Package ‘MultiStatM’

January 31, 2024

Type Package

Title Multivariate Statistical Methods

Version 1.2.1

Imports arrangements, Matrix, MASS, stats, mvtnorm

Description Algorithms to build set partitions and commutator matrices and their use in the construction of multivariate d-
Hermite polynomials; estimation and derivation of theoretical vector moments and vector cumulants of multivariate distributions; conversion formulae for multivariate moments and cumulants. Applications to
estimation and derivation of multivariate measures of skewness and kurtosis; estimation and derivation of asymptotic
covariances for d-variate Hermite polynomials, multivariate moments and cumulants and measures of skewness and kurtosis.
The formulae implemented are discussed in Terdik (2021, ISBN:9783030813925), ``Multivariate Statistical Methods''.

License GPL-3

Encoding UTF-8

RoxygenNote 7.3.0

Depends R (>= 2.10)

Suggests rmarkdown, knitr

VignetteBuilder knitr

NeedsCompilation no

Author Gyorgy Terdik [aut],
Emanuele Taufer [aut, cre]

Maintainer Emanuele Taufer <emanuele.taufer@unitn.it>

Repository CRAN

Date/Publication 2024-01-31 08:30:02 UTC

R topics documented:

`conv_Cum2Mom` *Convert cumulants to moments (univariate)*

Description

Obtains a vector of univariate moments from a vector of univariate cumulants

Usage

```
conv_Cum2Mom(cum_x)
```

Arguments

`cum_x`	the r-vector of cumulants starting from the first - the mean - and arriving to the r-th order cumulant

Value

`mu_x` the vector of univariate moments

References

Gy. Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021. Section 3.4 formula 3.23

See Also

Other Moments and cumulants: `conv_Cum2MomMulti()`, `conv_Mom2CumMulti()`, `conv_Mom2Cum()`

Examples

```
cum_x<- c(1,2,3,4)
conv_Cum2Mom(cum_x)
```

conv_Cum2MomMulti *Convert T-cumulants to T-moments (multivariate)*

Description

Obtains a vector of d-variate moments from a vector of d-variate cumulants

Usage

```
conv_Cum2MomMulti(cum)
```

Arguments

cum	the list of r d-variate cumulants in vector form starting from the first cumulant - the vector of means - and arriving to the r-th order d-variate cumulant in vector form

Value

Mom the list of r vectors of d-variate moments

References

Gy. Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021. Section 3.4 formula 3.40

See Also

Other Moments and cumulants: conv_Cum2Mom(), conv_Mom2CumMulti(), conv_Mom2Cum()

Examples

```
#cum contains the  T-vector cumulants up to the fifth order of the bivariate
#standard normal distribution
cum<-list(c(0,0),c(1,0,0,1),c(0,0,0,0,0,0,0,0),c(0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0),
c(rep(0,32)))
conv_Cum2MomMulti(cum)
```

conv_Mom2Cum *Convert moments to cumulants (univariate)*

Description

Obtains a vector of univariate cumulants from a vector of univariate moments

Usage

```
conv_Mom2Cum(mu_x)
```

Arguments

mu_x	the r-vector of moments starting from the first moment - the mean - and arriving to the r-th order moment

Value

cum_x the vector of univariate cumulants

References

Gy. Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021. Section 3.4 formula 3.20

See Also

Other Moments and cumulants: conv_Cum2MomMulti(), conv_Cum2Mom(), conv_Mom2CumMulti()

Examples

```
mu_x<- c(1,2,3,4)
conv_Mom2Cum(mu_x)
```

conv_Mom2CumMulti *Convert T-moments to T-cumulants (multivariate)*

Description

Obtains a vector of d-variate cumulants from a vector of d-variate moments

Usage

```
conv_Mom2CumMulti(mu)
```

Arguments

mu — the list of r d-variate moments in vector form starting from the first moment - the vector of means - and arriving to the r-th order d-variate moment in vector form

Value

Cum the list of n vectors of d-variate cumulants

References

Gy. Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021. Section 3.4 formula 3.29

See Also

Other Moments and cumulants: conv_Cum2MomMulti(), conv_Cum2Mom(), conv_Mom2Cum()

Examples

```
#Mu contains the  T-vector moments up to the fifth order of the bivariate
#standard normal distribution
mu<-list(c(0,0),c(1,0,0,1),c(0,0,0,0,0,0,0,0),c(3,0,0,1,0,1,1,0,0,1,1,0,1,0,0,3),
c(rep(0,32)))
conv_Mom2CumMulti(mu)
```

---

conv_Stand_Multi — *Standardize multivariate data*

Description

For data formed by d-variate vectors x with sample covariance S and sample mean M, it computes the values $z = S^{-1/2}(x - M)$

Usage

```
conv_Stand_Multi(x)
```

Arguments

x — a multivariate data matrix, sample size is the number of rows

Value

a matrix of multivariate data with null mean vector and identity sample covariance matrix

Examples

```
x<-MASS::mvrnorm(1000,c(0,0,1,3),diag(4))
z<-conv_Stand_Multi(x)
mu_z<- apply(z,2,mean)
cov_z<- cov(z)
```

---

distr_CFUSN_MomCum_Th *Moments and cumulants CFUSN*

Description

Provides the theoretical cumulants of the multivariate Canonical Fundamental Skew Normal distribution

Usage

```
distr_CFUSN_MomCum_Th(r, d, p, Delta, nMu = FALSE)
```

Arguments

r	The highest cumulant order
d	The multivariate dimension and number of rows of the skewness matrix Delta
p	The number of cols of the skewness matrix Delta
Delta	The skewness matrix
nMu	If set to TRUE, the list of the first r d-variate moments is provided

Value

The list of theoretical cumulants in vector form

References

Gy.Terdik, Multivariate statistical methods - Going beyond the linear, Springer 2021, Lemma 5.3 p.251

See Also

Other Theoretical Moments and Cumulants: distr_SkewNorm_EVSK_Th(), distr_SkewNorm_MomCum_Th(), distr_UniAbs_EVSK_Th(), distr_Uni_EVSK_Th(), distr_Uni_MomCum_Th(), distr_ZabsM_MomCum_Th(), distr_Zabs_MomCum_Th()

Other Multivariate distributions: distr_CFUSN_Rand(), distr_CFUSSD_Rand(), distr_SkewNorm_EVSK_Th(), distr_SkewNorm_MomCum_Th(), distr_SkewNorm_Rand(), distr_UniAbs_EVSK_Th(), distr_Uni_EVSK_Th(), distr_Uni_MomCum_Th(), distr_Uni_Rand(), distr_ZabsM_MomCum_Th(), distr_Zabs_MomCum_Th()

Examples

```
r <- 4; d <- 2; p <- 3
Lamd <-  matrix(sample(1:50-25, d*p), nrow=d)
ieg<- eigen(diag(p)+t(Lamd)%*%Lamd)
V <- ieg$vectors
Delta <-Lamd %*% V %*% diag(1/sqrt(ieg$values)) %*% t(V)
MomCum_CFUSN <- distr_CFUSN_MomCum_Th (r,d,p,Delta)
```

---

distr_CFUSN_Rand *Random multivariate CFUSN*

Description

Generate random d-vectors from the multivariate Canonical Fundamental Skew-Normal (CFUSN) distribution

Usage

```
distr_CFUSN_Rand(n, Delta)
```

Arguments

`n`	The number of variates to be generated
`Delta`	Correlation matrix, the skewness matrix Delta

Value

A random matrix $n \times d$

References

Gy.Terdik, Multivariate statistical methods - Going beyond the linear, Springer 2021 (5.5) p.247

S. R. Jammalamadaka, E. Taufer, Gy. Terdik. On multivariate skewness and kurtosis. Sankhya A, 83(2), 607-644.

See Also

Other Random generation: `distr_CFUSSD_Rand()`, `distr_SkewNorm_Rand()`, `distr_Uni_Rand()`

Other Multivariate distributions: `distr_CFUSN_MomCum_Th()`, `distr_CFUSSD_Rand()`, `distr_SkewNorm_EVSK_Th()`, `distr_SkewNorm_MomCum_Th()`, `distr_SkewNorm_Rand()`, `distr_UniAbs_EVSK_Th()`, `distr_Uni_EVSK_Th()`, `distr_Uni_MomCum_Th()`, `distr_Uni_Rand()`, `distr_ZabsM_MomCum_Th()`, `distr_Zabs_MomCum_Th()`

