

# Package ‘sn’

January 17, 2010

**Version** 0.4-14

**Date** 2010-01-16

**Title** The skew-normal and skew-t distributions

**Author** Adelchi Azzalini <adelchi.azzalini@unipd.it>

**Maintainer** Adelchi Azzalini <adelchi.azzalini@unipd.it>

**Depends** R (>= 2.2.0), mnormt (>= 1.3-1)

**Description** Functions for manipulating skew-normal and skew-t probability distributions, and for fitting them to data, in the scalar and in the multivariate case.

**Encoding** latin1

**License** GPL-2

**URL** <http://azzalini.stat.unipd.it/SN>

**Repository** CRAN

**Date/Publication** 2010-01-17 10:52:52

## R topics documented:

ais . . . . .	2
cp.to.dp . . . . .	3
dmsn . . . . .	4
dmst . . . . .	6
dsn . . . . .	7
dsn2.plot . . . . .	9
dst . . . . .	10
dst2.plot . . . . .	12
frontier . . . . .	13
gamma1.to.lambda . . . . .	14
msn.affine . . . . .	15
msn.cond.plot . . . . .	16

msn.conditional . . . . .	17
msn.fit . . . . .	18
msn.marginal . . . . .	21
msn.mle . . . . .	22
msn.quantities . . . . .	24
mst.fit . . . . .	26
mst.mle . . . . .	28
sample.centralmoments . . . . .	30
SN . . . . .	31
sn.2logL.profile . . . . .	33
sn.cumulants . . . . .	34
sn.Einfo . . . . .	36
sn.em . . . . .	37
sn.mle . . . . .	39
sn.mle.grouped . . . . .	41
sn.mmle . . . . .	42
st.2logL.profile . . . . .	44
st.cumulants . . . . .	46
T.Owen . . . . .	48
zeta . . . . .	49

## Index 50

---

ais *Australian Institute of Sport data*

---

### Description

Data on 102 male and 100 female athletes collected at the Australian Institute of Sport, courtesy of Richard Telford and Ross Cunningham.

### Usage

```
data(ais)
```

### Format

A data frame with 202 observations on 13 variables.

[, 1]	sex	sex
[, 2]	sport	sport
[, 3]	rcc	red cell count
[, 4]	wcc	white cell count
[, 5]	Hc	Hematocrit
[, 6]	Hg	Hemoglobin
[, 7]	Fe	plasma ferritin concentration
[, 8]	bmi	body mass index, $\text{weight}/(\text{height})^2$
[, 9]	ssf	sum of skin folds

[,10]	Bfat	body fat percentage
[,11]	lbm	lean body mass
[,12]	Ht	height (cm)
[,13]	Wt	weight (Kg)

### Source

Cook and Weisberg (1994), *An Introduction to Regression Graphics*. John Wiley & Sons, New York.

### Examples

```
data(ais, package="sn")
attach(ais)
pairs(ais[,c(3:4,10:13)], main = "AIS data")
plot(Wt~sport)
```

---

cp.to.dp

*Conversion between equivalent parametrizations*

---

### Description

Convert direct parameters (DP) to centred parameters (CP) of the one-dimensional skew-normal distribution and *vice versa*

### Usage

```
cp.to.dp(param)
dp.to.cp(param)
```

### Arguments

`param` a vector of length at least three. If `length(param)` is  $m+2$ , then the first  $m$  components refer to the regression coefficients (or the location parameter, in case  $m$  is 1), and the remaining two components refer to scale and shape, respectively; their role is preserved across parametrizations.

### Details

For a description of the two parametrizations, see the reference below.

### Value

a vector of the same length of `param`, representing `param` in the alternative parametrization; `cp.to.dp` converts centred to direct parameters, `dp.to.cp` converts direct to centred parameters.

## References

Azzalini, A. and Capitanio, A. (1999). Statistical applications of the multivariate skew-normal distribution. *J.Roy.Statist.Soc. B* **61**, 579–602.

## See Also

[sn.mle](#), [sn.em](#)

## Examples

```
cp <- dp.to.cp(c(30,30,2,4))
dp <- cp.to.dp(cp)
```

---

dmsn

*Multivariate skew-normal distribution*

---

## Description

Probability density function, distribution function and random number generation for the multivariate skew-normal (MSN) distribution.

## Usage

```
dmsn(x, xi = rep(0, length(alpha)), Omega, alpha, dp = NULL, log = FALSE)
pmsn(x, xi = rep(0, length(alpha)), Omega, alpha, dp = NULL, ...)
rmsn(n = 1, xi = rep(0, length(alpha)), Omega, alpha, dp = NULL)
```

## Usage

```
dmsn(x, xi=rep(0,length(alpha)), Omega, alpha, log=FALSE)
dmsn(x, dp=, log=FALSE)
pmsn(x, xi=rep(0,length(alpha)), Omega, alpha, ...)
pmsn(x, dp=)
rmsn(n=1, xi=rep(0,length(alpha)), Omega, alpha)
rmsn(n=1, dp=)
```

## Arguments

<code>x</code>	for <code>dmsn</code> , this is either a vector of length <code>d</code> , where <code>d=length(alpha)</code> , or a matrix with <code>d</code> columns, giving the coordinates of the point(s) where the density must be evaluated; for <code>pmsn</code> , only a vector of length <code>d</code> is allowed.
<code>xi</code>	a numeric vector of length <code>d</code> representing the location parameter of the distribution. In a call to <code>dmsn</code> , <code>xi</code> can be a matrix; in this case, its dimensions must agree with those of <code>x</code> .
<code>Omega</code>	a positive-definite covariance matrix of dimension $(d, d)$ .
<code>alpha</code>	a numeric vector which regulates the shape of the density.

dp	a list with three elements named xi, Omega and alpha containing quantities as described above. If dp is specified, this overrides the individual parameter specification.
n	a numeric value which represents the number of random vectors to be drawn.
log	logical; if TRUE, densities are given as log-densities.
...	additional parameters passed to pmnorm

### Details

The positive-definiteness of Omega is not tested for efficiency reasons. Function pmsn requires pmnorm from package mnormt; the accuracy of its computation can be controlled via use of ...

### Value

A vector of density values (dmsn), or a single probability (pmsn) or a matrix of random points (rmsn).

### Background

The multivariate skew-normal distribution is discussed by Azzalini and Dalla Valle (1996); the (Omega, alpha) parametrization adopted here is the one of Azzalini and Capitanio (1999).

### References

Azzalini, A. and Dalla Valle, A. (1996). The multivariate skew-normal distribution. *Biometrika* **83**, 715–726.

Azzalini, A. and Capitanio, A. (1999). Statistical applications of the multivariate skew-normal distribution. *J.Roy.Statist.Soc. B* **61**, 579–602.

### See Also

[dsn](#), [dmst](#), [dmnorm](#)

### Examples

```
x <- seq(-3,3,length=15)
xi <- c(0.5, -1)
Omega <- diag(2)
Omega[2,1] <- Omega[1,2] <- 0.5
alpha <- c(2,-6)
pdf <- dmsn(cbind(x,2*x-1), xi, Omega, alpha)
rnd <- rmsn(10, xi, Omega, alpha)
p1 <- pmsn(c(2,1), xi, Omega, alpha)
p2 <- pmsn(c(2,1), xi, Omega, alpha, abseps=1e-12, maxpts=10000)
```

dmst

*Multivariate skew-t distribution***Description**

Probability density function, distribution function and random number generation for the multivariate skew-t (MST) distribution.

**Usage**

```
dmst(x, xi=rep(0,length(alpha)), Omega, alpha, df=Inf, dp = NULL, log=FALSE)
pmst(x, xi=rep(0,length(alpha)), Omega, alpha, df=Inf, dp = NULL, ...)
rmst(n=1, xi=rep(0,length(alpha)), Omega, alpha, df=Inf, dp = NULL)
```

**Usage**

```
dmst(x, xi=rep(0,length(alpha)), Omega, alpha, df=Inf, log=FALSE)
dmst(x, dp=, log=FALSE)
pmst(x, xi=rep(0,length(alpha)), Omega, alpha, df=Inf, ...)
pmst(x, dp=, ...)
rmst(n=1, xi=rep(0,length(alpha)), Omega, alpha, df=Inf)
rmst(n=1, dp=)
```

**Arguments**

<code>x</code>	for <code>dmst</code> , this is either a vector of length <code>d</code> , where <code>d=length(alpha)</code> , or a matrix with <code>d</code> columns, giving the coordinates of the point(s) where the density must be evaluated; for <code>pmst</code> , only a vector of length <code>d</code> is allowed.
<code>xi</code>	a numeric vector of length <code>d</code> , or a matrix with <code>d</code> columns, representing the location parameter of the distribution. If <code>xi</code> is a matrix, its dimensions must agree with those of <code>x</code> .
<code>Omega</code>	a positive-definite covariance matrix of dimension $(d, d)$ .
<code>alpha</code>	a numeric vector which regulates the shape of the density.
<code>df</code>	degrees of freedom (scalar); default is <code>df=Inf</code> which corresponds to the multivariate skew-normal distribution.
<code>dp</code>	a list with three elements named <code>xi</code> , <code>Omega</code> , <code>alpha</code> and <code>df</code> , containing quantities as described above. If <code>dp</code> is specified, this overrides the individual parameter specification.
<code>n</code>	a numeric value which represents the number of random vectors to be drawn.
<code>log</code>	logical; if <code>TRUE</code> , densities are given as log-densities.
<code>...</code>	additional parameters passed to <code>pmt</code>

## Details

The positive-definiteness of  $\Omega$  is not tested for efficiency reasons. Function `pmst` requires `pmt` from package `mnormt`; the accuracy of its computation can be controlled via use of `...`

## Value

A vector of density values (`dmst`) or a single probability (`pmst`) or a matrix of random points (`rmst`).

## Background

The family of multivariate skew-t distributions is an extension of the multivariate Student's t family, via the introduction of a `shape` parameter which regulates skewness; when `shape=0`, the skew-t distribution reduces to the usual t distribution. When `df=Inf` the distribution reduces to the multivariate skew-normal one; see `dmsn`. See the reference below for additional information.

## References

Azzalini, A. and Capitanio, A. (2003). Distributions generated by perturbation of symmetry with emphasis on a multivariate skew *t* distribution. *J.Roy. Statist. Soc. B* **65**, 367–389.

## See Also

[dst](#), [dmsn](#), [dmt](#)

## Examples

```
x <- seq(-4,4,length=15)
xi <- c(0.5, -1)
Omega <- diag(2)
Omega[2,1] <- Omega[1,2] <- 0.5
alpha <- c(2,2)
pdf <- dmst(cbind(x,2*x-1), xi, Omega, alpha, df=5)
rnd <- rmst(10, xi, Omega, alpha, 6)
p1 <- pmst(c(2,1), xi, Omega, alpha, df=5)
p2 <- pmst(c(2,1), xi, Omega, alpha, df=5, abseps=1e-12, maxpts=10000)
```

## Description

Density function, distribution function, quantiles and random number generation for the skew-normal (SN) distribution.

