

# Package ‘extrememix’

July 8, 2021

**Type** Package

**Title** Bayesian Estimation of Extreme Value Mixture Models

**Version** 1.0

**Date** 2021-06-21

**Author** Manuele Leonelli

**Maintainer** Manuele Leonelli <manuele.leonelli@ie.edu>

**Description** Fits extreme value mixture models, which are models for tails not requiring selection of a threshold, for continuous data. It includes functions for model comparison, estimation of quantity of interest in extreme value analysis and plotting.

**License** GPL (>= 2)

**Imports** evd, ggplot2, gridExtra, mixtools, Rcpp, threshr

**LinkingTo** Rcpp

**Encoding** UTF-8

**RoxygenNote** 7.1.1

**Suggests** rmarkdown,  
knitr

**VignetteBuilder** knitr

**URL** <https://github.com/manueleleonelli/extrememix>

## R topics documented:

extrememix-package	2
check_convergence	3
claim	4
DIC	4
ES	5
fggpd	6
fmgpd	8
ggpd	10
logLik.evmm	11
mgamma	11
mgpd	13
plot.evmm	14
plot.upper_bound	15
plot_summaries	15

pred . . . . .	16
print . . . . .	17
quant . . . . .	18
rainfall . . . . .	19
return_level . . . . .	20
summary.evmm . . . . .	21
upper_bound . . . . .	21
VaR . . . . .	22
WAIC . . . . .	23

<b>Index</b>	<b>25</b>
--------------	-----------

---

extrememix-package	<i>Bayesian Estimation of Extreme Value Mixture Models</i>
--------------------	--

---

## Description

Bayesian estimation of various extreme value mixture models using MCMC algorithms. The package provides capabilities for model selection and comparison, estimation of quantities of interest in extreme value analyses and plotting functions taking advantage of the ggplot2 package.

## Details

The functions in extrememix can be divided into four groups according to their purpose:

- Density functions, probability functions, quantile functions and random number generation of many extreme value mixture models:
  - Gamma bulk and GPD tail ([ggpd](#))
  - Mixture of Gammas bulk and GPD tail ([mgpd](#))
- Bayesian estimation using MCMC algorithms:
  - [fggpd](#): estimation of the [ggpd](#) model
  - [fmgpd](#): estimation of the [mgpd](#) model
- Model selection functions:
  - AIC: Aikiake information criterion
  - BIC: Bayesian information criterion
  - [DIC](#): Deviance information criterion
  - [WAIC](#): Widely applicable information criterion
- Summaries for extreme value analysis:
  - [pred](#): plot of the predictive distribution
  - [quant](#): estimated and empirical quantiles
  - [return\\_level](#): estimated and empirical return levels
  - [VaR](#): estimated and empirical value-at-risk
  - [ES](#): estimated and empirical expected shortfall

All estimated models as well as summaries have their own printing and plotting method.

## Author(s)

Manuele Leonelli <[manuele.leonelli@ie.edu](mailto:manuele.leonelli@ie.edu)>

## References

Behrens, Cibele N., Hedibert F. Lopes, and Dani Gamerman. "Bayesian analysis of extreme events with threshold estimation." *Statistical Modelling* 4.3 (2004): 227-244.

do Nascimento, Fernando Ferraz, Dani Gamerman, and Hedibert Freitas Lopes. "A semiparametric Bayesian approach to extreme value estimation." *Statistics and Computing* 22.2 (2012): 661-675.

Li, Yunxian, Niansheng Tang, and Xuejun Jiang. "Bayesian approaches for analyzing earthquake catastrophic risk." *Insurance: Mathematics and Economics* 68 (2016): 110-119.

Scarrott, Carl, and Anna MacDonald. "A review of extreme value threshold estimation and uncertainty quantification." *REVSTAT—Statistical Journal* 10.1 (2012): 33-60.

## Examples

```
## Not run:
  data(rainfall)
  model <- fggpd(rainfall, it = 25000, burn = 5000, thin = 25)
  plot(model)
  WAIC(model)
  quant(model)

## End(Not run)
```

---

check\_convergence

*Convergence Assessment of MCMC Algorithms*

---

## Description

Plot of the traceplot and autocorrelation function for the 0.99 quantile from the posterior sample.

