MultiStatM: Multivariate Statistical Methods in R

by György Terdik and Emanuele Taufer

Abstract The package MultiStatM presents a vectorial approach to multivariate moments and cumulants. Functions provided concern algorithms to build set partitions and commutator matrices, multivariate d-Hermite polynomials; theoretical vector moments and vector cumulants of multivariate distributions and their conversion formulae. Applications discussed concern multivariate measures of skewness and kurtosis; asymptotic covariances for d-variate Hermite polynomials and multivariate moments and cumulants; Gram-Charlier approximations.

1 Introduction and background

The package **MultiStatM** (Terdik and Taufer (2024)) provides a set of tools such as set partitions, commutator matrices and vector Hermite polynomials which are needed in developing general computational and conversion formulae for multivariate moments and cumulants of any order. Applications are provided for theoretical moments and cumulants of some important symmetric and skew-symmetric multivariate distributions; measures of multivariate skewness and kurtosis; asymptotic covariance formulae for vector Hermite polynomials; last but not least, estimation functions for all theoretical results discussed are provided. All the methods implemented in **MultiStatM** are discussed in detail in Terdik (2021).

A careful study of the cumulants is a necessary and typical part of nonlinear statistics. Dealing with cumulants for multivariate distributions is made complicated by the index notations: McCullagh (2018) provides quite an elegant approach using tensor methods; however, tensor methods are not very well known and computationally not so simple. Kollo (2008) and Móri et al. (1994) provide formulae for cumulants in terms of matrices; however, retaining a matrix structure for all higher-order cumulants leads to high-dimensional matrices with special symmetric structures which are quite hard to follow notionally and computationally. Cumulant matrices, with special attention to the third and fourth order, are ubiquitous in the literature. The use of decompositions and approximations allow for dimensionality reduction and find several statistical applications; see Loperfido (2015) and Loperfido (2017) for analysis and discussion of skewness and kurtosis matrices and their applications.

MultiStatM offers an alternative method, which we believe is simpler to follow: the higher-order cumulants of the same degree for some random d-vector $\mathbf{X}$ are collected together and kept as a vector (see Terdik (2003)). In order to do so, a particular differential operator on a multivariate function, called the T-derivative, is introduced. Although the tensor product of Euclidean vectors is not commutative, it has the advantage of permutation equivalence and allows one to obtain general formulae for cumulants and moments of any order. Methods based on a matrix approach do not provide this type of result; see also Ould-Baba et al. (2015), which goes at most up to the sixth-order moment matrices, whereas there is no such limitation in our derivations and formulae. The vectorial approach has also been advocated by Holmquist (1996) and Chacón and Duong (2015); for further discussion, one can see also Kolda and Bader (2009) and Qi (2006).

More formally, with the symbol $\otimes$ denoting the Cartesian tensor product, consider the operator $D_{\boldsymbol{\lambda}}^{\otimes}$, which we refer to as the T-derivative; see Jammalamadaka et al. (2006) for details. For any function $\phi(\boldsymbol{\lambda})$, the T-derivative is defined as

$$D_{\boldsymbol{\lambda}}^{\otimes}\phi(\boldsymbol{\lambda}) = \operatorname{vec}\left(\left(\frac{\partial\phi(\boldsymbol{\lambda})}{\partial\boldsymbol{\lambda}'}\right)'\right) = \phi(\boldsymbol{\lambda}) \otimes \frac{\partial}{\partial\boldsymbol{\lambda}}. \tag{1}$$

If ϕ is k-times differentiable, its k-th T-derivative is simply $D_{\boldsymbol{\lambda}}^{\otimes k}\phi(\boldsymbol{\lambda}) = D_{\boldsymbol{\lambda}}^{\otimes}\left(D_{\boldsymbol{\lambda}}^{\otimes k-1}\phi(\boldsymbol{\lambda})\right)$. When ϕ is the characteristic function of some random d-vector, k-times application of the T-derivative will allow to retrieve the so-called multivariate T-moments vector of order k, which lists all moments of the same order and has dimension d^k. In the same way, multivariate T-cumulants and T-Hermite polynomials can be derived. **MultiStatM** aims at exploiting the advantages of this method for multivariate analysis and dealing with the high-dimensionality implied by this approach.

We could not find any **R** (R Core Team (2023)) package using this approach although there is a partial overlapping between functions in **MultiStatM** and functions that can be found in other **R** packages; when this happens, the functions in **MultiStatM** provide useful generalizations.

The package **arrangements** (Lai (2020)) provides, among other things, all possible partitions of a set (or an integer) in lexicographical order. In **MultiStatM** all possible partitions in terms of partition matrices together with information about types, sizes and groups of partitions are given. Furthermore **MultiStatM** provides other partition functions which produce indecomposable partitions, partitions that bring to a closed graph, partitions in only two groups.

The package **matrixcalc** (Novomestky (2021)) provides the commutation, elimination and duplication matrices for Cartesian tensor products of two vectors, which are particular cases of those provided in the package **MultiStatM**. Furthermore, in **MultiStatM** a commutator matrix for any T-product of vectors of any size is given; also, a fast algorithm to produce a symmetrizer of T-products of vectors is implemented (see Holmquist (1996)). One of the problems of commutators matrices is their high-dimensionality which can become soon quite cumbersome; on the other hand, commutator matrices are useful if one needs to work with linear combinations of matrices. To deal with the high-dimensionality issue, index versions of the commutators are also provided (more details below).

The package **calculus** (Guidotti (2022)) dealing with high dimensional numerical and symbolic calculus provides, up to a given order, the list of all symbolic entries of the multivariate Hermite polynomials with given covariance matrix used here. **MultiStatM** does not deal with single symbolic expressions rather provides full numerical evaluation of multivariate T-Hermite polynomials suitable for immediate use in further formulas and applications.

kStatistics (Di Nardo and Guarino (2022a)) gives string expressions for converting single entries of multivariate cumulants or moments which need to be joined and evaluated for numerical computations, for details and examples see Di Nardo and Guarino (2022b). This differs to a large degree from the formulae provided in **MultiStatM** which are always in numerical form and ready for applications; for example, conversion of multivariate moments to multivariate cumulants and viceversa, given any list of (numerical) multivariate moments up to order r, or multivariate cumulants up to order r are provided.

In **MultiStatM**, exploiting the conversion formulae and some results on uniform spherical and folded multivariate normal distributions due to Jammalamadaka et al. (2021b) and Terdik (2021), the multivariate moments and cumulants of any order of the multivariate skew normal and its generalization (Azzalini and Dalla Valle (1996) , Arellano-Valle and Genton (2005)) are provided. The package **sn** (Azzalini (2022)) provides moments and cumulants of any order for the univariate skew-normal and the skew-t distributions; statistical methods are provided for data fitting and model diagnostics, in the univariate and the multivariate case. In **MultiStatM** random numbers generator for multivariate skew distributions are provided also.

The package **moments** (Komsta and Novomestky (2022)) deals with functions to calculate, cumulants, Pearson's kurtosis, Geary's kurtosis and skewness; tests related to them from univariate data. However the multivariate approach is not considered. The packages **mpoly** (Kahle (2013)) and **orthopolynom** (Novomestky (2022)) deal with Hermite polynomials in the univariate case.

MultiStatM provides some common measures of cumulant-based multivariate skewness and kurtosis (Mardia (1970), Móri et al. (1994)) which are provided in a number of packages, for example **MaxSkew** (Franceschini and Loperfido (2017a)), **MultiSkew** (Franceschini and Loperfido (2017b)) and **Kurt** (Franceschini and Loperfido (2021)). Indeed the vectorial approach provides a unifying framework for several cumulant-based skewness and kurtosis indexes, as discussed in Jammalamadaka et al. (2021b) where, beyond the above mentioned ones, indexes proposed by Malkovich and Afifi (1973), Kollo (2008), Koziol (1987), Koziol (1989) and others are discussed. When going higher than the fourth order, the vectorial approach shows it full power; see Jammalamadaka et al. (2021a) for an application in deriving the asymptotic variances of skewness and kurtosis vectors and indexes which require cumulants up to the 8-th order. Below, exploiting **MultiStatM** applications to the construction of new measures of multivariate skewness and kurtosis and derivation of Gram-Charlier approximation to densities will be presented.

As far as non R-universe is concerned, the proprietary software Mathematica (Inc.) provides partial overlapping of the multivariate methods discussed in **MultiStatM**. There, functions to compute, numerically and symbolically, multivariate moments and cumulants of common distributions and general conversion functions from moments to cumulants and viceversa are available. Most of the multivariate distributions discussed in **MultiStatM** however are not available there. Mathematica also provides several functions for univariate Hermite polynomials which however need to be adapted for the multivariate case based on the T-derivative approach.

2 Package features

MultiStatM imports **arrangements**, **MASS** (Venables and Ripley (2002)), **Matrix** (Bates et al. (2024)), **mvtnorm** (Genz and Bretz (2009)) and **stats**. We discuss here some examples on the use of the functions; considerations about timing, objects size and algorithms will be collected in Section 4.

2.1 Set partitions

MultiStatM provides several functions dealing with set partitions. Such functions provide some basic tools used to build the multivariate formulae for moments and cumulants in the following sections.