**Usage**

```
dsn(x, location = 0, scale = 1, shape = 0, dp = NULL, log = FALSE)
psn(x, location = 0, scale = 1, shape = 0, dp = NULL, engine, ...)
qsn(p, location = 0, scale = 1, shape = 0, dp = NULL, tol = 1e-8, engine, ...)
rsn(n = 1, location = 0, scale = 1, shape = 0, dp = NULL)
```

**Usage**

```
dsn(x, location=0, scale=1, shape=0, log=FALSE)
dsn(x, dp=, log=FALSE)
psn(x, location=0, scale=1, shape=0, engine, ...)
psn(x, dp=, engine, ...)
qsn(p, location=0, scale=1, shape=0, tol=1e-8, ...)
qsn(x, dp=, ...)
rsn(n=1, location=0, scale=1, shape=0)
rsn(x, dp=)
```

**Arguments**

<code>x</code>	vector of quantiles. Missing values (NAs) and Inf's are allowed.
<code>p</code>	vector of probabilities. Missing values (NAs) are allowed.
<code>location</code>	vector of location parameters.
<code>scale</code>	vector of (positive) scale parameters.
<code>shape</code>	vector of shape parameters. With <code>psn</code> and <code>qsn</code> , it must be of length 1.
<code>dp</code>	a vector of length 3, whose elements represent location, scale (positive) and shape, respectively. If <code>dp</code> is specified, the individual parameters cannot be set.
<code>n</code>	sample size.
<code>tol</code>	a scalar value which regulates the accuracy of the result of <code>qsn</code> .
<code>log</code>	logical flag used in <code>dsn</code> (default FALSE). When TRUE, the logarithm of the density values is returned.
<code>engine</code>	character string to select the computing engine, which is either "T.Owen" or "biv.nt.prob" (the latter from package <code>mnormt</code> ). If the parameter is missing, a default selection rule is applied.
<code>...</code>	additional parameters passed to T.Owen

**Value**

density (`dsn`), probability (`psn`), quantile (`qsn`) or random sample (`rsn`) from the skew-normal distribution with given `location`, `scale` and `shape` parameters.

**Background**

The family of skew-normal distributions is an extension of the normal family, via the introduction of a `shape` parameter which regulates skewness; when `shape=0`, the skew-normal distribution reduces to the normal one. The density of the SN distribution in the "standard" case having

location=0 and scale=1 is  $2 \cdot dnorm(x) \cdot pnorm(shape \cdot x)$ . A multivariate version of the distribution exists. See the reference below for additional information.

### Details

psn and qsn make use either of function [T.Owen](#) or [biv.nt.prob](#)

### References

Azzalini, A. (1985). A class of distributions which includes the normal ones. *Scand. J. Statist.* **12**, 171-178.

### See Also

[dmsn](#), [dst](#), [T.Owen](#), [biv.nt.prob](#)

### Examples

```
pdf <- dsn(seq(-3,3,by=0.1), shape=3)
cdf <- psn(seq(-3,3,by=0.1), shape=3)
qu <- qsn(seq(0.1,0.9,by=0.1), shape=-2)
rn <- rsn(100, 5, 2, 5)
```

---

 dsn2.plot

---

*Plot of Bivariate Skew-normal Density Function*


---

### Description

Produces a contour plot of the density function of a bivariate skew-normal variate.

### Usage

```
dsn2.plot(x, y, xi, Omega, alpha, dp = NULL, ...)
```

### Usage

```
dsn2.plot(x, y, xi, Omega, alpha, ...)
dsn2.plot(x, y, dp=, ...)
```

### Arguments

x	vector of values of the first component.
y	vector of values of the second component.
xi	a vector of length 2 containing the location parameter.
Omega	a 2 by 2 matrix containing a covariance matrix.
alpha	a vector of length 2 containing the shape parameter.

dp a list with components named `xi`, `Omega`, `alpha`, containing quantities as described above. If this parameter is set, then the individual parameters must not be.

... additional parameters to be passed to `contour`.

### Details

The density function is evaluated at the grid of points whose coordinates are given by vectors `x` and `y`. The actual computation is done by the function `dmsn`. A contour level plot is produced on the graphical window.

### Value

A list containing the original input parameters plus a matrix containing the density function evaluated at the grid formed by the `x` and `y` values.

### Background

The multivariate skew-normal distribution is discussed by Azzalini and Dalla Valle (1996); the `(Omega, alpha)` parametrization adopted here is the one of Azzalini and Capitanio (1999).

### References

Azzalini, A. and Dalla Valle, A. (1996). The multivariate skew-normal distribution. *Biometrika* **83**, 715–726.

Azzalini, A. and Capitanio, A. (1999). Statistical applications of the multivariate skew-normal distribution. *J.Roy.Statist.Soc. B* **61**, 579–602.

### See Also

[dmsn](#)

### Examples

```
x <- y <- seq(-5, 5, length=35)
dsn2.plot(x, y, c(-1,2), diag(c(1,2.5)), c(2,-3))
```

---

dst

*Skew-t Distribution*

---

### Description

Density function, distribution function and random number generation for the skew-t (ST) distribution.

**Usage**

```
dst(x, location = 0, scale = 1, shape = 0, df = Inf, dp = NULL, log = FALSE)
pst(x, location = 0, scale = 1, shape = 0, df = Inf, dp = NULL, ...)
qst(p, location = 0, scale = 1, shape = 0, df = Inf, tol = 1e-08, dp = NULL, ...)
rst(n = 1, location = 0, scale = 1, shape = 0, df = Inf, dp = NULL)
```

**Usage**

```
dst(x, location=0, scale=1, shape=0, df=Inf, log=FALSE)
dst(x, dp=, log=FALSE)
pst(x, location=0, scale=1, shape=0, df=Inf, ...)
pst(x, dp=, log=FALSE)
qst(p, location=0, scale=1, shape=0, df=Inf, tol=1e-8, ...)
qst(x, dp=, log=FALSE)
rst(n=1, location=0, scale=1, shape=0, df=Inf)
rst(x, dp=, log=FALSE)
```

**Arguments**

x	vector of quantiles. Missing values (NAs) are allowed.
p	vector of probabilities
location	vector of location parameters.
scale	vector of (positive) scale parameters.
shape	vector of shape parameters. With <code>pst</code> and <code>qst</code> , it must be of length 1.
df	degrees of freedom (scalar); default is <code>df=Inf</code> which corresponds to the skew-normal distribution.
dp	a vector of length 4, whose elements represent location, scale (positive), shape and df, respectively. If <code>dp</code> is specified, the individual parameters cannot be set.
n	sample size.
log	logical; if TRUE, densities are given as log-densities.
tol	a scalar value which regulates the accuracy of the result of <code>qsn</code> .
...	additional parameters passed to <code>integrate</code> .

**Value**

Density (`dst`), probability (`pst`), quantiles (`qst`) and random sample (`rst`) from the skew-t distribution with given `location`, `scale`, `shape` and `df` parameters.

**Background**

The family of skew-t distributions is an extension of the Student's t family, via the introduction of a shape parameter which regulates skewness; when `shape=0`, the skew-t distribution reduces to the usual Student's t distribution. When `df=Inf`, it reduces to the skew-normal distribution. A multivariate version of the distribution exists. See the reference below for additional information.

**References**

Azzalini, A. and Capitanio, A. (2003). Distributions generated by perturbation of symmetry with emphasis on a multivariate skew- $t$  distribution. *J.Roy. Statist. Soc. B* **65**, 367–389.

**See Also**

[dmst](#), [dsn](#), [psn](#)

**Examples**

```
pdf <- dst(seq(-4,4,by=0.1), shape=3, df=5)
rnd <- rst(100, 5, 2, -5, 8)
q <- qst(c(0.25,0.5,0.75), shape=3, df=5)
pst(q, shape=3, df=5) # must give back c(0.25,0.5,0.75)
```

---

dst2.plot

---

*Plot of bivariate skew-t density function*


---

**Description**

Produces a contour plot of the density function of a bivariate skew- $t$  variate.

**Usage**

```
dst2.plot(x, y, xi, Omega, alpha, df, dp = NULL, ...)
```

**Usage**

```
dst2.plot(x, y, xi, Omega, alpha, df, ...)
dst2.plot(x, y, dp=, ...)
```

**Arguments**

x	vector of values of the first component.
y	vector of values of the second component.
xi	a vector of length 2 containing the location parameter.
Omega	a 2 by 2 matrix containing a covariance matrix.
alpha	a vector of length 2 containing the shape parameter.
df	a positive number, representing the degrees of freedom .
dp	a list with components named xi, Omega, alpha, df, containing quantities as described above. If this parameter is set, then the individual parameters must not be.
...	additional parameters to be passed to <code>contour</code> .

**Details**

The density function is evaluated at the grid of points whose coordinates are given by vectors  $x$  and  $y$ . The actual computation is done by the function `dmst`. A contour level plot is produced on the graphical window.

**Value**

A list containing the original input parameters plus a matrix containing the density function evaluated at the grid formed by the  $x$  and  $y$  values.

**Background**

The family of multivariate skew- $t$  distributions is an extension of the multivariate Student's  $t$  family, via the introduction of a `shape` parameter which regulates skewness; when `shape=0`, the skew- $t$  distribution reduces to the usual  $t$  distribution. When `df=Inf` the distribution reduces to the multivariate skew-normal one; see `dmsn`. See the reference below for additional information.

**References**

Azzalini, A. and Capitanio, A. (2003). Distributions generated by perturbation of symmetry with emphasis on a multivariate skew  $t$  distribution. *J.Roy. Statist. Soc. B* **65**, 367–389.

**See Also**

`dmst`, `dsn2.plot`

**Examples**

```
x <- y <- seq(-5, 5, length=35)
dst2.plot(x, y, c(-1,2), diag(c(1,2.5)), c(2,-3), df=5)
```

---

frontier

*Simulated sample from a skew-normal distribution*

---

**Description**

A sample simulated from the SN(0,1,5) distribution having sample index of skewness in the admissible range (-0.9952719,0.9952719) but maximum likelihood estimate on the frontier of the parameter space.

**Usage**

```
data(frontier)
```

**Format**

A vector of length 50.

**Source**

Generated by a run of `rsn(50, 0, 1, 5)`.

**Examples**

```
data(frontier, package="sn")
a <- sn.2logL.profile(y=frontier)
a <- sn.2logL.profile(y=frontier, param.range=c(0.8, 1.6, 10, 30),
  use.cp=FALSE, npts=11)
```

---

`gamma1.to.lambda`    *Converts skewness to shape parameter of skew-normal distribution*

---

**Description**

For a given value of the index of skewness (standardized third cumulant), the function finds the corresponding shape parameter of a skew-normal distribution.

**Usage**

```
gamma1.to.lambda(gamma1)
```

**Arguments**

`gamma1`            a numeric vector of indices of skewness.

**Details**

Feasible values for input must have  $\text{abs}(\text{gamma1}) < 0.5 * (4 - \pi) * (2 / (\pi - 2))^{1.5}$ , which is about 0.99527. If some values of `gamma1` are not in the feasible region, a warning message is issued, and NAs are returned.

See the reference below for the expression of the index of skewness of a skew-normal distribution.

**Value**

A numeric vector of the corresponding shape parameters.

**References**

Azzalini, A. (1985). A class of distributions which includes the normal ones. *Scand. J. Statist.* **12**, 171-178.