Examples

```
d <- 2; p <- 3
Lamd <-  matrix(sample(1:50-25, d*p), nrow=d)
ieg<- eigen(diag(p)+t(Lamd)%*%Lamd)
V <- ieg$vectors
Delta <-Lamd %*% V %*% diag(1/sqrt(ieg$values)) %*% t(V)
x<-distr_CFUSN_Rand(20,Delta)
```

`distr_CFUSSD_Rand` *Random multivariate CFUSSD*

Description

Generate random d-vectors from the multivariate Canonical Fundamental Skew-Spherical distribution (CFUSSD) with Gamma generator

Usage

```
distr_CFUSSD_Rand(n, d, p, a, b, Delta)
```

Arguments

`n`	sample size
`d`	dimension
`p`	dimension of the first term of (5.5)
`a`	shape parameter of the Gamma generator
`b`	scale parameter of the Gamma generator
`Delta`	skewness matrix

Value

A matrix of $n \times d$ random numbers

References

Gy.Terdik, Multivariate statistical methods - Going beyond the linear, Springer 2021, (5.36) p. 266, (see p.247 for Delta)

See Also

Other Random generation: `distr_CFUSN_Rand()`, `distr_SkewNorm_Rand()`, `distr_Uni_Rand()`

Other Multivariate distributions: `distr_CFUSN_MomCum_Th()`, `distr_CFUSN_Rand()`, `distr_SkewNorm_EVSK_Th()`, `distr_SkewNorm_MomCum_Th()`, `distr_SkewNorm_Rand()`, `distr_UniAbs_EVSK_Th()`, `distr_Uni_EVSK_Th()`, `distr_Uni_MomCum_Th()`, `distr_Uni_Rand()`, `distr_ZabsM_MomCum_Th()`, `distr_Zabs_MomCum_Th()`

Examples

```
n <- 10^3; d <- 2; p <- 3 ; a <- 1; b <- 1
Lamd <-  matrix(sample(1:50-25, d*p), nrow=d)
ieg<- eigen(diag(p)+t(Lamd)%*%Lamd)
V <- ieg$vectors
Delta <-Lamd %*% V %*% diag(1/sqrt(ieg$values)) %*% t(V)
distr_CFUSSD_Rand(20,d,p,1,1,Delta)
```

---

distr_SkewNorm_EVSK_Th

EVSK multivariate Skew Normal

Description

Computes the theoretical values of the mean vector, covariance, skewness vector, total skenwness, kurtosis vector and total kurtosis for the multivariate Skew Normal distribution

Usage

```
distr_SkewNorm_EVSK_Th(omega, alpha)
```

Arguments

omega	A $d \times d$ correlation matrix
alpha	shape parameter d-vector

Value

A list of theoretical values for the mean vector, covariance, skewness vector, total skenwness, kurtosis vector and total kurtosis

References

Gy.Terdik, Multivariate statistical methods - Going beyond the linear, Springer 2021 (5.5) p.247

S. R. Jammalamadaka, E. Taufer, Gy. Terdik. On multivariate skewness and kurtosis. Sankhya A, 83(2), 607-644.

See Also

Other Theoretical Moments and Cumulants: distr_CFUSN_MomCum_Th(), distr_SkewNorm_MomCum_Th(), distr_UniAbs_EVSK_Th(), distr_Uni_EVSK_Th(), distr_Uni_MomCum_Th(), distr_ZabsM_MomCum_Th(), distr_Zabs_MomCum_Th()

Other Multivariate distributions: distr_CFUSN_MomCum_Th(), distr_CFUSN_Rand(), distr_CFUSSD_Rand(), distr_SkewNorm_MomCum_Th(), distr_SkewNorm_Rand(), distr_UniAbs_EVSK_Th(), distr_Uni_EVSK_Th(), distr_Uni_MomCum_Th(), distr_Uni_Rand(), distr_ZabsM_MomCum_Th(), distr_Zabs_MomCum_Th()

Examples

```
alpha<-c(10,5,0)
omega<-diag(3)
distr_SkewNorm_EVSK_Th(omega,alpha)
```

---

distr_SkewNorm_MomCum_Th

Moments and cumulants d-variate Skew Normal

Description

Computes the theoretical values of moments and cumulants up to the r-th order. Warning: if nMu = TRUE it can be very slow

Usage

```
distr_SkewNorm_MomCum_Th(r = 4, omega, alpha, nMu = FALSE)
```

Arguments

r	the highest moment and cumulant order
omega	A $d \times d$ correlation matrix
alpha	shape parameter d-vector
nMu	if it is TRUE then moments are calculated as well

Value

A list of theoretical moments and cumulants

References

Gy.Terdik, Multivariate statistical methods - Going beyond the linear, Springer 2021 (5.5) p.247, Lemma 5.1 p. 246

S. R. Jammalamadaka, E. Taufer, Gy. Terdik. On multivariate skewness and kurtosis. Sankhya A, 83(2), 607-644.

See Also

Other Theoretical Moments and Cumulants: distr_CFUSN_MomCum_Th(), distr_SkewNorm_EVSK_Th(), distr_UniAbs_EVSK_Th(), distr_Uni_EVSK_Th(), distr_Uni_MomCum_Th(), distr_ZabsM_MomCum_Th(), distr_Zabs_MomCum_Th()

Other Multivariate distributions: distr_CFUSN_MomCum_Th(), distr_CFUSN_Rand(), distr_CFUSSD_Rand(), distr_SkewNorm_EVSK_Th(), distr_SkewNorm_Rand(), distr_UniAbs_EVSK_Th(), distr_Uni_EVSK_Th(), distr_Uni_MomCum_Th(), distr_Uni_Rand(), distr_ZabsM_MomCum_Th(), distr_Zabs_MomCum_Th()

Examples

```
alpha<-c(10,5,0)
omega<-diag(3)
distr_SkewNorm_MomCum_Th(r=4,omega,alpha)
```

---

distr_SkewNorm_Rand *Random Multivariate Skew Normal*

Description

Generate random d-vectors from the multivariate Skew Normal distribution

Usage

```
distr_SkewNorm_Rand(n, omega, alpha)
```

Arguments

n	sample size
omega	correlation matrix with d dimension
alpha	shape parameter vector of dimension d

Value

A random matrix $n \times d$

References

Azzalini, A. with the collaboration of Capitanio, A. (2014). The Skew-Normal and Related Families. Cambridge University Press, IMS Monographs series.

Gy.H.Terdik, Multivariate statistical methods - Going beyond the linear, Springer 2021, Section 5.1.2

See Also

Other Random generation: distr_CFUSN_Rand(), distr_CFUSSD_Rand(), distr_Uni_Rand()

Other Multivariate distributions: distr_CFUSN_MomCum_Th(), distr_CFUSN_Rand(), distr_CFUSSD_Rand(), distr_SkewNorm_EVSK_Th(), distr_SkewNorm_MomCum_Th(), distr_UniAbs_EVSK_Th(), distr_Uni_EVSK_Th(), distr_Uni_MomCum_Th(), distr_Uni_Rand(), distr_ZabsM_MomCum_Th(), distr_Zabs_MomCum_Th()

Examples

```
alpha<-c(10,5,0)
omega<-diag(3)
x<-distr_SkewNorm_Rand(20,omega,alpha)
```

distr_UniAbs_EVSK_Th *Moments of the modulus of the Uniform distribution on the sphere*

Description

Moments (up to the 4th order) of the modulus of the d-variate Uniform distribution on the sphere on (d-1)

Usage

```
distr_UniAbs_EVSK_Th(d, nCum = FALSE)
```

Arguments

d	vector-dimension
nCum	if it is TRUE then cumulants, skewness an kurtosis are calculated

Value

The list of the first four moments in vector form

References

Gy.Terdik, Multivariate statistical methods - Going beyond the linear, Springer 2021, Lemma 5.12 p.298

See Also

Other Theoretical Moments and Cumulants: distr_CFUSN_MomCum_Th(), distr_SkewNorm_EVSK_Th(), distr_SkewNorm_MomCum_Th(), distr_Uni_EVSK_Th(), distr_Uni_MomCum_Th(), distr_ZabsM_MomCum_Th(), distr_Zabs_MomCum_Th()