Generally a set of N elements can be split into a set of disjoint subsets, i.e. it can be partitioned. The set of N elements will correspond to set $1:N=\{1,2,\ldots,N\}$. If $\mathcal{K}=\{b_1,b_2,\ldots,b_r\}$ where each $b_j\subset 1:N$, then $\mathcal{K}$ is a partition provided $\cup b_j = 1:N$, each b_j is non-empty and $b_j\cap b_i=\emptyset$ (the empty set) is disjoint whenever $j\neq i$. The subsets b_j, $j=1,2,\ldots,r$ are called the blocks of $\mathcal{K}$. We will call r (the number of the blocks in partition $\mathcal{K}$), the size of $\mathcal{K}$, and denote it by $|\mathcal{K}|=r$, and a partition with size r will be denoted by $\mathcal{K}_{\{r\}}$. Let us denote the set of all partitions of the numbers $1:N$ by $\mathcal{P}_N$.

Consider next a partition $\mathcal{K}_{\{r\}}=\{b_1,b_2,\ldots,b_r\}\in\mathcal{P}_N$, with size r. Denote the cardinality k_j of a block in the partition $\mathcal{K}_{\{r\}}$, i.e. $k_j=|b_j|$. The type of a partition $\mathcal{K}_{\{r\}}$ is $l=[l_1,\ldots,l_N]$, if $\mathcal{K}_{\{r\}}$ contains exactly l_j blocks with cardinality j. The type l is with length N always. A partition with size r and type l will be denoted by $\mathcal{K}_{\{r|l\}}$. It is clear that $l_j\geq 0$, and $\sum_j jl_j=N$, and $\sum_j l_j=r$. Naturally, some l_j's are zero. A block constitutes a row vector of entries 0's and 1's with length N. The places of 1's correspond to the elements of the block. A partition matrix collects the rows of its blocks, it is an $r\times N$ matrix with column-sums 1.

The basic function is `PartitionTypeAll` which provides complete information on the partition of a set of `N` elements, namely:

- `S_N_r`: a vector with the number of partitions of size `r=1`, `r=2`, etc. (Stirling numbers of the second kind); `S_N_r[r]` denotes the number of partition matrices of size `r`.
- `Part.class`: the list of all possible partitions given as partition matrices. This list is enumerated according to `S_N_r[r]`, $r=1,2,\ldots N$, such that the partition matrices with size `R` are listed from $\sum_{r<R}$`S_N_r`$[r]+1$ up to $\sum_{r\leq R}$`S_N_r`$[r]$. The order of the partition matrices within a fixed size is called canonical.
- `S_r_j`: a list of vectors of number of partitions with given types grouped by partitions of size `r=1`, `r=2`, etc.; an entry is the number of partitions with that type.
- `eL_r`: a list of partition types with respect to partitions of size `r=1`, `r=2`, etc. ; since a partition type is a row vector with length `N` this list includes matrices of types (row vectors), the number of rows are the length of vectors of `S_r_j` of a given size `r`.

Example 1. Consider the case where `N=4` and run the following

```
PTA <- PartitionTypeAll(4)
```

`S_N_r` provides the number of partitions with `r=1` to `r=4` blocks:

```
PTA$S_N_r
```

```
#> [1] 1 7 6 1
```

From the results above we see that there are: 1 partition of 1 block (`r=1`); 7 partitions of two blocks (`r=2`); 6 partitions of 3 blocks and 1 partition of 4 blocks. All the partition matrices are listed in `Part.class`, for example the first partition matrix of size 2 among 7 is

```
PTA$Part.class[[2]]
```

```
#>      [,1] [,2] [,3] [,4]
#> [1,]    1    1    1    0
#> [2,]    0    0    0    1
```

The second partition matrix of size 3 is

```
PTA$Part.class[[10]]
```

```
#>      [,1] [,2] [,3] [,4]
#> [1,]    1    0    1    0
#> [2,]    0    1    0    0
#> [3,]    0    0    0    1
```

etc.. If one interested in the number of partitions with different types for r=2 then consider the list S_r_j, i.e.

```
PTA$S_r_j[[2]]
```

```
#> [1] 4 3
```

That is, for partitions with r=2 blocks, there are 2 possible types with 4 and 3 partitions each. These types will show up in the list eL_r:

```
PTA$eL_r
```

```
#> [[1]]
#> [1] 0 0 0 1
#>
#> [[2]]
#>      [,1] [,2] [,3] [,4]
#> [1,]    1    0    1    0
#> [2,]    0    2    0    0
#>
#> [[3]]
#> [1] 2 1 0 0
#>
#> [[4]]
#> [1] 4 0 0 0
```

From PTA$eL_r[[2]] we see that there are two types of partition with r=2: the first is of type $[1,0,1,0] = (l_1 = 1, l_2 = 0, l_3 = 1, l_4 = 0)$ with 4 partitions and the second is of type $[0,2,0,0] = (l_1 = 0, l_2 = 2, l_3 = 0, l_4 = 0)$ with 3 partitions.

Another general function in this class is Partitions which has a Type argument in order to specify the type of partition to compute: 2Perm which provides the permutation of N elements according to a partition matrix L; Diagram which provides the list of partition matrices indecomposable with respect to L, representing diagrams without loops; Indecomp, which provides the list of all indecomposable partitions with respect to a partition matrix L; Pairs, which provides the list of partitions dividing into pairs the set of N elements.

2.2 Commutators, symmetrizer and selection matrices

The Commutators family of functions produce commutators and selection matrices. The use of matrices allows to use linear combinations to represent problems such as the permutation of powers of T-products or the selection of certain elements in a vector. On the other hand, the size of these matrices can quickly become quite important. To deal with this issues an option for sparse matrices is always provided; also, a corresponding Indx group of functions which give equivalent results to the corresponding functions in the group Matr is provided.

Two general functions in this family are CommutatorMatr and CommutatorIndx, both with a Type argument selected from: Kmn, Kperm, Mixing, Moment. Kmn produces a commutation matrix, with usual notation $\mathbf{K}_{m \cdot n}$, of dimension $mn \times mn$ such that, given a matrix $\mathbf{A}$ $m \times n$, $\mathbf{K}_{m \cdot n} \operatorname{vec} \mathbf{A} = \operatorname{vec} \mathbf{A}'$ (see Terdik (2021), p.8) while Kperm produce any permutation of Kronecker products of vectors of any length.

Example 2. Consider the product of vectors $\mathbf{a}_1 \otimes \mathbf{a}_2 \otimes \mathbf{a}_3$ of dimensions d_1 to d_3 respectively.

CommutatorMatr(Type="Kperm",c(3,1,2),c(d1,d2,d3)) produces $\mathbf{a}_3 \otimes \mathbf{a}_1 \otimes \mathbf{a}_2$.

```
a1 <- c(1, 2)
a2 <- c(2, 3, 4)
a3 <- c(1, 3)
x <- a1 %x% a2 %x% a3
c(CommutatorMatr(Type = "Kperm", c(3, 1, 2), c(2, 3, 2)) %*% x)
```

```
#>  [1]  2  3  4  4  6  8  6  9 12 12 18 24
```

```
a3 %x% a1 %x% a2
```

```
#>  [1]  2  3  4  4  6  8  6  9 12 12 18 24
```

The same result can be obtained by using CommutatorIndx

```
x[CommutatorIndx(Type = "Kperm", c(3, 1, 2), c(2, 3, 2))]
```

```
#>  [1]  2  3  4  4  6  8  6  9 12 12 18 24
```

Note that in the example above CommutatorMatr produces the matrix $\mathbf{K}_p$ of dimension 12×12 by which $\mathbf{y} = \mathbf{K}_p\mathbf{x}$, while CommutatorIndx produces the permutation $p_{\mathbf{y}}$ of 1:12 such that $\mathbf{y} = \mathbf{x}[p_{\mathbf{y}}]$. Recall that from computational point of view CommutatorIndx is always preferable however, if working with linear combinations of matrices, like in Example 3 and in Section 3.1 and the discussion leading to formulae (10) and (11), the use of CommutatorMatr is unavoidable. For this reason **MultiStatM** offers both possibilities. The idea of avoiding the use of commutator, selection (and symmetrizer) matrices can be traced back to Chacón and Duong (2015).

The Elimination and Qplication matrices- related functions respectively eliminate and restore duplicated or q-plicated elements in powers of T-products. For examples of applications of these functions we defer to Section 2.4.

The symmetrizer matrix, a $d^n \times d^n$ matrix for the symmetrization of a T-product of n vectors with the same dimension d which overcomes the difficulties arising from the non commutative property of the Kronecker product, and simplifies considerably the computation formulae for multivariate polynomials and their derivatives (see Holmquist (1996) for details). The symmetrizer for a T-product of q vectors of dimension d is defined as

$$\mathbf{S}_{d\mathbf{1}q} = \frac{1}{q!} \sum_{p \in \mathcal{P}_q} \mathbf{K}_p \tag{2}$$

where $\mathcal{P}_q$ is the set of all permutations of the numbers $1 : q$ and $\mathbf{K}_p$ is the commutator matrix for the permutation $p \in \mathcal{P}_{\text{II}}$, (i.e. the CommutatorMatr with Type="Kperm" of the package). For an application of the symmetrizer via SymMatr with d-variate Hermite polynomials, see formula (4) and Example 3. We defer to Section 4 for a discussion about computational time and memory allocation of these matrices.

