Package: rTensor2 (via r-universe)

October 12, 2024

Type Package

Title MultiLinear Algebra

Version 2.0.0

Maintainer Kyle Caudle <kyle.caudle@sdsmt.edu>

Description A set of tools for basic tensor operators. A tensor in the context of data analysis in a multidimensional array. The tools in this package rely on using any discrete transformation (e.g. Fast Fourier Transform (FFT)). Standard tools included are the Eigenvalue decomposition of a tensor, the QR decomposition and LU decomposition. Other functionality includes the inverse of a tensor and the transpose of a symmetric tensor. Functionality in the package is outlined in Kernfeld et al. (2015) <https://www.sciencedirect.com/science/article/pii/S0024379515004358>.

Depends R (>= 4.2.0)

Imports methods,raster,png,rTensor, wavethresh, gsignal, Matrix, matrixcalc

License GPL-3

Encoding UTF-8

LazyData true

LazyDataCompression xz

Config/testthat/edition 3

RoxygenNote 7.2.3

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Author Kyle Caudle [aut, cre], Randy Hoover [ctb], Jackson Cates [ctb]

Date/Publication 2024-03-29 16:30:02 UTC

Repository https://kylecaudle.r-universe.dev

RemoteUrl https://github.com/cran/rTensor2

RemoteRef HEAD

RemoteSha f15710b17eb635eb7317eb10db3c4edf2fbf34ab

Contents

as.tensor 3
LU 4
Mnist 4
polar 5
QR 6
rand_tensor 6
raytrace 7
tDWT 8
tEIG 9
tEIGdct 10
tEIGdht 11
tEIGdst 12
tEIGdwht 12
tEIGdwt 13
tEIGfft 14
tIDWT 15
tINV 16
tINVdct 17
tINVdht 17
tINVdst 18
tINVdwht 19
tINVdwt 19
tINVfft 20
tLDA 21
tLU 22
tLUdct 23
tLUdht 24
tLUdst 24
tLUdwht 25
tLUdwt 26
tLUfft 26
tmean 27
tmult 28
tQR 29
tQRdct 30
tQRdht 31
tQRdst 31
tQRdwht 32
tQRdwt 33
tQRfft 34
tSVD 34
tSVDdct 36

tSVDdht 37
tSVDdst 38
tSVDdwht 39
tSVDdwt 40
tSVDfft 41
t_tpose 42

Index **43**

`as.tensor` *Tensor Conversion*

Description

Create a Tensor-class object from an array, matrix, or vector.

Usage

```
as.tensor(x, drop = FALSE)
```

Arguments

`x`	: an instance of array, matrix, or vector
`drop`	: whether or not modes equal to 1 should be dropped

Value

a Tensor-class object

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

References

Imported from rTensor package version 1.4.8.

Examples

```
#From vector
vec <- runif(3); vecT <- as.tensor(vec); vecT
#From matrix
mat <- matrix(runif(2*3),nrow=2,ncol=3)
matT <- as.tensor(mat); matT
#From array
indices <- c(2,3,4)
arr <- array(runif(prod(indices)), dim = indices)
arrT <- as.tensor(arr); arrT
```

LU — *LU Decomposition of a Complex Matrix*

Description

Decompose a square matrix A into the product of a lower triangular matrix L and an upper triangular matrix U.

Usage

```
LU(A)
```

Arguments

A	: an n x n matrix

Value

a lower triangular matrix L and an upper triangular matrix U so that A = LU.

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

Examples

```
z <- complex(real = rnorm(16), imag = rnorm(16))
A <- matrix(z,nrow=4)
LU(A)
```

Mnist — *Subset of MNIST training and testing data.*

Description

10000 MNIST training images (1000 of every digit), reformatted into a tensor: 28 x 10000 x 28. 1000 MNIST test images (100 of every digit), reformatted into a tensor: 28 x 1000 x 28

Usage

```
data("Mnist")
```

Format

The format is:

Mnist$train$images, Mnist$train$labels

Mnist$test$images, Mnist$test$labels

References

Deng L (2012). "The mnist database of handwritten digit images for machine learning research." IEEE Signal Processing Magazine, 29(6), 141–142

Examples

```
data(tensor)
```

`polar` *Polar/Jordan Form of a Matrix*

Description

Creates the polar/Jordan form of the P and D matrices after performing eigenvalue decomposition where the eigenvalue values are complex.

Usage

```
polar(P,D)
```

Arguments

`P`	: the eigenvectors from an eigenvalue decomposition.
`D`	: the eigenvalues from an eigenvalue decomposition.

Value

P the polar form (real-valued) matrix of eigenvectors.

D the polar form (real-valued) matrix of eigenvalues.

