Package 'evdbayes'

July 22, 2025

Version 1.1-3

Date 2023-02-27

Title Bayesian Analysis in Extreme Value Theory

Author Alec Stephenson [aut, cre],
Mathieu Ribatet [aut]

Maintainer Alec Stephenson <alec_stephenson@hotmail.com>

Depends R (>= 1.8.0)

Description Provides functions for the Bayesian analysis of extreme value models, using Markov chain Monte Carlo methods. Allows the construction of both uninformative and informed prior distributions for common statistical models applied to extreme event data, including the generalized extreme value distribution.

License GPL (>= 2)

NeedsCompilation yes

Repository CRAN

Date/Publication 2023-03-01 09:00:05 UTC

Contents

`Accept Rate` *Compute Suited Proposal Standard Deviations*

Description

Compute suited proposal standard deviations for the MCMC algorithm.

Usage

```
ar.choice(init, prior, lh = c("none","gev","gpd","pp","os"), ..., psd,
ar = rep(.4, npar), n = 1000, tol = rep(.05, npar))
```

Arguments

`init`	a numeric vector for the starting value of the MCMC algorithm.
`prior`	A prior model. See function `prior.prob`, `prior.quant`, `prior.norm` and `prior.loglognorm`.
`lh`	The likelihood function. Should be one of "none", "gev", "gpd", "pp" and "os".
`...`	Optional arguments to be passed to the `posterior` function.
`psd`	The initials proposal standard deviations.
`ar`	Optional. The objective accept rates - default is `rep(.4, npar)`.
`n`	Optional. The length of the simulated Markov Chains.
`tol`	Optional. The tolerance for the convergence test.

Details

The suited proposal standard deviations (psd) are computed through trial and error processes. Proposal standard deviations are fundamental to ensure good mixing properties for the Markov Chains.

For this purpose, there exits a thumb rule: *"In small dimensions, aim at an average acceptance rate of 50. In large dimensions, at an average acceptance rate of 25. (Gelman et al., 1995)"*.

For numerical conveniences, the trial and error process is more accurate with small initial starting psd.

Value

Return a list with two arguments. "psd": the suited proposal standard deviations and "ar": the accept rates related to these proposal standard deviations.

Author(s)

Mathieu Ribatet

References

Gelman, A. and Roberts, G. and Gilks, W. (1995) *Efficient Metropolis Jumping Rules.* Oxford University Press.

Examples

```
data(rainfall)
prrain <- prior.quant(shape = c(38.9, 7.1, 47), scale = c(1.5, 6.3,
  2.6))
n <- 10000; t0 <- c(43.2, 7.64, 0.32);
s <- ar.choice(init = t0, prior = prrain, lh = "pp", data = rainfall,
  thresh = 40, noy = 54, psd = rep(0.01, 3))
```

---

dinfo *Information for Beta and Gamma Distributions*

Description

Show means, variances and modes for beta and gamma distributions.

Usage

```
ibeta(mean, var, shape1, shape2)
igamma(mean, var, shape, scale)
```

Arguments

mean, var	Numeric vectors giving means and variances.
shape1, shape2	Numeric vectors. See dbeta.
shape, scale	Numeric vectors. See dgamma.

Details

For ibeta, either both of mean and var or both of shape1 and shape2 must be specified. For igamma, either both of mean and var or both of shape and scale must be specified. The pair of vectors that are passed to each function define a set of beta/gamma distributions. If one vector is shorter than the other, the shorter vector is replicated.

Value

A matrix with five columns and n rows, where n is the length of the longest argument. If $n = 1$ the dimension is dropped (i.e. a vector of length five is returned). The columns contain the means, variances, modes, and the shape/scale parameters of the specified distributions. If a mode is NA, it does not exist, or it is not unique, or it does not occur in the interior of the support. If an entire row is NA, the corresponding arguments do not lead to a valid distribution.

See Also

dbeta, dgamma

Examples

```
ibeta(shape1 = 5, shape2 = 4)
ibeta(mean = seq(0.1,0.9,0.2), var = 0.03)
igamma(shape=c(38.9,7.1,47), scale=c(1.5,6.3,2.6))
```

---

lh *Calculate Log-likelihoods*

Description

Calculate log-likelihoods for the gev, order statistics or point process models.

Usage

```
pplik(par, data, thresh, noy, trend, exact = FALSE)
gevlik(par, data, trend)
gpdlik(par, data, trend)
oslik(par, data, trend)
```

Arguments

par	If trend is missing, should be a numeric vector of length three, containing the location, scale and shape parameters. If trend is not missing, should be a numeric vector of length four, containing the location intercept, scale, shape and location trend parameters, in that order.
data	For pplik, gevlik and gpdlik; a non-empty numeric vector containing the data at which the likelihood is evaluated, possibly containing missing values. For oslik; a numeric matrix (see the user's guide).
thresh	Threshold. Typically a single number or a vector of the same length as data.
noy	Number of years/periods of observations, excluding any missing values.
trend	Trend vector (optional). If given, should be the same length as data for pplik and gevlik. For oslik, should contain one value for each row of data.
exact	In general, the point process likelihood includes an approximation to an integral. If exact is TRUE, every value in trend and thresh is used for the approximation.

