Package ‘rTensor2’

October 14, 2022

Type Package

Title MultiLinear Algebra

Version 0.1.0

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

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,rTensor, wavethresh, gsignal, Matrix, matrixcalc

License GPL-3

Encoding UTF-8

Suggests testthat (>= 3.1.4)

Config/testthat/edition 3

NeedsCompilation no

Repository CRAN

Date/Publication 2022-08-14 07:40:01 UTC

R topics documented:

`LU` *LU Decomposition of a Complex Matrix*

Description

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

Usage

```
LU(A)
```

Arguments

A	: an nxn matrix

Value

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

Author(s)

Kyle Caudle kyle.caudle@sdsmt.edu

Examples

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

---

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 kyle.caudle@sdsmt.edu

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 nxn matrix

Value

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

Author(s)

Kyle Caudle kyle.caudle@sdsmt.edu

Examples

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

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 kyle.caudle@sdsmt.edu

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

```
require(rTensor)
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
`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 $nxnxk$ tensor, the components in the returned value are:

P: A tensor of Eigenvectors ($nxnxk$)

D: An diagonal tensor of Eigenvalues ($nxnxk$)

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

```
require(rTensor)
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 $nxnxk$ tensor, the components in the returned value are:

P: A tensor of Eigenvectors ($nxnxk$)

D: An diagonal tensor of Eigenvalues ($nxnxk$)

Author(s)

Kyle Caudle kyle.caudle@sdsmt.edu

Examples

```
require(rTensor)
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 $nxnxk$ tensor, the components in the returned value are:

P: A tensor of Eigenvectors $(nxnxk)$

D: An diagonal tensor of Eigenvalues $(nxnxk)$

Author(s)

Kyle Caudle kyle.caudle@sdsmt.edu

Examples

```
require(rTensor)
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 $nxnxk$ tensor, the components in the returned value are:

P: A tensor of Eigenvectors $(nxnxk)$

D: An diagonal tensor of Eigenvalues $(nxnxk)$

Author(s)

Kyle Caudle kyle.caudle@sdsmt.edu

Examples

```
require(rTensor)
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 $nxnxk$ tensor, the components in the returned value are:

P: A tensor of Eigenvectors ($nxnxk$)

D: An diagonal tensor of Eigenvalues ($nxnxk$)

Author(s)

Kyle Caudle kyle.caudle@sdsmt.edu

Examples

```
require(rTensor)
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 $nxnxk$ tensor, the components in the returned value are:

P: A tensor of Eigenvectors ($nxnxk$)

D: An diagonal tensor of Eigenvalues ($nxnxk$)

Author(s)

Kyle Caudle kyle.caudle@sdsmt.edu

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

```
require(rTensor)
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 $nxnxk$ tensor, the components in the returned value are:

P: A tensor of Eigenvectors ($nxnxk$)

D: An diagonal tensor of Eigenvalues ($nxnxk$)

Author(s)

Kyle Caudle kyle.caudle@sdsmt.edu

Examples

```
require(rTensor)
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 kyle.caudle@sdsmt.edu

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

```
require(rTensor)
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 kyle.caudle@sdsmt.edu

Examples

```
require(rTensor)
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 kyle.caudle@sdsmt.edu

Examples

```
require(rTensor)
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 kyle.caudle@sdsmt.edu

Examples

```
require(rTensor)
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 kyle.caudle@sdsmt.edu

Examples

```
require(rTensor)
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 kyle.caudle@sdsmt.edu

Examples

```
require(rTensor)
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 kyle.caudle@sdsmt.edu

Examples

```
require(rTensor)
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 kyle.caudle@sdsmt.edu

Examples

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

---

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 $nxnxk$ 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

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

```
require(rTensor)
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

```
require(rTensor)
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

```
require(rTensor)
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

```
require(rTensor)
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