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

Examples

```
z <- complex(real = rnorm(16), imag = rnorm(16))
M <- matrix(z,nrow=4)
decomp <- eigen(M)
polar(decomp$vectors,decomp$values)
```

QR — *QR Decomposition of a Complex Matrix Without Pivoting*

Description

Performs QR Decomposition of a Complex Matrix without pivoting.

Usage

```
QR(A)
```

Arguments

`A`	: an n x n matrix

Value

an orthogonal matrix Q and an upper triangular matrix R so that A = QR.

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

Examples

```
z <- complex(real = rnorm(16), imag = rnorm(16))
A <- matrix(z,nrow=4)
QR(A)
```

rand_tensor — *Tensor with Random Entries*

Description

Generate a Tensor with specified modes with iid normal(0,1) entries.

Usage

```
rand_tensor(modes = c(3, 4, 5), drop = FALSE)
```

Arguments

`modes`	: the modes of the output Tensor
`drop`	: whether or not modes equal to 1 should be dropped

Value

a Tensor object with modes given by modes

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

References

Imported from rTensor package version 1.4.8.

Examples

```
rand_tensor()
rand_tensor(c(4,4,4))
rand_tensor(c(10,2,1),TRUE)
```

---

raytrace *Subset of raytrace data*

Description

4 tensors (128 x 128 x 128) for 4 different gray scale images. boat, flashlight, keyboard, scooter.

Usage

```
data("raytrace")
```

Format

The format is:

raytrace$boat

raytrace$flashlight

raytrace$keyboard

raytrace$scooter

References

Hoover RC, Braman KS, Hao N (2011b). "Pose estimation from a single image using tensor decomposition and an algebra of circulants." In 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 2928–2934. IEEE.

Examples

```
data(raytrace)
```

`tDWT` *Discrete Wavelet Transform of a 3-D Tensor*

Description

Performs the Discrete Wavelet Transform of a 3-D Tensor.

Usage

```
tDWT(tnsr)
```

Arguments

`tnsr`	: a 3-mode tensor

Value

a Tensor-class object

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

References

G. Strang and T. Nguyen, Wavelets and filter banks. SIAM, 1996.

A. Haar, "Zur theorie der orthogonalen funktionensysteme", Mathematische annalen, vol. 69, no. 3, pp. 331-371, 1910.

Examples

```
T <- rand_tensor(modes=c(2,3,4))
print(tDWT(T))
```

tEIG *Tensor Eigenvalue Decomposition Using any Discrete Transform*

Description

Performs a Eigenvalue decomposition of 3-mode tensor using any discrete transform.

Usage

```
tEIG(tnsr, tform)
```

Arguments

`tnsr`	: a 3-mode tensor, n x n x k
`tform`	: Any discrete transform. Supported transforms are: fft: Fast Fourier Transform dwt: Discrete Wavelet Transform (Haar Wavelet) dct: Discrete Cosine transform dst: Discrete Sine transform dht: Discrete Hadley transform dwht: Discrete Walsh-Hadamard transform

Value

a Tensor-class object

If Eigenvalue decomposition is performed on a n x n x k tensor, the components in the returned value are:

P: A tensor of Eigenvectors (n x n x k)

D: An diagonal tensor of Eigenvalues (n x n x k)

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

References

Kernfeld, E., Kilmer, M., & Aeron, S. (2015). Tensor-tensor products with invertible linear transforms. Linear Algebra and its Applications, 485, 545-570.

M. E. Kilmer, C. D. Martin, and L. Perrone, “A third-order generalization of the matrix svd as a product of third-order tensors,” Tufts University, Department of Computer Science, Tech. Rep. TR-2008-4, 2008

K. Braman, "Third-order tensors as linear operators on a space of matrices", Linear Algebra and its Applications, vol. 433, no. 7, pp. 1241-1253, 2010.

Examples

```
T <- rand_tensor(modes=c(2,2,4))
tEIG(T,"dst")
```

---

tEIGdct *Tensor Eigenvalue Decomposition Using the Discrete Cosine Transform*

Description

Performs a Eigenvalue decomposition of 3-mode tensor using the discrete Cosine transform.

Usage

```
tEIGdct(tnsr)
```

Arguments

tnsr	: a 3-mode tensor

Value

a Tensor-class object

If Eigenvalue decomposition is performed on a n x n x k tensor, the components in the returned value are:

P: A tensor of Eigenvectors (n x n x k)

D: An diagonal tensor of Eigenvalues (n x n x k)

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

Examples

```
T <- rand_tensor(modes=c(2,2,4))
print(tEIGdct(T))
```

tEIGdht *Tensor Eigenvalue Decomposition Using the Discrete Hadley Transform*

Description

Performs a Eigenvalue decomposition of 3-mode tensor using the discrete Hadley transform.