**See Also**

[dsn](#)

**Examples**

```
gamma1.to.lambda(seq(-0.95, 0.95, length=11))
```

msn.affine

*Affine transformation of a multivariate skew-normal or skew-t variable***Description**

Computes the parameters of an affine transformation  $a + A Y$  of a multivariate skew-normal or skew-t variable  $Y$

**Usage**

```
msn.affine(dp, a=0, A, drop=TRUE)
mst.affine(dp, a=0, A, drop=TRUE)
```

**Arguments**

dp	a list containing the parameters of the variable being transformed; it must include components xi, Omega, alpha as described for dmsn; for mst.affine, also a component df is expected
A	a matrix with ncol(A) equal to nrow(dp\$Omega)
a	a vector with length(a) equal to nrow(dp\$Omega)
drop	a logical flag (default value is TRUE) operating when nrow(A) equals 1. If these conditions are met, the output is provided in the form of parameters of a scalar distribution, dsn or dst, depending in the case.

**Value**

A list containing the same components of the input parameter dp

**Background**

For background information about the skew-normal and skew-t distributions, their parameters and the properties of affine transformations, see the references below. The specific formulae implemented by this function are given in Appendix A.2 of Capitanio et al.(2003).

**References**

- Azzalini, A. and Capitanio, A. (1999). Statistical applications of the multivariate skew-normal distribution. *J.Roy.Statist.Soc. B* **61**, 579–602.
- Azzalini, A. and Capitanio, A. (2003). Distributions generated by perturbation of symmetry with emphasis on a multivariate skew-t distribution. *J.Roy. Statist. Soc. B* **65**, 367–389.
- Capitanio, A. et al. (2003). Graphical models for skew-normal variates. *Scand.\ J.\ Statist.* **30**, 129–144.

**See Also**

[dsn](#), [dst](#), [dmsn](#), [dmst](#)

**Examples**

```

dp<- list(xi=c(1,1,2), Omega=toeplitz(1/1:3), alpha=c(3,-1,2))
A <- matrix(c(1,-1,1,3,0,-2), 2, 3, byrow=TRUE)
dp1 <- msn.affine(dp, 1:2, A)
#
dp$df <- 5
dp2<- mst.affine(dp,,A[1,,drop=FALSE])
dp3<- mst.affine(dp,,A[1,,drop=FALSE], drop=FALSE)
if(zapsmall(dp2$scale^2 - dp3$Omega)) print("something wrong here!")

```

---

msn.cond.plot

*Plot of the density of a conditional skew-normal variate*


---

**Description**

Plot of the exact and of the approximate density function of a multivariate skew-normal variate conditionally on the values taken on by some components.

**Usage**

```
msn.cond.plot(xi, Omega, alpha, fixed.comp, fixed.values, n=35)
```

**Arguments**

<code>xi</code>	a numeric vector of length $k$ , say, giving the location parameter.
<code>Omega</code>	a covariance matrix of dimension $(k, k)$ .
<code>alpha</code>	a numeric vector of length $k$ , which regulates the shape of the density.
<code>fixed.comp</code>	a vector containing a subset of $1:k$ which selects the components whose values are to be fixed; it must be of length $k-2$ .
<code>fixed.values</code>	a numeric vector of values taken on by the components <code>fixed.comp</code> ; it must be of the same length of <code>fixed.comp</code> .
<code>n</code>	an integer value which determines the grid size of the density computations and plot.

**Details**

See Section 4.2 of the reference given below for background details

**Value**

A list containing the following elements:

<code>cumulants</code>	two lists as returned by <code>msn.conditional</code> .
<code>pdf</code>	a list containing the coordinates $x$ and $y$ of the points where the densities have been evaluated, and the matrices <code>f.exact</code> and <code>f.fitted</code> of the exact and fitted conditional densities.
<code>rel.error</code>	summary statistics of relative and absolute error of the approximation.

**Side Effects**

A contour plot of the exact and approximate densities is produced on a graphical device.

**References**

Azzalini, A. and Capitanio, A. (1999). Statistical applications of the multivariate skew-normal distribution. *J.Roy.Statist.Soc. B* **61**, 579–602.

**See Also**

[msn.conditional](#), [dmsn](#)

**Examples**

```
Omega <- diag(3)+0.5*outer(rep(1,3),rep(1,3))
a<- msn.cond.plot(rep(0,3), Omega, 1:3, 3, -0.75)
```

---

msn.conditional	<i>Cumulants and distribution of a skew-normal variate after conditioning</i>
-----------------	---

---

**Description**

Finds cumulants up to 3rd order of a multivariate skew-normal distribution conditionally on the values taken on by some of its components, and finds a multivariate skew-normal distribution with the same cumulants.

**Usage**

```
msn.conditional(xi = rep(0, length(alpha)), Omega, alpha, fixed.comp,
               fixed.values, dp = NULL)
```

**Usage**

```
msn.conditional(xi, Omega, alpha, fixed.comp, fixed.values)
msn.conditional(dp=, fixed.comp, fixed.values)
```

**Arguments**

xi	a numeric vector of length d, say, giving the location parameter.
Omega	a covariance matrix of dimension (d, d).
alpha	a numeric vector of length d, which regulates the shape of the density.
fixed.comp	a vector containing a subset of 1:d which selects the components whose values are to be fixed; it must be of length d-2.
fixed.values	a numeric vector of values taken on by the components fixed.comp; it must be of the same length of fixed.comp.

`dp` a list containing the components `xi`, `Omega`, `alpha`, containing quantities as described above

### Details

See the reference below for details and background.

### Value

A list containing the following elements:

`cumulants` a list containing mean vector, variance matrix, and indices of skewness of the conditional distribution.

`fit` a list containing the parameters of the fitted skew-normal distribution in the  $(\mathbf{xi}, \mathbf{Omega}, \mathbf{alpha})$  parametrization, plus the vector `delta`.

### References

Azzalini, A. and Capitanio, A. (1999). Statistical applications of the multivariate skew-normal distribution. *J.Roy.Statist.Soc. B* **61**, 579–602.

### See Also

[msn.cond.plot](#), [msn.marginal](#)

### Examples

```
Omega <- diag(3)+0.5*outer(rep(1,3),rep(1,3))
a<- msn.conditional(rep(0,3), Omega, 1:3, 3, -0.75)
```

---

`msn.fit`

*Fitting multivariate skew-normal distributions*

---

### Description

Fits a multivariate skew-normal (MSN) distribution to data, or fits a linear regression model with multivariate skew-normal errors, using maximum likelihood estimation. The outcome is then displayed in graphical form.

### Usage

```
msn.fit(X, y, freq, plot.it=TRUE, trace=FALSE, ... )
```

**Arguments**

<code>y</code>	a matrix or a vector. If <code>y</code> is a matrix, its rows refer to observations, and its columns to components of the multivariate distribution. If <code>y</code> is a vector, it is converted to a one-column matrix, and a scalar skew-normal distribution is fitted.
<code>X</code>	a matrix of covariate values. If missing, a one-column matrix of 1's is created; otherwise, it must have the same number of rows of <code>y</code> .
<code>freq</code>	a vector of weights. If missing, a one-column matrix of 1's is created; otherwise it must have the same number of rows of <code>y</code> .
<code>plot.it</code>	logical value which controls the graphical output (default=TRUE); see below for description.
<code>trace</code>	logical value which controls printing of the algorithm convergence. If <code>trace=TRUE</code> , details are printed. Default value is FALSE.
<code>...</code>	additional parameters passed to <code>msn.mle</code> ; in practice, the <code>start</code> , the <code>algorithm</code> and the <code>control</code> parameters can be passed.

**Details**

For computing the maximum likelihood estimates, `msn.fit` invokes `msn.mle` which does the actual computational work; then, `msn.fit` displays the results in graphical form. The documentation of `msn.mle` gives details of the numerical procedure for maximum likelihood estimation.

Although the function accepts a vector `y` as input, the use of `sn.mle` is recommended in the scalar case.

**Value**

A list containing the following components:

<code>call</code>	a string containing the calling statement.
<code>dp</code>	a list containing the direct parameters <code>beta</code> , <code>Omega</code> , <code>alpha</code> . Here, <code>beta</code> is a matrix of regression coefficients with <code>dim(beta)=c(nrow(X), ncol(y))</code> , <code>Omega</code> is a covariance matrix of order <code>ncol(y)</code> , <code>alpha</code> is a vector of shape parameters of length <code>ncol(y)</code> .
<code>logL</code>	log-likelihood evaluated at <code>dp</code> .
<code>se</code>	a list containing the components <code>beta</code> , <code>alpha</code> , <code>info</code> . Here, <code>beta</code> and <code>alpha</code> are the standard errors for the corresponding point estimates; <code>info</code> is the observed information matrix for the working parameter, as explained below.
<code>algorithm</code>	see the documentation of <code>msn.mle</code> for its explanation
<code>test.normality</code>	a list of with elements <code>test</code> and <code>p.value</code> , which are the value of the likelihood ratio test statistic for normality (i.e. test that all components of the shape parameter are 0), and the corresponding p-value.

### Side Effects

Graphical output is produced if `(plot.it & missing(freq))=TRUE` and a suitable device is active. Three plots are produced, and the programs pauses between each two of them, waiting for the <Enter> key to be pressed.

The first plot uses the variable  $y$  if  $X$  is missing, otherwise it uses the residuals from the regression. The form of this plot depends on the value of `k=ncol(y)`; if `k=1`, an histogram is plotted with the fitted distribution superimposed. If `k>1`, a matrix of scatterplots is produced, with superimposed the corresponding bivariate densities of the fitted distribution.

The second plot has two panels, each representing a QQ-plot of Mahalanobis distances. The first of these refers to the fitting of a multivariate normal distribution, a standard statistical procedure; the second panel gives the corresponding QQ-plot of suitable Mahalanobis distances for the multivariate skew-normal fit.

The third plot is similar to the previous one, except that PP-plots are produced.

### Background

The multivariate skew-normal distribution is discussed by Azzalini and Dalla Valle (1996); the  $(\Omega, \alpha)$  parametrization adopted here is the one of Azzalini and Capitanio (1999).

### Note

This function may be removed in future versions of the package, and (some of) its functionality transferred somewhere else

### References

Azzalini, A. and Dalla Valle, A. (1996). The multivariate skew-normal distribution. *Biometrika* **83**, 715–726.

Azzalini, A. and Capitanio, A. (1999). Statistical applications of the multivariate skew-normal distribution. *J.Roy.Statist.Soc. B* **61**, 579–602.

### See Also

`msn.mle`, `mst.fit`, `dmsn`,

### Examples

```
data(ais, package="sn")
attach(ais)
# a simple-sample case
b <- msn.fit(y=cbind(Ht,Wt))
#
# a regression case:
a <- msn.fit(X=cbind(1,Ht,Wt), y=bmi, control=list(x.tol=1e-6))
#
# refine the previous outcome
a1 <- msn.fit(X=cbind(1,Ht,Wt), y=bmi, control=list(x.tol=1e-9), start=a$dp)
```

---

msn.marginal                      *Marginal components of a multivariate skew-normal distribution*

---

**Description**

Computes the marginal distribution of a subset of components of a multivariate skew-normal distribution.

**Usage**

```
msn.marginal(xi=rep(0, length(alpha)), Omega, alpha, comp=1:d, dp=NULL)
```

**Usage**

```
msn.marginal(xi, Omega, alpha, comp)
msn.marginal(dp=, comp)
```

**Arguments**

xi	a numeric vector of length $d$ , say, giving the location parameter.
Omega	a covariance matrix of dimension $(d, d)$ .
alpha	a numeric vector of length $d$ , which regulates the shape of the density.
comp	a vector containing a subset of $1:d$ selecting the components of the marginal distribution. A permutation of $1:d$ is allowed, and the components of <code>comp</code> do not need to be sorted.
dp	a list containing the components <code>xi</code> , <code>Omega</code> , <code>alpha</code> , containing quantities as described above; if <code>dp</code> is specified, then the individual components must not be in the calling statement

**Value**

A list containing components `xi`, `Omega`, `alpha` with the parameters of the marginal distribution. If `length(comp)` is equal to  $m$ , say, then the new components are of size  $m$ ,  $(m, m)$ ,  $n$ , respectively.