Details

See the user's guide.

Value

A numeric vector.

Note

These functions are essentially internal, and need not be called by the user. They are documented only because their arguments (excluding par) can be passed to posterior.

See Also

posterior, prior.prob

---

mc.quant *Compute GEV Quantiles from Markov Chains*

Description

Compute gev quantiles from samples stored within a Markov chain, corresponding to specified probabilities in the upper tail.

Usage

```
mc.quant(post, p, lh = c("gev", "gpd"))
```

Arguments

post	A Markov chain generated using posterior, containing samples of gev parameters.
p	A numeric vector of upper tail probabilities.
lh	Specify "gev" or "gpd" likelihood.

Details

See the user's guide.

Value

A matrix with n rows and m columns, where n is the number of samples stored within the chain, and m is the length of the vector p. If $m = 1$ the dimension is dropped (i.e. a vector of length n is returned). The (i,j)th entry contains the gev quantile coresponding to the upper tail probability p[j], evaluated at the parameters within sample i.

If a linear trend on the location has been implemented, the quantiles correspond to the distribution obtained when the trend parameter is zero.

See Also

posterior

mposterior *Maximizing Posterior Distributions*

Description

Maximizing prior and posterior distibutions for the location (with optional trend), scale and shape parameters under the gev, order statistics or point process models.

Usage

```
mposterior(init, prior, lh = c("none", "gev", "gpd", "pp", "os"),
    method = c("Nelder-Mead", "BFGS", "CG", "L-BFGS-B", "SANN"),
    lower = -Inf, upper = Inf, control = list(), hessian = FALSE, ...)
```

Arguments

`init`	Numeric vector of length three/four, giving the initial values for the optimization.
`prior`	An object of class `"evprior"`, constructed using `prior.prob`, `prior.quant` or `prior.norm`.
`lh`	A character string specifying the likelihood; either `"gev"` for gev, `"gpd"` for gpd, `"os"` for order statistics, `"pp"` for Poisson process or `"none"` for none (the default). The latter can be used to maximize the prior distribution.
`method`	The method to be used. See `optim`.
`lower, upper`	Bounds on the variables for the `"L-BFGS-B"` method. See `optim`.
`control`	A list of control parameters. See `optim`.
`hessian`	Logical. See `optim`.
`...`	Arguments to the likelihood. Should include `data` unless `lh` is `"none"`. Should also include `thresh` and `noy` if `lh` is `"pp"`. Should include the vector `trend` if a linear trend on the location is implemented. See `pplik` for details.

Value

A list. See `optim`.

See Also

`pplik`, `posterior`, `prior.prob`

posterior *MCMC Sampling of Posterior Distributions*

Description

Constructing MCMC samples of prior and posterior distibutions for the location (with optional trend), scale and shape parameters under the gev, order statistics or point process models.

Usage

```
posterior(n, init, prior, lh = c("none", "gev", "gpd", "pp","os"), ..., psd,
    burn = 0, thin = 1)
```

Arguments

n	The run-length; the number of sampled vectors (excluding init).
init	Numeric vector of length three/four, giving the initial values for the chain, taken to be iteration zero.
prior	An object of class "evprior", constructed using prior.prob, prior.quant or prior.norm.
lh	A character string specifying the likelihood; either "gev" for gev, "gpd" for gpd, "os" for order statistics, "pp" for Poisson process or "none" for none (the default). The latter can be used to sample from the prior distribution.
...	Arguments to the likelihood. Should include data unless lh is "none". Should also include thresh and noy if lh is "pp". Should include the vector trend if a linear trend on the location is implemented. See pplik for details.
psd	A vector of length three/four containing standard deviations for proposal distributions.
burn	The burn-in period (an integer); the first burn iterations (including init) are excluded from the chain.
thin	The thinning interval (an integer); iteration k is stored only if k mod thin is zero (and if k greater than or equal to burn).

Details

See the user's guide.

Value

A matrix with 1+floor(n/thin)-burn rows. Row labels give the iteration numbers. Column labels give parameter names.

An attribute ar is also returned. This is a matrix containing acceptence rates in the first row (the number of proposals accepted divided by the number of iterations) and "external rates" in the second (the number of proposals that resulted in a zero likelihood, divided by the number of iterations).