Usage

```
tEIGdht(tnsr)
```

Arguments

`tnsr`	: a 3-mode tensor

Value

a Tensor-class object

If Eigenvalue decomposition is performed on a n x n x k tensor, the components in the returned value are:

P: A tensor of Eigenvectors (n x n x k)

D: An diagonal tensor of Eigenvalues (n x n x k)

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

Examples

```
T <- rand_tensor(modes=c(2,2,4))
print(tEIGdht(T))
```

tEIGdst — *Tensor Eigenvalue Decomposition Using the Discrete Sine Transform*

Description

Performs a Eigenvalue decomposition of 3-mode tensor using the discrete Sine transform.

Usage

```
tEIGdst(tnsr)
```

Arguments

`tnsr`	: a 3-mode tensor

Value

a Tensor-class object

If Eigenvalue decomposition is performed on a n x n x k tensor, the components in the returned value are:

P: A tensor of Eigenvectors (n x n x k)

D: An diagonal tensor of Eigenvalues (n x n x k)

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

Examples

```
T <- rand_tensor(modes=c(2,2,4))
print(tEIGdst(T))
```

tEIGdwht — *Tensor Eigenvalue Decomposition Using the Discrete Walsh-Hadamard Transform*

Description

Performs a Eigenvalue decomposition of 3-mode tensor using the discrete Walsh-Hadamard transform.

Usage

```
tEIGdwht(tnsr)
```

Arguments

`tnsr`	: a 3-mode tensor

Value

a Tensor-class object

If Eigenvalue decomposition is performed on a n x n x k tensor, the components in the returned value are:

P: A tensor of Eigenvectors (n x n x k)

D: An diagonal tensor of Eigenvalues (n x n x k)

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

Examples

```
T <- rand_tensor(modes=c(2,2,4))
print(tEIGdwht(T))
```

`tEIGdwt` *Tensor Eigenvalue Decomposition Using the Discrete Wavelet Transform*

Description

Performs a Eigenvalue decomposition of 3-mode tensor using the discrete Wavelet transform (Haar Wavelet).

Usage

```
tEIGdwt(tnsr)
```

Arguments

`tnsr`	: a 3-mode tensor

Value

a Tensor-class object

If Eigenvalue decomposition is performed on a n x n x k tensor, the components in the returned value are:

P: A tensor of Eigenvectors (n x n x k)

D: An diagonal tensor of Eigenvalues (n x n x k)

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

References

G. Strang and T. Nguyen, Wavelets and filter banks. SIAM, 1996.

A. Haar, “Zur theorie der orthogonalen funktionensysteme,” Mathema- tische annalen, vol. 69, no. 3, pp. 331–371, 1910.

Examples

```
T <- rand_tensor(modes=c(2,2,4))
print(tEIGdwt(T))
```

`tEIGfft` *Tensor Eigenvalue Decomposition Using the Discrete Fourier Transform*

Description

Performs a Eigenvalue decomposition of 3-mode tensor using the discrete Fourier transform.

Usage

```
tEIGfft(tnsr)
```

Arguments

`tnsr`	: a 3-mode tensor

Value

a Tensor-class object

If Eigenvalue decomposition is performed on a n x n x k tensor, the components in the returned value are:

P: A tensor of Eigenvectors (n x n x k)

D: An diagonal tensor of Eigenvalues (n x n x k)

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

Examples

```
T <- rand_tensor(modes=c(2,2,4))
print(tEIGfft(T))
```

---

tIDWT *Discrete Inverse Wavelet Transform of a 3-D Tensor*

Description

Performs the Discrete Inverse Wavelet Transform of a 3-D Tensor.

Usage

```
tIDWT(tnsr)
```

Arguments

tnsr	: a 3-mode tensor

Value

a Tensor-class object

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

References

G. Strang and T. Nguyen, Wavelets and filter banks. SIAM, 1996.

A. Haar, "Zur theorie der orthogonalen funktionensysteme", Mathematische annalen, vol. 69, no. 3, pp. 331-371, 1910.

Examples

```
T <- rand_tensor(modes=c(2,3,4))
print(tIDWT(T))
```

`tINV` *Inverse of a 3-mode Tensor Using any Discrete Transform*

Description

Performs the inverse of a tensor using the any discrete transform.

Usage

```
tINV(tnsr,tform)
```

Arguments

`tnsr`	: a 3-mode tensor
`tform`	: Any discrete transform. Supported transforms are: fft: Fast Fourier Transform dwt: Discrete Wavelet Transform (Haar Wavelet) dct: Discrete Cosine transform dst: Discrete Sine transform dht: Discrete Hadley transform dwht: Discrete Walsh-Hadamard transform

Value

a Tensor-class object

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

Examples

```
T <- rand_tensor(modes=c(2,2,4))
print(tINV(T,"dst"))
```

tINVdct — *Inverse of a 3-mode Tensor Using the Discrete Cosine Transform*

Description

Performs the inverse of a tensor using the discrete cosine transform.