2.3 Multivariate T-Hermite polynomials

Consider a Gaussian vector $\mathbf{X}$ of dimension d with $\operatorname{E}\mathbf{X} = 0$ and $\boldsymbol{\Sigma} = \operatorname{Cov}(\mathbf{X}) = \operatorname{E}\mathbf{X}\mathbf{X}'$ and define the generator function

$$\begin{aligned} \psi(\mathbf{X}; \mathbf{a}) &= \exp\left(\mathbf{a}'\mathbf{X} - \frac{1}{2}\mathbf{a}'\boldsymbol{\Sigma}\mathbf{a}\right) \\ &= \exp\left(\mathbf{a}'\mathbf{X} - \frac{1}{2}\boldsymbol{\kappa}_2^{\otimes\prime}\mathbf{a}^{\otimes 2}\right) \end{aligned} \tag{3}$$

where $\mathbf{a}$ is a d-vector of constants and $\boldsymbol{\kappa}_2^{\otimes} = \operatorname{vec}\boldsymbol{\Sigma}$. The vector Hermite polynomials is defined via the T-derivative of the generator function, viz.

$$\mathbf{H}_n(\mathbf{X}) = D_{\mathbf{a}}^{\otimes n}\psi(\mathbf{X}; \mathbf{a})\big|_{\mathbf{a}=0}.$$

For example one has

$$\mathbf{H}_1(\mathbf{X}) = \mathbf{X}, \quad \mathbf{H}_2(\mathbf{X}) = \mathbf{X}^{\otimes 2} - \boldsymbol{\kappa}_2^{\otimes}.$$

Note that the multivariate T-Hermite polynomial $\mathbf{H}_n(\mathbf{X})$ is a vector of dimension d^n which contains the n-th order polynomials of the vector $\mathbf{X}^{\otimes n}$. For example the entries of $\mathbf{H}_2(\mathbf{X})$ are the second order Hermite polynomials $H_2(X_i, X_j)$, $i, j = 1, 2, \ldots d$; for $d = 2$

$$\mathbf{H}_2(\mathbf{X}) = \left((X_1^2 - \sigma_{11}), (X_1X_2 - \sigma_{12}), (X_2X_1 - \sigma_{21}), (X_2^2 - \sigma_{22})\right)'.$$

Note that $\mathbf{H}_n(\mathbf{X})$ is n-symmetric, i.e. $\mathbf{H}_2(\mathbf{X}) = \mathbf{S}_{d\mathbf{1}_n}\mathbf{H}_2(\mathbf{X})$ where $\mathbf{S}_{d\mathbf{1}_n}$ is the symmetrizer defined in (2). From this one can get useful recursion formulae

$$\mathbf{H}_n(\mathbf{X}) = \mathbf{S}_{d\mathbf{1}_n}\left(\mathbf{H}_{n-1}(\mathbf{X}) \otimes \mathbf{X} - (n-1)\mathbf{H}_{n-2}(\mathbf{X}) \otimes \boldsymbol{\kappa}_2^{\otimes}\right). \tag{4}$$

For further details, consult Terdik (2021), Section 4.5.

The definition of the d-variate Hermite polynomial requires the variance-covariance matrix $\mathbf{\Sigma}$ of the vector $\mathbf{X}$. `HermiteN` and `HermiteN2X` with `Type=c("Univariate","Multivariate")` compute respectively the univariate and d-variate Hermite polynomials and their inverses up to a given order `N` evaluated at `x` for a given covariance matrix `Sig2`. By default `Sig2=`$I_{\mathbf{d}}$.

Example 3. Consider a vector x of dimension $d = 2$ and compute the Hermite polynomials up to order `N=3` with identity variance matrix $\mathbf{I}_2$; `HermiteN(x,3,Type="Multivariate")` will contain the list of d-variate Hermite polynomials up to order 3 at value x.

```
x <- c(1, 2)
HN <- HermiteN(x, 3, Type = "Multivariate")
```

Retrieve the third Hermite polynomial

```
HN[[3]]

#> [1] -2  0  0  3  0  3  3  2
```

Use now the symmetrizer in matrix form to verify formula (4) with $n = 2$. Note that in this case $\kappa_2^{\otimes} = \operatorname{vec}\mathbf{I}_2$.

```
k2 <- c(1, 0, 0, 1)
c(SymMatr(2, 3) %*% (kronecker(HN[[2]], x) - (3 - 1) * kronecker(HN[[1]], k2)))

#> [1] -2  0  0  3  0  3  3  2
```

2.4 Multivariate T-moments and T-cumulants

Multivariate moments and cumulants of all orders of a random vector $\mathbf{X}$ in d-dimensions, with mean vector $\boldsymbol{\mu}$ and covariance matrix $\mathbf{\Sigma}$ can be obtained by applying the T-derivative respectively to the characteristic function and the log of the CF. More formally, let $\boldsymbol{\lambda}$ a d-vector of real constants; $\phi_{\mathbf{X}}(\boldsymbol{\lambda})$ and $\psi_{\mathbf{X}}(\boldsymbol{\lambda}) = \log \phi_{\mathbf{X}}(\boldsymbol{\lambda})$ denote, respectively, the characteristic function and the cumulant function of $\mathbf{X}$. Then the k-th order moments and cumulants of the vector $\mathbf{X}$ are obtained as

$$\boldsymbol{\mu}_{\mathbf{X},k}^{\otimes} = (-\mathbf{i})^k D_{\boldsymbol{\lambda}}^{\otimes k} \phi_{\mathbf{X}}(\boldsymbol{\lambda})\big|_{\boldsymbol{\lambda}=0}.$$

$$\boldsymbol{\kappa}_{\mathbf{X},k}^{\otimes} = \underline{\operatorname{Cum}}_k(\mathbf{X}) = (-\mathbf{i})^k D_{\boldsymbol{\lambda}}^{\otimes k} \psi_{\mathbf{X}}(\boldsymbol{\lambda})\big|_{\boldsymbol{\lambda}=0}.$$

Note that $\boldsymbol{\mu}_{\mathbf{X},k} = \operatorname{E}\mathbf{X}^{\otimes k}$ is a vector of dimension d^k that contains all possible moments of order k formed by $X_1, \ldots, X_d$. This approach has the advantage of being straightforwardly extendable to any k-th order moment. An analogous discussion can be done for cumulants. Note that one has $\boldsymbol{\kappa}_{\mathbf{X},2} = \operatorname{vec}\mathbf{\Sigma}$. **MultiStatM** contains functions which obtains moments from cumulants and vice-versa as well as functions which provide theoretical moments and cumulants for some important multivariate distributions.

The `Cum2Mom` and `Mom2Cum` either for the univariate or multivariate cases provide conversion formulae for cumulants from moments and viceversa given any list of moments (or cumulants).

Example 4. Consider the case of the 2-variate standard normal distribution with unit mean vector and covariance matrix with unit elements on the main diagonal and off-diagonal elements equal to 0.5; in this case the first four moments are given in the vector `mu` below

```
mu <- list(
  c(1, 1), c(2, 1.5, 1.5, 2), c(4, 3, 3, 3, 3, 3, 3, 4),
  +c(10, 7, 7, 6.5, 7, 6.5, 6.5, 7, 7, 6.5, 6.5, 7, 6.5, 7, 7, 10)
)
cum <- Mom2Cum(mu, Type = "Multivariate")
cum

#> [[1]]
#> [1] 1 1
#>
#> [[2]]
#> [1] 1.0 0.5 0.5 1.0
#>
```

```
#> [[3]]
#> [1] 0 0 0 0 0 0 0 0
#>
#> [[4]]
#>  [1] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

Getting back to moments

```
Cum2Mom(cum, Type = "Multivariate")
```

```
#> [[1]]
#> [1] 1 1
#>
#> [[2]]
#> [1] 2.0 1.5 1.5 2.0
#>
#> [[3]]
#> [1] 4 3 3 3 3 3 3 4
#>
#> [[4]]
#>  [1] 10.0  7.0  7.0  6.5  7.0  6.5  6.5  7.0  7.0  6.5  6.5  7.0  6.5  7.0  7.0
#> [16] 10.0
```

If one wishes to select only the distinct moments from the vector of third moments, then

```
r.mu <- mu[[3]][EliminIndx(2, 3)]
r.mu
```

```
#> [1] 4 3 3 4
```

Note that EliminIndx (and EliminMatr) does not correspond to the function unique, rather it individuates the distinct elements from the symmetry of the Kronecker product. Here the values 4 and 3 could appear to be duplicates however in general they could be different. This allows to recover the whole vector when needed by using QplicIndx (or QplicMatr)

