Package ‘alphastable’

October 12, 2022

Title  Inference for Stable Distribution
Version  0.2.1
Author  Mahdi Teimouri, Adel Mohammadpour, and Saralees Nadarajah
Maintainer  Mahdi Teimouri <teimouri@aut.ac.ir>
Description  Developed to perform the tasks given by the following. 1-computing the probability density function and distribution function of a univariate stable distribution; 2- generating from univariate stable, truncated stable, multivariate elliptically contoured stable, and bivariate strictly stable distributions; 3- estimating the parameters of univariate symmetric stable, skew stable, Cauchy, multivariate elliptically contoured stable, and multivariate strictly stable distributions; 4- estimating the parameters of the mixture of symmetric stable and mixture of Cauchy distributions.
License  GPL (>= 2)
Encoding  UTF-8
LazyData  true
Depends  R(>= 3.4.0)
Imports  mvtnorm, nnls, stabledist, nlme
Suggests  Matrix, fBasics, FMStable, RUnit, Rmpfr, sfsmisc
RoxygenNote  6.0.1
NeedsCompilation  no
Repository  CRAN
Date/Publication  2019-09-10 19:00:02 UTC

R topics documented:

mdstab.elliptical .......................................................... 2
mfitstab.elliptical ....................................................... 3
mfitstab.ustat .......................................................... 4
mrstab ................................................................. 5
mrstab.elliptical ....................................................... 6
udstab ................................................................. 7
ufitstab.cauchy ......................................................... 8
mdstab.elliptical

Description
computes the probability density function of a d-dimensional elliptically contoured stable distribution at a given point in \( \mathbb{R}^d \), see Teimouri et al. (2018).

Usage
\[
\text{mdstab.elliptical}(x, \alpha, \Sigma, \mu)
\]

Arguments
- \( x \) : vector of real values in \( \mathbb{R}^d \)
- \( \alpha \) : tail index parameter
- \( \Sigma \) : \( d \) by \( d \) positive definite dispersion matrix
- \( \mu \) : location vector in \( \mathbb{R}^d \)

Value
a numeric value

Note
\text{mdstab.elliptical()} computes the probability density function of an \( d \)-dimensional elliptically contoured stable distribution using either asymptotic series or Monte Carlo approximation.

Author(s)
Mahdi Teimouri, Adel Mohammadpour, and Saralees Nadarajah

References
Examples

```r
library("stabledist")
mdstab.elliptical(c(5,5),1.2,matrix(c(1,0.5,0.5,1),2,2),c(0,0))
```

Description

estimates the parameters of a \(d\)-dimensional elliptically contoured stable distribution, see Teimouri et al. (2018).

Usage

```r
mfitstab.elliptical(yy, alpha0, Sigma0, Mu0)
```

Arguments

- `yy`: vector of \(d\)-dimensional realizations
- `alpha0`: initial value of the tail index parameter to start the EM algorithm
- `Sigma0`: initial value of the dispersion matrix to start the EM algorithm
- `Mu0`: initial value of the location vector to start the EM algorithm

Value

- `alpha`: estimated value of the tail index parameter
- `Sigma`: estimated value of the dispersion matrix
- `Mu`: estimated value of the location vector

Author(s)

Mahdi Teimouri, Adel Mohammadpour, and Saralees Nadarajah

References

Examples

# Here we follow for applying the EM algorithm to Z=(x1, x2)^T using the
# initial values alpha0=1, Sigma0=matrix(c(0.75,0.25,0.25,0.75),2,2), and
# Mu0=(0.5,0.5)^T.
library("stabledist")
x1<-urstab(100,1.2,0,1,2,0)
x2<-urstab(100,1.2,0,0.5,2,0)
z<-cbind(x1,x2)
mfitstab.elliptical(z,1,matrix(c(0.75,0.25,0.25,0.75),2,2),c(0.5,0.5))

Description

estimates the parameters of a strictly bivariate stable distribution using approaches proposed by Mohammadi et al. (2015)<doi.org/10.1007/s00184-014-0515-7> and Teimouri et al. (2017)<doi.org/10.1155/2017/3483827>. The estimated parameters are tail index and discretized spectral measure on m equidistant points located on unit sphere in R^2.