Usage

```
tINVdct(tnsr)
```

Arguments

`tnsr` : a 3-mode tensor

Value

a Tensor-class object

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

Examples

```
T <- rand_tensor(modes=c(2,2,4))
print(tINVdct(T))
```

tINVdht — *Inverse of a 3-mode Tensor Using the Discrete Hartley Transform*

Description

Performs the inverse of a tensor using the discrete Hartley transform.

Usage

```
tINVdht(tnsr)
```

Arguments

`tnsr` : a 3-mode tensor

Value

a Tensor-class object

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

Examples

```
T <- rand_tensor(modes=c(2,2,4))
print(tINVdht(T))
```

---

tINVdst *Inverse of a 3-mode Tensor Using the Discrete Sine Transform*

Description

Performs the inverse of a tensor using the discrete sine transform.

Usage

```
tINVdst(tnsr)
```

Arguments

`tnsr`	: a 3-mode tensor

Value

a Tensor-class object

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

Examples

```
T <- rand_tensor(modes=c(2,2,4))
print(tINVdst(T))
```

tINVdwht — *Inverse of a 3-mode Tensor Using the Discrete Walsh-Hadamard Transform*

Description

Performs the inverse of a tensor using the discrete Walsh-Hadamard transform.

Usage

```
tINVdwht(tnsr)
```

Arguments

`tnsr`	: a 3-mode tensor

Value

a Tensor-class object

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

Examples

```
T <- rand_tensor(modes=c(2,2,4))
print(tINVdwht(T))
```

tINVdwt — *Inverse of a 3-mode Tensor Using the Discrete Wavelet Transform*

Description

Performs the inverse of a tensor using the discrete wavelet transform (Haar Wavelet).

Usage

```
tINVdwt(tnsr)
```

Arguments

`tnsr`	: a 3-mode tensor

Value

a Tensor-class object

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

Examples

```
T <- rand_tensor(modes=c(2,2,4))
print(tINVdwt(T))
```

---

tINVfft — *Inverse of a 3-mode Tensor Using the Discrete Fourier Transform*

Description

Performs the inverse of a tensor using the discrete Fourier transform.

Usage

```
tINVfft(tnsr)
```

Arguments

`tnsr`	: a 3-mode tensor

Value

a Tensor-class object

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

Examples

```
T <- rand_tensor(modes=c(2,2,4))
print(tINVfft(T))
```

`tLDA` *Linear Discriminate Analysis of a 3-mode Tensor Using any Discrete Transform*

Description

Performs linear discriminate analysis on a tensor using any discrete transform. Assumes tensor is sorted by classes.

Usage

```
tLDA(tnsr,nClass,nSamplesPerClass,tform)
```

Arguments

`tnsr`	: a 3-mode tensor
`nClass`	: Number of classes
`nSamplesPerClass`	: Samples in each class
`tform`	: one of six-discrete transforms. Supported transforms are: fft: Fast Fourier Transform dwt: Discrete Wavelet Transform (Haar Wavelet) dct: Discrete Cosine transform dst: Discrete Sine transform dht: Discrete Hadley transform dwht: Discrete Walsh-Hadamard transform

Value

a Tensor-class object

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

Examples

```
data("Mnist")
T <- Mnist$train$images
myorder <- order(Mnist$train$labels)
# tLDA need to be sorted by classes
T_sorted <- as.tensor(T[,myorder,])
# Using small tensor, 2 images for each class for demonstration
T <- T_sorted[,c(1:2,1001:1002,2001:2002,3001:3002,
```

```
        4001:4002,5001:5002,6001:6002,7001:7002,
        8001:8002,9001:9002),]
tLDA(T,10,2,"dct")
```

---

tLU *Tensor LU Decomposition Using Using Any Discrete Transform*

Description

Performs a tensor LU decomposition on any 3-mode tensor using any discrete transform.

Usage

```
tLU(tnsr,tform)
```

Arguments

tnsr	: a 3-mode tensor
tform	: Any discrete transform. Supported transforms are: fft: Fast Fourier Transform dwt: Discrete Wavelet Transform (Haar Wavelet) dct: Discrete Cosine transform dst: Discrete Sine transform dht: Discrete Hadley transform dwht: Discrete Walsh-Hadamard transform

Value

a Tensor-class object

If LU decomposition is performed on a n x n x k tensor, the components in the returned value are:

L: The lower triangular tensor object (n x n x k)

U: The upper triangular tensor object (n x n x k)

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

References

Kernfeld, E., Kilmer, M., & Aeron, S. (2015). Tensor-tensor products with invertible linear transforms. Linear Algebra and its Applications, 485, 545-570.