Other Multivariate distributions: distr_CFUSN_MomCum_Th(), distr_CFUSN_Rand(), distr_CFUSSD_Rand(), distr_SkewNorm_EVSK_Th(), distr_SkewNorm_MomCum_Th(), distr_SkewNorm_Rand(), distr_Uni_EVSK_Th(), distr_Uni_MomCum_Th(), distr_Uni_Rand(), distr_ZabsM_MomCum_Th(), distr_Zabs_MomCum_Th()

distr_Uni_EVSK_Th *EVSK Uniform on the sphere*

Description

Computes the theoretical values of the mean vector, covariance, skewness vector, total skenwness, kurtosis vector and total kurtosis for the Uniform distribution on the sphere. Note that Skewness is ZERO

Usage

```
distr_Uni_EVSK_Th(d, nCum = TRUE)
```

Arguments

d	dimensions
nCum	if it is TRUE then cumulants, skewness an kurtosis are calculated

Value

The list with mean vector, covariance, skewness vector, total skenwness, kurtosis vector and total kurtosis

References

Gy.Terdik, Multivariate statistical methods - Going beyond the linear, Springer 2021 Proposition 5.3 p.297

S. R. Jammalamadaka, E. Taufer, Gy. Terdik. On multivariate skewness and kurtosis. Sankhya A, 83(2), 607-644.

See Also

Other Theoretical Moments and Cumulants: distr_CFUSN_MomCum_Th(), distr_SkewNorm_EVSK_Th(), distr_SkewNorm_MomCum_Th(), distr_UniAbs_EVSK_Th(), distr_Uni_MomCum_Th(), distr_ZabsM_MomCum_Th(), distr_Zabs_MomCum_Th()

Other Multivariate distributions: distr_CFUSN_MomCum_Th(), distr_CFUSN_Rand(), distr_CFUSSD_Rand(), distr_SkewNorm_EVSK_Th(), distr_SkewNorm_MomCum_Th(), distr_SkewNorm_Rand(), distr_UniAbs_EVSK_Th(), distr_Uni_MomCum_Th(), distr_Uni_Rand(), distr_ZabsM_MomCum_Th(), distr_Zabs_MomCum_Th()

---

distr_Uni_MomCum_Th *Moments and cumulants Uniform Distribution on the Sphere*

Description

By default, only moments are provided

Usage

```
distr_Uni_MomCum_Th(r, d, nCum = FALSE)
```

Arguments

r	highest order of moments and cumulants
d	dimension
nCum	if it is TRUE then cumulants are calculated

Value

The list of moments and cumulants in vector form

References

Gy.Terdik, Multivariate statistical methods - Going beyond the linear, Springer 2021 Proposition 5.3 p.297

See Also

Other Theoretical Moments and Cumulants: distr_CFUSN_MomCum_Th(), distr_SkewNorm_EVSK_Th(), distr_SkewNorm_MomCum_Th(), distr_UniAbs_EVSK_Th(), distr_Uni_EVSK_Th(), distr_ZabsM_MomCum_Th(), distr_Zabs_MomCum_Th()

Other Multivariate distributions: distr_CFUSN_MomCum_Th(), distr_CFUSN_Rand(), distr_CFUSSD_Rand(), distr_SkewNorm_EVSK_Th(), distr_SkewNorm_MomCum_Th(), distr_SkewNorm_Rand(), distr_UniAbs_EVSK_Th(), distr_Uni_EVSK_Th(), distr_Uni_Rand(), distr_ZabsM_MomCum_Th(), distr_Zabs_MomCum_Th()

Examples

```
# The first four moments for d=3
distr_Uni_MomCum_Th(4,3,nCum=0)
# The first four moments and cumulants for d=3
distr_Uni_MomCum_Th(4,3,nCum=4)
```

---

distr_Uni_Rand *Random Uniform on the sphere*

Description

Generate random d-vectors from the Uniform distribution on the sphere

Usage

```
distr_Uni_Rand(n, d)
```

Arguments

n	sample size
d	dimension

Value

A random matrix $n \times d$

References

Gy.Terdik, Multivariate statistical methods - Going beyond the linear, Springer 2021

S. R. Jammalamadaka, E. Taufer, Gy. Terdik. On multivariate skewness and kurtosis. Sankhya A, 83(2), 607-644.

See Also

Other Random generation: distr_CFUSN_Rand(), distr_CFUSSD_Rand(), distr_SkewNorm_Rand()

Other Multivariate distributions: distr_CFUSN_MomCum_Th(), distr_CFUSN_Rand(), distr_CFUSSD_Rand(), distr_SkewNorm_EVSK_Th(), distr_SkewNorm_MomCum_Th(), distr_SkewNorm_Rand(), distr_UniAbs_EVSK_Th(), distr_Uni_EVSK_Th(), distr_Uni_MomCum_Th(), distr_ZabsM_MomCum_Th(), distr_Zabs_MomCum_Th()

---

distr_ZabsM_MomCum_Th *Moments and cumulants multivariate central folded Normal distribution*

Description

Provides the theoretical moments and cumulants of the multivariate central Folded Normal distribution. By default only cumulants are provided.

Usage

```
distr_ZabsM_MomCum_Th(r, d, nMu = FALSE)
```

Arguments

r	The highest cumulant (moment) order
d	dimension
nMu	if True then moments are calculated as well.

Value

The list of cumulants (and moments) in vector form.

References

Gy.Terdik, Multivariate statistical methods - Going beyond the linear, Springer 2021, Lemma 5.2 p. 249

See Also

Other Theoretical Moments and Cumulants: distr_CFUSN_MomCum_Th(), distr_SkewNorm_EVSK_Th(), distr_SkewNorm_MomCum_Th(), distr_UniAbs_EVSK_Th(), distr_Uni_EVSK_Th(), distr_Uni_MomCum_Th(), distr_Zabs_MomCum_Th()

Other Multivariate distributions: distr_CFUSN_MomCum_Th(), distr_CFUSN_Rand(), distr_CFUSSD_Rand(), distr_SkewNorm_EVSK_Th(), distr_SkewNorm_MomCum_Th(), distr_SkewNorm_Rand(), distr_UniAbs_EVSK_Th(), distr_Uni_EVSK_Th(), distr_Uni_MomCum_Th(), distr_Uni_Rand(), distr_Zabs_MomCum_Th()

distr_Zabs_MomCum_Th *Moments and cumulants Central folded Normal distribution*

Description

Provides the theoretical moments and cumulants of the univariate Central Folded Normal distribution. By default only moments are provided.

Usage

```
distr_Zabs_MomCum_Th(r, nCum = FALSE)
```

Arguments

r	The highest moment (cumulant) order
nCum	if it is TRUE then cumulants are calculated

Value

The list of moments and cumulants

References

Gy.Terdik, Multivariate statistical methods - Going beyond the linear, Springer 2021, Proposition 5.1 p.242 and formula: p. 301

See Also

Other Multivariate distributions: distr_CFUSN_MomCum_Th(), distr_CFUSN_Rand(), distr_CFUSSD_Rand(), distr_SkewNorm_EVSK_Th(), distr_SkewNorm_MomCum_Th(), distr_SkewNorm_Rand(), distr_UniAbs_EVSK_Th(), distr_Uni_EVSK_Th(), distr_Uni_MomCum_Th(), distr_Uni_Rand(), distr_ZabsM_MomCum_Th()

Other Theoretical Moments and Cumulants: distr_CFUSN_MomCum_Th(), distr_SkewNorm_EVSK_Th(), distr_SkewNorm_MomCum_Th(), distr_UniAbs_EVSK_Th(), distr_Uni_EVSK_Th(), distr_Uni_MomCum_Th(), distr_ZabsM_MomCum_Th()

Examples

```
# The first three moments
distr_Zabs_MomCum_Th(3, nCum = FALSE)
# The first three moments and cumulants
distr_Zabs_MomCum_Th(3, nCum = TRUE)
```

Esti_EVSK — *Estimation of multivariate Mean, Variance, T-Skewness and T-Kurtosis vectors*

Description

Provides estimates of mean, variance, skewness and kurtosis vectors for d-variate data

Usage

```
Esti_EVSK(X)
```

Arguments

X	d-variate data vector

Value

The list of the estimated mean, variance, skewness and kurtosis vectors

References

Gy.Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021, Sections 6.4.1 and 6.5.1

See Also

Other Estimation: Esti_Kurt_Variance_Th(), Esti_MMom_MCum(), Esti_Skew_Mardia(), Esti_Skew_Variance_Th()

Examples

```
x<- MASS::mvrnorm(100,rep(0,3), 3*diag(rep(1,3)))
EVSK<-Esti_EVSK(x)
names(EVSK)
EVSK$estSkew
```

Esti_Gram_Charlier — *Gram-Charlier approximation to a multivariate density*

Description

Provides the truncated Gram-Charlier approximation to a multivariate density. Approximation can be up to the first k=8 cumulants.