```
require(rTensor)
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

```
require(rTensor)
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

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

---

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

"transform" (e.g. "dst")

Author(s)

Kyle Caudle kyle.caudle@sdsmt.edu

Examples

```
require(rTensor)
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 $nxnxk$ tensor, the components in the returned value are:

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

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

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

```
require(rTensor)
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 $nxnxk$ tensor, the components in the returned value are:

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

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

Author(s)

Kyle Caudle kyle.caudle@sdsmt.edu

Examples

```
require(rTensor)
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 $nxnxk$ tensor, the components in the returned value are:

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

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

Author(s)

Kyle Caudle kyle.caudle@sdsmt.edu

Examples

```
require(rTensor)
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 $nxnxk$ tensor, the components in the returned value are:

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

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

Author(s)

Kyle Caudle kyle.caudle@sdsmt.edu

Examples

```
require(rTensor)
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 $nxnxk$ tensor, the components in the returned value are:

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

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

Author(s)

Kyle Caudle kyle.caudle@sdsmt.edu

Examples

```
require(rTensor)
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 Fourier transform.

Usage

```
tQRdwt(tnsr)
```

Arguments

`tnsr`	: a 3-mode tensor

Value

a Tensor-class object

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

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

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

Author(s)

Kyle Caudle kyle.caudle@sdsmt.edu

Examples

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

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 $nxnxk$ tensor, the components in the returned value are:

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

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

Author(s)

Kyle Caudle kyle.caudle@sdsmt.edu

Examples

```
require(rTensor)
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 $mxnxk$ tensor, the components in the returned value are:

U: The left singular value tensor object ($mxmxk$)

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

S: A diagonal tensor ($mxnxk$)

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

```
require(rTensor)
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 $mxnxk$ tensor, the components in the returned value are:

U: The left singular value tensor object ($mxmxk$)

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

S: A diagonal tensor ($mxnxk$)

Author(s)

Kyle Caudle kyle.caudle@sdsmt.edu

Examples

```
require(rTensor)
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 $mxnxk$ tensor, the components in the returned value are:

U: The left singular value tensor object ($mxmxk$)

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

S: A diagonal tensor ($mxnxk$)

Author(s)

Kyle Caudle kyle.caudle@sdsmt.edu

Examples

```
require(rTensor)
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 $mxnxk$ tensor, the components in the returned value are:

U: The left singular value tensor object $(mxmxk)$

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

S: A diagonal tensor $(mxnxk)$

Author(s)

Kyle Caudle kyle.caudle@sdsmt.edu

Examples

```
require(rTensor)
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 $mxnxk$ tensor, the components in the returned value are:

U: The left singular value tensor object ($mxmxk$)

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

S: A diagonal tensor ($mxnxk$)

Author(s)

Kyle Caudle kyle.caudle@sdsmt.edu

Examples

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

---

tSVDdwt *Tensor Singular Value Decomposition Using the Discrete Fourier 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 $mxnxk$ tensor, the components in the returned value are:

U: The left singular value tensor object ($mxmxk$)

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

S: A diagonal tensor ($mxnxk$)

Author(s)

Kyle Caudle kyle.caudle@sdsmt.edu

Examples

```
require(rTensor)
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 $mxnxk$ tensor, the components in the returned value are:

U: The left singular value tensor object ($mxmxk$)

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

S: A diagonal tensor ($mxnxk$)

Author(s)

Kyle Caudle kyle.caudle@sdsmt.edu

Examples

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

`tsym` *Transpose of a Symmetric 3-mode Tensor*

Description

Performs the transpose of a symmetric 3-mode tensor.

Usage

```
tsym(tnsr)
```

Arguments

`tnsr`	: a 3-mode tensor

Value

a Tensor-class object

Author(s)

Kyle Caudle kyle.caudle@sdsmt.edu

Examples

```
require(rTensor)
T <- rand_tensor(modes=c(2,3,4))
print(tsym(T))
```

Index