Usage

mfitstab.ustat(u,m,method=method)

Arguments

u an n by 2 vector of observations
m number of masses located on unit circle in R^2
method integer values 1 or 2, respectively, corresponds to the method given by Teimouri et al. (2017) and Mohammadi et al. (2015)

Value

alpha estimated value of tail index
mass estimated value of discrete spectral measure

Author(s)

Mahdî Teimouri, Adel Mohammadpour, and Saralees Nadarajah

References

Examples

# Here, for example, we are interested to estimate the parameters of a bivariate
# stable distribution. For this, two sets of n=400 iid realizations which are
# assumed to distributed jointly as a strictly bivariate stable distribution with
# tail index alpha=1.2 are simulated. Considering m=4, masses of the discrete spectral
# measure are addressed by s_j=(cos(2*pi(j-1)/m), sin (2*pi(j-1)/m)); for j=1,...,4.
library("nnls")
x1<-urstab(400,1.2,-0.50,1,0,0)
x2<-urstab(400,1.2,0.50,0.5,0,0)
z<-cbind(x1,x2)
mfitstab.ustat(z,4,1)

Description

generates iid realizations from bivariate stable vectors using the methodology proposed by Modares and Nolan (1994).

Usage

mrstab(n, m, alpha, Gamma, Mu)

Arguments

n sample size
m number of masses
alpha tail index parameter
Gamma vector of masses
Mu location vector

Value

a vector of n numeric values

Note

mrstab() assumes that masses are located at unit sphere with addresses s_j=(cos(2*pi(j-1)/m),
sin(2*pi(j-1)/m)); for j=1,...,4.

Author(s)

Mahdi Teimouri, Adel Mohammadpour, and Saralees Nadarajah
References

Examples
# We use the following command to simulate n=200 iid vectors from a two-dimensional stable
# distribution with alpha=1.3, with a vector of 4 masses as gamma=(0.1,0.5,0.5,0.1)^T, 
# and mu=(0,0)^T.
library("stabledist")
mrstab(200,4,1.3,c(0.1,0.5,0.5,0.1),c(0,0))

Description

Usage
mrstab.elliptical(n, alpha, Sigma, Mu)

Arguments
n sample size
alpha tail index parameter
Sigma d by d positive definite dispersion matrix
Mu location vector in R^d

Details
mrstab.elliptical() needs to install the mvtnorm package

Value
an n by d matrix of numeric values

Note
mrstab.elliptical() generates iid realizations from d-dimensional elliptically contoured stable
distribution based on definitions given by Nolan (2013) and Samorodnitsky and Taqqu (1994)

Author(s)
Mahdi Teimouri, Adel Mohammadpour, and Saralees Nadarajah
References


Examples

# In the following example, we simulate n=200 iid vectors of a two-dimensional elliptically contoured stable distribution with parameters alpha=1.3, Sigma=matrix(c(1,.5,.5,1),2,2), # and mu=(0,0)^T.
library("mvtnorm")
library("stabledist")
mrstab.elliptical(200,1.3,matrix(c(1,.5,.5,1),ncol=2,nrow=2),c(0,0))

udstab

Description

computes the probability density function (pdf) of the univariate stable distribution based on formulas given by Nolan (1997) <doi.org/10.1080/15326349708807450> and asymptotic series, see Teimouri and Amindavar (2008).

Usage

udstab(x, alpha, beta, sigma, mu, param)

Arguments

x point at which the pdf is computed
alpha tail index parameter
beta skewness parameter
sigma scale parameter
mu location parameter
param kind of parameterization; must be 0 or 1 for S_0 and S_1 parameterizations, respectively

Value

a numeric value

Note

udstab() computes the pdf of univariate stable distribution using asymptotic series within their convergence regions. For points outside of convergence regions, the pdf is computed using stabledist package based on formulas given by Nolan (1997). So, to compute the pdf using the upstab() we may need to install stabledist package.
ufitstab.cauchy

Author(s)

Mahdi Teimouri, Adel Mohammadpour, and Saralees Nadarajah

References


Examples

# In the following, we compute the pdf of a univariate stable distribution at point 2
# with parameters alpha=1.2, beta=0.9, sigma=1, and mu=0 in S_0 parameterization.
library("stabledist")
udstab(2,1.2,0.9,1,0,1)

ufitstab.cauchy

Description

estimates the parameters of the Cauchy distribution. Given the initial values of the skewness, scale, and location parameters, it uses the EM algorithm to estimate the parameters of the Cauchy distribution.