Usage

```
Esti_Gram_Charlier(X, k = 4, cum = NULL)
```

Arguments

X	A matrix of d-variate data
k	the order of the approximation, by default set to 4; (k must not be smaller than 3 or greater than 8)
cum	if NULL (default) the cumulant vector is estimated from X. If cum is provided no estimation of cumulants is performed.

Value

The vector of the Gram-Charlier density evaluated at X

References

Gy.Terdik, Multivariate statistical methods - Going beyond the linear, Springer 2021. Section 4.7.

Examples

```
# Gram-Charlier density approximation (k=4) of data generated from
# a bivariate skew-gaussian distribution
n<-50
alpha<-c(10,0)
omega<-diag(2)
X<-distr_SkewNorm_Rand(n,omega,alpha)
EC<-Esti_EVSK(X)
fy4<-Esti_Gram_Charlier(X[1:5,],cum=EC)
```

Esti_Hermite_Poly_HN_Multi

Estimate the N-th d-variate Hermite polynomial

Description

The vector x is standardized and the N-th d-variate polynomial is computed

Usage

```
Esti_Hermite_Poly_HN_Multi(x, N)
```

Arguments

x	a d-variate data vector
N	the order of the d-variate Hermite polynomial

Value

The vector of the N-th d-variate polynomial

Examples

```
x<-MASS::mvrnorm(100,rep(0,3),diag(3))
H3<-Esti_Hermite_Poly_HN_Multi(x,3)
```

`Esti_Kurt_Mardia` *Estimation of Mardia's Kurtosis Index*

Description

Estimation of Mardia's Kurtosis Index

Usage

```
Esti_Kurt_Mardia(x)
```

Arguments

`x`	A matrix of multivariate data

Value

`Mardia.Kurtosis` The kurtosis index

`p.value` The p-value under the Gaussian hypothesis

References

Gy.Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021. Example 6.1

See Also

Other Indexes: `Esti_Kurt_MRSz()`, `Esti_Kurt_Total()`, `Esti_Skew_MRSz()`, `Esti_Skew_Mardia()`

`Esti_Kurt_MRSz` *Estimation of Mori, Rohatgi, Szekely (CMRSz's) kurtosis matrix*

Description

Estimation of Mori, Rohatgi, Szekely (CMRSz's) kurtosis matrix

Usage

```
Esti_Kurt_MRSz(x)
```

Arguments

`x`	A matrix of multivariate data

Value

`MRSz.Kurtosis` The kurtosis matrix

`p.value` The p-value for the hypothesis of symmetry under the Gaussian assumption

References

Gy.Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021. Example 6.9

See Also

Other Indexes: `Esti_Kurt_Mardia()`, `Esti_Kurt_Total()`, `Esti_Skew_MRSz()`, `Esti_Skew_Mardia()`

`Esti_Kurt_Total` *Estimation of the Total Kurtosis Index*

Description

Estimation of the Total Kurtosis Index

Usage

```
Esti_Kurt_Total(x)
```

Arguments

`x`	A matrix of multivariate data

Value

`Total.Kurtosis` The total kurtosis index

`p.value` The p-value under the Gaussian hypothesis

References

Gy.Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021. Example 6.1

See Also

Other Indexes: `Esti_Kurt_MRSz()`, `Esti_Kurt_Mardia()`, `Esti_Skew_MRSz()`, `Esti_Skew_Mardia()`

Esti_Kurt_Variance_Th *Asymptotic Variance of the estimated kurtosis vector*

Description

Warning: the function requires 8! computations, for d>3, the timing required maybe large.

Usage

```
Esti_Kurt_Variance_Th(cum)
```

Arguments

cum	The theoretical/estimated cumulants up to the 8th order in vector form

Value

The matrix of theoretical/estimated variance

References

Gy.Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021. Ch. 6, formula (6.26)

See Also

Other Estimation: Esti_EVSK(), Esti_MMom_MCum(), Esti_Skew_Mardia(), Esti_Skew_Variance_Th()

Esti_MMom_MCum *Estimation of multivariate T-Moments and T-Cumulants*

Description

Provides estimates of univariate and multivariate moments and cumulants up to order r. By default data are standardized; using only demeaned or raw data is also possible.

Usage

```
Esti_MMom_MCum(X, r, centering = FALSE, scaling = TRUE)
```

Arguments

X	d-vector data
r	The highest moment order (r >2)
centering	set to T (and scaling = F) if only centering is needed
scaling	set to T (and centering=F) if standardization of multivariate data is needed

Value

`estMu.r`: the list of the multivariate moments up to order `r`

`estCum.r`: the list of the multivariate cumulants up to order `r`

References

Gy.Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021.

See Also

Other Estimation: `Esti_EVSK()`, `Esti_Kurt_Variance_Th()`, `Esti_Skew_Mardia()`, `Esti_Skew_Variance_Th()`

Examples

```
## generate random data from a 3-variate skew normal distribution
alpha<-c(10,5,0)
omega<-diag(3)
x<-distr_SkewNorm_Rand(50,omega,alpha)
## estimate the first three moments and cumulants from raw (uncentered and unstandardized) data
Esti_MMom_MCum(x,3,centering=FALSE,scaling=FALSE)
## estimate the first three moments and cumulants from standardized data
Esti_MMom_MCum(x,3,centering=FALSE,scaling=TRUE)
```

Esti_Skew_Mardia — *Estimation of Mardia's Skewness index*

Description

Compute the multivariate Mardia's skewness index and provides the p-value for the hypothesis of zero symmetry under the Gaussian assumption

Usage

```
Esti_Skew_Mardia(x)
```

Arguments

`x`	A matrix of multivariate data

Value

`Mardia.Skewness` The skewness index

`p.value` The p-value under the Gaussian hypothesis

References

Gy.Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021. Example 6.1

See Also

Other Indexes: Esti_Kurt_MRSz(), Esti_Kurt_Mardia(), Esti_Kurt_Total(), Esti_Skew_MRSz()

Other Estimation: Esti_EVSK(), Esti_Kurt_Variance_Th(), Esti_MMom_MCum(), Esti_Skew_Variance_Th()

Esti_Skew_MRSz *Estimation of Mori, Rohatgi, Szekely (MRSz's) skewness vector*

Description

Estimation of Mori, Rohatgi, Szekely (MRSz's) skewness vector

Usage

```
Esti_Skew_MRSz(x)
```

Arguments

x	A matrix of multivariate data

Value

MRSz.Skewness.Vector The skewness vector

MRSz.Skewness.Index The skewness index

p.value The p-value for the hypothesis of symmetry under the Gaussian assumption

References

Gy.Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021. Example 6.2

S. R. Jammalamadaka, E. Taufer, Gy. Terdik. On multivariate skewness and kurtosis. Sankhya A, 83(2), 607-644

See Also

Other Indexes: Esti_Kurt_MRSz(), Esti_Kurt_Mardia(), Esti_Kurt_Total(), Esti_Skew_Mardia()

Esti_Skew_Variance_Th *Asymptotic Variance of the estimated skewness vector*

Description

Asymptotic Variance of the estimated skewness vector

Usage

```
Esti_Skew_Variance_Th(cum)
```

Arguments

cum	The theoretical/estimated cumulants up to order 6 in vector form

Value

The matrix of theoretical/estimated variance

References

Gy.Terdik, Multivariate statistical methods - Going beyond the linear, Springer 2021. Ch.6, formula (6.13)

See Also

Other Estimation: Esti_EVSK(), Esti_Kurt_Variance_Th(), Esti_MMom_MCum(), Esti_Skew_Mardia()

Examples

```
alpha<-c(10,5)
omega<-diag(rep(1,2))
MC <- distr_SkewNorm_MomCum_Th(r = 6,omega,alpha)
cum <- MC$CumX
VS <- Esti_Skew_Variance_Th(cum)
```

Esti_Variance_Skew_Kurt
Estimated Variance of skewness and kurtosis vectors

Description

Provides the estimated covariance matrices of the data-estimated skewness and kurtosis vectors.

Usage

```
Esti_Variance_Skew_Kurt(X)
```

Arguments

`X`	A matrix of d-variate data

Value

The list of covariance matrices of the skewness and kurtosis vectors

References

Gy.Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021.

