Package 'MCMC4Extremes'

July 14, 2016

Type Package

Title Posterior Distribution of Extreme Value Models in R

Version 1.1

Author Fernando Ferraz do Nascimento [aut, cre], Wyara Vanesa Moura e Silva [aut, ctb]

Maintainer Fernando Ferraz do Nascimento <fernandofn@ufpi.edu.br>

Depends R (>= 3.1), evir

Description Provides some function to perform posterior estimation for some distribution, with emphasis to extreme value distributions. It contains some extreme datasets, and functions that perform the runs of posterior points of the GPD and GEV distribution. The package calculate some important extreme measures like return level for each t periods of time, and some plots as the predictive distribution, and return level plots.

License GPL-2

NeedsCompilation no

Repository CRAN

Date/Publication 2016-07-14 07:18:48

R topics documented:

`barcelos` *30-day maxima rainfall at Barcelos Station*

Description

These data are the 30-day maxima rainfall at Barcelos Station, in Portugal, from 1931 to 2008. The data are contained in a numeric vector

Usage

```
data(barcelos)
```

Format

A numeric vector containing 918 observations.

Examples

```
data(barcelos)
hist(barcelos, main=NULL)
```

`dggev` *Dual Gamma Generalized Extreme Value Distribution*

Description

Cumulative probability, quantiles, density and random generation from the dual gamma generalized extreme value distribution.

Usage

```
pggev(q, xi, mu, sigma, delta)
qggev(p, xi, mu, sigma, delta)
dggev(x, xi, mu, sigma, delta)
rggev(n, xi, mu, sigma, delta)
```

Arguments

q	vector of quantiles
p	vector of probabilities
x	vector of values at which to evaluate density
n	sample size
xi	shape parameter
mu	location parameter
sigma	scale parameter
delta	additional shape parameter of GGEV extension

Value

Probability (pggev), quantile (qggev), density (dggev) or random sample (rggev) for the GGEV distribution.

References

Nascimento, F. F.; Bourguigon, M. ; Leao, J. S. (2015). Extended generalized extreme value distribution with applications in environmental data. *HACET J MATH STAT*.

See Also

ggevp

---

fajardo *15-day maxima river food at Fajardo River*

Description

These data are the 15-day maxima river food at Fajardo River, in Porto Rico, from 1967 to 2008. The data are contained in a numeric vector

Usage

```
data(fajardo)
```

Format

A numeric vector containing 864 observations.

Examples

```
data(fajardo)
hist(fajardo, main=NULL)
```

gammap *Posterior Distribution with Gamma Density*

Description

MCMC runs of posterior distribution of data with `Gamma(alpha,beta)` density.

Usage

```
gammap(data, int=1000)
```

Arguments

`data`	data vector
`int`	number of iteractions selected in MCMC. The program selects 1 in each 10 iteractions, then `thin=10`. The first `thin*int/3` iteractions is used as burn-in. After that, is runned `thin*int` iteraction, in which 1 of thin is selected for the final MCMC chain, resulting the number of int iteractions.

Value

An object of class `gammap` that gives a list containing the points of posterior distributions of `alpha` and `beta` of the gamma distribution, the data, mean posterior, median posterior and the credibility interval of the parameters.

Note

The non-informative prior distribution of these parameters are both `Gamma(0.0001,0.0001)`. During the MCMC runs, screen shows the proportion of iteractions made

Examples

```
# Vector of maxima return for each 10 days for ibovespa data
data(ibovespa)
ibmax=gev(ibovespa[,4],10)$data
# obtaining 500 points of posterior distribution
ibovpost=gammap(ibmax,300)
```

gevp *Posterior Distribution with Parameters of GEV*

Description

MCMC runs of posterior distribution of data with parameters of Generalized Extreme Value (GEV) density, with parameters `mu`, `sigma` and `xi`.

Usage

```
gevp(data, block, int=1000)
```

Arguments

	X
	data vector
block	the block size. A numeric value is interpreted as the number of data values in each successive block. All the data is used, so the last block may not contain block observations
`int`	Number of iteractions selected in MCMC. The program selects 1 in each 10 iteraction, then `thin=10`. The first `thin*int/3` iteractions is used as burn-in. After that, is runned `thin*int` iteraction, in which 1 of thin is selected for the final MCMC chain, resulting the number of int iteractions.