```
mu3 <- r.mu[QplicIndx(2, 3)]
mu3
```

```
#> [1] 4 3 3 3 3 3 3 4
```

The MomCum functions provide theoretical moments and cumulants for some common multivariate distributions: Skew-normal, Canonical Fundamental Skew-normal (CFUSN), Uniform distribution on the sphere, central folded Normal distribution (univariate and multivariate); for detail on the multivariate formulae used see Jammalamadaka et al. (2021b). Some more details on the multivariate distributions considered are reported in the list below.

- A d-vector $\mathbf{U}$ having uniform distribution on the sphere $\mathbb{S}_{d-1}$. Moments and cumulants of all orders are provided for $\mathbf{U}$ by the function MomCumUniS; the function EVSKUniS can compute moments and cumulants (up to the 4th order), skewness, and kurtosis of $\mathbf{U}$ (Type="Standard") and its modulus (Type="Modulus"). Recall that any d-vector, say $\mathbf{W}$, has a spherically symmetric distribution if that distribution is invariant under the group of rotations in $\mathbb{R}^d$. This is equivalent to saying that $\mathbf{W}$ has the stochastic representation $\mathbf{W} = R\mathbf{U}$ where R is a non negative random variable. Moments and cumulants of $\mathbf{W}$ can be obtained by its stochastic representation as discussed in Jammalamadaka et al. (2021b), Theorem 1 and Lemma 1. Furthermore a d-vector $\mathbf{X}$ has an elliptically symmetric distribution if it has the representation

$$\mathbf{X} = \boldsymbol{\mu} + \boldsymbol{\Sigma}^{1/2}\mathbf{W},$$

where $\boldsymbol{\mu} \in \mathbb{R}^d$, $\boldsymbol{\Sigma}$ is a variance-covariance matrix and $\mathbf{W}$ is spherically distributed. Hence the cumulants of $\mathbf{X}$ are just constant times the cumulants of $\mathbf{W}$ except for the mean i.e.

$$\underline{\mathrm{Cum}}_m\left(\mathbf{X}\right) = \left(\boldsymbol{\Sigma}^{1/2}\right)^{\otimes m} \underline{\mathrm{Cum}}_m\left(\mathbf{W}\right).$$

- If $\mathbf{Z}$ denotes a d-vector with d-variate standard normal distribution, the function `MomCumZabs` provides the moments and cumulants of $|\mathbf{Z}|$ respectively in the univariate (`Type="univariate"`) and multivariate case (`Type="Multivariate"`).
- The multivariate skew-normal distribution introduced by Azzalini and Dalla Valle (1996), whose marginal densities are scalar skew-normals. A d-dimensional random vector $\mathbf{X}$ is said to have a multivariate skew-normal distribution, $\mathrm{SN}_d\left(\boldsymbol{\mu}, \boldsymbol{\Omega}, \boldsymbol{\alpha}\right)$ with shape parameter $\boldsymbol{\alpha}$ if it has the density function
$$2\varphi\left(\mathbf{X}; \boldsymbol{\mu}, \boldsymbol{\Omega}\right) \Phi\left(\boldsymbol{\alpha}'\left(\mathbf{X} - \boldsymbol{\mu}\right)\right), \quad \mathbf{X} \in \mathbb{R}^d,$$
where $\varphi\left(\mathbf{X}; \boldsymbol{\mu}, \boldsymbol{\Omega}\right)$ is the d-dimensional normal density with mean $\boldsymbol{\mu}$ and correlation matrix $\boldsymbol{\Omega}$; here φ and Φ denote the univariate standard normal density and the cdf. For a general formula for cumulants, see Jammalamadaka et al. (2021b), Lemma 4. For this distribution are available the functions `MomCumSkewNorm`, which computes the theoretical values of moments and cumulants up to the r-th order and `EVSKSkewNorm` which gives mean vector, covariance, skewness and kurtosis vectors and other features.
- Arellano-Valle and Genton (2005) introduced the CFUSN distribution (cf. their Proposition 2.3), to include all existing definitions of skew-normal distributions. The marginal stochastic representation of $\mathbf{X}$ with distribution $\mathrm{CFUSN}_{d,m}\left(\boldsymbol{\Delta}\right)$ is given by
$$\mathbf{X} = \boldsymbol{\Delta}\left|\mathbf{Z}_1\right| + \left(\mathbf{I}_d - \boldsymbol{\Delta}\boldsymbol{\Delta}'\right)^{1/2}\mathbf{Z}_2,$$
where $\boldsymbol{\Delta}$, is the $d \times m$ skewness matrix such that $\|\boldsymbol{\Delta}\mathbf{a}\| < 1$, for all $\|\mathbf{a}\| = 1$, and $\mathbf{Z}_1 \in \mathcal{N}\left(0, \mathbf{I}_m\right)$ and $\mathbf{Z}_2 \in \mathcal{N}\left(0, \mathbf{I}_d\right)$ are independent (Proposition 2.2. Arellano-Valle and Genton (2005)). A simple construction of $\boldsymbol{\Delta}$ is $\boldsymbol{\Delta} = \boldsymbol{\Lambda}\left(\mathbf{I}_m + \boldsymbol{\Lambda}'\boldsymbol{\Lambda}\right)^{-1/2}$ with some real matrix $\boldsymbol{\Lambda}$ with dimensions $d \times m$. The $\mathrm{CFUSN}_{d,m}\left(\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\Delta}\right)$ can be defined via the linear transformation $\boldsymbol{\mu} + \boldsymbol{\Sigma}^{1/2}\mathbf{X}$. For a general formula for cumulants, see Jammalamadaka et al. (2021b), Lemma 5. For this distributions `MomCumCFUSN` provides moments and cumulants up to the r-th order.

Example 5. For a skew-normal distribution with $\alpha = (10, 5, 0)$ and correlation function $\Omega = \mathrm{diag}(1, 1, 1)$ the third moments and cumulants are

```
alpha <- c(10, 5, 0)
omega <- diag(3)
MSN <- MomCumSkewNorm(r = 3, omega, alpha, nMu = TRUE)
round(MSN$Mu[[3]], 3)

#>  [1] 1.568 0.073 0.000 0.073 0.570 0.000 0.000 0.000 0.711 0.073 0.570 0.000
#> [13] 0.570 0.996 0.000 0.000 0.000 0.355 0.000 0.000 0.711 0.000 0.000 0.355
#> [25] 0.711 0.355 0.000

round(MSN$CumX[[3]], 3)

#>  [1] 0.154 0.077 0.000 0.077 0.039 0.000 0.000 0.000 0.000 0.077 0.039 0.000
#> [13] 0.039 0.019 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
#> [25] 0.000 0.000 0.000
```

As another example, for the uniform distribution on the sphere, the kurtosis vector is:

```
EVSKUniS(3, Type = "Standard")$Kurt.U

#>  [1] -1.2  0.0  0.0  0.0 -0.4  0.0  0.0  0.0 -0.4  0.0 -0.4  0.0 -0.4  0.0  0.0
#> [16]  0.0  0.0  0.0  0.0  0.0 -0.4  0.0  0.0  0.0 -0.4  0.0  0.0  0.0 -0.4  0.0
#> [31] -0.4  0.0  0.0  0.0  0.0  0.0 -0.4  0.0  0.0  0.0 -1.2  0.0  0.0  0.0 -0.4
#> [46]  0.0  0.0  0.0  0.0  0.0 -0.4  0.0 -0.4  0.0  0.0  0.0 -0.4  0.0  0.0  0.0
#> [61] -0.4  0.0  0.0  0.0  0.0  0.0  0.0  0.0 -0.4  0.0 -0.4  0.0 -0.4  0.0  0.0
#> [76]  0.0 -0.4  0.0  0.0  0.0 -1.2
```

2.5 Analyzing multivariate skewness and kurtosis

A complete picture of multivariate skewness is provided by the third-order T-cumulant (skewness vector) of a standardized $\mathbf{X}$; set $\mathbf{Y} = \boldsymbol{\Sigma}^{-1/2}(\mathbf{X} - \boldsymbol{\mu})$, then the skewness vector is

$$\boldsymbol{\kappa}_{\mathbf{Y},3}^{\otimes} = \underline{\mathrm{Cum}}_3\left(\mathbf{Y}\right) = \left(\boldsymbol{\Sigma}^{-1/2}\right)^{\otimes 3}\boldsymbol{\kappa}_{\mathbf{X},3}^{\otimes}. \tag{5}$$

A definition of skewness which is invariant under shifting and orthogonal transformations or, in other words affine invariant, is provided by the total skewness of $\mathbf{X}$, i.e. the square norm of the skewness vector:

$$\gamma_{1,d} = \|\boldsymbol{\kappa}^{\otimes}_{\mathbf{Y},3}\|^2 = \beta_{1,d} \tag{6}$$

where $\beta_{1,d}$ is Mardia's multivariate skewness index (Mardia, 1970) which coincides with the total skewness $\gamma_{1,d}$ since the third-order central moments and third-order cumulants are equal. Móri et al. (1994) (MRSz's) skewness vector can also be recovered from the skewness vector as

$$\mathbf{b}(\mathbf{Y}) = \left(\operatorname{vec}' \mathbf{I}_d \otimes \mathbf{I}_d\right) \boldsymbol{\kappa}^{\otimes}_{\mathbf{Y},3}. \tag{7}$$

Note that $\operatorname{vec}' \mathbf{I}_d \otimes \mathbf{I}_d$ is a matrix of dimensions $d \times d^3$, which contains d unit values per-row, whereas all the others are 0; as a consequence, $\mathbf{b}(\mathbf{Y})$ does not take into account the contribution of cumulants of the type $\operatorname{Cum}_3(X_j, X_k, X_l)$, where all the three indices j, k, l are different from each other. The corresponding scalar measure of multivariate skewness is $b(\mathbf{Y}) = \|\mathbf{b}(\mathbf{Y})\|^2$.

The fourth-order T-cumulant of the standardized $\mathbf{X}$, i.e. $\boldsymbol{\kappa}^{\otimes}_{\mathbf{Y},4} = \mathbf{H}_4(\mathbf{Y})$, will be called kurtosis vector of $\mathbf{X}$; its square norm will be called the total kurtosis of $\mathbf{X}$,

$$\gamma_{2,d} = \|\boldsymbol{\kappa}^{\otimes}_{\mathbf{Y},4}\|^2. \tag{8}$$

Mardia's kurtosis index $\beta_{2,d} = \operatorname{E}\left(\mathbf{Y}'\mathbf{Y}\right)^2$ (Mardia (1970)) is related to the kurtosis vector by the formula

$$\beta_{2,d} = \left(\operatorname{vec}' \mathbf{I}_{d^2}\right) \boldsymbol{\kappa}^{\otimes}_{\mathbf{Y},4} + d(d+2).$$

A consequence of this is that Mardia's measure does not depend on all the entries of $\boldsymbol{\kappa}^{\otimes}_{\mathbf{Y},4}$ which has $d(d+1)(d+2)(d+3)/24$ distinct elements, while $\beta_{2,d}$ includes only d^2 elements among them. We note that if $\mathbf{X}$ is Gaussian, then $\boldsymbol{\kappa}^{\otimes}_{\mathbf{Y},4} = \mathbf{0}$. Note that Koziol (1989), for $\mathbf{Y}$ and an independent copy $\tilde{\mathbf{Y}}$, considered $\operatorname{E}(\mathbf{Y}'\tilde{\mathbf{Y}})^4$ as an index of kurtosis which is the next higher degree analogue of Mardia's skewness index $\beta_{1,d}$. Note that (see Jammalamadaka et al. (2021b))

$$\operatorname{E}(\mathbf{Y}'\tilde{\mathbf{Y}})^4 = \gamma_{2,d} + 6\beta_{2,d} - d^2.$$

Móri et al. (1994) define what we will call the MRS kurtosis matrix

$$\mathbf{B}(Y) = \operatorname{E}\left(\mathbf{Y}\mathbf{Y}'\mathbf{Y}\mathbf{Y}'\right) - (d+2)\mathbf{I}_d,$$

which can be expressed in terms of the kurtosis vector as (note also that $\operatorname{tr}\mathbf{B}(Y) = \beta_{2,d}$),

$$\operatorname{vec}\mathbf{B}(Y) = \left(\mathbf{I}_{d^2} \otimes \operatorname{vec}' \mathbf{I}_d\right) \boldsymbol{\kappa}^{\otimes}_{\mathbf{Y},4}.$$

$\mathbf{B}(Y)$ got a prominent role in multivariate statistics due to invariant coordinate selection (Tyler et al. (2009)), model-based clustering (Peña et al. (2010)), and multivariate outlier detection (Archimbaud et al. (2018)). For further connections between $\boldsymbol{\kappa}^{\otimes}_{\mathbf{Y},3}$, $\boldsymbol{\kappa}^{\otimes}_{\mathbf{Y},4}$ and other measures of skewness and kurtosis, see Terdik (2021), Section 6 and Jammalamadaka et al. (2021b). For Malkovich and Afifi (1973), where directional measures of multivariate skewness and kurtosis measures are proposed, one can refer to **MaxSkew**, **Kurt** and Loperfido (2015) for discussions and applications; see also Arevalillo and Navarro (2020) and Arevalillo and Navarro (2021).

For examples on computing skewness and kurtosis measures with **MultiStatM** and relate tests see the following sections.

2.6 Estimation

Estimating functions starting from a vector of multivariate data are available: multivariate Hermite polynomials, moments, cumulants, skewness and kurtosis vectors, Mardia's skewness and kurtosis indexes, MRSz's skewness vector and kurtosis matrices. The `Estimation` family of functions include `SampleEVSK`, `SampleGC`, `SampleHermiteN`, `SampleKurt`, `SampleSkew`, `SampleMomCum`, see the examples in this and the next section for their application.

The basic estimation method relies on sample averages: given a random sample of d-vectors $\mathbf{X}_1, \ldots, \mathbf{X}_n$, for any function g let

$$\widehat{g(\mathbf{X})} = \frac{1}{n}\sum_{i=1}^{n} g(\mathbf{X}_i). \tag{9}$$

To estimate moments of any order $k = 1, \ldots, r$ the function `SampleMomCum` sets $g(\mathbf{X}_i) = \mathbf{X}_i^{\otimes k}$ in (9), while cumulants are obtained from moments via the function `Mom2Cum`. Options for centering, i.e. $(\mathbf{X_i} - \bar{\mathbf{X}})$

or scaling (default), i.e. $\mathbf{S}^{-1/2}(\mathbf{X_i} - \bar{\mathbf{X}})$, where $\bar{\mathbf{X}}$ and $\mathbf{S}$ are the usual sample estimates of mean and variance, are provided. For any sample $\mathbf{X}_1, \ldots, \mathbf{X}_n$ the mean zero and identity covariance matrix data $\mathbf{Z}_i = \mathbf{S}^{-1/2}(\mathbf{X_i} - \bar{\mathbf{X}})$ can always be obtained using `MultiStatM::MVStandardize`.

d-variate Hermite polynomials of any order $k = 1, \ldots, N$ are evaluated at standardized data points setting $g(\mathbf{X}_i) = \mathbf{H}_k(\mathbf{X}_i)$ in (9). The estimators in terms of standard Hermite polynomials proved to be useful for finding the asymptotic distribution of skewness and kurtosis estimates, see Jammalamadaka et al. (2021a). For estimation of skewness and kurtosis vectors, see Terdik (2021) (6.20), p. 331 and (6.28), p. 340. Formulae for evaluating the variance of estimators are also available. For example `SampleVarianceSkewKurt` provides estimates of the covariance matrix of the data-estimated skewness and kurtosis vectors; see (Terdik (2021), (6.13) and (6.22)).

MultiStatM does not make use of multivariate k-statistics in estimation. For these one can use **kStatistics**; for tutorials see Di Nardo and Guarino (2022b) and Smith (2020).

Example 6. Consider a multivariate data vector of dimension $d = 3$ and $n = 1000$ from the multivariate skew-normal distribution of Example 5. The estimated first four cumulants are listed in the object `EsMSN` obtained by `SampleEVSK`; the corresponding theoretical values are in the object `ThMSN` obtained by `MomCumSkewNorm`.

```
data <- rSkewNorm(1000, omega, alpha)
EsMSN <- SampleEVSK(data)
ThMSN <- EVSKSkewNorm(omega, alpha)
```

Compare the distinct elements of the estimated skewness vector and the theoretical ones using `EliminIndx`.

```
EsMSN$estSkew[EliminIndx(3, 3)]

#>  [1]  0.560791793  0.314183327 -0.056347906  0.234568223 -0.017589732
#>  [6]  0.007504290  0.015390429 -0.013230420  0.003719084 -0.064462086

ThMSN$SkewX[EliminIndx(3, 3)]

#>  [1] 0.68927167 0.34463583 0.00000000 0.17231792 0.00000000 0.00000000
#>  [7] 0.08615896 0.00000000 0.00000000 0.00000000
```

If one wishes to recover the estimated univariate skewness and kurtosis of the univariate components X_1, X_2 and X_3 of X, then, using `UnivMomCum`,

```
EsMSN$estSkew[UnivMomCum(3, 3)]

#> [1]  0.56079179  0.01539043 -0.06446209

EsMSN$estKurt[UnivMomCum(3, 4)]

#> [1]  0.13840605  0.04696035 -0.14484549
```

An estimate of Mardia's skewness index is provided together with the p-value under the null hypothesis of normality. The theoretical value of Mardia's skewness can be recovered from the element `SkewX.tot` in the object `ThMSN`.

```
SampleSkew(data, Type = "Mardia")

#> $Mardia.Skewness
#> [1] 0.7921971
#>
#> $p.value
#> [1] 1.796947e-23

ThMSN$SkewX.tot

#> [1] 0.9279208
```

The MRSz skewness vector and index are provided together with the p-value for the skewness index under the null hypothesis of normality. The theoretical value, for the distribution at hand, can be computed using formula (7).

```
SampleSkew(data, Type = "MRSz")

#> $MRSz.Skewness.Vector
#> [1]  0.8028643  0.3332928 -0.1340404
#>
#> $MRSz.Skewness.Index
#> [1] 0.773642
#>
#> $p.value
#> [1] 1.127964e-16

c(t(c(diag(3)) %x% diag(3)) %*% ThMSN$SkewX)

#> [1] 0.8615896 0.4307948 0.0000000
```

3 Applications

3.1 Weighted measures of multivariate skewness and kurtosis

The skewness and kurtosis vectors, respectively $\mathbf{H}_q(\mathbf{Y})$, $q = 3, 4$, contain d^q elements, which are not all distinct. Just as the covariance matrix of a d-dimensional vector contains only $d(d+1)/2$ distinct elements, a simple computation shows that $\mathbf{H}_q(\underline{Y})$ contains $\binom{d+q-1}{q}$ distinct elements. To avoid duplicated information, it makes sense to work only with these *distinct* elements of the cumulant vector. The selection of distinct elements can be accomplished via linear transformations and through the so-called elimination matrix which we denote here as $\mathbf{E}^{+}_{d,q}$. The linear transformation

$$\mathbf{H}_{q,D}\left(\mathbf{Y}\right) = \mathbf{E}^{+}_{d,q}\mathbf{H}_{q}\left(\mathbf{Y}\right) \tag{10}$$

contains only the distinct values of $\mathbf{H}_q(\mathbf{Y})$ and has covariance matrix

$$\mathbf{C}_{\mathbf{H}_{q,D}} = \mathbf{E}^{+}_{d,q}\mathbf{C}_{\mathbf{H}_q}\mathbf{E}^{+\top}_{d,q}. \tag{11}$$

Given the availability of explicit formulae for the covariance matrices, it is natural to consider the weighted skewness and kurtosis measures based on $\boldsymbol{\kappa}_{\mathbf{Y}^{\otimes},q,D} = \mathbf{E}^{+}_{d,q}\boldsymbol{\kappa}_{\mathbf{Y}^{\otimes},q}$, $q = 3, 4$ the third and fourth order cumulant vectors with distinct elements, respectively; denote these as $\beta_{1,T,d}$ (skewness) and $\beta_{2,T,d}$ (kurtosis); then

$$\beta_{q-2,T,d} = \left\|\mathbf{C}^{-1/2}_{\mathbf{H}_{q,D}}\boldsymbol{\kappa}_{\mathbf{Y}^{\otimes},q,D}\right\|^{2}, \quad q = 3, 4. \tag{12}$$

The corresponding sample versions, $b_{q-2,T,d}$, $q = 3, 4$, are simply obtained by replacing the cumulant vectors and the covariance with their estimated versions. See Jammalamadaka et al. (2021a) for detailed results about their asymptotic distributions which, under mild conditions are non-central χ^2 distributions with $\binom{d+q-1}{q}$ degrees of freedom and non-centrality parameter $\beta_{q-2,T,d}$, $q = 3, 4$.

Since the distribution of $b_{1,T,d}$ with estimated covariance matrix is asymptotically a χ^2 with $\binom{d+2}{3}$ degrees of freedom either in the Gaussian or in the more general case of elliptically symmetric multivariate distributions, a consistent test for elliptical symmetry can be based on $b_{1,T,d}$ with the estimated covariance $\mathbf{C}_{\mathbf{H}_{3,D}}$. Under the alternative hypothesis of asymmetry, the asymptotic distribution will involve a non-centrality parameter $\beta_{1,T,d}$.

Example 7. Suppose we have 3-variate data from a normal distribution with unknown mean and covariance matrix. We want to estimate $b_{1,T,d}$ from formula (12). Since we are considering the skewness vector $\mathbf{H}_3(\mathbf{Y})$, without loss of generality we can consider a null mean vector and $\boldsymbol{\Sigma} = \mathbf{I}_d$. Hence, in order to compute the covariance matrix of $\mathbf{H}_3(\mathbf{Y})$ we use `HermiteCov12`:

```
d <- 3
Cov <- diag(d)
CovH3 <- HermiteCov12(Cov, 3)
```

From formula (11) one can use `EliminMatr` to compute the covariance (and its inverse) of the distinct vectors elements of $\mathbf{Y}$ as

```
ME <- EliminMatr(3, 3)
CovH3D <- ME %*% CovH3 %*% t(ME)
InvCovH3D <- solve(CovH3D)
```

Given a sample of data, compute $b_{1,T,3}$. First estimate the skewness vector,

```
x <- MASS::mvrnorm(150, rep(0, d), diag(d))
EH3 <- SampleHermiteN(x, 3)
```

As a second step obtain the skewness vector with only distinct elements

```
EH3D <- EH3[EliminIndx(d, 3)]
```

In the third step we estimate $b_{1,T,3}$

```
b1T <- EH3D %*% InvCovH3D %*% EH3D
b1T
```

```
#>            [,1]
#> [1,] 0.04174866
```

If we compare $b_{1,T,3}$ with Mardia's skewness index $b_{1,3}$,

```
b1 <- SampleSkew(x, Type = "Mardia")$Mardia.Skewness
b1
```

```
#> [1] 0.250492
```

We note that $b_{1,3} = 6 \cdot b_{1,T,3}$. This fact was already noted and formalized in Jammalamadaka et al. (2021a), i.e. Mardia's index can also be seen to be equivalent, up to a constant, to the weighted index $b_{1,T,d}$ for the Gaussian case.

One can also build a weighted indexes for more general cases, e.g. for an elliptically symmetric distribution, by computing the covariance matrices for $\mathbf{H}_3(\mathbf{Y})$ and $\mathbf{H}_4(\mathbf{Y})$ using the results of Theorem 1 in Jammalamadaka et al. (2021a) and the functions provided in **MultiStatM**.

3.2 A multivariate Gram-Charlier density

The Gram-Charlier expansion expresses the density function of a random variable in terms of its cumulants (see e.g. Brenn and Anfinsen (2017) and Section 4.7 in Terdik (2021)).

Given a d-vector from some distribution with density $f_{\mathbf{X}}(\mathbf{x})$ having mean $\mathbf{0}$ and covariance matrix $\boldsymbol{\Sigma}$, under appropriate moment conditions, the density can be expanded into a Gram-Charlier series as follows

$$f_{\mathbf{X}}(\mathbf{x}) = \left(1 + \sum_{k=3}^{\infty} \frac{1}{k!} \left(\mathbf{B}_k\left(0, 0, \boldsymbol{\kappa}_{\mathbf{Y},3}, \ldots \boldsymbol{\kappa}_{\mathbf{Y},k}\right)\right)^{\top} \mathbf{H}_k\left(\boldsymbol{\Sigma}^{-1/2}\left(\mathbf{x} - \boldsymbol{\mu}\right)\right)\right) \varphi\left(\mathbf{x}; \mathbf{0}, \boldsymbol{\Sigma}\right) \tag{13}$$

with φ denoting the d-variate standard normal density function and $\mathbf{B}_k$ indicating the (complete) T-Bell polynomial of k-th order (see, e.g. Terdik (2021), Section 1.4.2 and Section 4.7). The polynomials $\mathbf{B}_3$, $\mathbf{B}_4$, and $\mathbf{B}_5$ have the following simple forms

$$\begin{aligned}
\mathbf{B}_3\left(0, 0, \boldsymbol{\kappa}^{\otimes}_{\mathbf{Y},3}\right) &= \boldsymbol{\kappa}^{\otimes}_{\mathbf{Y},3}, \\
\mathbf{B}_4\left(0, 0, \boldsymbol{\kappa}^{\otimes}_{\mathbf{Y},3}, \boldsymbol{\kappa}^{\otimes}_{\mathbf{Y},4}\right) &= \boldsymbol{\kappa}^{\otimes}_{\mathbf{Y},4}, \\
\mathbf{B}_5\left(0, 0, \boldsymbol{\kappa}^{\otimes}_{\mathbf{Y},3}, \boldsymbol{\kappa}^{\otimes}_{\mathbf{Y},4}, \boldsymbol{\kappa}^{\otimes}_{\mathbf{Y},5}\right) &= \boldsymbol{\kappa}^{\otimes}_{\mathbf{Y},5}.
\end{aligned}$$

Using the above results and considering only the first few order terms in the expansion (13), one can provide an approximation to $f_{\mathbf{X}}(\mathbf{x})$:

$$f_{\mathbf{X}}(\mathbf{x}) = \left(1 + \sum_{k=3}^{4} \frac{1}{k!} \boldsymbol{\kappa}^{\top}_{\mathbf{Y},k} \mathbf{H}_k\left(\boldsymbol{\Sigma}^{-1/2}(\mathbf{x} - \boldsymbol{\mu})\right)\right) \varphi\left(\mathbf{x}, \mathbf{0}, \boldsymbol{\Sigma}\right) + \mathcal{O} \tag{14}$$

where $\mathcal{O}$ includes the remaining terms. Starting from d-variate data, a Gram-Charlier approximation to an unkonw multivariate density can be constructed using the multivariate function for cumulants and Hermite polynomials provided in **MultiStatM**. In the package, the function `SampleGC` is available in order to estimate the truncated Gram-Charlier density for k from 3 to 8. For higher order generalized multivariate Bell polynomials see **kstatistics**. `SampleGC` can estimate directly the needed cumulants from the data or use a given list of cumulants to evaluate the Gram-Charlier density at the given data. If large data is used it will be more efficient to estimate the needed cumulants using `SampleMomCum` and then evaluate the density at the points needed for analysis; note that in (14) $\mathbf{\Sigma}$ refers to $\mathbf{X}$ while the higher order cumulants are computed for the standardized data $\mathbf{Y}$. These aspects can be handled efficiently with `SampleMomCum` which provides options for centering and scaling of the data.

Example 8. For $n = 500$ randomly generated data from a bivariate skew-normal distribution with $\alpha = (4, 12)$ and correlation function $\Omega = \text{diag}(1, 1)$, the Gram-Charlier approximation (for $k = 4$ and $k = 6$) is evaluated: a) using the cumulants estimated from data; b) using the true cumulants of the given skew-normal. For this distribution the mean can be computed as $\sqrt{2/\pi}\,\Omega\alpha\,(1 + \alpha'\Omega\alpha)^{-1/2}$.

```
alpha <- c(4, 12)
omega <- diag(2)
set.seed(1234)
X <- rSkewNorm(500, omega, alpha)
X <- X - sqrt(2 / pi) * c(omega %*% alpha) / c(sqrt(1 + alpha %*% omega %*% alpha))
```

For k all the needed cumulants (in proper form as argument for `SampleGC`) can be efficiently estimated using (e.g. $k = 4$)

```
EC <- SampleMomCum(X, r = 4, centering = TRUE, scaling = FALSE)
```

The theoretical cumulants (up to any order) can be obtained using `MomCumSkewNorm`.

```
TC <- MomCumSkewNorm(r = 4, omega, alpha, nMu = FALSE)
```

Compare directly the theoretical and data-estimated skewness and kurtosis vectors (use only the distinct values) in the code below

```
EC$estCum.r[[3]][EliminIndx(2, 3)]

#> [1]  0.215266065 -0.007213623  0.032072834  0.207076236

TC$CumX[[3]][EliminIndx(2, 3)]

#> [1] 0.006830064 0.020490192 0.061470576 0.184411729

EC$estCum.r[[4]][EliminIndx(2, 4)]

#> [1] -0.028502396  0.047460791  0.010992685  0.001601845  0.010988540

TC$CumX[[4]][EliminIndx(2, 4)]

#> [1] 0.001133494 0.003400482 0.010201445 0.030604334 0.091813003
```

In order to produce the plots in Figure 1, the Gram-Charlier approximation with $k = 4$ (and $k = 6$) are evaluated using the estimated cumulants and the true ones; for example:

```
fy4 <- SampleGC(X, cum = EC$estCum.r)
```

Figure 1 shows the contours of true skew-normal density and the Gram-Charlier densities based on the estimated and true cumulants for the cases $k = 4$ and $k = 6$. The approximation, as expected, appears to improve when true cumulants are used and k is larger.

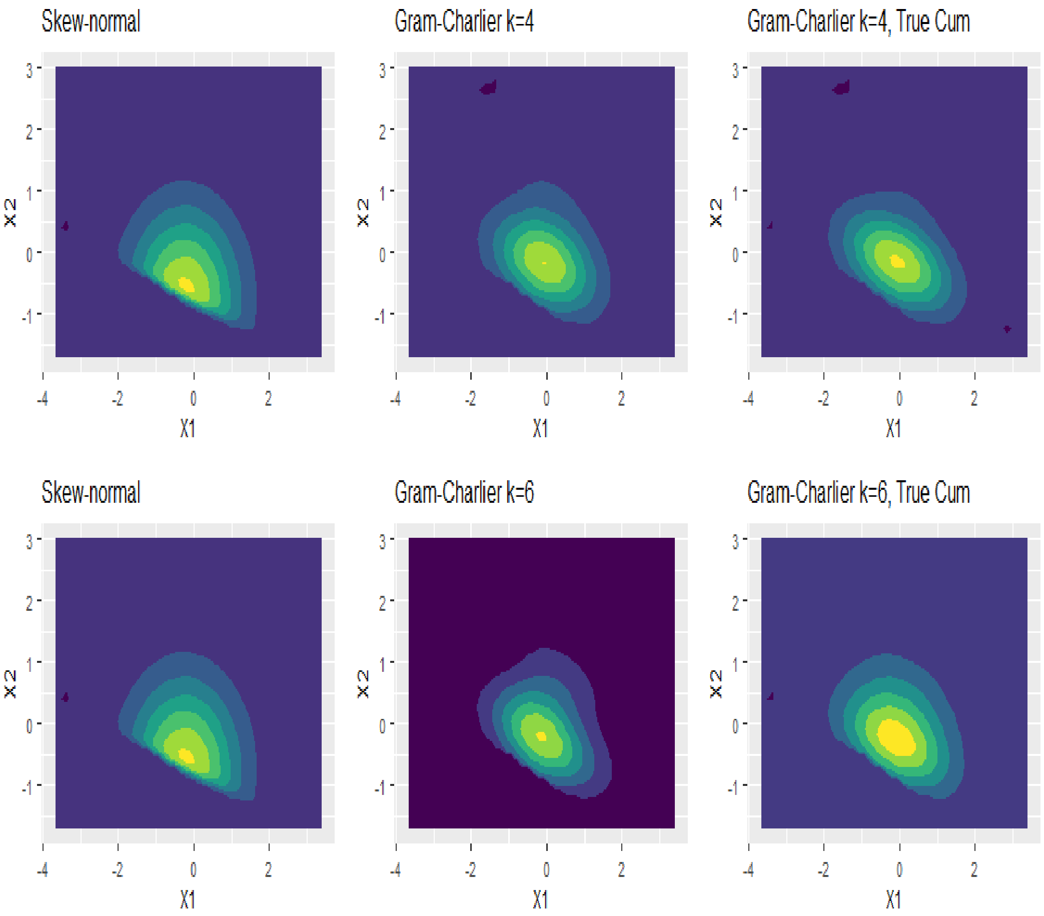

Figure 1: Contour plots of the skew-normal true density (left), the Gram-Charlier approximation with estimated cumulants (center) and the Gram-Charlier approximation with true cumulants (right). Case with $k = 4$ (first row) and $k = 6$ (second row).

4 Computational details

One of the basic functions of **MultiStatM** is `PartitionTypeAll` (Section 2.1) whose algorithm to generate all partitions of the set $1:N$ is based on the recursive generation of inclusive and exclusive partition matrices as described in Terdik (2021), section 1.4. For example, the number of partitions of a given type `S_r_j` and `eL_r` - see Section 2.1 produced by `PartitionTypeAll` are exploited in conversion formulae from moments to cumulants and viceversa. Recall that the conversion formula from moments to cumulants (Terdik (2021), Section 3.4)) is given by

$$\begin{aligned}\boldsymbol{\mu}_n^{\otimes} &= \sum_{\mathcal{K}\in\mathcal{P}_n} \mathbf{K}^{-1}_{p(\mathcal{K})} \prod_{b_j\in\mathcal{K}}^{\otimes} \kappa^{\otimes}_{|b_j|} \\ &= \mathbf{S}_{d\mathbf{1}_n}\left(\sum_{r=1}^{n}\sum_{\sum l_j=r,\sum jl_j=n}\frac{n!}{\prod_{j=1}^{n} l_j!(j!)^{l_j}}\prod_{j=1:n-r+1}^{\otimes}\kappa_j^{\otimes l_j}\right)\end{aligned} \tag{15}$$

where the summation is over all partitions $\mathcal{K}=\{b_1,b_2,\ldots,b_k\}$ of $1:n$; $|b_j|$ denotes the cardinality of block b_j. The simpler second formula, exploiting the symmetrizer matrix, derives from symmetry of $\boldsymbol{\mu}_n^{\otimes}$.

As far as the formula from cumulants to moments (Terdik (2021), Section 3.4) is concerned,

$$\boldsymbol{\kappa}_n^{\otimes} = \mathbf{S}_{d\mathbf{1}_n}\left(\sum_{r=1}^{n}(-1)^{r-1}(r-1)!\sum_{\sum l_j=r,\sum jl_j=n}\prod_{j=1:n-r+1}^{\otimes}\frac{1}{l_j!}\left(\frac{1}{j!}\boldsymbol{\mu}_j^{\otimes}\right)^{l_j}\right). \tag{16}$$

One can also use T-Bell polynomials $\mathbf{B}_n$, and incomplete T-Bell polynomial $\mathbf{B}_{n,r}$ (see, e.g. Section 1.4.2 and Section 4.7 in Terdik (2021)) to show the structure of partitions in (15) as

$$\boldsymbol{\mu}_n^{\otimes} = \mathbf{S}_{d\mathbf{1}_n}\mathbf{B}_n\left(\boldsymbol{\kappa}_1^{\otimes},\boldsymbol{\kappa}_2^{\otimes},\ldots,\boldsymbol{\kappa}_n^{\otimes}\right) \tag{17}$$

and in (16),

$$\boldsymbol{\kappa}_n^{\otimes} = \mathbf{S}_{d\mathbf{1}_n}\sum_{r=1}^{n}(-1)^{r-1}(r-1)!\mathbf{B}_{n,r}\left(\boldsymbol{\mu}_1^{\otimes},\ldots,\boldsymbol{\mu}_n^{\otimes}\right). \tag{18}$$

Consider now the symmetrizer matrix (Section 2.2), note that, by definition, its computation requires $q!$ operations; in the package, the computational complexity is overcome by exploiting the Chacón and Duong (2015) efficient recursive algorithms for functionals based on higher order derivatives and the prime factorization theorem. Note however that even though the computational requirements are greatly reduced, the symmetrizer is a square matrix of dimension d^n where d is the vector dimension and n the power of the Kronecker products. Since `SymMatr` produces essentially sparse matrices, the option `useSparse=T,F` can be selected (`F` is the default). Overall however, if the use of matrices is not strictly necessary, the function `SymIndx` will directly produce a symmetrized version of the T-product of vectors. Table 1 reports the computational times as well as memory allocation of the objects created. Computation times are approximate and obtained on a desktop computer with an Intel-Core-i7-7700 CPU at 3.60GHz and 16GB RAM. From the table it can be noted that using sparse matrices can be much faster and more efficient in terms of memory allocation. The use of `SymIndx` should be the standard way however, as it much less demanding in terms of memory allocation. Recall Example 3 for an instance where the use of `SymMatr` is preferable. The numerical results are in line with those of Chacón and Duong (2015).

In general the functions of the group `Indx` should be used wherever possible, as they are typically much faster than the corresponding `Matr` group since the former avoids constructing possibly large matrices. All code in **MultiStatM** wherever possible, exploits the `Indx` group of functions.

Table 1: Approximate memory allocation (in Kb or Mb) and computational times in secs (s) or mins (m) of `SymMatr` and `SymIndx` under different settings. For $d=3$, n indicates the power of the Kronecker products.

Function	**n**	**Time**	**Object size**
SymMatr()	4	0.003s	55Kb
useSparse=F	6	0.720s	4Mb
	8	9.2m	344Mb
SymMatr()	4	0.03s	9Kb
useSparse=T	6	0.12s	0.4Mb
	8	0.46s	25Mb

Function	n	Time	Object size
Symindx()	4	0.06	0.6Kb
	6	0.02	0.6Kb
	8	0.06	52Kb

A further discussion is needed for `CommutatorMatr` with `Type="Kperm"` (Section 2.2) which produces permutations of any T-products of vectors of any dimensions: beyond the considerations concerning using `useSparse=T` or not, when using `CommutatorMatr` with `Type="Kperm"`, the dimensions of the vectors in the argument may have a large impact on computing time.

Recall that `CommutatorMatr` with `Type="Kperm"` produces a permutation `perm` of vectors of dimensions specified in `dims`. As an example consider the case of the product of $n = 8$ vectors all with the same dimension d; using `dims <- d` instead of `dims <- c(d,d,d,d,d,d,d,d)` will require much less computing time. In the evaluation below

```
CommutatorMatr(Type = "Kperm", perm = c(2, 4, 6, 1, 3, 8, 5, 7), dims = 3)
```

requires approximately 2.11 seconds while