**Background**

See the reference below for background information.

**References**

Azzalini, A. and Capitanio, A. (1999). Statistical applications of the multivariate skew-normal distribution. *J.Roy.Statist.Soc. B* **61**, 579–602.

**See Also**

[dmsn](#), [msn.conditional](#), [msn.affine](#)

**Examples**

```
xi <- c(10,0,-30)
Omega <- 5*diag(3)+outer(1:3,1:3)
alpha <- c(1,-3,5)
msn.marginal(xi,Omega,alpha,c(3,1))
msn.marginal(dp=list(xi=xi,Omega=Omega,alpha=alpha), comp=3)
```

---

msn.mle	<i>Maximum likelihood estimation for a multivariate skew-normal distribution</i>
---------	--

---

**Description**

Fits a multivariate skew-normal (MSN) distribution to data, or fits a linear regression model with multivariate skew-normal errors, using maximum likelihood estimation.

**Usage**

```
msn.mle(X, y, freq, start, trace=FALSE,
        algorithm=c("nlminb", "Nelder-Mead", "BFGS", "CG", "SANN"),
        control=list())
```

**Arguments**

y	a matrix or a vector. If y is a matrix, rows refer to observations, and columns to components of the multivariate distribution. If y is a vector, it is converted to a one-column matrix, and a scalar skew-normal distribution is fitted.
X	a matrix of covariate values. If missing, a one-column matrix of 1's is created; otherwise, it must have the same number of rows of y, and it must include a column of 1's if an intercept term is required.
freq	a vector of weights. If missing, a vector of 1's is created; otherwise it must have length equal to the number of rows of y.
start	a list containing the components beta, Omega, alpha, of the type described below. The dp component of the returned list from a previous call has the required format.
trace	logical value which controls printing of the algorithm convergence. If trace=TRUE, details are printed. Default value is FALSE.
algorithm	a character string which selects the numerical optimization procedure used to maximize the log-likelihood function. If this string is set equal to "nlminb", then this function is called; in all other cases, optim is called, with method set equal to the given string. Default value is "nlminb".
control	this parameter is passed to the optimizer selected via algorithm; see the documentation of nlminb or optim for its usage.

## Details

The parameter `freq` is intended for use with grouped data, setting the values of `y` equal to the central values of the cells; in this case the resulting estimate is an approximation to the exact maximum likelihood estimate. If `freq` is not set, exact maximum likelihood estimation is performed.

The working parameter used in the maximization stage is `c(beta, alpha/omega)`, since a profile ‘deviance’  $-2 \times \log\text{likelihood}$  for this parameter is actually used; see Azzalini and Capitanio (1999, section 6.1) for details. The selected optimizer (`nlminb` or `optim`) is called, supplying the gradient of the profile deviance. In case the optimizer is `optim`, the gradient may or may not be used, depending on which specific method has been selected.

The function can take a vector `y` as input; however the use of `sn.mle` is recommended in the scalar case.

## Value

A list containing the following components:

<code>call</code>	a string containing the calling statement.
<code>dp</code>	a list containing terms named <code>beta</code> , <code>Omega</code> , <code>alpha</code> , where <code>beta</code> is a matrix of regression coefficients with $\text{dim}(\text{beta}) = \text{c}(\text{nrow}(X), \text{ncol}(y))$ , <code>Omega</code> is a covariance matrix of order <code>ncol(y)</code> , <code>alpha</code> is a vector of shape parameters of length <code>ncol(y)</code> .
<code>se</code>	a list containing the components <code>beta</code> , <code>alpha</code> , <code>info</code> . Here, <code>beta</code> and <code>alpha</code> are the standard errors for the corresponding point estimates; <code>info</code> is the observed information matrix for the working parameter, as explained below.
<code>algorithm</code>	the list returned by the chosen optimizer, either <code>nlminb</code> or <code>optim</code> , plus an item with the name of the selected algorithm; see the documentation of either <code>nlminb</code> or <code>optim</code> for explanation of the other components.

## Background

The multivariate skew-normal distribution is discussed by Azzalini and Dalla Valle (1996); the  $(\text{Omega}, \text{alpha})$  parametrization adopted here is the one of Azzalini and Capitanio (1999).

## References

Azzalini, A. and Dalla Valle, A. (1996). The multivariate skew-normal distribution. *Biometrika* **83**, 715–726.

Azzalini, A. and Capitanio, A. (1999). Statistical applications of the multivariate skew-normal distribution. *J.Roy.Statist.Soc. B* **61**, 579–602.

## See Also

[dmsn](#), [msn.fit](#), [nlminb](#), [optim](#)

**Examples**

```

data(ais, package="sn")
attach(ais)
# a simple-sample case
a <- msn.mle(y=cbind(Ht,Wt))
#
# a regression case:
b <- msn.mle(X=cbind(1,Ht,Wt), y=ssf)
b1 <- msn.mle(X=cbind(1,Ht,Wt), y=ssf, algorithm="Nelder-Mead")
b2 <- msn.mle(X=cbind(1,Ht,Wt), y=ssf, start=b1$dp)

```

---

msn.quantities

*Quantities related to the multivariate skew-normal distribution.*


---

**Description**

Computes mean vector, variance matrix and other relevant quantities of a given multivariate skew-normal distribution.

**Usage**

```
msn.quantities(xi=rep(0,length(alpha)), Omega, alpha, dp = NULL)
```

**Usage**

```
msn.quantities(xi=rep(0,length(alpha)), Omega, alpha)
msn.quantities(dp= )
```

**Arguments**

xi	numeric vector giving the location parameter, of length d, say. Missing values are not allowed.
Omega	a covariance matrix of size d by d. Missing values are not allowed.
alpha	numeric vector of shape parameter of length d. Missing values are not allowed.
dp	a list with three components named xi, Omega, alpha, containing quantities as described above. If dp is set, then the component parameters must not be.

**Details**

The meaning of the parameters is explained in the references below, especially Azzalini and Capitanio (1999).

**Value**

A list containing the following components:

<code>xi</code>	the input parameter <code>xi</code> .
<code>Omega</code>	the input parameter <code>Omega</code> .
<code>alpha</code>	the input parameter <code>alpha</code> .
<code>omega</code>	vector of scale parameters.
<code>mean</code>	the mean value of the distribution (vector)
<code>variance</code>	variance-covariance matrix of the distribution.
<code>Omega.conv</code>	concentration matrix associated to <code>Omega</code> , i.e. its inverse.
<code>Omega.cor</code>	correlation matrix associated to <code>Omega</code> .
<code>Omega.pcor</code>	partial correlations matrix associated to <code>Omega</code> .
<code>lambda</code>	shape parameters of the marginal distributions
<code>Psi</code>	correlation matrix of the equivalent <code>(lambda, Psi)</code> parametrization.
<code>delta</code>	the parameter <code>delta</code> which determines the shape of the marginal distributions; this is related to <code>lambda</code>
<code>skewness</code>	numeric vector with marginal indices of skewness (the standardised third cumulant).

**References**

Azzalini, A. and Dalla Valle, A. (1996). The multivariate skew-normal distribution. *Biometrika* **83**, 715–726.

Azzalini, A. and Capitanio, A. (1999). Statistical applications of the multivariate skew-normal distribution. *J.Roy.Statist.Soc. B* **61**, 579–602.

**See Also**

[dmsn](#)

**Examples**

```
Omega <- 5*diag(3)+outer(1:3,1:3)
msn.quantities(c(0,0,1), Omega, c(-2,2,3))
```

---

mst.fit

*Fitting multivariate skew-t distributions*


---

### Description

Fits a multivariate skew-t (MST) distribution to data, or fits a linear regression model with multivariate skew-t errors, using maximum likelihood estimation. The outcome is then displayed in graphical form.

### Usage

```
mst.fit(X, y, freq, start, fixed.df=NA, plot.it=TRUE, trace=FALSE, ...)
```

### Arguments

<code>y</code>	a matrix or a vector. If <code>y</code> is a matrix, its rows refer to observations, and its columns to components of the multivariate distribution. If <code>y</code> is a vector, it is converted to a one-column matrix, and a scalar skew-t distribution is fitted.
<code>X</code>	a matrix of covariate values. If missing, a one-column matrix of 1's is created; otherwise, it must have the same number of rows of <code>y</code> .
<code>freq</code>	a vector of weights. If missing, a vector of 1's is created; otherwise it must have the same number of rows of <code>y</code> .
<code>fixed.df</code>	a scalar value containing the degrees of freedom (df), if these must be taken as fixed, or NA (default value) if df is a parameter to be estimated.
<code>start</code>	a list containing the components <code>beta</code> , <code>Omega</code> , <code>alpha</code> , <code>df</code> of the type described below. The <code>dp</code> component of the returned list from a previous call has the required format.
<code>plot.it</code>	logical value which controls the graphical output (default=TRUE); see below for description.
<code>trace</code>	logical value which controls printing of the algorithm convergence. If <code>trace=TRUE</code> , details are printed. Default value is FALSE.
<code>...</code>	additional parameters passed to <code>msn.mle</code> ; in practice, the <code>start</code> , the algorithm and the <code>control</code> parameters can be passed.

### Details

For computing the maximum likelihood estimates, `mst.fit` invokes `mst.mle`, while `mst.fit` displays the results in graphical form. See the documentation of `mst.mle` for details of the numerical procedure for maximum likelihood estimation.

### Value

A list containing the following components:

<code>call</code>	a string containing the calling statement.
-------------------	--

<code>dp</code>	a list containing the direct parameters <code>beta</code> , <code>Omega</code> , <code>alpha</code> , <code>df</code> . Here, <code>beta</code> is a matrix of regression coefficients with $\dim(\text{beta}) = c(\text{nrow}(X), \text{ncol}(y))$ , <code>Omega</code> is a covariance matrix of order <code>ncol(y)</code> , <code>alpha</code> is a vector of shape parameters of length <code>ncol(y)</code> , <code>df</code> is a positive scalar.
<code>logL</code>	log-likelihood evaluated at <code>dp</code> .
<code>se</code>	a list containing the components <code>beta</code> , <code>alpha</code> , <code>info</code> . Here, <code>beta</code> and <code>alpha</code> are the standard errors for the corresponding point estimates; <code>info</code> is the observed information matrix for the working parameter, as explained below.
<code>algorithm</code>	see the documentation of <code>mst.mle</code> for its explanation
<code>test.normality</code>	a list with elements <code>test</code> and <code>p.value</code> , which are the value of the likelihood ratio test statistic for normality (i.e. <code>test</code> that all components of the shape parameter are 0 and <code>df=Inf</code> ), and the corresponding p-value.

### Side Effects

Graphical output is produced if `(plot.it & missing(freq)) == TRUE`. Three plots are produced, and the programs pauses between each two of them, waiting for the <Enter> key to be pressed.

The first plot uses the variable `y` if `X` is missing, otherwise it uses the residuals from the regression. The form of this plot depends on the value of `d=ncol(y)`; if `d=1`, an histogram is plotted with the fitted distribution superimposed. If `d>1`, a matrix of scatter-plots is produced, with superimposed the corresponding bivariate densities of the fitted distribution.

