

Package ‘ccTensor’

August 12, 2021

Type Package

Title CUR/CX Tensor Decomposition

Version 1.0.2

Date 2021-08-02

Depends R (>= 4.1.0)

Imports methods, fields, MASS, igraph, rTensor

Suggests testthat, nnTensor

Description CUR/CX decomposition factorizes a matrix into two factor matrices and Multidimensional CX Decomposition factorizes a tensor into a core tensor and some factor matrices. See the reference section of GitHub README.md <<https://github.com/rikenbit/ccTensor>>, for details of the methods.

License Artistic-2.0

URL <https://github.com/rikenbit/ccTensor>

NeedsCompilation no

Author Koki Tsuyuzaki [aut, cre],
Itoshi Nikaido [aut]

Maintainer Koki Tsuyuzaki <k.t.the-answer@hotmail.co.jp>

Repository CRAN

Date/Publication 2021-08-12 04:30:07 UTC

R topics documented:

ccTensor-package	2
CUR	3
CX	4
MultiCX	5
Index	7

ccTensor-package *CUR/CX Tensor Decomposition*

Description

CUR/CX decomposition factorizes a matrix into two factor matrices and Multidimensional CX Decomposition factorizes a tensor into a core tensor and some factor matrices. See the reference section of GitHub README.md <<https://github.com/rikenbit/ccTensor>>, for details of the methods.

Details

The DESCRIPTION file:

```
Package:      ccTensor
Type:         Package
Title:        CUR/CX Tensor Decomposition
Version:      1.0.2
Date:         2021-08-02
Authors@R:   c(person("Koki", "Tsuyuzaki", role = c("aut", "cre"), email = "k.t.the-answer@hotmail.co.jp"), person("Itoshi", "Nikaido", role = c("aut", "cre"), email = "itoshi.nikaido@gmail.com"))
Depends:      R (>= 4.1.0)
Imports:      methods, fields, MASS, igraph, rTensor
Suggests:    testthat, nnTensor
Description:  CUR/CX decomposition factorizes a matrix into two factor matrices and Multidimensional CX Decomposition factorizes a tensor into a core tensor and some factor matrices.
License:      Artistic-2.0
URL:         https://github.com/rikenbit/ccTensor
Author:       Koki Tsuyuzaki [aut, cre], Itoshi Nikaido [aut]
Maintainer:  Koki Tsuyuzaki <k.t.the-answer@hotmail.co.jp>
```

Index of help topics:

```
ccTensor-package      CUR/CX Tensor Decomposition
CUR                    CUR Matrix Decomposition
CX                     CX Matrix Decomposition
MultiCX                MultiCX Tensor Decomposition
```

Author(s)

NA
Maintainer: NA

References

Michael W. Mahoney, et. al., (2009). CUR matrix decompositions for improved data analysis. *PNAS*

Petros Drineas et.al., (2008). Relative-error CUR Matrix Decompositions. *SIAM J. Matrix Anal. Appl.*

Maria F. K. B. et. al. (2019). Multidimensional CX Decomposition of Tensors. *WCNPS*

See Also

[CX,CUR,MulticX](#)

Examples

```
ls("package:ccTensor")
```

CUR

CUR Matrix Decomposition

Description

The input data is assumed to be a matrix. CUR decomposes the matrix to three low-dimensional factor matrices. C and R are not estimated values but the actual column and row vectors sampled from the matrix.

Usage

```
CUR(A, c.rank=NULL, r.rank=NULL, thr=0.9,
     c.method=c("best.match", "random", "exact.num.random", "top.scores"),
     u.method=c("invCR", "invW"),
     r.method=c("best.match", "random", "exact.num.random", "top.scores"))
```

Arguments

A	The input matrix which has N-rows and M-columns.
c.rank	The number of low-dimension of C ($J1 < N,M$). If this argument is not specified or specified as NULL, the low-dimension is estimated based on the cumulative singular value (Default: NULL).
r.rank	The number of low-dimension of R ($J2 < N,M$). If this argument is not specified or specified as NULL, the low-dimension is estimated based on the cumulative singular value (Default: NULL).
thr	The threshold to determine the low-dimension J1 and J2. The value must be range 0 to 1 (Default: 0.9).
c.method	The column sampling algorithm (Default: best.match).
u.method	The algorithm to calculate U (Default: invCR).
r.method	The row sampling algorithm (Default: best.match).

