

Package ‘nipals’

October 24, 2018

Title Principal Components Analysis using NIPALS with Gram-Schmidt Orthogonalization

Version 0.5

Date 2018-10-24

Description Principal Components Analysis of a matrix using Non-linear Iterative Partial Least Squares with Gram-Schmidt orthogonalization of the scores and loadings. Optimized for speed. See Andrecut (2009) <doi:10.1089/cmb.2008.0221>.

Depends R (>= 3.4.0)

License GPL-3

Encoding UTF-8

LazyData true

Suggests knitr, rmarkdown, testthat

URL <https://github.com/kwstat/nipals>

BugReports <https://github.com/kwstat/nipals/issues>

VignetteBuilder knitr

RoxygenNote 6.1.0

NeedsCompilation no

Author Kevin Wright [aut, cre] (<<https://orcid.org/0000-0002-0617-8673>>)

Maintainer Kevin Wright <kw.stat@gmail.com>

Repository CRAN

Date/Publication 2018-10-24 17:20:03 UTC

R topics documented:

nipals	2
uscrime	3
Index	5

nipals	<i>Principal component analysis by NIPALS, non-linear iterative partial least squares</i>
--------	---

Description

Used for finding principal components of a numeric matrix. Missing values in the matrix are allowed. Principal Components are extracted one a time. The algorithm computes $x = TP'$, where T is the 'scores' matrix and P is the 'loadings' matrix.

Usage

```
nipals(x, ncomp = min(nrow(x), ncol(x)), center = TRUE, scale = TRUE,
       maxiter = 500, tol = 1e-06, startcol = 0, fitted = FALSE,
       force.na = FALSE, gramschmidt = TRUE, verbose = FALSE)
```

Arguments

x	Numerical matrix for which to find principal components. Missing values are allowed.
ncomp	Maximum number of principal components to extract from x.
center	If TRUE, subtract the mean from each column of x.
scale	if TRUE, divide the standard deviation from each column of x.
maxiter	Maximum number of NIPALS iterations for each principal component.
tol	Default 1e-9 tolerance for testing convergence of the NIPALS iterations for each principal component.
startcol	Determine the starting column of x for the iterations of each principal component. If 0, use the column of x that has maximum variation. If a number, use that column of x. If a function, apply the function to each column of x and choose the column with the maximum value of the function.
fitted	Default FALSE. If TRUE, return the fitted (reconstructed) value of x.
force.na	Default FALSE. If TRUE, force the function to use the method for missing values, even if there are no missing values in x.
gramschmidt	Default TRUE. If TRUE, perform Gram-Schmidt orthogonalization at each iteration.
verbose	Default FALSE. Use TRUE or 1 to show some diagnostics.

Value

A list with components eig, scores, loadings, ncomp, R2, xhat, iter.

Author(s)

Kevin Wright

References

- Wold, H. (1966) Estimation of principal components and related models by iterative least squares. In *Multivariate Analysis* (Ed., P.R. Krishnaiah), Academic Press, NY, 391-420.
- Andrecut, Mircea (2009). Parallel GPU implementation of iterative PCA algorithms. *Journal of Computational Biology*, 16, 1593-1599.

Examples

```
B <- matrix(c(50, 67, 90, 98, 120,
             55, 71, 93, 102, 129,
             65, 76, 95, 105, 134,
             50, 80, 102, 130, 138,
             60, 82, 97, 135, 151,
             65, 89, 106, 137, 153,
             75, 95, 117, 133, 155), ncol=5, byrow=TRUE)
rownames(B) <- c("G1", "G2", "G3", "G4", "G5", "G6", "G7")
colnames(B) <- c("E1", "E2", "E3", "E4", "E5")
dim(B) # 7 x 5
p1 <- nipals(B)
dim(p1$scores) # 7 x 5
dim(p1$loadings) # 5 x 5

B2 = B
B2[1,1] = B2[2,2] = NA
p2 = nipals(B2, fitted=TRUE)
```

uscrime

U.S. Crime rates per 100,00 people

Description

U.S. Crime rates per 100,00 people for 7 categories in each of the 50 U.S. states in 1977.

Usage

```
uscrime
```

Format

A data frame with 50 observations on the following 8 variables.

state U.S. state
murder murders
rape rapes
robbery robbery
assault assault

burglary burglary

larceny larceny

autotheft automobile thefts

Details

There are two missing values.

Source

Documentation Example 3 for PROC HPPRINCOMP. <http://documentation.sas.com/api/docsets/stathpug/14.2/content/stathpug142ex301.html>

References

SAS/STAT User's Guide: High-Performance Procedures. The HPPRINCOMP Procedure. <http://support.sas.com/documentation/cdl/en/hpprocs/64302.pdf>

Examples

```
library(nipals)
head(uscrime)

# SAS deletes rows with missing values
dat <- uscrime[complete.cases(uscrime), ]
dat <- as.matrix(dat[ , -1])
m1 <- nipals(dat) # complete-data method

# Traditional NIPALS with missing data
dat <- uscrime
dat <- as.matrix(dat[ , -1])
m2 <- nipals(dat, gramschmidt=FALSE) # missing
round(crossprod(m2$loadings),3) # Prin Comps not quite orthogonal

# Gram-Schmidt corrected NIPALS
m3 <- nipals(dat, gramschmidt=TRUE) # TRUE is default
round(crossprod(m3$loadings),3) # Prin Comps are orthogonal
```

Index

*Topic **datasets**

uscrime, [3](#)

nipals, [2](#)

uscrime, [3](#)