The second plot has two panels, each representing a QQ-plot of Mahalanobis distances. The first of these refers to the fitting of a multivariate normal distribution, a standard statistical procedure; the second panel gives the corresponding QQ-plot of suitable Mahalanobis distances for the multivariate skew-normal fit.

The third plot is similar to the previous one, except that PP-plots are produced.

### Background

The family of multivariate skew-t distributions is an extension of the multivariate Student's *t* family, via the introduction of a `shape` parameter which regulates skewness; when `shape=0`, the skew-t distribution reduces to the regular symmetric *t*-distribution. When `df=Inf` the distribution reduces to the multivariate skew-normal one; see `dmsn`. See the reference below for additional information.

### Note

This function may be removed in future versions of the package, and (some of) its functionality transferred somewhere else

### References

Azzalini, A. and Capitanio, A. (2003). Distributions generated by perturbation of symmetry with emphasis on a multivariate skew *t* distribution. *J.Roy. Statist. Soc. B* **65**, 367–389.

**See Also**

[mst.mle](#), [msn.fit](#), [dmst](#), [dmsn](#)

**Examples**

```
data(ais, package="sn")
attach(ais)
# a simple-sample case
b <- mst.fit(y=cbind(Ht,Wt))
#
# a regression case:
a <- mst.fit(X=cbind(1,Ht,Wt), y=bmi)
#
# refine the previous outcome
a1 <- mst.fit(X=cbind(1,Ht,Wt), y=bmi, start=a$dp)
```

---

mst.mle

*Maximum likelihood estimation for a (multivariate) skew-t distribution*

---

**Description**

Fits a skew-t (ST) or multivariate skew-t (MST) distribution to data, or fits a linear regression model with (multivariate) skew-t errors, using maximum likelihood estimation.

**Usage**

```
mst.mle(X, y, freq, start, fixed.df=NA, trace=FALSE,
        algorithm = c("nlminb", "Nelder-Mead", "BFGS", "CG", "SANN"), control=list())
st.mle(X, y, freq, start, fixed.df=NA, trace=FALSE,
       algorithm = c("nlminb", "Nelder-Mead", "BFGS", "CG", "SANN"), control=list())
```

**Arguments**

<code>y</code>	a matrix (for <code>mst.mle</code> ) or a vector (for <code>st.mle</code> ). If <code>y</code> is a matrix, rows refer to observations, and columns to components of the multivariate distribution.
<code>X</code>	a matrix of covariate values. If missing, a one-column matrix of 1's is created; otherwise, it must have the same number of rows of <code>y</code> . If <code>X</code> is supplied, then it must include a column of 1's.
<code>freq</code>	a vector of weights. If missing, a vector of 1's is created; otherwise it must have length equal to the number of rows of <code>y</code> .
<code>start</code>	for <code>mst.mle</code> , a list containing the components <code>beta</code> , <code>Omega</code> , <code>alpha</code> , <code>df</code> of the type described below; for <code>st.mle</code> , a vector whose components contain analogous ingredients as before, with the exception that the scale parameter is the square root of <code>Omega</code> . In both cases, the <code>dp</code> component of the returned list from a previous call has the required format and it can be used as a new <code>start</code> . If the <code>start</code> parameter is missing, initial values are selected by the function.

<code>fixed.df</code>	a scalar value containing the degrees of freedom (df), if these must be taken as fixed, or NA (default value) if df is a parameter to be estimated.
<code>trace</code>	logical value which controls printing of the algorithm convergence. If <code>trace=TRUE</code> , details are printed. Default value is <code>FALSE</code> .
<code>algorithm</code>	a character string which selects the numerical optimization procedure used to maximize the loglikelihood function. If this string is set equal to "nlminb", then this function is called; in all other cases, <code>optim</code> is called, with <code>method</code> set equal to the given string. Default value is "nlminb".
<code>control</code>	this parameter is passed to the chosen optimizer, either <code>nlminb</code> or <code>optim</code> ; see the documentation of this function for its usage.

### Details

If `y` is a vector and it is supplied to `mst.mle`, then it is converted to a one-column matrix, and a scalar skew-t distribution is fitted. This is also the mechanism used by `st.mle` which is simply an interface to `mst.mle`.

The parameter `freq` is intended for use with grouped data, setting the values of `y` equal to the central values of the cells; in this case the resulting estimate is an approximation to the exact maximum likelihood estimate. If `freq` is not set, exact maximum likelihood estimation is performed.

Numerical search of the maximum likelihood estimates is performed in a suitable re-parameterization of the original parameters with aid of the selected optimizer (`nlminb` or `optim`) which is supplied with the derivatives of the log-likelihood function. Notice that, in case the optimizer is `optim`, the gradient may or may not be used, depending on which specific method has been selected. On exit from the optimizer, an inverse transformation of the parameters is performed. For a specific description on the re-parametrization adopted, see Section 5.1 and Appendix B of Azzalini & Capitanio (2003).

### Value

A list containing the following components:

<code>call</code>	a string containing the calling statement.
<code>dp</code>	for <code>mst.mle</code> , this is a list containing the direct parameters <code>beta</code> , <code>Omega</code> , <code>alpha</code> . Here, <code>beta</code> is a matrix of regression coefficients with $\dim(\text{beta}) = c(\text{ncol}(X), \text{ncol}(y))$ , <code>Omega</code> is a covariance matrix of order <code>ncol(y)</code> , <code>alpha</code> is a vector of shape parameters of length <code>ncol(y)</code> . For <code>st.mle</code> , <code>dp</code> is a vector of length <code>ncol(X)+3</code> , containing <code>c(beta, omega, alpha, df)</code> , where <code>omega</code> is the square root of <code>Omega</code> .
<code>se</code>	a list containing the components <code>beta</code> , <code>alpha</code> , <code>info</code> . Here, <code>beta</code> and <code>alpha</code> are the standard errors for the corresponding point estimates; <code>info</code> is the observed information matrix for the working parameter, as explained below.
<code>algorithm</code>	the list returned by the chosen optimizer, either <code>nlminb</code> or <code>optim</code> , plus an item with the name of the selected algorithm; see the documentation of either <code>nlminb</code> or <code>optim</code> for explanation of the other components.

## Background

The family of multivariate skew- $t$  distributions is an extension of the multivariate Student's  $t$  family, via the introduction of a `shape` parameter which regulates skewness; when `shape=0`, the skew- $t$  distribution reduces to the usual  $t$  distribution. When `df=Inf` the distribution reduces to the multivariate skew-normal one; see `dmsn`. See the reference below for additional information.

## References

Azzalini, A. and Capitanio, A. (2003). Distributions generated by perturbation of symmetry with emphasis on a multivariate skew  $t$  distribution. The full version of the paper published in abridged form in *J.Roy. Statist. Soc. B* **65**, 367–389, is available at <http://azzalini.stat.unipd.it/SN/se-ext.ps>

## See Also

`dmst`, `msn.mle`, `mst.fit`, `nlminb`, `optim`

## Examples

```
data(ais, package="sn")
attach(ais)
X.mat <- model.matrix(~lbm+sex)
b <- sn.mle(X.mat, bmi)
#
b <- mst.mle(y=cbind(Ht,Wt))
#
# a multivariate regression case:
a <- mst.mle(X=cbind(1,Ht,Wt), y=bmi, control=list(x.tol=1e-6))
#
# refine the previous outcome
a1 <- mst.mle(X=cbind(1,Ht,Wt), y=bmi, control=list(x.tol=1e-9), start=a$dp)
```

---

```
sample.centralmoments
```

*Sample centralmoments*

---

## Description

Computes sample central moments up to a given order and the first moment from the origin

## Usage

```
sample.centralmoments(x, w = rep(1, length(x)), order=4)
```

**Arguments**

<code>x</code>	a vector of sample values
<code>w</code>	an optional vector of weights
<code>order</code>	the maximal order of the central moments to be computed; it must be a positive integer (default value 4)

**Details**

NA's are allowed but removed. Averaging of appropriate quantities is actually performed by `weighted.mean`

**Value**

A vector containing the first sample central moments, in position `[2:order]`, and the first moment from the origin, in the first position of the returned vector

**Note**

The second component of the returned vector (if `order>1`) gives the sample variance; notice that it differs from the value returned by `var(x)`, since this gives the corrected sample variance.

Used in conjunction with `st.cumulants.inversion`, this function allows to fit a skew-t distribution by the methods of moments; see the example below. Note however, that for stability reasons, this is *not* adopted as the standard method for producing initial values of MLE search.

**Author(s)**

Adelchi Azzalini

**See Also**

[st.cumulants.inversion](#), [weighted.mean](#)

**Examples**

```
data(ais, package='sn')
mom <- sample.centralmoments(ais[, "bmi"])
st.cumulants.inversion(cum=c(mom[1:3], mom[4]-3*mom[2]^2))
# parameters of the fitted ST distribution
```

**Description**

This package provides functions related to the skew-normal (SN) and the skew-t (ST) probability distributions, both for the univariate and for the the multivariate case, including regression models.

## Functions

The package includes several functions, a number of which are intended as services to other functions, not really to be called directly by the users. The following ones are those more relevant for practical use.

*SN distribution:* `dsn`, `psn`, `qsn`, `rsn`, `cp.to.dp`, `dp.to.cp`, `sn.mle`, `sn.em`,

for the univariate case, and `dmsn`, `pmsn`, `rmsn`, `msn.fit`, `msn.mle`, `dsn2.plot` for the multivariate case.

*ST distribution:* `dst`, `pst`, `rst`, `st.mle`,

for the univariate case, and `dmst`, `pmst`, `rmst`, `mst.fit`, `mst.mle`, `dst2.plot` for the multivariate case.

It is suggested that a user starts by reading the documentation of (some of) these functions.

## Requirements

R 2.2.0. A few functions make use of package `mnormt`: `psn`, `pmsn`, `pmst`.

## Version

The version level of the package is given by the command `print(.sn.version)`. The most recent version of the package can be obtained from the WWW page: <http://azzalini.stat.unipd.it/SN> which also contains other related material

## Author

Adelchi Azzalini, Dipart. Scienze Statistiche, Università di Padova, Italia. Please send comments, error reports, etc. to the author whose WWW page is <http://azzalini.stat.unipd.it/>.

## Licence

This package and its documentation are usable under the terms of the "GNU General Public License", a copy of which is distributed with the package. While the software is freely usable, it would be appreciated if a reference is inserted in publications or other work which makes use of it; for this purpose, see the command `citation("sn")`.

## Acknowledgements

The package has evolved through several versions, developed over some years.

For versions up to 0.20, the following people and institutions have contributed. Many thanks go to Antonella Capitanio for testing the procedures, and to Brian Ripley, Kurt Hornik and Martin Maechler for useful advice on R. The function `num.deriv2` is based on a similar function written by Monica Chiogna. The first version of this software and part of the associated theoretical work has been developed while the author was at the Nuffield College, Oxford, under the Jemolo Fellowship scheme; the generous support of the college is gratefully acknowledged. Additional support for the development of the theoretical research work has been provided by the "Consiglio Nazionale delle Ricerche" of Italy, grant no.97.01331.CT10.

Versions 0.21 to 0.30 have been supported by "MIUR", Italy, under grant scheme PRIN 2000. Again, thanks to Antonella Capitanio for additional testing of those versions of the package.

---

sn.2logL.profile     *Profile twice loglikelihood for skew-normal models*

---

### Description

Computation and plot of 1-dimensional and 2-dimensional profile 2\*loglikelihood for skew-normal regression models.