```
CommutatorMatr(Type = "Kperm", perm = c(2, 4, 6, 1, 3, 8, 5, 7),
            dims = c(3, 3, 3, 3, 3, 3, 3, 3))
```

requires approximately 1326.47 seconds.

Acknowledgement. The work of Gy. H. Terdik has been supported by the project TKP2021-NKTA of the University of Debrecen, Hungary. Project no. TKP2021-NKTA-34 has been implemented with the support provided by the Ministry of Culture and Innovation of Hungary from the National Research, Development and Innovation Fund, financed under the TKP2021-NKTA funding scheme.

References

A. Archimbaud, K. Nordhausen, and A. Ruiz-Gazen. Ics for multivariate outlier detection with application to quality control. *Computational Statistics & Data Analysis*, 128:184–199, 2018. ISSN 0167-9473. doi: https://doi.org/10.1016/j.csda.2018.06.011. [p9]

R. B. Arellano-Valle and M. G. Genton. On fundamental skew distributions. *Journal of Multivariate Analysis*, 96(1):93–116, 2005. [p2, 8]

J. M. Arevalillo and H. Navarro. Data projections by skewness maximization under scale mixtures of skew-normal vectors. *Advances in Data Analysis and Classification*, 14(2):435–461, 2020. [p9]

J. M. Arevalillo and H. Navarro. Skewness-based projection pursuit as an eigenvector problem in scale mixtures of skew-normal distributions. *Symmetry*, 13(6):1056, 2021. [p9]

A. Azzalini. *The R package* `sn`*: The skew-normal and related distributions such as the skew-t and the SUN (version 2.1.0).* Università degli Studi di Padova, Italia, 2022. URL `https://cran.r-project.org/package=sn`. Home page: `http://azzalini.stat.unipd.it/SN/`. [p2]

A. Azzalini and A. Dalla Valle. The multivariate skew-normal distribution. *Biometrika*, 83(4):715–726, 1996. [p2, 8]

D. Bates, M. Maechler, and M. Jagan. *Matrix: Sparse and Dense Matrix Classes and Methods*, 2024. URL `https://CRAN.R-project.org/package=Matrix`. R package version 1.6-5. [p2]

T. Brenn and S. N. Anfinsen. A revisit of the Gram-Charlier and Edgeworth series expansions. Artikler, rapporter og annet (fysikk og teknologi), Department of Physics and Technology, University of Tromsø, The Arctic University of Norway,, 2017. [p12]

J. E. Chacón and T. Duong. Efficient recursive algorithms for functionals based on higher order derivatives of the multivariate gaussian density. *Statistics and Computing*, 25(5):959–974, 2015. [p1, 5, 15]

E. Di Nardo and G. Guarino. *kStatistics: Unbiased Estimators for Cumulant Products and Faa Di Bruno's Formula*, 2022a. URL https://CRAN.R-project.org/package=kStatistics. R package version 2.1.1. [p2]

E. Di Nardo and G. Guarino. kstatistics: Unbiased estimates of joint cumulant products from the multivariate faà di bruno's formula. *The R Journal*, 14:208–228, 2022b. ISSN 2073-4859. doi: 10.32614/RJ-2022-033. https://doi.org/10.32614/RJ-2022-033. [p2, 10]

C. Franceschini and N. Loperfido. *MaxSkew: Orthogonal Data Projections with Maximal Skewness*, 2017a. URL https://CRAN.R-project.org/package=MaxSkew. R package version 1.1. [p2]

C. Franceschini and N. Loperfido. *MultiSkew: Measures, Tests and Removes Multivariate Skewness*, 2017b. URL https://CRAN.R-project.org/package=MultiSkew. R package version 1.1.1. [p2]

C. Franceschini and N. Loperfido. *Kurt: Performs Kurtosis-Based Statistical Analyses*, 2021. URL https://CRAN.R-project.org/package=Kurt. R package version 1.1. [p2]

A. Genz and F. Bretz. *Computation of Multivariate Normal and t Probabilities*. Lecture Notes in Statistics. Springer-Verlag, Heidelberg, 2009. ISBN 978-3-642-01688-2. [p2]

E. Guidotti. calculus: High-dimensional numerical and symbolic calculus in R. *Journal of Statistical Software*, 104(5):1–37, 2022. doi: 10.18637/jss.v104.i05. [p2]

B. Holmquist. The d-variate vector Hermite polynomial of order. *Linear Alg. and its Appl.*, 237/238: 155–190, 1996. [p1, 2]

W. R. Inc. Mathematica, Version 14.0. URL https://www.wolfram.com/mathematica. Champaign, IL, 2024. [p2]

S. R. Jammalamadaka, T. Subba Rao, and G. H. Terdik. Higher order cumulants of random vectors and applications to statistical inference and time series. *Sankhya (Ser. A, Methodology)*, 68:326–356, 2006. [p1]

S. R. Jammalamadaka, E. Taufer, and G. H. Terdik. Asymptotic theory for statistics based on cumulant vectors with applications. *Scandinavian Journal of Statistics*, 48(2):708–728, 2021a. [p2, 10, 11, 12]

S. R. Jammalamadaka, E. Taufer, and G. H. Terdik. On multivariate skewness and kurtosis. *Sankhya A*, 83-A:607–644, 2021b. [p2, 7, 8, 9]

D. Kahle. mpoly: Multivariate polynomials in R. *The R Journal*, 5(1):162–170, 2013. URL https://journal.r-project.org/archive/2013-1/kahle.pdf. [p2]

T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *SIAM review*, 51(3):455–500, 2009. [p1]

T. Kollo. Multivariate skewness and kurtosis measures with an application in ICA. *Journal of Multivariate Analysis*, 99(10):2328–2338, 2008. [p1, 2]

L. Komsta and F. Novomestky. *moments: Moments, Cumulants, Skewness, Kurtosis and Related Tests*, 2022. URL https://CRAN.R-project.org/package=moments. R package version 0.14.1. [p2]

J. A. Koziol. An alternative formulation of neyman's smooth goodness of fit tests under composite alternatives. *Metrika*, 34(1):17–24, 1987. [p2]

J. A. Koziol. A note on measures of multivariate kurtosis. *Biometrical journal*, 31(5):619–624, 1989. [p2, 9]

R. Lai. *arrangements: Fast Generators and Iterators for Permutations, Combinations, Integer Partitions and Compositions*, 2020. URL https://CRAN.R-project.org/package=arrangements. R package version 1.1.9. [p1]

N. Loperfido. Singular value decomposition of the third multivariate moment. *Linear Algebra and its Applications*, 473:202–216, 2015. ISSN 0024-3795. doi: https://doi.org/10.1016/j.laa.2014.05.043. Special issue on Statistics. [p1, 9]

N. Loperfido. A new kurtosis matrix, with statistical applications. *Linear Algebra and its Applications*, 512:1–17, 2017. ISSN 0024-3795. doi: https://doi.org/10.1016/j.laa.2016.09.033. [p1]

J. F. Malkovich and A. A. Afifi. On tests for multivariate normality. *Journal of the american statistical association*, 68(341):176–179, 1973. [p2, 9]

K. V. Mardia. Measures of multivariate skewness and kurtosis with applications. *Biometrika*, 57: 519–530, 1970. [p2, 9]

P. McCullagh. *Tensor methods in statistics*. Chapman and Hall/CRC, 2018. [p1]

T. F. Móri, V. K. Rohatgi, and G. J. Székely. On multivariate skewness and kurtosis. *Theory of Probability & Its Applications*, 38(3):547–551, 1994. [p1, 2, 9]

F. Novomestky. *matrixcalc: Collection of Functions for Matrix Calculations*, 2021. URL https://CRAN.R-project.org/package=matrixcalc. R package version 1.0-5. [p2]

F. Novomestky. *orthopolynom: Collection of Functions for Orthogonal and Orthonormal Polynomials*, 2022. URL https://CRAN.R-project.org/package=orthopolynom. R package version 1.0-6.1. [p2]

H. Ould-Baba, V. Robin, and J. Antoni. Concise formulae for the cumulant matrices of a random vector. *Linear Algebra and its Applications*, 485:392–416, 2015. [p1]

D. Peña, F. J. Prieto, and J. Viladomat. Eigenvectors of a kurtosis matrix as interesting directions to reveal cluster structure. *Journal of Multivariate Analysis*, 101(9):1995–2007, 2010. [p9]

L. Qi. Rank and eigenvalues of a supersymmetric tensor, the multivariate homogeneous polynomial and the algebraic hypersurface it defines. *Journal of Symbolic Computation*, 41(12):1309–1327, 2006. [p1]

R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2023. URL https://www.R-project.org/. [p1]

K. D. Smith. A tutorial on multivariate k-statistics and their computation. *arXiv preprint arXiv:2005.08373*, 2020. [p10]

G. H. Terdik. Higher order statistics and multivariate vector Hermite polynomials for nonlinear analysis of multidimensional time series. *Theory of Probability and Mathematical Statistics*, (66): 165–186, 2003. [p1]

G. H. Terdik. *Multivariate Statistical Methods: Going Beyond the Linear*. Springer Nature, 2021. [p1, 2, 4, 5, 9, 10, 12, 15]

G. H. Terdik and E. Taufer. *MultiStatM: Multivariate Statistical Methods*, 2024. URL https://CRAN.R-project.org/package=MultiStatM. R package version 2.0.0. [p1]

D. E. Tyler, F. Critchley, L. Dümbgen, and H. Oja. Invariant co-ordinate selection. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 71(3):549–592, 2009. [p9]

W. N. Venables and B. D. Ripley. *Modern Applied Statistics with S*. Springer, New York, fourth edition, 2002. URL https://www.stats.ox.ac.uk/pub/MASS4/. ISBN 0-387-95457-0. [p2]

György Terdik
University of Debrecen
Department of Informatics
Debrecen, Hungary
ORCiD: 0000-0002-9663-6892
terdik.gyorgy@inf.unideb.hu

Emanuele Taufer
University of Trento
Department of Economics and Management
Trento, Italy
ORCiD: 0000-0001-5140-9912
emanuele.taufer@unitn.it