Value

An object of class `gevp` that gives a list containing the points of posterior distributions of `mu`, `sigma` and `xi` of the gev distribution, the data, mean posterior, median posterior and the credibility interval of the parameters.

Note

The non-informative prior distribution of these parameters are `Normal(0,1000)` for the parameter `mu`, `Gamma(0.001,0.001)` for the parameter `sigma` and `Normal(0,100)` for parameter `xi`. During the MCMC runs, screen shows the proportion of iteractions made.

See Also

`plot.gevp`, `summary.gevp`

Examples

```
# Obtaining posterior distribution of a vector of simulated points
x=rgev(300,xi=0.1,mu=10,sigma=5)

# Obtaning 600 points of posterior distribution
ajuste=gevp(x,1,200)
```

```
# Obtaining 300 points of ponterior distribution of river nidd data
## Not run: data(nidd.annual)
## Not run: out=gevp(nidd.annual,1,300)

# Vector of maxima return for each 15 days for ibovespa data
## Not run: data(ibovespa)
## Not run: postibv=gevp(ibovespa[,4],15,300)
## Not run: plot.ts(postibv$posterior)
```

---

ggevp *Posterior Distribution with Parameters of Dual Gamma Generalized Extreme Value Distribution*

Description

MCMC runs of posterior distribution of data with parameters of Dual Gamma Generalized Extreme Value Distribution density, with parameters `mu`, `sigma` and `xi`.

Usage

```
ggevp(data, block, int=1000, delta)
```

Arguments

`data`	data vector
`block`	the block size. A numeric value is interpreted as the number of data values in each successive block. All the data is used, so the last block may not contain block observations
`int`	Number of iteractions selected in MCMC. The program selects 1 in each 10 iteraction, then `thin=10`. The first `thin*int/3` iteractions is used as burn-in. After that, is runned `thin*int` iteraction, in which 1 of thin is selected for the final MCMC chain, resulting the number of int iteractions.
`delta`	additional shape parameter of GGEV extension

Value

An object of class `ggevp` that gives a list containing the points of posterior distributions of `mu`, `sigma` and `xi` of the dual gamma generalized extreme value distribution, the data, mean posterior, median posterior and the credibility interval of the parameters.

References

Nascimento, F. F.; Bourguigon, M. ; Leao, J. S. (2015). Extended generalized extreme value distribution with applications in environmental data. *HACET J MATH STAT*.

See Also

plot.ggevp, summary.ggevp

Examples

```
# Obtaining posterior distribution of a vector of simulated points
w=rggev(300,0.1,10,5,0.5)

# Obtaning 500 points of posterior distribution with delta=0.5
ajust=ggevp(w,1,200,0.5)
```

---

gpdp *Posterior Distribution with Parameters of GPD*

Description

MCMC runs of posterior distribution of data with parameters of Generalized Pareto Distribution (GPD), with parameters `sigma` and `xi` .

Usage

```
gpdp(data, threshold, int=1000)
```

Arguments

`data`	data vector
`threshold`	a threshold value
`int`	number of iteractions selected in MCMC. The program selects 1 in each 10 iteraction, then `thin=10`. The first `thin*int/3` iteractions is used as burn-in. After that, is runned `thin*int` iteraction, in which 1 of thin is selected for the final MCMC chain, resulting the number of int iteractions.

Value

An object of class gpdp that gives a list containing the points of posterior distributions of `sigma` and `xi` of the gpd distribution, the data, mean posterior, median posterior and the credibility interval of the parameters.

Note

The joint priordistribution for these parameters is the Jeffreys prior Given as Castellanos and Cabras (2007).

References

Castellanos, M. A. and Cabras, S. (2007). A default Bayesian procedure for the generalized Pareto distribution, *Journal of Statistical Planning and Inference*, 137, 473-483.