### Usage

```
sn.2logL.profile(X=matrix(rep(1, n)), y,
  param.range=c(sqrt(var(y)) * c(2/3, 3/2), -0.95, 0.95),
  use.cp=TRUE, npts=51 %% d, plot.it=TRUE, ...)
```

### Arguments

y	a numeric vector. Missing values (NA's) are not allowed.
X	a matrix of explanatory variables. It must have <code>col(X)</code> equal to <code>length(y)</code> , and it must include a column of 1's if an intercept term is required. Missing values (NA's) are not allowed. If X is missing, a one-column matrix of 1's is created.
param.range	a numeric vector of length either 2 or 4. If the length is 2, the dimensional parameter d is set to 1, and a 1-dimensional profile is computed and plotted, for the shape or skewness parameter (depending on the parametrization adopted; see below); in this case the two value represent the minimum and maximum value for the range of the parameter. If the length of <code>param.range</code> is 4, the first two values determine the range of the scale parameter, the last two give the range of the shape (or skewness) parameter; in this case, <code>d=2</code> .
use.cp	logical value which selects the parametrization adopted. If <code>use.cp=TRUE</code> (default value), the centred parametrization is used, otherwise the direct parametrization is adopted.
npts	number of points (in the scalar case) or grid size (in the 2-dimensional case).
plot.it	logical value which determines if plotting takes place; default is T.
...	any additional parameter is passed to <code>sn.em</code> .

### Details

Likelihood maximization is performed by `sn.em`.

See the reference below for explanation of the two possible parametrizations.

### Value

A list containing the following components

param1	vectors of the parameters values where the function has been evaluated. If <code>d=2</code> , the second vector contains NAs.
--------	---

param.names a character vector of two elements with the names of the param1 and param2.  
 2logL a vector or a matrix which represents twice the profile loglikelihood; this is in the "relative" version, i.e. setting the maximum value to be 0.  
 maximum a numeric value with the maximum which has been subtracted to obtain the "relative" version of 2logL.

### Side Effects

If `plot.it=TRUE`, a plot of the profile twice relative loglikelihood is produced on a graphical device. If `length(fixed.comp)=2`, this plot takes the form of a contour plot whose contour lines correspond to confidence regions of approximate probability levels  $c(0.25, 0.5, 0.75, 0.90, 0.95, 0.99)$ .

### References

Azzalini, A. and Capitanio, A. (1999). Statistical applications of the multivariate skew-normal distribution. *J.Roy.Statist.Soc. B* **61**, 579–602.

### See Also

[sn.em](#), [sn.mle](#)

### Examples

```
data(ais, package="sn")
attach(ais)
a <- sn.2logL.profile(y=bmi)
a <- sn.2logL.profile(y=bmi, use.cp=FALSE, param.range=c(3,6,1,5))
a <- sn.2logL.profile(X=cbind(1,lbm), y=bmi, param.range=c(0,0.9),
  npts=50)
#
data(frontier, package="sn")
a <- sn.2logL.profile(y=frontier, param.range=c(0.8,1.6,10,30),
  use.cp=FALSE, npts=11)
```

---

sn.cumulants

*Cumulants of the skew-normal distribution*

---

### Description

Cumulants of the skew-normal distribution.

### Usage

```
sn.cumulants(location = 0, scale = 1, shape = 0, dp = NULL, n = 4)
```

**Usage**

```
sn.cumulants(location = 0, scale = 1, shape = 0, n = 4)
sn.cumulants(dp=, n = 4)
```

**Arguments**

location	location parameter (vector)
scale	scale parameter (vector)
shape	shape parameter (vector)
dp	a vector of three elements, whose elements are (location, scale, shape) respectively. If dp is specified, then the individual parameters must not be.
n	a scalar integer of the maximal order of cumulants required

**Details**

The moment generating function (hence the cumulant generating function) of the distribution is given in the reference below. The computations method used is proved analytically up to  $n=3$  but it is seen to behave correctly up to the order which was checked ( $n=8$ ).

**Value**

the cumulants up to order  $n$  of the skew-normal distribution with `location=0`, `scale=1` and `shape` as selected.

**References**

Azzalini, A. (1985). A class of distributions which includes the normal ones. *Scand. J. Statist.* **12**, 171-178.

**See Also**

[dsn,zeta](#)

**Examples**

```
sn.cumulants(shape=c(0, 2.5, 5, 10), n=5)
sn.cumulants(dp=c(10, 3, -8), n=6)
```

---

 sn.Einfo

*Expected Fisher information for SN distribution parameters*


---

### Description

Computes expected Fisher information for parameters of simple sample having one-dimensional skew-normal (SN) distribution or regression model having SN errors, in the DP and CP parametrizations.

### Usage

```
sn.Einfo(dp=NULL, cp=NULL, n=1, x=NULL)
```

### Arguments

dp, cp	vector of direct or centred parameters, respectively; one of the two vectors must be supplied, but not both. See below for more details.
n	sample size; if this parameter is supplied, then x must not be.
x	design matrix of the regression model; if this parameter is supplied, then n must not be.

### Value

a list containing the following components:

dp, cp	DP and CP parameters; one of the two vectors is the one supplied on input, the other one matches the previous one in the alternative parametrization
info.dp, info.cp	matrices of Fisher expected information in the two parametrizations
se.dp, se.cp	vectors of standard errors in the two parametrizations
D	matrix of derivatives of DP parameters with respect to CP parameters

### Details

When x is not set, then a simple random sample is assumed and a matrix x with a single column of all 1's is constructed; in this case, the supplied vector dp or cp must have length 3. If x is set, then the supplied vector of parameters must have length  $n \text{col}(x) + 2$ .

### Background

For the description of the DP and CP parametrizations and for the expression of the expected Fisher information of the DP parametrizations, see Azzalini (1985). Further discussion is in Azzalini and Capitanio (1999).

## References

- Azzalini, A. (1985). A class of distributions which includes the normal ones. *Scand. J. Statist.* **12**, 171-178.
- Azzalini, A. and Capitanio, A. (1999). Statistical applications of the multivariate skew-normal distribution. *J.Roy.Statist.Soc. B* **61**, 579–602.

## See Also

[dsn](#), [cp.to.dp](#), [dp.to.cp](#)

## Examples

```
info <- sn.Einfo(dp=c(0,1,5), n=3)
#
data(ais, package="sn")
M <- model.matrix(~ais$"Ht")
mle <- sn.mle(X=M, y=ais$"Wt", plot.it=FALSE)
info <- sn.Einfo(cp=mle$cp, x=M)
```

---

sn.em

*Fitting Skew-normal variables using the EM algorithm*

---

## Description

Fits a skew-normal (SN) distribution to data, or fits a linear regression model with skew-normal errors, using the EM algorithm to locate the MLE estimate. The estimation procedure can be global or it can fix some components of the parameters vector.

## Usage

```
sn.em(X, y, fixed, p.eps=0.0001, l.eps=0.01, trace=FALSE, data=FALSE)
```

## Arguments

- |                    |   |
|--------------------|---|
| <code>y</code>     | a vector containing the observed variable. This is the response variable in case of linear regression.  |
| <code>X</code>     | a matrix of explanatory variables. If <code>X</code> is missing, then a one-column matrix of all 1's is created. If <code>X</code> is supplied, and an intercept term is required, then it must include a column of 1's.  |
| <code>fixed</code> | a vector of length 3, indicating which components of the parameter vector must be regarded as fixed. In <code>fixed=c(NA, NA, NA)</code> , which is the default setting, a global maximization is performed. If the 3rd component is given a value, then maximization is performed keeping that value fixed for the shape parameter. If the 3rd and 2nd parameters are fixed, then the scale and the shape parameter are kept fixed. No other patterns of the fixed values are allowed. |
| <code>p.eps</code> | numerical value which regulates the parameter convergence tolerance.  |

<code>l.eps</code>	numerical value which regulates the log-likelihood convergence tolerance.
<code>trace</code>	logical value which controls printing of the algorithm convergence. If <code>trace=TRUE</code> , details are printed. Default value is <code>F</code> .
<code>data</code>	logical value. If <code>data=TRUE</code> , the returned list includes the original data. Default value is <code>data=FALSE</code> .

### Details

The function works using the direct parametrization; on convergence, the output is then given in both parametrizations.

This function is based on the EM algorithm; it is generally quite slow, but it appears to be very robust. See `sn.mle` for an alternative method, which also returns standard errors.

### Value

a list with the following components:

<code>dp</code>	a vector of the direct parameters, as explained in the references below.
<code>cp</code>	a vector of the centred parameters, as explained in the references below.
<code>logL</code>	the log-likelihood at convergence.
<code>data</code>	optionally (if <code>data=TRUE</code> ), a list containing $X$ and $y$ , as supplied on input, and a vector of <code>residuals</code> , which should have an approximate SN distribution with <code>location=0</code> and <code>scale=1</code> , in the direct parametrization.

### Background

Background information on the SN distribution is given by Azzalini (1985). See Azzalini and Capitanio (1999) for a more detailed discussion of the direct and centred parametrizations.

### References

- Azzalini, A. (1985). A class of distributions which includes the normal ones. *Scand. J. Statist.* **12**, 171-178.
- Azzalini, A. and Capitanio, A. (1999). Statistical applications of the multivariate skew-normal distribution. *J.Roy.Statist.Soc. B* **61**, 579-602.

### See Also

[dsn](#), [sn.mle](#), [cp.to.dp](#)

### Examples

```
data(ais, package="sn")
attach(ais)
#
a<-sn.em(y=bmi)
#
a<-sn.em(X=cbind(1, lbm, lbm^2), y=bmi)
```

```
#
M<-model.matrix(~lbm+I(ais$sex))
b<-sn.em(M,bmi)
#
fit <- sn.em(y=bmi, fixed=c(NA, 2, 3), l.eps=0.001)
```

sn.mle

*Maximum likelihood estimation for skew-normal models***Description**

Fits a skew-normal (SN) distribution to data, or fits a linear regression model with skew-normal errors, using maximum likelihood estimation.

**Usage**

```
sn.mle(X, y, cp, plot.it=TRUE, trace=FALSE, method="L-BFGS-B",
       control=list(maxit=100))
```

**Arguments**

<code>y</code>	a vector containing the observed variable. This is the response variable in case of linear regression. Missing values (NAs) are not allowed.
<code>X</code>	a matrix of explanatory variables. If <code>X</code> is missing, then a one-column matrix of all 1's is created. If <code>X</code> is supplied, then it must include a column of 1's. Missing values (NAs) are not allowed.
<code>cp</code>	a vector of initial values for the centred parameters, with <code>length(cp) = ncol(X) + 2</code>
<code>plot.it</code>	logical value, If <code>plot.it=TRUE</code> (default), a plot of the nonparametric estimate of variable <code>y</code> (or the residuals, in the case of regression), and the parametric fit is superimposed. See below for details.
<code>trace</code>	logical value which controls printing of the algorithm convergence. If <code>trace=TRUE</code> , details are printed. Default value is <code>FALSE</code> .
<code>method</code>	this parameter is just passed to the optimizer <code>optim</code> ; see the documentation of this function for its usage. Default value is "L-BFGS-B".
<code>control</code>	this parameter is just passed to the optimizer <code>optim</code> ; see the documentation of this function for its usage.