`Hermite_Coeff` *Coefficients of univariate Hermite polynomials*

Description

Provides the vector of coefficients of the univariate Hermite polynomial $H_N(x)$ with variance 1 and order N.

Usage

```
Hermite_Coeff(N)
```

Arguments

`N`	The order of polynomial

Value

The vector of coefficients of x^N, x^{N-2}...

References

Gy.Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021. Section 4.4 (4.24)

See Also

Other Hermite: `Hermite_CoeffMulti()`, `Hermite_N_Cov_X1_X2()`, `Hermite_Nth()`, `Hermite_Poly_HN_Multi()`, `Hermite_Poly_HN()`, `Hermite_Poly_NH_Inv()`, `Hermite_Poly_NH_Multi_Inv()`

Hermite_CoeffMulti — *Coefficients of multivariate T-Hermite polynomials for standardized variate*

Description

Provides the matrix of coefficients of $x^{\otimes N}$, $\kappa_2^{\otimes} x^{\otimes (N-2)}$... for the d-variate T-Hermite polynomials up to order N.

Usage

```
Hermite_CoeffMulti(N, d)
```

Arguments

N	the maximum order of polynomials
d	the dimension of d-variate X

Value

The list of matrices of coefficients for the d-variate polynomials from 1 to N

References

Gy.Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021. Section 4.6.2, p. 223, Remark 4.8,

See Also

Other Hermite: Hermite_Coeff(), Hermite_N_Cov_X1_X2(), Hermite_Nth(), Hermite_Poly_HN_Multi(), Hermite_Poly_HN(), Hermite_Poly_NH_Inv(), Hermite_Poly_NH_Multi_Inv()

Examples

```
N <- 5; d <- 3
H_N_Xc <- Hermite_CoeffMulti(N,d) # coefficients
X <- c(1:3);
X3 <- kronecker(X,kronecker(X,X));
X5 <- kronecker(X3,kronecker(X,X))
Idv <- as.vector(diag(d)) # vector of variance matrix
# value of H5 at X is
vH5<-H_N_Xc[[1]] %*% X5 + H_N_Xc[[2]] %*%kronecker(Idv,X3) +
  H_N_Xc[[3]] %*%kronecker(kronecker(Idv,Idv),X)
```

Hermite_Nth — *T-Hermite polynomial with order N at standardized vector x*

Description

Computes the N-th d-variate T-Hermite polynomial at standardized vector x

Usage

```
Hermite_Nth(x, N)
```

Arguments

`x`	multivariate data of size d
`N`	degree of T-Hermite polynomial

Value

d-variate T-Hermite polynomial of order N evaluated at vector x

References

Gy.Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021. Section 4.6.2, (4.73), p.223

See Also

Other Hermite: `Hermite_CoeffMulti()`, `Hermite_Coeff()`, `Hermite_N_Cov_X1_X2()`, `Hermite_Poly_HN_Multi()`, `Hermite_Poly_HN()`, `Hermite_Poly_NH_Inv()`, `Hermite_Poly_NH_Multi_Inv()`

Hermite_N_Cov_X1_X2 — *Covariance matrix for multivariate T-Hermite polynomials*

Description

Computation of the covariance matrix between d-variate T-Hermite polynomials $H_N(X_1)$ and $H_N(X_2)$.

Usage

```
Hermite_N_Cov_X1_X2(SigX12, N)
```

Arguments

`SigX12`	Covariance matrix of the Gaussian vectors X1 and X2 respectively of dimensions d1 and d2
`N`	Common degree of the multivariate Hermite polynomials

Value

Covariance matrix of $H_N(X_1)$ and $H_N(X_2)$

References

Gy.Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021. (4.59), (4.66),

See Also

Other Hermite: `Hermite_CoeffMulti()`, `Hermite_Coeff()`, `Hermite_Nth()`, `Hermite_Poly_HN_Multi()`, `Hermite_Poly_HN()`, `Hermite_Poly_NH_Inv()`, `Hermite_Poly_NH_Multi_Inv()`

Examples

```
Covmat<-matrix(c(1,0.8,0.8,1),2,2)
Cov_X1_X2 <- Hermite_N_Cov_X1_X2(Covmat,3)
```

`Hermite_Poly_HN` *Univariate Hermite polynomials*

Description

Provides the vector of univariate Hermite polynomials up to order N evaluated at x

Usage

```
Hermite_Poly_HN(x, N, sigma2 = 1)
```

Arguments

`x`	A scalar at which to evaluate the Hermite polynomials
`N`	The maximum order of the polynomials
`sigma2`	The variance, by default is set to 1

Value

`H_N_x` The vector of Hermite polynomials with degrees from 1 to N evaluated at x

References

Gy.Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021. Section 4.1

See Also

Other Hermite: `Hermite_CoeffMulti()`, `Hermite_Coeff()`, `Hermite_N_Cov_X1_X2()`, `Hermite_Nth()`, `Hermite_Poly_HN_Multi()`, `Hermite_Poly_NH_Inv()`, `Hermite_Poly_NH_Multi_Inv()`

Hermite_Poly_HN_Multi *Multivariate T-Hermite polynomials*

Description

Computes the multivariate T-Hermite polynomials up to order N at vector variate x with covariance matrix Sig2

Usage

```
Hermite_Poly_HN_Multi(x, N, Sig2 = diag(length(x)))
```

Arguments

x	the d-vector of values at which to evaluate the polynomials
N	the maximum order of polynomials
Sig2	the covariance matrix; default value is the unit matrix diag(length(x))

Value

The list of d-variate polynomials of order from 1 to N evaluated at vector x

References

Gy.Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021. Section 4.6.2, (4.73), p.223

See Also

Other Hermite: Hermite_CoeffMulti(), Hermite_Coeff(), Hermite_N_Cov_X1_X2(), Hermite_Nth(), Hermite_Poly_HN(), Hermite_Poly_NH_Inv(), Hermite_Poly_NH_Multi_Inv()

Examples

```
x<-c(1,3)
N<-3
Sig2<- diag(length(x)) # matrix(c(1,0,0,1),2,2,byrow = T)
Hermite_Poly_HN_Multi(x,N)
```

Hermite_Poly_NH_Inv *Inverse univariate Hermite polynomial*

Description

Inverse univariate Hermite polynomial

Usage

```
Hermite_Poly_NH_Inv(H_N_x, sigma2 = 1)
```

Arguments

H_N_x	The vector of Hermite Polynomials from 1 to N evaluated at x
sigma2	The variance, by default is set to 1

Value

The vector of x powers: x^n, $n = 1 : N$

References

Gy.Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021. Section 4.4, (4.23), p.198

See Also

Other Hermite: Hermite_CoeffMulti(), Hermite_Coeff(), Hermite_N_Cov_X1_X2(), Hermite_Nth(), Hermite_Poly_HN_Multi(), Hermite_Poly_HN(), Hermite_Poly_NH_Multi_Inv()

Hermite_Poly_NH_Multi_Inv *Inverse of d-variate T-Hermite Polynomial*

Description

Compute the powers of vector variate x when Hermite polynomials are given

Usage

```
Hermite_Poly_NH_Multi_Inv(H_N_X, N, Sig2 = diag(length(H_N_X[[1]])))
```

Arguments

H_N_X	The list of d-variate T-Hermite Polynomials of order from 1 to N evaluated at X
N	the highest polynomial order
Sig2	The variance matrix of x, the default is set to unit matrix

Value

The list of x, $x^{\otimes 2}$,... $x^{\otimes N}$

References

Gy.Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021. Section 4.6.2, (4.72), p.223

See Also

Other Hermite: Hermite_CoeffMulti(), Hermite_Coeff(), Hermite_N_Cov_X1_X2(), Hermite_Nth(), Hermite_Poly_HN_Multi(), Hermite_Poly_HN(), Hermite_Poly_NH_Inv()

Examples

```
x<-c(1,3)
Sig2 <- diag(length(x)) # matrix(c(1,0,0,1),2,2,byrow=T)
N<-4
H_N_X<-Hermite_Poly_HN_Multi(x,N,Sig2)
x_ad_n <- Hermite_Poly_NH_Multi_Inv(H_N_X,N,Sig2)
```

---

indx_Commutator_Kmn *Index vector for commutation of T-products of two vectors*

Description

Transforms vec A to vec of the transposed A. Same results as matr_Commutator_Kmn.