See Also

plot.gpdp, summary.gpdp

Examples

```
# Obtaining posterior distribution of a vector of simulated points
x=rgpd(300,xi=0.1,mu=9,beta=2) # in this case beta is the scale parameter sigma

# Obtaning 1000 points of posterior distribution
ajuste=gpdp(x,9, 200)

# Histogram of posterior distribution of the parameters,with 95% credibility intervals
# Danish data for evir package, modelling losses over 10
## Not run data(danish)
## Not run out=gpdp(danish,10,300)
```

---

gumbelp *Posterior Distribution with GEV, where xi=0*

Description

MCMC runs of posterior distribution of data with parameters of Generalized Extreme Value (GEV) density, in the particular case where `xi=0` with parameters `mu`, `sigma`.

Usage

```
gumbelp(data, block, int=1000)
```

Arguments

`data`	data vector
`block`	the block size. A numeric value is interpreted as the number of data values in each successive block. All the data is used, so the last block may not contain block observations.
`int`	number of iteractions selected in MCMC. The program selects 1 in each 10 iteraction, then `thin=10`. The first `thin*int/3` iteractions is used as burn-in. After that, is runned `thin*int` iteraction, in which 1 of thin is selected for the final MCMC chain, resulting the number of int iteractions

Value

An object of class `gumbelp` that gives a list containing the points of posterior distributions of `mu` and `sigma` of the gev distribution, the data, mean posterior, median posterior and the credibility interval of the parameters.

Note

The non-informative prior distribution of these parameters are `Normal(0,1000)` for the parameter `mu` and `Gamma(0.001,0.001)` for the parameter `sigma`. During the MCMC runs, screen shows the proportion of iteractions made.

See Also

`plot.gumbelp`, `summary.gumbelp`

Examples

```
# Obtaining posterior distribution of a vector of simulated points
x=rgev(200,xi=0.0001,mu=10,sigma=5)
# Obtaning 600 points of posterior distribution
ajuste=gumbelp(x,1,600)

# Maxima of each month in river nidd data
## Not run: data(nidd.annual)
## Not run: out=gumbelp(nidd.annual,1,500)

# Predictive distribution for 15 day maxima ibovespa returns
## Not run: data(ibovespa)
## Not run: postibv=gumbelp(ibovespa[,4],15,500)
```

---

gurgueia *Daily river quota of Gurgueia River.*

Description

These data are the monthly maximum river quota of Gurgueia River, in Brazil, from 1975 to 2012.

Usage

```
data(gurgueia)
```

Format

A data frame with 415 observations on the following 2 variables.

`date` month/year

`maximum` a numeric vector with monthly maximum

Examples

```
data(gurgueia)
hist(gurgueia[,2], main=NULL)
```

ibovespa *Daily returns of ibovespa*

Description

These data are the daily returns of ibovespa from 2000 to 2009.

Usage

```
data(ibovespa)
```

Format

A data frame with 2369 observations on the following 4 variables.

`month` a numeric vector with month

`day` a numeric vector with day

`year` a numeric vector with year

`returns` a numeric vector with returns

Examples

```
data(ibovespa)
hist(ibovespa[,4], main=NULL)
```

normalp *Posterior Distribution with Normal Density*

Description

MCMC runs of posterior distribution of data with `Normal(mu,1/tau)` density, where `tau` is the inverse of variance.

Usage

```
normalp(data, int=1000)
```

Arguments

`data`	data vector
`int`	number of iteractions selected in MCMC. The program selects 1 in each 10 iteraction, then `thin=10`. The first `thin*int/3` iteractions is used as burn-in. After that, is runned `thin*int` iteraction, in which 1 of thin is selected for the final MCMC chain, resulting the number of int iteractions

Value

An object of class gumbelp that gives a list containing the points of posterior distributions of mu and tau of the normal distribution, the data, mean posterior, median posterior and the credibility interval of the parameters.

Note

The non-informative prior distribution of these parameters are Normal(0,10000000) for the parameter mu and Gamma(0.001,0.001) for the parameter tau . During the MCMC runs, screen shows the proportion of iteractions made.