## Usage

```
check_convergence(x, ...)
```

## S3 method for class 'evmm'

```
check_convergence(x, ...)
```

## Arguments

x                    the output of a model estimated with `extrememix`.

...                   additional arguments for compatibility.

## Examples

```
## Not run:
  data(rainfall)
  model1 <- fggpd(rainfall, it = 25000, burn = 5000, thin = 25)
  check_convergence(model1)

## End(Not run)
```

---

claim	<i>Automobile Claims Data</i>
-------	-------------------------------

---

**Description**

Claim amounts of one-year vehicle insurance policies taken out in 2004 or 2005 in Australia

**Usage**

```
data(claim)
```

**Format**

A numeric vector of 4624 insurance claims

**Source**

Charpentier, A. (Ed.). (2014). Computational actuarial science with R. CRC press.

Goegebeur, Yuri, Armelle Guillou, and Jing Qin. "Extreme value estimation of the conditional risk premium in reinsurance." *Insurance: Mathematics and Economics* 96 (2021): 68-80.

---

DIC	<i>Deviance Information Criterion</i>
-----	---------------------------------------

---

**Description**

Computation of the DIC for an extreme value mixture model

**Usage**

```
DIC(x, ...)
```

```
## S3 method for class 'evmm'
DIC(x, ...)
```

**Arguments**

x	the output of a model estimated with <code>extrememix</code>
...	additional arguments for compatibility.

**Details**

Let  $y$  denote a dataset and  $p(y|\theta)$  the likelihood of a parametric model with parameter  $\theta$ . The deviance is defined as  $D(\theta) = -2\log p(y|\theta)$ . The deviance information criterion (DIC) is defined as

$$DIC = D(\hat{\theta}) + 2p_D,$$

where  $\hat{\theta}$  is the posterior estimate of  $\theta$  and  $p_D$  is referred to as the effective number of parameters and defined as

$$E_{\theta|y}(D(\theta)) - D(\hat{\theta}).$$

Models with a smaller DIC are favored.

**Value**

The DIC of a model estimated with `extrememix`

**References**

Spiegelhalter, David J., et al. "Bayesian measures of model complexity and fit." *Journal of the Royal Statistical Society: Series B* 64.4 (2002): 583-639.

**See Also**

[WAIC](#)

**Examples**

```
## Not run:
data(rainfall)
model1 <- fggpd(rainfall, it = 25000, burn = 5000, thin = 25)
DIC(model1)

## End(Not run)
```

---

 ES

---

*Expected Shortfall*


---

**Description**

Computation of the expected shortfall for an extreme value mixture model

**Usage**

```
ES(x, ...)

## S3 method for class 'evmm'
ES(x, values = NULL, cred = 0.95, ...)
```

**Arguments**

<code>x</code>	the output of a model estimated with <code>extrememix</code> .
<code>...</code>	additional arguments for compatibility.
<code>values</code>	numeric vector of values of which to compute the expected shortfall.
<code>cred</code>	amplitude of the posterior credibility interval.

**Details**

The expected shortfall is the expectation of a random variable conditional of being larger of a specific Value-at-Risk (quantile). For an extreme value mixture model this is equal to:

$$ES_p = \frac{VaR_p}{1 - \xi} + \frac{\sigma - \xi u}{1 - \xi}$$

**Value**

A list with the following entries:

- `quantiles`: a matrix containing the estimated shortfall, the posterior credibility intervals and the empirical estimate.
- `data`: the dataset used to estimate the expected shortfall.
- `complete`: a matrix with the expected shortfall for each value in the posterior sample.

**References**

Lattanzi, Chiara, and Manuele Leonelli. "A changepoint approach for the identification of financial extreme regimes." *Brazilian Journal of Probability and Statistics*.

**See Also**

[quant](#), [return\\_level](#), [VaR](#)

**Examples**

```
## Not run:
data(rainfall)
model1 <- fggpd(rainfall, it = 25000, burn = 5000, thin = 25)
ES(model1)

## End(Not run)
```

---

fggpd

*GGPD Estimation*


---

**Description**

Fit of the GGPD model using an MCMC algorithm.