Usage

ufitstab.cauchy(y, beta0, sigma0, mu0, param)

Arguments

y          vector of observations
beta0      initial value of skewness parameter to start the EM algorithm
sigma0     initial value of scale parameter to start the EM algorithm
mu0        initial value of location parameter to start the EM algorithm
param      kind of parameterization; must be 0 or 1 for S_0 and S_1 parameterizations, respectively
Details

Generally the EM algorithm seeks for the ML estimations when the log-likelihood function is not tractable mathematically. This is done by considering an extra missing (or latent) variable when the conditional expectation of the complete data log-likelihood given observed data and a guess of unknown parameter(s) is maximized. So, first we look for a stochastic representation. The representation given by the following proposition is valid for Cauchy distribution. Suppose \( Y \sim S_0(1, \beta, \sigma, \mu) \) and \( T \sim S_1(1, 1, 1, 0) \) (here \( S_0 \) and \( S_1 \) refer to parameterizations \( S_0 \) and \( S_1 \), respectively). Then \( Y = \sigma(1-|\beta|)N/Z + \sigma\beta T + \mu \) where \( N \sim Z \sim N(0, 1) \). The random variables \( N, Z, \) and \( T \) are mutually independent.

Value

- **beta**: estimated value of the skewness parameter
- **sigma**: estimated value of the scale parameter
- **mu**: estimated value of the location parameter

Note

The set of data considered here is large recorded intensities (in Richter scale) of the earthquake at seismometer locations in western North America between 1940 and 1980, see Davidian and Giltinan (1995). Among the features, we focus on the 182 distances from the seismological measuring station to the epicenter of the earthquake (in km) as the variable of interest. This set of data can be found in package `nlme`. We note that `ufitstab.cauchy()` is robust with respect to the initial values.

Author(s)

Mahdi Teimouri, Adel Mohammadpour, and Saralees Nadarajah

References


Examples

```r
# In the following example, using the initial values beta.0=0.5, sigma.0=5, and mu.0=10, # we apply the EM algorithm to estimate the parameters of Cauchy distribution fitted to # the earthquake data given by the vector y.
y<-c(7.5, 8.8, 8.9, 9.4, 9.7, 10.5, 10.5, 12.0, 12.2, 12.8, 14.6, 14.9, 17.6, 23.9, 25.0, 2.9, 7.6, 17.0, 8.0, 10.0, 10.0, 8.0, 19.0, 21.0, 13.0, 22.0, 29.0, 31.0, 5.8, 12.0, 12.1, 20.5, 20.5, 25.3, 35.9, 36.1, 36.3, 38.5, 41.4, 43.6, 44.4, 46.1, 47.1, 47.7, 49.2, 53.1, 4.0, 10.1, 11.1, 17.7, 22.5, 26.5, 29.0, 30.9, 37.8, 48.3, 62.0, 50.0, 16.0, 62.0, 1.2, 1.6, 9.1, 3.7, 5.3, 7.4, 17.9, 19.2, 23.4, 30.0, 38.9, 10.8, 15.7, 16.7, 20.8, 28.5, 33.1, 40.3, 8.0, 32.0, 30.0, 31.0, 16.1, 63.6, 6.6, 9.3, 13.0, 17.3, 105.0, 112.0, 123.0, 5.0, 23.5, 26.0, 0.5, 0.6, 1.3, 1.4, 2.6, 3.8, 4.0, 5.1, 6.2, 6.8, 7.5, 7.6, 8.4, 8.5, 8.5, 10.6, 12.6, 12.7, 12.9, 14.0, 15.0, 16.0, 17.7, 18.0, 22.0, 22.0, 23.0, 23.2, 29.0, 32.0, 32.7, 36.0, 43.5, 49.0, 60.0, 64.0),
```

ufitstab.cauchy.mix

Description

estimates the parameters of a k-component mixture of Cauchy distributions. Assuming that k is known, given the vector of initial values of entire parameter space, it uses the EM algorithm to estimate the parameters of the k-component mixture of Cauchy distributions.