See Also

plot.normalp

Examples

```
# Obtaining posterior distribution of a vector of simulated points
x=rnorm(300,2,sqrt(10))

# Obtaning 1000 points of posterior distribution
ajuste=normalp(x, 200)

# Posterior distribution of river Nile dataset
## Not run: data(Nile)
## Not run: postnile=normalp(Nile,1000)
```

---

plot.gevp *Plot Fitted GEV Model*

Description

The plot method plot.gevp provides three differents plots: a histogram of the gev parameters, a plot of predictive density resulting of posterior distribution of GEV parameters, and a return level plot of GEV distribution.

Usage

```
## S3 method for class 'gevp'
plot(x, type = c("histogram", "predictive", "retlevel"), t=2, k=100, ...)
```

Arguments

x	a gevp object
type	which chosen plot
t	start return level
k	end return level
...	other graphics parameters

See Also

gevp

Examples

```
# Return level of river nidd data
data(nidd.annual)
out=gevp(nidd.annual,1,300)
## Not run: plot(out,"histogram")
plot(out,"predictive")
## Not run: plot(out,"retlevel", 10, 50)
```

`plot.ggevp` *Plot Fitted for the Dual Gamma Generalized Extreme Value Distribution (GGEV) Model*

Description

The plot method plot.ggevp provides three differents plots: a histogram of the GGEV parameters, a plot of predictive density resulting of posterior distribution of GGEV parameters, and a return level plot of GGEV distribution.

Usage

```
## S3 method for class 'ggevp'
plot(x, type = c("histogram", "predictive", "retlevel"), t=2, k = 100, ...)
```

Arguments

`x`	a ggevp object
`type`	which chosen plot
`t`	start return level
`k`	end return level
`...`	other graphics parameters

References

Nascimento, F. F.; Bourguigon, M. ; Leao, J. S. (2015). Extended generalized extreme value distribution with applications in environmental data. *HACET J MATH STAT*.

See Also

ggevp

Examples

```
# Obtaining posterior distribution of a vector of simulated points
w=rggev(300,0.4,10,5,0.5)

# Obtaning 300 points of posterior distribution with delta=0.5
fit=ggevp(w,1,200,0.5)
## Not run: plot(fit,"histogram")
plot(fit,"predictive")
## Not run: plot(fit,"retlevel", 10, 50)
```

`plot.gpdp` *Plot Fitted GPD Model*

Description

The plot method plot.gpdp provides three differents plots: a histogram of the GPD parameters, a plot of predictive density resulting of posterior distribution of GPD parameters, and a return level plot of GPD distribution.

Usage

```
## S3 method for class 'gpdp'
plot(x, type = c("histogram", "predictive", "retlevel"), t=2, k=100, ...)
```

Arguments

`x`	a gpdp object
`type`	which chosen plot
`t`	start return level
`k`	end return level
`...`	other graphics parameters

See Also

gpdp

Examples

```
data(danish)
out=gpdp(danish,10,300)
## Not run: plot(out,"histogram")
## Not run: plot(out,"predictive")
plot(out,"retlevel", 10, 50)
```

plot.gumbelp *Plot Fitted Gumbel Model*

Description

The plot method plot.gumbelp provides three differents plots: a histogram of the gumbel parameters, a plot of predictive density resulting of posterior distribution of gumbel parameters, and a return level plot of gumbel distribution.

Usage

```
## S3 method for class 'gumbelp'
plot(x, type = c("histogram", "predictive", "retlevel"), t=2, k=100, ...)
```

Arguments

x	a gumbelp object
type	which chosen plot
t	start return level
k	end return level
...	other graphics parameters

See Also

gumbelp

Examples

```
data(nidd.annual)
out=gumbelp(nidd.annual,1,500)
## Not run: plot(out,"histogram")
## Not run: plot(out,"predictive")
plot(out,"retlevel", 10)
```

plot.normalp *Plot Fitted Normal Model*

Description

The plot method plot.normalp provides three differents plots: a histogram of the normal parameters, a plot of predictive density resulting of posterior distribution of normal parameters, and a return level plot of normal distribution.