See Also

pplik, prior.prob

Examples

```
data(rainfall)
prrain <- prior.quant(shape = c(38.9, 7.1, 47), scale = c(1.5, 6.3, 2.6))

n <- 100 ; t0 <- c(50.8, 1.18, 0.65) ; s <- c(25, .35, .07) ; b <- 20
rn.prior <- posterior(n, t0, prrain, "none", psd = s, burn = b)

t0 <- c(43.2, 7.64, 0.32) ; s <- c(2, .2, .07)
rn.post <- posterior(n, t0, prrain, "pp", data = rainfall, thresh = 40,
  noy = 54, psd = s, burn = b)
```

---

prior *Construction of Prior Distributions*

Description

Constructing prior distibutions for the location, scale and shape parameters using normal, beta or gamma distributions. A linear trend for the location can also be specified, using a prior normal distribution centered at zero for the trend parameter.

Usage

```
prior.prob(quant, alpha, trendsd = 0)
prior.quant(prob = 10^-(1:3), shape, scale, trendsd = 0)
prior.norm(mean, cov, trendsd = 0)
prior.loglognorm(mean, cov, trendsd = 0)
```

Arguments

quant, alpha	Numeric vectors of length three and four respectively. Beta prior distibutions are placed on probability ratios corresponding to the quantiles given in quant.
prob, shape, scale	Numeric vectors of length three. Gamma prior distibutions, with parameters shape and scale, are placed on quantile differences corresponding to the probabilities given in prob.
mean, cov	The prior distibution for the location, log(scale) and shape is taken to be trivariate normal, with mean mean (a numeric vector of length three) and covariance matrix cov (a symmetric positive definite three by three matrix).
trendsd	The standard deviation for the marginal normal prior distribution (with mean zero) placed on the linear trend parameter for the location. If this is zero (the default) a linear trend is not implemented.

Details

See the user's guide.

Value

Returns an object of class `"evprior"`, which is essentially just a list of the arguments passed.

See Also

`posterior`, `pplik`

Examples

```
mat <- diag(c(10000, 10000, 100))
prior.norm(mean = c(0,0,0), cov = mat, trendsd = 10)
prior.quant(shape = c(38.9,7.1,47), scale = c(1.5,6.3,2.6))
prior.prob(quant = c(85,88,95), alpha = c(4,2.5,2.25,0.25))
```

---

rainfall *Daily Aggregate Rainfall*

Description

A numeric vector of length 20820 containing daily aggregate rainfall observations, in millimetres, recorded at a rain gauge in England over a 57 year period, beginning on a leap year. Three of these years contain only missing values.

Usage

```
data(rainfall)
```

Format

A vector containing 20820 observations.

Source

Unknown.

`rl.pred` *Return Level Plots for GEV Predictive Distributions*

Description

Produce return level plots depicting prior and posterior predictive gev distributions.

Usage

```
rl.pred(post, qlim, npy, lh = c("gev", "gpd"), period = 1, lty = 1, col = 1,
xlab = "return period", ylab = "return level", ...)
```

Arguments

`post`	A Markov chain generated using `posterior`, containing samples from the corresponding prior/posterior distribution.
`qlim`	A vector of length two, giving the limits for the quantiles at which the predictive probabilities are calculated.
`npy`	The Number of observation Per Year (in average). If "gev" likelihood, "npy" is supposed to be equal to 1 i.e. annual maxima.
`lh`	The likelihood.
`period`	A vector of integers. One curve is plotted for each element of `period`. The ith curve depicts the probabilities that that quantiles will be exceeded over the next `period[i]` periods.
`lty`	Passed to `matplot`.
`col`	Passed to `matplot`.
`xlab`, `ylab`	Labels for the x and y axes.
`...`	Other arguments passed to `matplot`.

Details

See the user's guide.

Value

The first two arguments to `matplot` are returned invisibly as a list.

If a linear trend on the location has been implemented, the plot corresponds to the distribution obtained when the trend parameter is zero.

See Also

`matplot`, `posterior`

rl.pst *Return Level Plots Depicting Distributions of GEV Quantiles*

Description

Produce return level plots depicting prior and posterior distributions of gev quantiles.

Usage

```
rl.pst(post, npy, lh = c("gev", "gpd"), ci = 0.9, lty = c(2,1), col = c(2,1),
xlab = "return period", ylab = "return level",  ...)
```

Arguments

post	A Markov chain generated using posterior, containing samples from the corresponding prior/posterior distribution.
npy	The Number of observation Per Year (in average). If "gev" likelihood, "npy" is supposed to be equal to 1 i.e. annual maxima.
lh	The likelihood.
ci	The confidence coefficient for the plotted prior/posterior probability interval.
lty	Passed to matplot. The first and second values specify the line type of the probability interval and the median line respectively.
col	Passed to matplot. The first and second values specify the colour of the probability interval and the median line respectively.
xlab, ylab	Labels for the x and y axes.
...	Other arguments passed to matplot.

Details

See the user's guide.

Value

The first two arguments to matplot are returned invisibly as a list.

If a linear trend on the location has been implemented, the plot corresponds to the distribution obtained when the trend parameter is zero.

See Also

matplot, posterior

Index