M. E. Kilmer, C. D. Martin, and L. Perrone, "A third-order generalization of the matrix svd as a product of third-order tensors," Tufts University, Department of Computer Science, Tech. Rep. TR-2008-4, 2008

K. Braman, "Third-order tensors as linear operators on a space of matrices", Linear Algebra and its Applications, vol. 433, no. 7, pp. 1241-1253, 2010.

Examples

```
T <- rand_tensor(modes=c(2,2,4))
tLU(T,"dst")
```

---

tLUdct — *Tensor LU Decomposition Using the Discrete Cosine Transform*

Description

Performs a LU decomposition of 3-mode tensor using the discrete Cosine transform.

Usage

```
tLUdct(tnsr)
```

Arguments

`tnsr`	: a 3-mode tensor

Value

a Tensor-class object

If LU decomposition is performed on a $nxnxk$ tensor, the components in the returned value are:

L: The lower triangular tensor object ($nxnxk$)

U: The upper triangular tensor object ($nxnxk$)

Author(s)

Kyle Caudle kyle.caudle@sdsmt.edu

Examples

```
T <- rand_tensor(modes=c(2,2,4))
print(tLUdct(T))
```

tLUdht — *Tensor LU Decomposition Using the Discrete Hartley Transform*

Description

Performs a LU decomposition of 3-mode tensor using the discrete Hartley transform.

Usage

```
tLUdht(tnsr)
```

Arguments

tnsr	: a 3-mode tensor

Value

a Tensor-class object

If LU decomposition is performed on a $nxnxk$ tensor, the components in the returned value are:

L: The lower triangular tensor object ($nxnxk$)

U: The upper triangular tensor object ($nxnxk$)

Author(s)

Kyle Caudle kyle.caudle@sdsmt.edu

Examples

```
T <- rand_tensor(modes=c(2,2,4))
print(tLUdht(T))
```

tLUdst — *Tensor LU Decomposition Using the Discrete Cosine Transform*

Description

Performs a LU decomposition of 3-mode tensor using the discrete Sine transform.

Usage

```
tLUdst(tnsr)
```

Arguments

tnsr	: a 3-mode tensor

Value

a Tensor-class object

If LU decomposition is performed on a $nxnxk$ tensor, the components in the returned value are:

L: The lower triangular tensor object ($nxnxk$)

U: The upper triangular tensor object ($nxnxk$)

Author(s)

Kyle Caudle kyle.caudle@sdsmt.edu

Examples

```
T <- rand_tensor(modes=c(2,2,4))
print(tLUdst(T))
```

---

tLUdwht — *Tensor LU Decomposition Using the Discrete Walsh-Hadamard Transform*

Description

Performs a LU decomposition of 3-mode tensor using the discrete Walsh-Hadamard transform.

Usage

```
tLUdwht(tnsr)
```

Arguments

`tnsr`	: a 3-mode tensor

Value

a Tensor-class object

If LU decomposition is performed on a $nxnxk$ tensor, the components in the returned value are:

L: The lower triangular tensor object ($nxnxk$)

U: The upper triangular tensor object ($nxnxk$)

Author(s)

Kyle Caudle kyle.caudle@sdsmt.edu

Examples

```
T <- rand_tensor(modes=c(2,2,4))
print(tLUdwht(T))
```

tLUdwt — *Tensor LU Decomposition Using the Discrete Wavelet Transform*

Description

Performs a LU decomposition of 3-mode tensor using the discrete Wavelet transform (Haar Wavelet).

Usage

```
tLUdwt(tnsr)
```

Arguments

`tnsr`	: a 3-mode tensor

Value

a Tensor-class object

If LU decomposition is performed on a $nxnxk$ tensor, the components in the returned value are:

L: The left singular value tensor object ($nxnxk$)

U: The right singular value tensor object ($nxnxk$)

Author(s)

Kyle Caudle kyle.caudle@sdsmt.edu

Examples

```
T <- rand_tensor(modes=c(2,2,4))
print(tLUdwt(T))
```

tLUfft — *Tensor LU Decomposition Using the Discrete Fourier Transform*

Description

Performs a LU decomposition of 3-mode tensor using the discrete Fourier transform.