Usage

```
indx_Commutator_Kmn(m, n)
```

Arguments

m	Row-dimension
n	Col-dimension

Value

A vector of indexes to provide the commutation

References

Gy. Terdik, Multivariate statistical methods - Going beyond the linear, Springer 2021 (p.8, (1.12)).

See Also

Other Matrices and commutators: indx_Commutator_Kperm(), indx_Commutator_Mixing(), indx_Commutator_Moment(), indx_Elimination(), indx_Qplication(), indx_Symmetry(), indx_UnivMomCum(), matr_Commutator_Kmn(), matr_Commutator_Kperm(), matr_Commutator_Mixing(), matr_Commutator_Moment(), matr_Elimination(), matr_Qplication(), matr_Symmetry()

Examples

```
A<-1:6
A[indx_Commutator_Kmn(3,2)]
## Same as
as.vector(matr_Commutator_Kmn(3,2)%*%A)
```

---

indx_Commutator_Kperm *Index vector for commutation of T-products of any number of vectors*

Description

Produces any permutation of kronecker products of vectors of any length. Same results as matr_Commutator_Kperm.

Usage

```
indx_Commutator_Kperm(perm, dims)
```

Arguments

perm	vector indicating the permutation of the order in the Kronecker product,
dims	vector indicating the dimensions of the vectors, use dims <- d if all dimensions are equal

Value

An index vector to produce the permutation

References

Holmquist B (1996) The d-variate vector Hermite polynomial of order. Linear Algebra and its Applications 237/238, 155-190.

Gy., Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021, 1.2.4 Commuting T-Products of Vectors.

See Also

Other Matrices and commutators: indx_Commutator_Kmn(), indx_Commutator_Mixing(), indx_Commutator_Moment(), indx_Elimination(), indx_Qplication(), indx_Symmetry(), indx_UnivMomCum(), matr_Commutator_Kmn(), matr_Commutator_Kperm(), matr_Commutator_Mixing(), matr_Commutator_Moment(), matr_Elimination(), matr_Qplication(), matr_Symmetry()

Examples

```
a1<-c(1,2)
a2<-c(2,3,4)
a3<-c(1,3)
p1<-a1%x%a2%x%a3
p1[indx_Commutator_Kperm(c(3,1,2),c(2,3,2))]
## Same as
a3%x%a1%x%a2
## Same as
as.vector(matr_Commutator_Kperm(c(3,1,2),c(2,3,2))%*%p1)
```

---

indx_Commutator_Mixing

Index commutator mixing

Description

Provides the product Kx where K is the moment commutator as produced by `matr_Commutator_Mixing` and x is a vector. It avoids the construction of large commutators matrices working much faster with respect to `matr_Commutator_Moment`.

Usage

```
indx_Commutator_Mixing(x, d1, d2)
```

Arguments

x	a vector of dimension `prod(d1)*prod(d2)`
d1	dimension of the first group of vectors
d2	dimension of the second group of vectors

Value

A vector Kx.

References

Gy.Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021. Formula (4.58) p. 218.

See Also

Other Matrices and commutators: `indx_Commutator_Kmn()`, `indx_Commutator_Kperm()`, `indx_Commutator_Moment()`, `indx_Elimination()`, `indx_Qplication()`, `indx_Symmetry()`, `indx_UnivMomCum()`, `matr_Commutator_Kmn()`, `matr_Commutator_Kperm()`, `matr_Commutator_Mixing()`, `matr_Commutator_Moment()`, `matr_Elimination()`, `matr_Qplication()`, `matr_Symmetry()`

Examples

```
d1 <- c(2, 3, 2)
d2<-  c(3 ,2, 2)
x<-1:(prod(d1)*prod(d2))
indx_Commutator_Mixing(x,d1,d2)
# Same as
MCM<-matr_Commutator_Mixing(d1,d2)
as.vector(MCM%*%x)
```

---

indx_Commutator_Moment

Linear combination of moments

Description

For a d-variate distribution it provides the product Kx where K is the moment commutator as produced by `matr_Commutator_Moment` and x is a vector of appropriate dimension. It avoids the construction of large commutators matrices working much faster with respect to `matr_Commutator_Moment`.

Usage

```
indx_Commutator_Moment(x, el_rm, d)
```

Arguments

`x`	a vector of length d^n where n is length of (el_rm)
`el_rm`	type of a partition
`d`	dimensionality of the underlying multivariate distribution

Value

A vector K x

References

Gy., Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021, Section 2.4.3, p.100, Sect. A.2.1, p. 353., Corollary 2.6., p.95

See Also

Other Matrices and commutators: `indx_Commutator_Kmn()`, `indx_Commutator_Kperm()`, `indx_Commutator_Mixing()`, `indx_Elimination()`, `indx_Qplication()`, `indx_Symmetry()`, `indx_UnivMomCum()`, `matr_Commutator_Kmn()`, `matr_Commutator_Kperm()`, `matr_Commutator_Mixing()`, `matr_Commutator_Moment()`, `matr_Elimination()`, `matr_Qplication()`, `matr_Symmetry()`

Examples

```
n=4;  r=2 ;  m=1 ;  d=2;
PTA<-Partition_Type_All(n)
el_r<-PTA$eL_r[[r]][m,]
## el_r is a given type (always a vector)
x<-1:d^n
indx_Commutator_Moment(x,el_r,d)
# Same as
MC<- matr_Commutator_Moment(el_r,d)
as.vector(MC%*%x)
```

---

indx_Elimination   *Distinct values selection vector*

Description

Eliminates the duplicated/q-plicated elements in a T-vector of multivariate moments and cumulants. Produces the same results as matr_Elimination. Note indx_Elimination does not provide the same results as unique()

Usage

```
indx_Elimination(d, q)
```

Arguments

d	dimension of a vector x
q	power of the Kronecker product

Value

A vector of indexes of the distinct elements in the T-vector

References

Gy. Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021. Section 1.3.2 Multi-Indexing, Elimination, and Duplication, p.21,(1.32)

See Also

Other Matrices and commutators: `indx_Commutator_Kmn()`, `indx_Commutator_Kperm()`, `indx_Commutator_Mixing()`, `indx_Commutator_Moment()`, `indx_Qplication()`, `indx_Symmetry()`, `indx_UnivMomCum()`, `matr_Commutator_Kmn()`, `matr_Commutator_Kperm()`, `matr_Commutator_Mixing()`, `matr_Commutator_Moment()`, `matr_Elimination()`, `matr_Qplication()`, `matr_Symmetry()`

Examples

```
x<-c(1,0,3)
y<-kronecker(x,kronecker(x,x))
y[indx_Elimination(3,3)]
## Not the same results as
unique(y)
```

---

indx_Qplication *Qplication vector*

Description

Restores the duplicated/q-plicated elements which are eliminated by matr_Elimination or indx_Elimination in a T-product of vectors of dimension d. It produces the same results as matr_Qplication.

Usage

```
indx_Qplication(d, q)
```

Arguments

d	dimension of the vectors in the T-product
q	power of the Kronecker product

Value

A vector (T-vector) with all elements previously eliminated by indx_Elimination

References

Gy. Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021, p.21, (1.31)

See Also

Other Matrices and commutators: indx_Commutator_Kmn(), indx_Commutator_Kperm(), indx_Commutator_Mixing(), indx_Commutator_Moment(), indx_Elimination(), indx_Symmetry(), indx_UnivMomCum(), matr_Commutator_Kmn(), matr_Commutator_Kperm(), matr_Commutator_Mixing(), matr_Commutator_Moment(), matr_Elimination(), matr_Qplication(), matr_Symmetry()

Examples

```
x<-c(1,2,3)
y<-kronecker(kronecker(x,x),x)
## Distinct elements of y
z<-y[indx_Elimination(3,3)]
## Restore eliminated elements in z
z[indx_Qplication(3,3)]
```

`indx_Symmetry` *Symmetrizing vector*

Description

Vector symmetrizing a T-product of vectors of the same dimension d. Produces the same results as matr_Symmetry

Usage

```
indx_Symmetry(x, d, n)
```

Arguments

x	the vector to be symmetrized of dimension d^n
d	size of the single vectors in the product
n	power of the T-product

Value

A vector with the symmetrized version of x of dimension d^n

References

Gy. Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021.Section 1.3.1 Symmetrization, p.14. (1.29)