**Details**

The optimizer `optim` is used, supplying the gradient of the log-likelihood. Convergence is generally fast and reliable, but inspection of the returned `message` from `optim` is always appropriate. In suspect cases, re-run the function changing the starting `cp` vector.

If plotting operates, the function `sm.density` of the package `sm` is searched; this library is associated with the book by Bowman and Azzalini (1997). If `sm.density` is not found, an histogram is plotted.

To fit a skew-normal distribution to grouped data by exact maximum likelihood estimation, use `sn.mle.grouped`.

**Value**

a list containing the following components:

<code>call</code>	a string containing the calling statement
<code>cp</code>	a vector of length <code>ncol(X) + 2</code> with the centred parameters
<code>logL</code>	the log-likelihood at convergence
<code>se</code>	a vector of standard errors for the <code>cp</code> component
<code>info</code>	the observed information matrix for the <code>cp</code> component
<code>optim</code>	the list returned by the optimizer <code>optim</code> ; see the documentation of this function for explanation of its components.

**Side Effects**

If `plot.it=TRUE` and a graphical device is active, a plot is produced, as described above.

**Background**

Background information on the SN distribution is given by Azzalini (1985). See also Azzalini and Capitanio (1999), for an additional discussion of the centred parametrization.

**References**

Azzalini, A. (1985). A class of distributions which includes the normal ones. *Scand. J. Statist.* **12**, 171-178.

Azzalini, A. and Capitanio, A. (1999). Statistical applications of the multivariate skew-normal distribution. *J.Roy.Statist.Soc. B* **61**, 579-602.

Bowman, A.W. and Azzalini, A. (1997). *Applied Smoothing Techniques for Data Analysis: the Kernel Approach with S-Plus Illustrations*. Oxford University Press, Oxford.

**See Also**

[dsn](#), [sn.em](#), [msn.mle](#), [optim](#), [sn.mmle](#), [sn.mle.grouped](#)

**Examples**

```
data(ais, package="sn")
attach(ais)
a<-sn.mle(y=bmi)
#
a<-sn.mle(X=cbind(1, lbm), y=bmi)
#
b<-sn.mle(X=model.matrix(~lbm+sex), y=bmi)
```

---

sn.mle.grouped	<i>Maximum likelihood estimation of SN and ST distribution for grouped data</i>
----------------	---

---

**Description**

Fits a skew-normal (SN) and skew-t (ST) distribution to grouped data using exact maximum likelihood estimation

**Usage**

```
sn.mle.grouped(breaks, freq, trace=FALSE, start=NA)
st.mle.grouped(breaks, freq, trace=FALSE, start=NA)
```

**Arguments**

<code>breaks</code>	a vector containing the cut points of the groups, given in ascending order. The last value can be <code>Inf</code> , the first one can be <code>-Inf</code>
<code>freq</code>	a vector containing the observed frequencies corresponding to the intervals defined by <code>breaks</code> ; it is required that <code>length(freq) = length(breaks) - 1</code>
<code>trace</code>	logical value which controls printing of the algorithm convergence. If <code>trace=TRUE</code> , details are printed. Default value is <code>FALSE</code>
<code>start</code>	vector of length with initial values for the working parameters: location, <code>log(scale)</code> , shape and (for the ST case) <code>log(df)</code> .

**Details**

The optimizer `optim` is used, supplying the log-likelihood function for grouped data, namely the multinomial expression whose probabilities are assigned by the SN or ST distribution to the given intervals.

**Value**

a list containing the following components:

<code>call</code>	a string containing the calling statement
<code>dp</code>	for the SN case, a vector of length 3 containing the location, scale and shape parameter; for the ST case, there is an extra parameter, the degrees of freedom
<code>end</code>	a vector of length 3 or 4 containing the working parameters; this vector can be used as a <code>start</code> for a new call to the function
<code>opt</code>	the list returned by the optimizer <code>optim</code> ; see the documentation of this function for explanation of its components.

**Background**

Background information on the SN distribution is given by Azzalini (1985); see also Azzalini and Capitanio (1999). For the ST distribution, see Azzalini and Capitanio (2003).

## References

- Azzalini, A. (1985). A class of distributions which includes the normal ones. *Scand. J. Statist.* **12**, 171-178.
- Azzalini, A. and Capitanio, A. (1999). Statistical applications of the multivariate skew-normal distribution. *J.Roy.Statist.Soc. B* **61**, 579–602.
- Azzalini, A. and Capitanio, A. (2003). Distributions generated by perturbation of symmetry with emphasis on a multivariate skew-*t* distribution. *J.Roy. Statist. Soc. B* **65**, 367–389.

## See Also

[dsn](#), [dst](#), [optim](#), [sn.mle](#)

## Examples

```
data(ais, package="sn")
attach(ais)
breaks<- c(130,160, seq(170, 190, by=2.5), 200, 230)
f <- cut(Ht[sex=="female"], breaks = breaks)
freq <- tabulate(f, length(levels(f)))
b1 <- sn.mle.grouped(breaks, freq)
b2 <- st.mle.grouped(breaks, freq, start=c(b1$end, log(5)))
print(b2$dp)
#
us.income <- c(0, seq(from=0.2, to=1.8, by=0.1), 2.0, 2.5, 5.0, Inf)
mid <- (us.income[-1]+us.income[-length(us.income)])/2
mid[length(mid)] <- 6.5
cum.freq<- c(1.78, 3.25, 5.56, 8.16, 11.12, 14.21, 17.54, 20.78, 24.00,
             27.52, 30.77, 34.21, 37.56, 40.70, 44.41, 47.85, 51.22,
             57.60, 72.12, 96.40, 100) / 100
freq<- round(diff(c(0, cum.freq*34660)))
a <- st.mle.grouped(breaks=log(us.income), freq, trace=TRUE,
                  start=c(1.2, log(0.9), -2.1, log(20)))
print(a$dp)
```

---

sn.mmle

*Modified maximum likelihood estimation for skew-normal and skew-t models*

---

## Description

Fits a one-dimensional skew-normal (SN) or skew-t (ST) distribution to data, or fits a linear regression model with skew-normal errors, using a modification of maximum likelihood estimation.

## Usage

```
sn.mmle(X, y, plot.it=TRUE, trace=FALSE, ...)
st.mmle(X, y, df, trace=FALSE)
```

### Arguments

<code>y</code>	a vector containing the observed variable. This is the response variable in case of linear regression. Missing values (NAs) are not allowed.
<code>X</code>	a matrix of explanatory variables. If <code>X</code> is missing, then a one-column matrix of all 1's is created. If <code>X</code> has only one column, then it is assumed to be made of 1's. Missing values (NAs) are not allowed.
<code>df</code>	a positive real value representing the degrees of freedom, in the ST case
<code>plot.it</code>	logical value, If <code>plot.it=TRUE</code> (default), a plot of the nonparametric estimate of variable <code>y</code> (or the residuals, in the case of regression), and the parametric fit is superimposed. See below for details.
<code>trace</code>	logical value which controls printing of the algorithm convergence. If <code>trace=TRUE</code> , details are printed. Default value is <code>FALSE</code> .
<code>...</code>	Additional arguments passed to <code>sn.mmle</code>

### Details

If plotting operates, the function `sm.density` of the library `sm` is searched. If `sm.density` is not found, an histogram is plotted.

### Value

from `sn.mmle`, a list containing the following components:

<code>call</code>	a string containing the calling statement
<code>dp</code>	a vector of length <code>ncol(X)+2</code> with estimates of the direct parameters
<code>se</code>	a vector of standard errors for the parameters
<code>Einfo</code>	the expected Fisher information matrix for the parameters

from `st.mmle` only the `call` and `dp` components are returned

### Background

Maximum likelihood estimation for SN and ST models can lead to estimates of the shape parameters on the frontier (that is `Inf` for the DP parameters); see Azzalini and Capitanio (1999) for a discussion of this aspect in the SN case. To avoid this situation, an alternative estimation criterion is the method of Sartori-Firth, which involves first regular maximum estimation and subsequent re-estimation of the shape parameter using a modified score function; see the references below for a full discussion. The effect of this modification is "negligible" for large sample size, but it avoids estimates of the frontier of the parameter space.

### References

- Azzalini, A. and Capitanio, A. (1999). Statistical applications of the multivariate skew-normal distribution. *J.Roy.Statist.Soc. B* **61**, 579–602.
- Firth, D. (1993). Bias reduction of maximum likelihood estimates. *Biometrika* **80**, 27–38. (Corr: 95V82 p.667).
- Sartori, N. (2006). Bias prevention of maximum likelihood estimates for scalar skew normal and skew  $t$  distributions. *J. Statist. Plann. Inf.* **136**, 4259–4275.

**See Also**

[sn.mle](#), [sn.Einfo](#)

**Examples**

```
data(ais, package="sn")
attach(ais)
a <- sn.mmle(y=bmi)
#
M <- model.matrix(~lbm+sex)
b <- sn.mmle(M,bmi)
```

---

st.2logL.profile    *Profile twice loglikelihood for skew-t models*

---

**Description**

Computation and plot of 1-dimensional and 2-dimensional profile (negative) twice log-likelihood function for skew-t regression models.

**Usage**

```
st.2logL.profile(X=matrix(rep(1, n)), y, freq, trace=FALSE,
  fixed.comp = c(ncol(X)+2, ncol(X)+3),
  fixed.values = cbind(c(-4, 4), log(c(1, 25))),
  npts=51/length(fixed.comp), plot.it=TRUE, ...)
```

**Arguments**

X	a matrix of explanatory variables; must have <code>nrow(X)</code> equal to <code>length(y)</code> . Missing values (NA) are not allowed. If X is missing, a one-column matrix of 1's is created.
y	a numeric vector. Missing values (NAs) are not allowed.
freq	a vector of weights. If missing, a vector of 1's is created; otherwise it must have the same length of y.
trace	logical value which controls printing of the algorithm convergence. If <code>trace=TRUE</code> , details are printed. Default value is <code>FALSE</code> .
fixed.comp	a vector containing the subset of the parameters for which the profile log-likelihood function is required; it can be of length 1 or 2. The set of components of the parameters are <code>beta</code> , <code>log(omega)</code> , <code>alpha</code> , <code>log(df)</code> , where <code>beta</code> represents the regression parameters and has <code>ncol(X)</code> components.
fixed.values	a numeric vector of values or a matrix with two columns, giving the range spanned by the selected parameters.
npts	number of points on each parameter interval for which the function must be evaluated.

plot.it	logical value; if plot.it=TRUE (default value) a graphical display is produced.
...	graphical parameter passed either to plot or to contour, depending on whether length(fixed.comp) is 1 or 2.

### Value

A list containing the following components:

call	the matched call.
param1	vector of values of the first parameters values where the function has been evaluated.
param2	vectors of the parameters values where the function has been evaluated. If length(fixed.comp)=1, the second vector contains NAs.
deviance	a vector or a matrix which represents twice the negative relative profile loglikelihood; this is in the "relative" version, i.e. setting the maximum value to be 0.
max.logL	a numeric value with the maximum which has been added to obtain the "relative" version of deviance.
best	a list with the output of optim at the best evaluated point, that is the one with higher log-likelihood.

### Side Effects

If plot.it=TRUE, a plot of the profile twice relative loglikelihood is produced on a graphical device. If length(fixed.comp)=1, a plot of the 'deviance' is produced as a function of the chosen parameter component. If length(fixed.comp)=2, this plot takes the form of a contour plot whose contour lines correspond to confidence regions of approximate probability levels  $c(0.25, 0.5, 0.75, 0.90, 0.95, 0.99)$ .