Usage

ufitstab.cauchy.mix(y, k, omega0, beta0, sigma0, mu0)

Arguments

- **y**: vector of observations
- **k**: number of components
- **omega0**: initial value for weight vector to start the EM algorithm
- **beta0**: initial value for skewness vector to start the EM algorithm
- **sigma0**: initial value for scale vector to start the EM algorithm
- **mu0**: initial value for location vector to start the EM algorithm

Details

Generally the EM algorithm seeks for the ML estimations when the log-likelihood function is not tractable mathematically. This is done by considering an extra missing (or latent) variable when the conditional expectation of the complete data log-likelihood given observed data and a guess of unknown parameter(s) is maximized. So, first we look for a stochastic representation. The representation given by the following proposition is valid for Cauchy distribution. Suppose \( Y \sim S_{0}(1, \beta, \sigma, \mu) \) and \( T\sim S_{1}(1, 1, 1, 0) \) (here \( S_{0} \) and \( S_{1} \) refer to parameterizations \( S_{0} \) and \( S_{1} \), respectively). Then \( Y=\sigma(1-|\beta|)*N/Z+\sigma\beta*T+\mu \) where \( N\sim N(0,1) \). The random variables \( N, Z, \) and \( P \) are mutually independent.

Value

- **omega-bar**: a k-component vector of estimated values for the weight vector
- **beta-bar**: a k-component vector of estimated values for the skewness vector
- **sigma-bar**: a k-component vector of estimated values for the scale vector
- **mu-bar**: a k-component vector of estimated values for the location vector

```r
library("stabledist")
ufitstab.cauchy(y, 0.5, 5, 10, 0)
```
Note

We use the survival times in days of 72 guinea pigs infected with different doses of tubercle bacilli, see Bjerkedal (1960). We note that the EM algorithm is robust with respect to the initial values.

Author(s)

Mahdi Teimouri, Adel Mohammadpour, and Saralees Nadarajah

References


Examples

# In the following, we give an example that fits a two-component mixture of Cauchy distributions
# to the survival times (in days) of 72 guinea pigs through the EM algorithm. For this, the initial
# values are: omega_0=(0.65,0.35), sigma_0=(20,50), beta_0=(0.20,0.05), and mu_0=(95,210).
library("stabledist")
y<-c(10,33,44,56,59,72,74,77,92,93,96,100,102,105,107,107,108,108,109,112,121,122,124,126,130,134,136,139,144,146,153,159,160,163,163,
168,171,172,176,115,116,120,123,183,195,197,202,213,215,216,222,
230,231,240,245,251,253,254,255,278,293,327,342,347,361,402,432,458,
555)
ufitstab.cauchy.mix(y,2,c(0.65,0.35),c(0.20,0.05),c(20,50),c(95,210))

ufitstab.skew(y, alpha0, beta0, sigma0, mu0, param)

Arguments

y vector of observations
alpha0 initial value of tail index parameter to start the EM algorithm
beta0 initial value of skewness parameter to start the EM algorithm
sigma0 initial value of scale parameter to start the EM algorithm
mu0 initial value of location parameter to start the EM algorithm
param kind of parameterization; must be 0 or 1 for S_0 and S_1 parameterizations, respectively
Details

For any skew stable distribution we give a new representation by the following. Suppose \( Y \sim S_0(\alpha, \beta, \sigma, \mu) \), \( P \sim S_1(\alpha/2, 1, (\cos(\pi \alpha/4))^{2/\alpha}, 0) \), and \( V \sim S_1(\alpha, 1, 1, 0) \). Then, \( Y = \eta(2P)^{1/2}N + \theta V + \mu - \lambda \), where \( \eta = \sigma(1 - |\beta|)^{1/\alpha} \), \( \theta = \sigma \text{sign}(\beta) |\beta|^{1/\alpha} \), \( \lambda = \sigma \beta \tan(\pi \alpha/2) \), and \( N \sim N(0, 1) \) follows a skew stable distribution. All random variables \( N, P, \) and \( V \) are mutually independent.

Value

- \( \alpha \): estimated value of the tail index parameter
- \( \beta \): estimated value of the skewness parameter
- \( \sigma \): estimated value of the scale parameter
- \( \mu \): estimated value of the location parameter

Note

Daily price returns of Abbey National shares between 31/7/91 and 8/10/91 (including \( n=50 \) business days). By assuming that \( p_{t} \) denotes the price at \( t \)-th day, the price return at \( t \)-th day is defined as \( (p_{t-1} - p_{t})/p_{t-1} \); for \( t=2, \ldots, n \), see Buckle (1995). We note that the EM algorithm is robust with respect to the initial values.