See Also

Other Matrices and commutators: `indx_Commutator_Kmn()`, `indx_Commutator_Kperm()`, `indx_Commutator_Mixing()`, `indx_Commutator_Moment()`, `indx_Elimination()`, `indx_Qplication()`, `indx_UnivMomCum()`, `matr_Commutator_Kmn()`, `matr_Commutator_Kperm()`, `matr_Commutator_Mixing()`, `matr_Commutator_Moment()`, `matr_Elimination()`, `matr_Qplication()`, `matr_Symmetry()`

Examples

```
a<-c(1,2)
b<-c(2,3)
c<-kronecker(kronecker(a,a),b)
## The symmetrized version of c is
indx_Symmetry(c,2,3)
```

`indx_UnivMomCum` *Univariate moments and cumulants from T-vectors*

Description

A vector of indexes to select the moments and cumulants of the single components of the random vector X for which a T-vector of moments and cumulants is available

Usage

```
indx_UnivMomCum(d, q)
```

Arguments

d	dimension of a vector X
q	power of the Kronecker product

Value

A vector of indexes

See Also

Other Matrices and commutators: `indx_Commutator_Kmn()`, `indx_Commutator_Kperm()`, `indx_Commutator_Mixing()`, `indx_Commutator_Moment()`, `indx_Elimination()`, `indx_Qplication()`, `indx_Symmetry()`, `matr_Commutator_Kmn()`, `matr_Commutator_Kperm()`, `matr_Commutator_Mixing()`, `matr_Commutator_Moment()`, `matr_Elimination()`, `matr_Qplication()`, `matr_Symmetry()`

Examples

```
## For a 3-variate skewness and kurtosis vectors estimated from data, extract
## the skewness and kurtosis estimates for each of the single components of the vector
alpha<-c(10,5,0)
omega<-diag(rep(1,3))
X<-distr_SkewNorm_Rand(200, omega, alpha)
EVSK<-Esti_EVSK(X)
## Get the univariate skewness and kurtosis for X1,X2,X3
EVSK$estSkew[indx_UnivMomCum(3,3)]
EVSK$estKurt[indx_UnivMomCum(3,4)]
```

`matr_Commutator_Kmn` *Commutation matrix*

Description

Transforms vec A to vec of the transposed A. An option for sparse matrix is provided, by default a non-sparse matrix is produced. Using sparse matrices increases computation times, but far less memory is required

Usage

```
matr_Commutator_Kmn(m, n, useSparse = FALSE)
```

Arguments

`m`	Row-dimension
`n`	Col-dimension
`useSparse`	T or F.

Value

A commutation matrix matrix of dimension $mn \times mn$. If `useSparse=TRUE` an object of the class "dgCMatrix" is produced.

References

Gy. Terdik, Multivariate statistical methods - Going beyond the linear, Springer 2021 (p.8, (1.12)).

See Also

Other Matrices and commutators: `indx_Commutator_Kmn()`, `indx_Commutator_Kperm()`, `indx_Commutator_Mixing()`, `indx_Commutator_Moment()`, `indx_Elimination()`, `indx_Qplication()`, `indx_Symmetry()`, `indx_UnivMomCum()`, `matr_Commutator_Kperm()`, `matr_Commutator_Mixing()`, `matr_Commutator_Moment()`, `matr_Elimination()`, `matr_Qplication()`, `matr_Symmetry()`

Examples

```
A<-matrix(1:6,3,2)
as.vector(matr_Commutator_Kmn(3,2)%*%c(A))
```

matr_Commutator_Kperm *Commutator for T-products of vectors*

Description

Produces any permutation of kronecker products of vectors of any length. An option for sparse matrix is provided, by default a non-sparse matrix is produced. Using sparse matrices increases computation times, but far less memory is required.

Usage

```
matr_Commutator_Kperm(perm, dims, useSparse = FALSE)
```

Arguments

perm	vector indicating the permutation of the order in the Kronecker product,
dims	vector indicating the dimensions of the vectors, use dims <- d if all dimensions are equal
useSparse	T or F.

Value

A square permutation matrix of size `prod(dims)`. If `useSparse=TRUE` an object of the class "dgCMatrix" is produced.

References

Holmquist B (1996) The d-variate vector Hermite polynomial of order. Linear Algebra and its Applications 237/238, 155-190.

Gy., Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021, 1.2.4 Commuting T-Products of Vectors.

See Also

Other Matrices and commutators: `indx_Commutator_Kmn()`, `indx_Commutator_Kperm()`, `indx_Commutator_Mixing()`, `indx_Commutator_Moment()`, `indx_Elimination()`, `indx_Qplication()`, `indx_Symmetry()`, `indx_UnivMomCum()`, `matr_Commutator_Kmn()`, `matr_Commutator_Mixing()`, `matr_Commutator_Moment()`, `matr_Elimination()`, `matr_Qplication()`, `matr_Symmetry()`

Examples

```
dims <- c(2,3,2)
perm  <-  c(1,3,2)
matr_Commutator_Kperm(perm,dims)
perm  <- c(3,1,4,2)
dims <- 4 # All vectors with dimension 4
# If all dimensions are equal, using dims <- d instead of
```

```
# dims <- c(d,d,d,d,d,d,d,d) will be much faster.
# For example, for perm <- c(2,4,6,1,3,8,5,7) and d <- 3
# matr_Commutator_Kperm(c(2,4,6,1,3,8,5,7),3)  ## requires 2.11 secs
# matr_Commutator_Kperm(c(2,4,6,1,3,8,5,7),c(3,3,3,3,3,3,3,3))  ## requires 1326.47 secs
```

---

matr_Commutator_Mixing *Mixing commutator*

Description

Used for the expected value of the T-product of two Hermite polynomials with dimensions d1 and d2 respectively. With an option for sparse matrices.

Usage

```
matr_Commutator_Mixing(d1, d2, useSparse = FALSE)
```

Arguments

d1	dimension of the first group of vectors
d2	dimension of the second group of vectors
useSparse	T or F.

Value

A square matrix of dimension `prod(d1)*prod(d2)`. If `useSparse=TRUE` an object of the class "dgCMatrix" is produced.

References

Gy.Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021. Formula (4.58) p. 218.

See Also

Other Matrices and commutators: `indx_Commutator_Kmn()`, `indx_Commutator_Kperm()`, `indx_Commutator_Mixing()`, `indx_Commutator_Moment()`, `indx_Elimination()`, `indx_Qplication()`, `indx_Symmetry()`, `indx_UnivMomCum()`, `matr_Commutator_Kmn()`, `matr_Commutator_Kperm()`, `matr_Commutator_Moment()`, `matr_Elimination()`, `matr_Qplication()`, `matr_Symmetry()`

Examples

```
d1 <- c(2, 3, 2)
d2<-  c(3 ,2, 2)
MCM<-matr_Commutator_Mixing(d1,d2)
```

matr_Commutator_Moment

Commutator matrix for moment formulae

Description

Commutator matrix for moment formulae

Usage

```
matr_Commutator_Moment(el_rm, d, useSparse = FALSE)
```

Arguments

el_rm	type of a partition
d	dimensions of the vector
useSparse	TRUE or FALSE

Value

A commutator matrix

References

Gy., Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021, Section 2.4.3, p.100, Sect. A.2.1, p. 353., Corollary 2.6., p.95

See Also

Other Matrices and commutators: indx_Commutator_Kmn(), indx_Commutator_Kperm(), indx_Commutator_Mixing(), indx_Commutator_Moment(), indx_Elimination(), indx_Qplication(), indx_Symmetry(), indx_UnivMomCum(), matr_Commutator_Kmn(), matr_Commutator_Kperm(), matr_Commutator_Mixing(), matr_Elimination(), matr_Qplication(), matr_Symmetry()

Examples

```
n=4;  r=2 ;  m=1 ;  d=2;
PTA<-Partition_Type_All(n)
el_r<-PTA$eL_r[[r]][m,]
## el_r is a given type (always a vector)
MC<- matr_Commutator_Moment(el_r,d)
MC
```

`matr_Elimination` *Elimination Matrix*

Description

Eliminates the duplicated/q-plicated elements in a T-vector of multivariate moments and cumulants.

Usage

```
matr_Elimination(d, q, useSparse = FALSE)
```

Arguments

d	dimension of a vector x
q	power of the Kronecker product
useSparse	TRUE or FALSE.

Value

Elimination matrix of order $\eta_{d,q} \times d^q = \binom{d+q-1}{q}$. If `useSparse=TRUE` an object of the class "dgCMatrix" is produced.