Value

C: A N-rows and J1-columns matrix contains the sampled column vectors from the input matrix A.
 U: A J1-rows and J2-columns matrix. R: A J2-rows and M-columns matrix contains the sampled row vectors from the input matrix A. indC: The sampled column indices. indR: The sampled row indices. RecError : The reconstruction error between data matrix and reconstructed matrix from C and X.

Author(s)

Koki Tsuyuzaki

References

Michael W. Mahoney, et. al., (2009). CUR matrix decompositions for improved data analysis. *PNAS*

Examples

```
library("ccTensor")
library("nnTensor")
# Test data
matdata <- toyModel(model = "NMF")
# Simple usage
out <- CUR(matdata, c.rank=3, r.rank=4)
```

 CX

CX Matrix Decomposition

Description

The input data is assumed to be a matrix. CX decomposes the matrix to two low-dimensional factor matrices. C is not an estimated values but the actual column vectors sampled from the matrix.

Usage

```
CX(A, rank=NULL, thr=0.9,
   c.method=c("best.match", "random", "exact.num.random", "top.scores"))
```

Arguments

A	The input matrix which has N-rows and M-columns.
rank	The number of low-dimension ($J < N, M$). If this argument is not specified or specified as NULL, the low-dimension is estimated based on the cumulative singular value (Default: NULL).
thr	The threshold to determine the low-dimension J. The value must be range 0 to 1 (Default: 0.9).
c.method	The column sampling algorithm (Default: best.match).

Value

C: A N-rows and J-columns matrix contains the sampled column vectors from the input matrix A. X: A J-rows and M-columns matrix. indC: The sampled column indices. RecError: The reconstruction error between data matrix and reconstructed matrix from C and X.

Author(s)

Koki Tsuyuzaki

References

Petros Drineas et.al., (2008). Relative-error CUR Matrix Decompositions. *SIAM J. Matrix Anal. Appl.*

Examples

```
library("ccTensor")
library("nnTensor")
# Test data
matdata <- toyModel(model = "NMF")
# Simple usage
out <- CX(matdata, rank=5)
```

MultiCX

*MultiCX Tensor Decomposition***Description**

The input data is assumed to be a tensor. MultiCX decomposes the tensor into a core tensor and some factor matrices. The factor matrices are not estimated values but the actual column vectors sampled from the unfolded matrix in each mode.

Usage

```
MultiCX(Y, rank=NULL, modes=1:3, thr=0.9,
        c.method=c("best.match", "random", "exact.num.random", "top.scores"))
```

Arguments

Y	The input tensor (e.g. N times M times L).
rank	The number of low-dimension of factor matrices (e.g. J1, J2, and J3). If this argument is not specified or specified as NULL, the low-dimension is estimated based on the cumulative singular value (Default: NULL).
modes	The vector of the modes on which to perform the decomposition (Default: 1:3 <all modes>).
thr	The threshold to determine the low-dimension of factor matrices. The value must be range 0 to 1 (Default: 0.9).
c.method	The column sampling algorithm (Default: best.match).

Value

U: Core tensor (e.g. J1 times J2 times J3). C: Factor matrices (e.g. C_1: ????????) RecError : The reconstruction error between data tensor and reconstructed tensor from C and X.

Author(s)

Koki Tsuyuzaki

References

Maria F. K. B. et. al. (2019). Multidimensional CX Decomposition of Tensors. *WCNPS*

Examples

```
library("ccTensor")
library("nnTensor")
# Test data
tensdata <- toyModel(model = "CP")
# Simple usage
out <- MultiCX(tensdata, rank=c(3,4,5))
```

Index

* **methods**

CUR, [3](#)

CX, [4](#)

MultiCX, [5](#)

* **package**

ccTensor-package, [2](#)

ccTensor (ccTensor-package), [2](#)

ccTensor-package, [2](#)

CUR, [3, 3](#)

CX, [3, 4](#)

MultiCX, [3, 5](#)