**Usage**

```
fggpd(x, it, start = NULL, var = NULL, prior = NULL, thin = 1, burn = 0)
```

**Arguments**

<code>x</code>	A vector of positive observations.
<code>it</code>	Number of iterations of the algorithm.
<code>start</code>	A list of starting parameter values.
<code>var</code>	A list of starting proposal variances.
<code>prior</code>	A list of hyperparameters for the prior distribution.
<code>thin</code>	Thinning interval.
<code>burn</code>	Burn-in length.

## Details

Estimation of the GGPD is carried out using an adaptive block Metropolis-Hastings algorithm. As standard, the user needs to specify the data to use during estimation, the number of iterations of the algorithm, the burn-in period (by default equal to zero) and the thinning interval (by default equal to one). To run the algorithm it is also needed the choice of the starting values, the starting values of the proposal variances, and the parameters of the prior distribution. If not provided, these are automatically set as follows:

- *starting values*:  $u$  is chosen by the function `ithresh` of the `threshr` package;  $\xi$  and  $\sigma$  are chosen by the `fpot` function of `evd` for data over the threshold;  $\mu$  and  $\eta$  are chosen as the maximum likelihood estimate of the Gamma distribution over data below the threshold.
- *variances*: variances are chosen as the standard deviation of the normal distribution whose 0.01 quantile is equal to 0.9 times the starting value of the associated parameter.
- *prior distributions*: the prior distribution for  $\xi$  and  $\sigma$  is the objective prior

$$p(\xi, \sigma) = \sigma^{-1}(1 + \xi)^{-1}(1 + 2\xi)^{-1/2}.$$

The prior for the threshold  $u$  is Normal with mean chosen as for the starting values and the standard deviation is chosen so that the 0.05 quantile of the prior is equal to the median of the data. The priors for the parameters  $\mu$  and  $\eta$  are Gammas with mean chosen as for the starting values and shapes equal to 0.001 so to give high prior variances.

The user can also select any of the three inputs above.

- The starting values `start` must be a list with entries `xi`, `sigma`, `u`, `mu`, `eta`.
- The variances `var` must be a list with entries `xi`, `sigma`, `u`, `mu`, `eta`.
- The prior `prior` must be a list with entries `u`, `mu`, `eta` all containing a vector of length two (for `u` giving the mean and the standard deviation of the Normal prior, for `mu` and `eta` giving the mean and shape of the Gamma prior).

## Value

`fggpd` returns a list with three elements:

- `chain`: a matrix of size  $(it - burn)/thin \times 5$ , reporting in each column the posterior sample for each parameter.
- `var`: a matrix of size  $it \times 5$  reporting the variance of the proposal distribution for each parameter.
- `data`: the dataset used for estimation.

## References

Behrens, Cibele N., Hedibert F. Lopes, and Dani Gamerman. "Bayesian analysis of extreme events with threshold estimation." *Statistical Modelling* 4.3 (2004): 227-244.

do Nascimento, Fernando Ferraz, Dani Gamerman, and Hedibert Freitas Lopes. "A semiparametric Bayesian approach to extreme value estimation." *Statistics and Computing* 22.2 (2012): 661-675.

## See Also

[fmgpd](#), [ggpd](#)

**Examples**

```
## Not run:
data(rainfall)
model1 <- fggpd(rainfall, it = 25000, burn = 5000, thin = 25)

start <- list(xi = 0.2, sigma = 2, u = 10, mu = 5, eta = 2)
var <- list(xi = 0.01, sigma = 1, u = 3, mu = 3, eta = 1)
prior <- list(u = c(22,5), mu = c(4,16), eta = c(0.001,0.001))
model2 <- fggpd(rainfall,it = 25000, start = start, var =var, prior = prior)

## End(Not run)
```

fmgpd

*MGPD Estimation***Description**

Fit of the MGPD model using an MCMC algorithm.