References

Gy. Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021. Section 1.3.2 Multi-Indexing, Elimination, and Duplication, p.21,(1.32)

See Also

Other Matrices and commutators: `indx_Commutator_Kmn()`, `indx_Commutator_Kperm()`, `indx_Commutator_Mixing()`, `indx_Commutator_Moment()`, `indx_Elimination()`, `indx_Qplication()`, `indx_Symmetry()`, `indx_UnivMomCum()`, `matr_Commutator_Kmn()`, `matr_Commutator_Kperm()`, `matr_Commutator_Mixing()`, `matr_Commutator_Moment()`, `matr_Qplication()`, `matr_Symmetry()`

Examples

```
x<-c(1,2,3)
y<-kronecker(kronecker(x,x),x)
## Distinct elements of y
z<-as.matrix(matr_Elimination(3,3))%*%y
## Restore eliminated elements in z
as.vector(matr_Qplication(3,3)%*%z)
```

`matr_Qplication` *Qplication Matrix*

Description

Restores the duplicated/q-plicated elements which are eliminated by matr_Elimination in a T-product of vectors of dimension d.

Usage

```
matr_Qplication(d, q, useSparse = FALSE)
```

Arguments

`d`	dimension of a vector x
`q`	power of the Kronecker product
`useSparse`	TRUE or FALSE.

Details

Note: since the algorithm of elimination is not unique, q-plication works together with the function matr_Elimination only.

Value

Qplication matrix of order $d^q \times \eta_{d,q}$, see (1.30), p.15. If `useSparse=TRUE` an object of the class "dgCMatrix" is produced.

References

Gy. Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021, p.21, (1.31)

See Also

Other Matrices and commutators: `indx_Commutator_Kmn()`, `indx_Commutator_Kperm()`, `indx_Commutator_Mixing()`, `indx_Commutator_Moment()`, `indx_Elimination()`, `indx_Qplication()`, `indx_Symmetry()`, `indx_UnivMomCum()`, `matr_Commutator_Kmn()`, `matr_Commutator_Kperm()`, `matr_Commutator_Mixing()`, `matr_Commutator_Moment()`, `matr_Elimination()`, `matr_Symmetry()`

Examples

```
x<-c(1,2,3)
y<-kronecker(kronecker(x,x),x)
## Distinct elements of y
z<-as.matrix(matr_Elimination(3,3))%*%y
## Restore eliminated elements in z
as.vector(matr_Qplication(3,3)%*%z)
```

matr_Symmetry *Symmetrizer Matrix*

Description

Based on Chacon and Duong (2015) efficient recursive algorithms for functionals based on higher order derivatives. An option for sparse matrix is provided. By using sparse matrices far less memory is required and faster computation times are obtained

Usage

```
matr_Symmetry(d, n, useSparse = FALSE)
```

Arguments

d	dimension of a vector x
n	power of the Kronecker product
useSparse	TRUE or FALSE. If TRUE an object of the class "dgCMatrix" is produced.

Value

A Symmetrizer matrix with order $d^n \times d^n$. If `useSparse=TRUE` an object of the class "dgCMatrix" is produced.

References

Chacon, J. E., and Duong, T. (2015). Efficient recursive algorithms for functionals based on higher order derivatives of the multivariate Gaussian density. Statistics and Computing, 25(5), 959-974.

Gy. Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021.Section 1.3.1 Symmetrization, p.14. (1.29)

See Also

Other Matrices and commutators: `indx_Commutator_Kmn()`, `indx_Commutator_Kperm()`, `indx_Commutator_Mixing()`, `indx_Commutator_Moment()`, `indx_Elimination()`, `indx_Qplication()`, `indx_Symmetry()`, `indx_UnivMomCum()`, `matr_Commutator_Kmn()`, `matr_Commutator_Kperm()`, `matr_Commutator_Mixing()`, `matr_Commutator_Moment()`, `matr_Elimination()`, `matr_Qplication()`

Examples

```
a<-c(1,2)
b<-c(2,3)
c<-kronecker(kronecker(a,a),b)
## The symmetrized version of c is
as.vector(matr_Symmetry(2,3)%*%c)
```

Partition_2Perm *Permutation of elements according to partition pU*

Description

Permutation of elements according to partition pU

Usage

```
Partition_2Perm(pU)
```

Arguments

pU	A partition matrix. For instance a matrix generated by Partition_Type_All.

Value

perm_pU A vector with the elements 1 to N permuted according to pU. The numbers of 1 : N are listed in the order of their occurrence in the blocks of pU.

References

Gy. Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021. Section 1.4.4

See Also

Other Partitions: Partition_Indecomposable(), Partition_Pairs(), Partition_Type_All(), Permutation_Inverse()

Examples

```
PA<-Partition_Type_All(4)
Partition_2Perm(PA$Part.class[[3]])
```

Partition_DiagramsClosedNoLoops *Closed Diagrams without Loops*

Description

Given a partition L looks for those partitions K of pairs which are indecomposable in accordance with L, (L, K) represent diagrams. It is used for getting higher order cumulants of Hermite polynomials defined by L, see Terdik (2021) Proposition 4.3, p.194.

Usage

```
Partition_DiagramsClosedNoLoops(L)
```

Arguments

L	a partition matrix

Value

The list of partition matrices indecomposable with respect to L

References

Gy. Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021. Section 1.4.8.1

Partition_Indecomposable
Building indecomposable partitions

Description

Produces the list of all indecomposable partitions with respect to the partition matrix L

Usage

```
Partition_Indecomposable(L)
```

Arguments

L	A partition matrix

Value

`IndecompK2L` A list of partition matrices indecomposable with respect to L

`numb_by_sizes` A vector indicating the number of indecomposable partitions with respect to L by sizes

References

Gy. Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021. Section 1.4.6

See Also

Other Partitions: `Partition_2Perm()`, `Partition_Pairs()`, `Partition_Type_All()`, `Permutation_Inverse()`

Examples

```
L<-matrix(c(1,1,0,0,0,0,1,1),2,4,byrow=TRUE)
IP<-Partition_Indecomposable(L)
IP$IndecompK2L
IP$num_by_sizes
```

Partition_Pairs *Partition into pairs of the set 1:N*

Description

Partition into pairs of the set 1:N

Usage

```
Partition_Pairs(N)
```

Arguments

N	The (integer) number of elements to be partitioned

Value

The list of partition matrices with blocks containing two elements. The list is empty if N is odd.

References

Gy. Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021. Section 1.4.8

See Also

Other Partitions: Partition_2Perm(), Partition_Indecomposable(), Partition_Type_All(), Permutation_Inverse()

Examples

```
PA<-Partition_Pairs(4)
```

`Partition_Type_All` *Partitions, type and number of partitions*

Description

Generates all partitions of `N` numbers and classify them by type

Usage

```
Partition_Type_All(N)
```

Arguments

`N`	The (integer) number of elements to be partitioned

Value

`Part.class` The list of all possible partitions given as partition matrices

`S_N_r` A vector with the number of partitions of size r=1, r=2, etc. (Stirling numbers of second kind)

`eL_r` A list of partition types with respect to partitions of size r=1, r=2, etc.

`S_r_j` Vectors of number of partitions with given types grouped by partitions of size r=1, r=2, etc.

References

Gy. Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021. Case 1.4, p.31 and Example 1.18, p.32.

See Also

Other Partitions: `Partition_2Perm()`, `Partition_Indecomposable()`, `Partition_Pairs()`, `Permutation_Inverse()`

Examples

```
# See Example 1.18, p. 32, reference below
PTA<-Partition_Type_All(4)
# Partitions generated
PTA$Part.class
# Partitions of size 2 includes two types
PTA$eL_r[[2]]
# Number of partitions with r=1 blocks, r=2 blocks, etc-
PTA$S_N_r
# Number of different types collected by  partitions of size r=1, r=2, etc.
PTA$S_r_j
# Partitions with size r=2, includes two types (above) each with number
PTA$S_r_j[[2]]
```

Permutation_Inverse *Inverse of a Permutation*

Description

Inverse of a Permutation

Usage

```
Permutation_Inverse(permutation0)
```

Arguments

permutation0	A permutation of numbers 1:n

Value

A vector containing the inverse permutation of permutation0

References

Gy. Terdik, Multivariate statistical methods - going beyond the linear, Springer 2021, Remark 1.1, p.2

See Also

Other Partitions: Partition_2Perm(), Partition_Indecomposable(), Partition_Pairs(), Partition_Type_All()

Index