Author(s)

Mahdi Teimouri, Adel Mohammadpour, and Saralees Nadarajah

References


Examples

```r
# For example, We use the daily price returns of Abbey National shares. Using the initial
# values as alpha(0)=0.8, beta(0)=0, sigma(0)=0.25, and mu(0)=0.25.
price<-c(296,296,300,302,300,304,303,299,293,294,294,293,295,287,288,297,
        305,307,304,303,304,304,309,309,307,306,304,300,296,301,298,
        295,295,293,292,307,297,294,293,306,303,301,303,308,305,302,301,
        297,299)
x<-c()
n<-length(price)
for(i in 2:n){x[[i]]<-(price[[i-1]]-price[i])/price[i-1]}
library("stabledist")
ufitstab.skew(x[2:n],0.8,0,0.25,0.25,1)
```
ufitstab.sym

Description

estimates the parameters of a symmetric stable distribution through the EM algorithm, see Teimouri et al. (2018).

Usage

ufitstab.sym(yy, alpha0, sigma0, mu0)

Arguments

yy a vector of observations
alpha0 initial value for the tail index parameter
sigma0 initial values for the scale parameter
mu0 initial values for the location parameter

Value

alpha estimated value of the tail index parameter
sigma estimated value of the scale parameter
mu estimated value of the location parameter

Author(s)

Mahdi Teimouri, Adel Mohammadpour, and Saralees Nadarajah

References


Examples

# By the following example, we apply the EM algorithm to n=50 iid realization of symmetric
# stable distribution with parameters alpha=1.2, sigma=1, and mu=1. The initial values
# are alpha.0=1.2, sigma.0=1, and mu.0=1.
library("stabledist")
y<-urstab(50,1.2,0,1,1,0)
ufitstab.sym(y,1.2,1,1)
ufitstab.sym.mix estimates the parameters of a $k$-component mixture of symmetric stable distributions, Teimouri et al. (2018) <doi.org/10.1080/03610918.2017.1288244>. Having $k$ and given the vector of initial values of entire parameter space, it uses some type of the EM algorithm (ECME) to estimate the parameters of mixture of symmetric stable distributions.

**Usage**

```r
ufitstab.sym.mix(yy, k, omega0, alpha0, sigma0, mu0)
```

**Arguments**

- `yy` vector of observations
- `k` number of components
- `omega0` vector of initial values for weights
- `alpha0` vector of initial values for tail indices
- `sigma0` vector of initial values for scale parameters
- `mu0` vector of initial values for location parameters

**Value**

- `omega-bar` a $k$-component vector of estimated values for the weight vector
- `alpha-bar` a $k$-component vector of estimated values for the tail index vector
- `sigma-bar` a $k$-component vector of estimated values for the scale vector
- `mu-bar` a $k$-component vector of estimated values for the location vector
- `membership` a $k$ by $n$ matrix whose entries are 1 and 0; for $i$-th row, if $j$-th column is one; then $i$-th observation belongs to the $j$-th component

**Note**

The set of data, used here, is the velocities of 82 distant galaxies, diverging from our own galaxy, called here galaxy. These data are available at https://people.maths.bris.ac.uk/mapjg/mixdata. We note that `ufitstab.sym.mix()` is robust with respect to the initial values.

**Author(s)**

Mahdi Teimouri, Adel Mohammadpour, and Saralees Nadarajah

**References**

Examples

# In what follows, we apply the EM algorithm to estimate the parameters of the
# mixture of symmetric stable distributions. For this, the initial values for
# fitting a three-component mixture of symmetric stable distribution to the
# galaxy data are: (0.1,0.35,0.55) for weight vector, (1.2,1.2,1.2) for tail
# index vector, (1,1,1) for scale vector, and (8,20,22) for the location vector.
          22.374,22.495,22.746,22.747,22.888,22.914,23.206,23.241,23.263,
          23.484,23.538,23.542,23.666,23.706,23.711,24.129,24.285,24.289,
library("stabledist")
ufitstab.sym.mix(galaxy,3,c(0.1,0.35,0.55),c(1.2,1.2,1.2),c(1,1,1),c(8,20,22))

ufitstab.ustat  ufitstab.ustat

Description

estimates the tail index and scale parameters of a symmetric and zero-location stable distribution using U-statistic proposed by Fan (2006) <DOI: 10.1080/03610920500439992>.