Usage

```
tLUfft(tnsr)
```

Arguments

`tnsr`	: a 3-mode tensor

Value

a Tensor-class object

If LU decomposition is performed on a $nxnxk$ tensor, the components in the returned value are:

L: The lower triangular tensor object $(nxnxk)$

U: The upper triangular tensor object $(nxnxk)$

Author(s)

Kyle Caudle kyle.caudle@sdsmt.edu

Examples

```
T <- rand_tensor(modes=c(2,2,4))
print(tLUfft(T))
```

---

tmean — *Find the mean of a tensor*

Description

Find the mean of a 3-mode tensor.

Usage

```
tmean(tnsr)
```

Arguments

`tnsr`	: a 3-mode tensor

Value

a Tensor-class object

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

Examples

```
tnsr <- rand_tensor(modes=c(3,4,5))
tmean(tnsr)
```

tmult — *Tensor Multiplication Using Any Discrete Transform*

Description

Multiplies two 3-mode tensors using any discrete transform.

Usage

```
tmult(x,y,tform)
```

Arguments

`x`	: a 3-mode tensor
`y`	: a 3-mode tensor
`tform`	: Any discrete transform. Supported transforms are: fft: Fast Fourier Transform dwt: Discrete Wavelet Transform (Haar Wavelet) dct: Discrete Cosine transform dst: Discrete Sine transform dht: Discrete Hadley transform dwht: Discrete Walsh-Hadamard transform

Value

a Tensor-class object

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

Examples

```
T1 <- rand_tensor(modes=c(2,2,4))
T2 <- rand_tensor(modes=c(2,3,4))
print(tmult(T1,T2,"dst"))
```

tQR *Tensor QR Decomposition Using Using Any Discrete Transform*

Description

Performs a tensor QR decomposition on any 3-mode tensor using any discrete transform.

Usage

```
tQR(tnsr,tform)
```

Arguments

`tnsr`	: a 3-mode tensor
`tform`	: Any discrete transform. Supported transforms are: fft: Fast Fourier Transform dwt: Discrete Wavelet Transform (Haar Wavelet) dct: Discrete Cosine transform dst: Discrete Sine transform dht: Discrete Hadley transform dwht: Discrete Walsh-Hadamard transform

Value

a Tensor-class object

If the QR decomposition is performed on a $n \times n \times k$ tensor, the components in the returned value are:

Q: The left singular value tensor object ($n \times n \times k$)

R: The right singular value tensor object ($n \times n \times k$)

Author(s)

Kyle Caudle kyle.caudle@sdsmt.edu

References

Kernfeld, E., Kilmer, M., & Aeron, S. (2015). Tensor-tensor products with invertible linear transforms. Linear Algebra and its Applications, 485, 545-570.

M. E. Kilmer, C. D. Martin, and L. Perrone, “A third-order generalization of the matrix svd as a product of third-order tensors,” Tufts University, Department of Computer Science, Tech. Rep. TR-2008-4, 2008

K. Braman, "Third-order tensors as linear operators on a space of matrices", Linear Algebra and its Applications, vol. 433, no. 7, pp. 1241-1253, 2010.

Examples

```
T <- rand_tensor(modes=c(2,2,4))
tQR(T,"dst")
```

---

tQRdct *Tensor QR Decomposition Using the Discrete Cosine Transform*

Description

Performs a QR decomposition of 3-mode tensor using the discrete Cosine transform.

Usage

```
tQRdct(tnsr)
```

Arguments

`tnsr`	: a 3-mode tensor

Value

a Tensor-class object

If QR decomposition is performed on a n x n x k tensor, the components in the returned value are:

Q: An orthogonal tensor (n x n x k).

R: An upper triangular tensor (n x n x k)

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

Examples

```
T <- rand_tensor(modes=c(2,2,4))
print(tQR(T,"dct"))
```

tQRdht *Tensor QR Decomposition Using the Discrete Hartley Transform*

Description

Performs a QR decomposition of 3-mode tensor using the discrete Hartley transform.

Usage

```
tQRdht(tnsr)
```

Arguments

`tnsr`	: a 3-mode tensor

Value

a Tensor-class object

If QR decomposition is performed on a n x n x k tensor, the components in the returned value are:

Q: An orthogonal tensor (n x n x k).

R: An upper triangular tensor (n x n x k)

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

Examples

```
T <- rand_tensor(modes=c(2,2,4))
print(tQRdht(T))
```

tQRdst *Tensor QR Decomposition Using the Discrete Sine Transform*

Description

Performs a QR decomposition of 3-mode tensor using the discrete Sine transform.

Usage

```
tQRdst(tnsr)
```

Arguments

`tnsr` : a 3-mode tensor

Value

a Tensor-class object

If QR decomposition is performed on a n x n x k tensor, the components in the returned value are:

Q: An orthogonal tensor (n x n x k).