### Background

The family of skew-t distributions is an extension of the Student's t family, via the introduction of a shape parameter which regulates skewness; when shape=0, the skew-t distribution reduces to the usual t distribution. A multivariate version of the distribution exists. See the reference below for additional information.

### References

Azzalini, A. and Capitanio, A. (2003). Distributions generated by perturbation of symmetry with emphasis on a multivariate skew *t* distribution. *J.Roy. Statist. Soc. B* **65**, 367–389.

### See Also

[st.mle](#), [sn.2logL.profile](#)

**Examples**

```

data(ais, package="sn")
attach(ais)
a <- st.2logL.profile(y=bmi, xlab="alpha", ylab="log(df)")
a <- st.2logL.profile(y=bmi, fixed.comp=4, fixed.values=log(c(1,25)))
a <- st.2logL.profile(X=cbind(1,lbm), y=bmi, fixed.comp=5,
  fixed.values=log(c(5,25)), xlab="log(df)", npts=50)
a <- st.2logL.profile(X=cbind(1,Ht), y=Wt, fixed.comp=c(4,5),
  fixed.values=cbind(c(-1,5), log(c(2,25))),
  xlab="alpha", ylab="log(df)")

```

---

st.cumulants

*Cumulants of the skew-t distribution*


---

**Description**

Cumulants of the skew-t distribution and inverse matching

**Usage**

```

st.cumulants(location = 0, scale = 1, shape = 0, df = Inf, dp = NULL, n = 4)
st.cumulants.inversion(cum, abstol = 1e-08)

```

**Usage**

```

st.cumulants(location = 0, scale = 1, shape = 0, df = Inf, n = 4)
st.cumulants(dp=, n = 4)
st.cumulants.inversion(cum, abstol = 1e-08)

```

**Arguments**

location	location parameter (vector)
scale	scale parameter (vector)
shape	shape parameter (vector)
df	degrees of freedom (scalar); default is df=Inf which corresponds to the skew-normal distribution.
dp	a vector of four elements, whose elements are (location, scale, shape, df) respectively. If dp is specified, then the individual parameters must not be.
n	a scalar integer of the maximal order of cumulants required; it must be from 1 to 4 and smaller than df
cum	a vector of 4 elements which are taken to represent the first 4 cumulants of a skew-t distribution (hence the second term must be positive)
abstol	a scalar which regulates the accuracy of the cumulants matching (default value 1e-08)

## Details

Expressions of the moments and other details on the skew-t distribution are given in the reference below. These formulae are used by `st.cumulants` to compute the cumulants.

`st.cumulants.inversion` searches the set of `shape` and `df` parameters of the skew-t family, attempting to match the third and fourth cumulants with those of the supplied vector `cum`. This search is done numerically twice, once using `optim` and a second time using `nlminb`, to the accuracy `abstol`; the best matching solution is retained. If the required accuracy of the matching is not achieved by any of the two methods, a warning message is issued. After this step, the other two parameters (`location` and `scale`) are computed via simple algebra.

## Value

`st.cumulants` computes the cumulants up to order `n` of the skew-t distribution with the selected parameters. The returned object is a vector of length `n` if the parameters are all scalar, otherwise a matrix with `n` columns.

`st.cumulants.inversion` returns a vector with the `dp` parameters of the matching skew-t distribution

## Note

The joint use `st.cumulants.inversion` and `sample.centralmoments` allows to fit a skew-t distribution by the methods of moments; see the example below. Note however, that for stability reasons, this is *not* adopted as the standard method for producing initial values of MLE search.

## References

Azzalini, A. and Capitanio, A. (2003). Distributions generated by perturbation of symmetry with emphasis on a multivariate skew-t distribution. *J. Roy. Statist. Soc. B* **65**, 367–389.

## See Also

[sn.cumulants](#), [dst](#), [sample.centralmoments](#), [optim](#), [nlminb](#)

## Examples

```
st.cumulants(shape=c(0,3,9), df=5)
cum <- st.cumulants(dp=c(10, 2, -8, 5.2))
st.cumulants.inversion(cum)
#
data(ais, package='sn')
mom <- sample.centralmoments(ais[, "bmi"])
st.cumulants.inversion(cum=c(mom[1:3], mom[4]-3*mom[2]^2))
# parameters of the ST distribution fitted by method of moments
```

---

T.Owen

*Owen's function*

---

### Description

Evaluates function  $T(h,a)$  studied by D.B.Owen

### Usage

```
T.Owen(h, a, jmax=50, cut.point=6)
```

### Arguments

<code>h</code>	a numerical vector. Missing values (NAs) and <code>Inf</code> are allowed.
<code>a</code>	a numerical scalar. <code>Inf</code> is allowed.
<code>jmax</code>	an integer scalar value which regulates the accuracy of the result. See the section Details below for explanation.
<code>cut.point</code>	a scalar value which regulates the behaviour of the algorithm, as explained by the details below.

### Details

If  $a > 1$  and  $0 < h \leq \text{cut.point}$ , a series expansion is used, truncated after `jmax` terms. If  $a > 1$  and  $h > \text{cut.point}$ , an asymptotic approximation is used. In the other cases, various reflection properties of the function are exploited. See the reference below for more information.

### Value

a numerical vector

### Background

The function  $T(h,a)$  is useful for the computation of the bivariate normal distribution function and related quantities, including the distribution function of a skew-normal variate, `psn`. See the reference below for more information on  $T(h,a)$ .

### References

Owen, D. B. (1956). Tables for computing bivariate normal probabilities. *Ann. Math. Statist.* **27**, 1075-1090.

### See Also

[psn](#)

### Examples

```
owen <- T.Owen(1:10, 2)
```

zeta

*Function 'log(2\*pnorm(x))' and its derivatives***Description**

The function  $\log(2 * (\text{pnorm}(x)))$  and its derivatives, including inverse Mills ratio.

**Usage**

```
zeta(k, x)
```

**Arguments**

`k` an integer scalar between 0 and 5.  
`x` a vector. Missing values (NAs) and Infs are allowed

**Details**

For `k` between 0 and 5, the derivative of order `k` of  $\log(2 * \text{pnorm}(x))$  is evaluated; the derivative of order `k=0` refers to the function itself. If `k` is not integer, it is converted to integer and a warning message is generated. If `k < 0` or `k > 5`, NULL is returned.

The computation for `k > 1` is reduced to the case `k=1`, making use of expressions given by Azzalini and Capitanio (1999). For numerical stability, the evaluation of `zeta(1, x)` when `x < -50` makes use of the asymptotic expansion (26.2.13) in Abramowitz and Stegun (1964).

`zeta(1, -x)` equals  $\text{dnorm}(x) / \text{pnorm}(-x)$  (in principle, apart from the asymptotic expansion mentioned above), called the *inverse Mills ratio*.

**Value**

a vector giving the `k`-th order derivative evaluated at `x`

**References**

Abramowitz, M. and Stegun, I. A., editors (1964). *Handbook of Mathematical Functions*. Dover Publications.

Azzalini, A. and Capitanio, A. (1999). Statistical applications of the multivariate skew-normal distribution. Technical report available at <http://azzalini.stat.unipd.it/SN>. An abridged version is published in *J.Roy.Statist.Soc. B* **61**, 579–602.

**Examples**

```
y <- zeta(2, seq(-20, 20, by=0.5))
#
for(k in 0:5) curve(zeta(k, x), from=-1.5, to=5, col = k+2, add = k > 0)
legend(3.5, -0.5, legend=as.character(0:5), col=2:7, lty=1)
```

# Index

## \*Topic **datasets**

ais, 1  
frontier, 13

## \*Topic **distribution**

cp.to.dp, 2  
dmsn, 3  
dmst, 5  
dsn, 7  
dsn2.plot, 8  
dst, 10  
dst2.plot, 11  
gammal.to.lambda, 13  
msn.affine, 14  
msn.cond.plot, 15  
msn.conditional, 17  
msn.fit, 18  
msn.marginal, 20  
msn.mle, 21  
msn.quantities, 23  
mst.fit, 25  
mst.mle, 27  
SN, 31  
sn.2logL.profile, 32  
sn.cumulants, 34  
sn.Einfo, 35  
sn.em, 36  
sn.mle, 38  
sn.mle.grouped, 40  
sn.mmle, 42  
st.2logL.profile, 44  
st.cumulants, 46

## \*Topic **math**

T.Owen, 47  
zeta, 48

## \*Topic **multivariate**

dmsn, 3  
dmst, 5  
msn.affine, 14  
msn.cond.plot, 15

msn.conditional, 17  
msn.marginal, 20  
msn.quantities, 23  
SN, 31

## \*Topic **regression**

msn.fit, 18  
msn.mle, 21  
mst.fit, 25  
mst.mle, 27  
SN, 31  
sn.em, 36  
sn.mle, 38  
sn.mmle, 42

## \*Topic **univar**

sample.centralmoments, 30

ais, 1

biv.nt.prob, 8

cp.to.dp, 2, 31, 36, 38

dmnorm, 4

dmsn, 3, 6, 8, 9, 15, 16, 20, 21, 23, 25, 27, 31

dmst, 4, 5, 11, 12, 15, 27, 29, 31

dmt, 6

dp.to.cp, 31, 36

dp.to.cp (*cp.to.dp*), 2

dsn, 4, 7, 11, 14, 15, 31, 35, 36, 38, 40, 41

dsn2.plot, 8, 12, 31

dst, 6, 8, 10, 15, 31, 41, 47

dst2.plot, 11, 31

frontier, 13

gammal.to.lambda, 13

msn.affine, 14, 21

msn.cond.plot, 15, 18

msn.conditional, 16, 17, 21

msn.fit, 18, 23, 27, 31

`msn.marginal`, 18, 20  
`msn.mle`, 20, 21, 29, 31, 40  
`msn.quantities`, 23  
`mst.affine` (`msn.affine`), 14  
`mst.fit`, 20, 25, 29, 31  
`mst.mle`, 27, 27, 31  
  
`nlminb`, 23, 29, 47  
  
`optim`, 23, 29, 40, 41, 47  
  
`pmsn`, 31  
`pmsn` (`dmsn`), 3  
`pmst`, 31  
`pmst` (`dmst`), 5  
`psn`, 11, 31, 48  
`psn` (`dsn`), 7  
`pst`, 31  
`pst` (`dst`), 10  
  
`qsn`, 31  
`qsn` (`dsn`), 7  
`qst` (`dst`), 10  
  
`rmsn`, 31  
`rmsn` (`dmsn`), 3  
`rmst`, 31  
`rmst` (`dmst`), 5  
`rsn`, 31  
`rsn` (`dsn`), 7  
`rst`, 31  
`rst` (`dst`), 10  
  
`sample.centralmoments`, 30, 47  
`SN`, 31  
`sn-package` (`SN`), 31  
`sn.2logL.profile`, 32, 45  
`sn.cumulants`, 34, 47  
`sn.Einfo`, 35, 43  
`sn.em`, 3, 31, 33, 36, 40  
`sn.mle`, 3, 31, 33, 38, 38, 41, 43  
`sn.mle.grouped`, 40, 40  
`sn.mmle`, 40, 42  
`st.2logL.profile`, 44  
`st.cumulants`, 46  
`st.cumulants.inversion`, 30  
`st.mle`, 31, 45  
`st.mle` (`mst.mle`), 27  
`st.mle.grouped` (`sn.mle.grouped`),  
40