Usage

ufitstab.ustat(x)

Arguments

  x  vector of observations

Value

  alpha  estimated value of the tail index parameter
  sigma  estimated value of the scale parameter

Note

The ufitstab.ustat() must be applied to a symmetric and zero-location stable distribution.

Author(s)

Mahdi Teimouri, Adel Mohammadpour, and Saralees Nadarajah
References


Examples

```r
# We are estimating the parameters of a symmetric stable distribution. For this, firstly,
# we simulate a sample of n=100 iid realizations from stable distribution in S_1 parameterization
# with parameters alpha=1.2, beta=0, sigma=1, and mu=0.
x<-urstab(100,1.2,0,1,0,1)
ufitstab.ustat(x)
```

Description

computes the cumulative distribution function (cdf) of the univariate stable distribution based on formulas given by Nolan (1997) and asymptotic series, see Teimouri and Amindavar (2008).

Usage

```
upstab(x, alpha, beta, sigma, mu, param)
```

Arguments

- `x` point at which the cdf is computed
- `alpha` tail index parameter
- `beta` skewness parameter
- `sigma` scale parameter
- `mu` location parameter
- `param` kind of parameterization; must be 0 or 1 for S_0 and S_1 parameterizations, respectively

Value

a numeric value

Note

`upstab()` computes the cdf of univariate stable distribution using asymptotic series within their convergence regions. For points outside of convergence regions, the cdf is computed using stabledist package based on formulas given by Nolan (1997). So, to compute the cdf using the `upstab()` we may need to install stabledist package.
Author(s)
Mahdi Teimouri, Adel Mohammadpour, and Saralees Nadarajah

References

Examples

In the following, we compute the cdf of a univariate stable distribution at point 2 with parameters alpha=1.2, beta=0.9, sigma=1, and mu=0 in S_0 parameterization.

upstab(2,1.2,0.9,1,0,1)

Description

Usage

urstab(n,alpha,beta,sigma,mu,param)

Arguments

n sample size
alpha tail index parameter
beta skewness parameter
sigma scale parameter
mu location parameter
param kind of parameterization must; be 0 or 1 for S_0 and S_1 parameterizations, respectively

Value
a vector of n numeric values
Author(s)

Mahdi Teimouri, Adel Mohammadpour, and Saralees Nadarajah

References


Examples

# By the following example, we simulate n=200 iid realizations from univariate stable
# distribution with parameters alpha=1.2, beta=0.5, sigma=2, and mu=0 in S_0 parameterization.
x <- urstab(200, 1.2, 0.5, 2, 0, 0)

Description

using the methodology given by Soltan and Shirvani (2010), Shirvani and Soltani (2013) for simulating iid truncated stable random variable, it simulates truncated stable realizations.

Usage

urstab.trunc(n, alpha, beta, sigma, mu, a, b, param)

Arguments

n | sample size
alpha | tail index parameter
beta | skewness parameter
sigma | scale parameter
mu | location parameter
a | lower bound of truncation
b | upper bound of truncation
param | kind of parameterization; must be 0 or 1 for S_0 and S_1 parameterizations, respectively

Value

a vector of n numeric values
Author(s)
Mahdi Teimouri, Adel Mohammadpour, and Saralees Nadarajah

References

Examples
# We simulate n=200 iid realizations from truncated stable distribution with parameters
# alpha=1.3, beta=0.5, sigma=2, and mu=0 which is truncated over (-5,5) in S_0 parameterization.
urstab.trunc(200,1.3,0.5,2,0,-5,5,0)
Index

mdstab.elliptical, 2
mfitstab.elliptical, 3
mfitstab.ustat, 4
mrstab, 5
mrstab.elliptical, 6
udstab, 7
ufitstab.cauchy, 8
ufitstab.cauchy.mix, 10
ufitstab.skew, 11
ufitstab.sym, 13
ufitstab.sym.mix, 14
ufitstab.ustat, 15
upstab, 16
urstab, 17
urstab.trunc, 18