Usage

```
## S3 method for class 'normalp'
plot(x, type = c("histogram"), ...)
```

Arguments

x	a normalp object
type	which chosen plot
...	other graphics parameters

See Also

normalp

Examples

```
data(Nile)
p=normalp(Nile,600)
plot(p,"histogram")
```

---

summary.gevp *Summarizing Posterior Distribution with Parameters of GEV*

Description

summary method for class "gevp"

Usage

```
## S3 method for class 'gevp'
summary(object, ...)
```

Arguments

object	an object of class "gevp", usually, a result of a call to gevp.
...	further arguments passed to or from other methods.

Value

The function summary.gevp computes and returns a list of summary statistics of the posterior distribution given in object.

postmean	mean posterior
postmedian	median posterior
postCI	credibility interval
fitm	fit measures for standard GEV model

See Also

gevp

Examples

```
# Return level of river nidd data
data(nidd.annual)
out=gevp(nidd.annual,1,300)
a=summary(out)
a
```

---

summary.ggevp *Summarizing Posterior Distribution with Parameters of GGEV*

Description

summary method for class "ggevp"

Usage

```
## S3 method for class 'ggevp'
summary(object, ...)
```

Arguments

object	an object of class "ggevp", usually, a result of a call to ggevp.
...	further arguments passed to or from other methods.

Value

The function summary.ggevp computes and returns a list of summary statistics of the posterior distribution given in object.

postmean	mean posterior
postmedian	median posterior
postCI	credibility interval
fitm	fit measures for standard GGEV model

References

Nascimento, F. F.; Bourguigon, M. ; Leao, J. S. (2015). Extended generalized extreme value distribution with applications in environmental data. *HACET J MATH STAT*.

See Also

ggevp

Examples

```
# Obtaining posterior distribution of a vector of simulated points
w=rggev(300,0.4,10,5,0.5)
# Obtaning 600 points of posterior distribution with delta=0.5
fit=ggevp(w,1,200,0.5)
a=summary(fit)

# Choice the best delta from a Grid of possible values as Nascimento et al. (2015)
## Not run: fitmeasures=summary(fit)$fitm
## Not run: delta=seq(0.1,2,0.2)
## Not run: results=array(0,c(length(delta),4))
## Not run: for (i in 1:length(delta))
## Not run:     {ajust=ggevp(w,1,200,delta[i])
## Not run:      results[i,]=summary(ajust)$fitm}

# As commented in Nascimento 2015 paper, a criteria to choice the best delta would be
# create a grid of values of theta and choose the best according the lowest fit measures
## Not run: resultsb=cbind(delta,results)
## Not run: colnames(resultsb)=c("delta","AIC","BIC","pD","DIC")
```

`summary.gpdp` *Summarizing Posterior Distribution with Parameters of GPD*

Description

summary method for class "gpdp"

Usage

```
## S3 method for class 'gpdp'
summary(object, ...)
```

Arguments

`object`	an object of class "gpdp", usually, a result of a call to gpdp.
`...`	further arguments passed to or from other methods.

Value

The function `summary.ggevp` computes and returns a list of summary statistics of the posterior distribution given in `object`.

`postmean`	mean posterior
`postmedian`	median posterior
`postCI`	credibility interval
`fitm`	fit measures for standard GPD model

See Also

gpdp

Examples

```
data(danish)
out=gpdp(danish,10,300)
a=summary(out)
a
```

---

summary.gumbelp *Summarizing Posterior Distribution with Parameters of Gumbel*

Description

summary method for class "gumbelp"

Usage

```
## S3 method for class 'gumbelp'
summary(object, ...)
```

Arguments

object	an object of class "gumbelp", usually, a result of a call to gumbelp.
...	further arguments passed to or from other methods.

Value

The function summary.gumbelp computes and returns a list of summary statistics of the posterior distribution given in object.

postmean	mean posterior
postmedian	median posterior
postCI	credibility interval
fitm	fit measures for standard Gumbel model

See Also

gumbelp

Examples

```
# Example with simulated datapoints
x=rgev(300,0.01,10,5)
fit=gumbelp(x,1,300)
fitgum=summary(fit)

# Compare if the fit measures of gumbel is better than measures using GEV
## Not run: fit2=gevp(x,1,300)
## Not run: fitgev=summary(fit2)
# the best model is that with lowest fit measures
```

Index