R: An upper triangular tensor (n x n x k)

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

Examples

```
T <- rand_tensor(modes=c(2,2,4))
print(tQRdst(T))
```

---

tQRdwht *Tensor QR Decomposition Using the Discrete Walsh-Hadamard Transform*

Description

Performs a QR decomposition of 3-mode tensor using the discrete Walsh-Hadamard transform.

Usage

```
tQRdwht(tnsr)
```

Arguments

`tnsr` : a 3-mode tensor

Value

a Tensor-class object

If QR decomposition is performed on a n x n x k tensor, the components in the returned value are:

Q: An orthogonal tensor (n x n x k).

R: An upper triangular tensor (n x n x k)

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

Examples

```
T <- rand_tensor(modes=c(2,2,4))
print(tQRdwht(T))
```

---

tQRdwt   *Tensor QR Decomposition Using the Discrete Wavelet Transform*

Description

Performs a QR decomposition of 3-mode tensor using the discrete wavelet transform.

Usage

```
tQRdwt(tnsr)
```

Arguments

`tnsr`	: a 3-mode tensor

Value

a Tensor-class object

If QR decomposition is performed on a n x n x k tensor, the components in the returned value are:

Q: An orthogonal tensor (n x n x k).

R: An upper triangular tensor (n x n x k)

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

Examples

```
T <- rand_tensor(modes=c(2,2,4))
print(tQRdwt(T))
```

tQRfft — *Tensor QR Decomposition Using the Discrete Fourier Transform*

Description

Performs a QR decomposition of 3-mode tensor using the discrete Fourier transform.

Usage

```
tQRfft(tnsr)
```

Arguments

`tnsr`	: a 3-mode tensor

Value

a Tensor-class object

If QR decomposition is performed on a n x n x k tensor, the components in the returned value are:

Q: An orthogonal tensor (n x n x k).

R: An upper triangular tensor (n x n x k)

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

Examples

```
T <- rand_tensor(modes=c(2,2,4))
print(tQRfft(T))
```

tSVD — *Tensor Singular Value Decomposition Using Any Discrete Transform*

Description

Performs a tensor singular value decomposition on any 3-mode tensor using any discrete transform.

Usage

```
tSVD(tnsr,tform)
```

Arguments

`tnsr`	: a 3-mode tensor
`tform`	: Any discrete transform. Supported transforms are: fft: Fast Fourier Transform dwt: Discrete Wavelet Transform (Haar Wavelet) dct: Discrete Cosine transform dst: Discrete Sine transform dht: Discrete Hadley transform dwht: Discrete Walsh-Hadamard transform

Value

a Tensor-class object

If the SVD is performed on a m x n x k tensor, the components in the returned value are:

U: The left singular value tensor object (m x m x k)

V: The right singular value tensor object (n x n x k)

S: A diagonal tensor (m x n x k)

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

References

Kernfeld, E., Kilmer, M., & Aeron, S. (2015). Tensor-tensor products with invertible linear transforms. Linear Algebra and its Applications, 485, 545-570.

M. E. Kilmer, C. D. Martin, and L. Perrone, “A third-order generalization of the matrix svd as a product of third-order tensors,” Tufts University, Department of Computer Science, Tech. Rep. TR-2008-4, 2008

K. Braman, "Third-order tensors as linear operators on a space of matrices", Linear Algebra and its Applications, vol. 433, no. 7, pp. 1241-1253, 2010.

Examples

```
T <- rand_tensor(modes=c(2,3,4))
print(tSVD(T,"dst"))
```

`tSVDdct` *Tensor Singular Value Decomposition Using the Discrete Cosine Transform*

Description

Performs a tensor singular value decomposition on any 3-mode tensor using the discrete cosine transform.

Usage

```
tSVDdct(tnsr)
```

Arguments

`tnsr`	: a 3-mode tensor

Value

a Tensor-class object

If the SVD is performed on a m x n x k tensor, the components in the returned value are:

U: The left singular value tensor object (m x m x k)

V: The right singular value tensor object (n x n x k)

S: A diagonal tensor (m x n x k)

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

Examples

```
T <- rand_tensor(modes=c(2,3,4))
print(tSVDdct(T))
```

`tSVDdht` *Tensor Singular Value Decomposition Using the Discrete Harley Transform*

Description

Performs a tensor singular value decomposition on any 3-mode tensor using the discrete Harley transform.

Usage

```
tSVDdht(tnsr)
```

Arguments

`tnsr`	: a 3-mode tensor

Value

a Tensor-class object

If the SVD is performed on a m x n x k tensor, the components in the returned value are:

U: The left singular value tensor object (m x m x k)

V: The right singular value tensor object (n x n x k)

S: A diagonal tensor (m x n x k)

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

Examples

```
T <- rand_tensor(modes=c(2,3,4))
print(tSVDdht(T))
```

`tSVDdst` *Tensor Singular Value Decomposition Using the Discrete Sine Transform*

Description

Performs a tensor singular value decomposition on any 3-mode tensor using the discrete Sine transform.

