2, c(4,4))
fraction_variance_explained(TestMat, mspcars$x_best)
```

fraction_variance_explained_perPC
Fraction of Variance Explained per PC

Description

Computes the fraction of variance explained (variance explained normalized by the trace of the covariance/correlation matrix) by each PC.

Usage

```
fraction_variance_explained_perPC(C, U)
```

Arguments

C A matrix. The correlation or covariance matrix (p x p).
U A matrix. The matrix containing the r PCs (p x r).

Value

An array.

mspca

Multiple Sparse PCA

Description

Returns multiple sparse principal component of a matrix using an iterative deflation heuristic.

Usage

```
mspca(
  Sigma,
  r,
  ks,
  feasibilityConstraintType = 0L,
  verbose = TRUE,
  maxIter = 200L,
  feasibilityTolerance = 1e-04,
  stallingTolerance = 1e-08,
  timeLimitTPM = 20L,
  maxRestartTPM = 20L,
  minRestartTPM = 10L
)
```

Arguments

Sigma	A matrix. The correlation or covariance matrix, whose sparse PCs will be computed.
r	An integer. Number of principal components (PCs) to be computed.
ks	A list of integers. Target sparsity of each PC.
feasibilityConstraintType	(optional) An integer. Type of feasibility constraints to be enforced. 0: orthogonality constraints; 1: uncorrelatedness constraints. Default 0.
verbose	(optional) A Boolean. Controls console output. Default TRUE.
maxIter	(optional) An integer. Maximum number of iterations of the algorithm. Default 200.
feasibilityTolerance	(optional) A float. Tolerance for the violation of the orthogonality constraints. Default 1e-4
stallingTolerance	(optional) A float. Controls the objective improvement below which the algorithm is considered to have stalled. Default 1e-8

- `timeLimitTPM` (optional) An integer. Maximum time in seconds for the truncated power method (inner iteration). Default 20.
- `maxRestartTPM` (optional) An integer. Number of random restarts of the truncated power method (inner iteration) for the first outer iteration. Default 20.
- `minRestartTPM` (optional) An integer. Number of random restarts of the truncated power method (inner iteration) for outer iterations ≥ 2 . Default 10.

Value

An object with 4 fields: 'x_best' (p x r array containing the sparse PCs), 'objective_value', 'feasibility_violation', 'runtime'.

Examples

```
TestMat <- cor(datasets::mtcars)
mspca(TestMat, 2, c(4,4))
```

```
print_mspca          Print msPCA Output
```

Description

Displays the output of the msPCA algorithm.

Usage

```
print_mspca(sol_object, C, digits = 3)
```

Arguments

- `sol_object` A list. The output of the mspca or tpm function.
- `C` A matrix. The correlation or covariance matrix (p x p).
- `digits` An integer. Number of digits used for rounded display. Default 3.

Value

None. Prints output to console.

Examples

```
TestMat <- cor(datasets::mtcars)
mspcars <- mspca(TestMat, 2, c(4,4))
print_mspca(mspcars, TestMat, digits = 3)
```

tpm

Truncated Power Method

Description

Returns the leading sparse principal component of a matrix using the truncated power method.

Usage

```
tpm(Sigma, k, maxIter = 200L, verbose = TRUE, timeLimit = 10L)
```

Arguments

Sigma	A matrix. The correlation or covariance matrix, whose sparse PCs will be computed.
k	An integer. Target sparsity of the PC.
maxIter	(optional) An integer. Maximum number of iterations of the algorithm. Default 200.
verbose	(optional) A Boolean. Controls console output. Default TRUE.
timeLimit	(optional) An integer. Maximum time in seconds. Default 10.

Value

An object with 3 fields: 'x_best' (p x 1 array containing the sparse PC), 'objective_value', 'runtime'.

References

Yuan, X. T., & Zhang, T. (2013). Truncated power method for sparse eigenvalue problems. *The Journal of Machine Learning Research*, 14(1), 899-925.

Examples

```
TestMat <- cor(datasets::mtcars)
tpm(TestMat, 4)
```

variance_explained_perPC
Variance Explained per PC

Description

Computes the variance explained by each PC.

Usage

variance_explained_perPC(C, U)

Arguments

- C A matrix. The correlation or covariance matrix (p x p).
- U A matrix. The matrix containing the r PCs (p x r).

Value

An array.

Index

feasibility_violation_off, [2](#)
fraction_variance_explained, [3](#)
fraction_variance_explained_perPC, [3](#)

mspca, [4](#)

print_mspca, [5](#)

tpm, [6](#)

variance_explained_perPC, [7](#)