Usage

```
tSVDdst(tnsr)
```

Arguments

`tnsr`	: a 3-mode tensor

Value

a Tensor-class object

If the SVD is performed on a m x n x k tensor, the components in the returned value are:

U: The left singular value tensor object (m x m x k)

V: The right singular value tensor object (n x n x k)

S: A diagonal tensor (m x n x k)

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

Examples

```
T <- rand_tensor(modes=c(2,3,4))
print(tSVDdst(T))
```

`tSVDdwht` *Tensor Singular Value Decomposition Using the Discrete Walsh-Hadamard Transform*

Description

Performs a tensor singular value decomposition on any 3-mode tensor using the discrete Walsh-Hadamard transform.

Usage

```
tSVDdwht(tnsr)
```

Arguments

`tnsr`	: a 3-mode tensor

Value

a Tensor-class object

If the SVD is performed on a m x n x k tensor, the components in the returned value are:

U: The left singular value tensor object (m x m x k)

V: The right singular value tensor object (n x n x k)

S: A diagonal tensor (m x n x k)

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

Examples

```
T <- rand_tensor(modes=c(2,3,4))
print(tSVDdwht(T))
```

`tSVDdwt` *Tensor Singular Value Decomposition Using the Discrete Wavelet Transform*

Description

Performs a tensor singular value decomposition on any 3-mode tensor using the discrete wavelet transform (Haar Wavelet).

Usage

```
tSVDdwt(tnsr)
```

Arguments

`tnsr`	: a 3-mode tensor

Value

a Tensor-class object

If the SVD is performed on a m x n x k tensor, the components in the returned value are:

U: The left singular value tensor object (m x m x k)

V: The right singular value tensor object (n x n x k)

S: A diagonal tensor (m x n x k)

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

Examples

```
T <- rand_tensor(modes=c(2,3,4))
print(tSVDdwt(T))
```

`tSVDfft` *Tensor Singular Value Decomposition Using the Discrete Fourier Transform*

Description

Performs a tensor singular value decomposition on any 3-mode tensor using the discrete Fourier transform.

Usage

```
tSVDfft(tnsr)
```

Arguments

`tnsr`	: a 3-mode tensor

Value

a Tensor-class object

If the SVD is performed on a m x n x k tensor, the components in the returned value are:

U: The left singular value tensor object (m x m x k)

V: The right singular value tensor object (n x n x k)

S: A diagonal tensor (m x n x k)

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

Examples

```
T <- rand_tensor(modes=c(2,3,4))
print(tSVDfft(T))
```

`t_tpose` *Transpose 3-mode Tensor*

Description

Performs the transpose of a symmetric 3-mode tensor using any discrete transform.

Usage

```
t_tpose(tnsr,tform)
```

Arguments

`tnsr`	: a 3-mode tensor
`tform`	: Any discrete transform. Supported transforms are: fft: Fast Fourier Transform dwt: Discrete Wavelet Transform (Haar Wavelet) dct: Discrete Cosine transform dst: Discrete Sine transform dht: Discrete Hadley transform dwht: Discrete Walsh-Hadamard transform

Value

a Tensor-class object

Author(s)

Kyle Caudle

Randy Hoover

Jackson Cates

Examples

```
T <- rand_tensor(modes=c(2,3,4))
print(t_tpose(T,"dct"))
```

Index

∗ **datasets**
  Mnist, 4
  raytrace, 7
∗ **misc**
  LU, 4

as.tensor, 3

LU, 4

Mnist, 4

polar, 5

QR, 6

rand_tensor, 6
raytrace, 7

t_tpose, 42
tDWT, 8
tEIG, 9
tEIGdct, 10
tEIGdht, 11
tEIGdst, 12
tEIGdwht, 12
tEIGdwt, 13
tEIGfft, 14
tIDWT, 15
tINV, 16
tINVdct, 17
tINVdht, 17
tINVdst, 18
tINVdwht, 19
tINVdwt, 19
tINVfft, 20
tLDA, 21
tLU, 22
tLUdct, 23
tLUdht, 24
tLUdst, 24
tLUdwht, 25
tLUdwt, 26
tLUfft, 26
tmean, 27
tmult, 28
tQR, 29
tQRdct, 30
tQRdht, 31
tQRdst, 31
tQRdwht, 32
tQRdwt, 33
tQRfft, 34
tSVD, 34
tSVDdct, 36
tSVDdht, 37
tSVDdst, 38
tSVDdwht, 39
tSVDdwt, 40
tSVDfft, 41