**Usage**

```
fmgpd(x, it, k, start = NULL, var = NULL, prior = NULL, thin = 1, burn = 0)
```

**Arguments**

x	A vector of positive observations.
it	Number of iterations of the algorithm.
k	number of mixture components for the bulk. Must be either 2, 3, or 4.
start	A list of starting parameter values.
var	A list of starting proposal variance.
prior	A list of hyperparameters for the prior distribution.
thin	Thinning interval.
burn	Burn-in.

**Details**

Estimation of the MGPD is carried out using an adaptive block Metropolis-Hastings algorithm. As standard, the user needs to specify the data to use during estimation, the number of mixture components for the bulk, the number of iterations of the algorithm, the burn-in period (by default equal to zero) and the thinning interval (by default equal to one). To run the algorithm it is also needed the choice of the starting values, the starting values of the proposal variances, and the parameters of the prior distribution. If not provided, these are automatically set as follows:

- *starting values*:  $u$  is chosen by the function `ithresh` of the `threshr` package;  $\xi$  and  $\sigma$  are chosen by the `fpot` function of `evd` for data over the threshold;  $\mu$  and  $\eta$  are chosen as estimates of the `gammamixEM` function from the `mixtools` package;  $w$  is chosen as the vector with entries  $1/k$ .



- *variances*: variances for  $\sigma$  and  $u$  are chosen as the standard deviation of the normal distribution whose 0.01 quantile is equal to 0.9 times the starting value of the associated parameter. The parameters  $\mu_i$  and  $\eta_i$  are sampled jointly and the proposal variance is chosen using the same method as for  $\sigma$  with respect to the parameter  $\mu$ . The proposal variance for  $w$  is 0.1 and the proposal variance for  $\xi$  is 0.1 if negative and 0.25 if positive.
- *prior distributions*: the prior distribution for  $\xi$  and  $\sigma$  is the objective prior

$$p(\xi, \sigma) = \sigma^{-1}(1 + \xi)^{-1}(1 + 2\xi)^{-1/2}.$$

The prior for the threshold  $u$  is Normal with mean chosen as for the starting values and the standard deviation is chosen so that the 0.05 quantile of the prior is equal to the median of the data. The priors for the parameters  $\mu_i$  and  $\eta_i$  are Gammas with mean chosen as for the starting values and shapes equal to 0.001 so to give high prior variances. The prior for the weights is the non-informative Dirichlet with parameter 1.

The user can also select any of the three inputs above.

- The starting values `start` must be a list with entries `xi`, `sigma`, `u`, `mu`, `eta` and `w`. The length of `mu`, `eta` and `w` must be `k`.
- The variances `var` must be a list with entries `xi`, `sigma`, `u`, `mu` and `w`. The length of `mu` must be `k`.
- The prior `prior` must be a list with entries `u`, `mu_mu`, `mu_eta`, `eta_mu` and `eta_eta`. `u` gives the mean and the standard deviation of the Normal prior for  $u$ . The vectors of length `k`, `mu_mu` and `eta_mu` give the prior means of  $\mu$  and  $\eta$ , whilst `mu_eta` and `eta_eta` give their prior shapes.

## Value

fmgpd returns a list with three elements:

- `chain`: a matrix of size  $(it - burn)/thin \times 5$ , reporting in each column the posterior sample for each parameter.
- `var`: a matrix of size  $it \times 5$  reporting the variance of the proposal distribution for each parameter.
- `data`: the dataset used for estimation.

## References

Behrens, Cibele N., Hedibert F. Lopes, and Dani Gamerman. "Bayesian analysis of extreme events with threshold estimation." *Statistical Modelling* 4.3 (2004): 227-244.

do Nascimento, Fernando Ferraz, Dani Gamerman, and Hedibert Freitas Lopes. "A semiparametric Bayesian approach to extreme value estimation." *Statistics and Computing* 22.2 (2012): 661-675.

## See Also

[fggpd](#), [mgpd](#)

## Examples

```
## Not run:
data(rainfall)
model1 <- fmgpd(rainfall, k = 2, it = 25000, burn = 5000, thin = 25)
start <- list(xi = 0.2, sigma = 2, u = 10, mu = c(2,5), eta = c(2,2) , w = c(0.4,0.6))
var <- list(xi = 0.01, sigma = 1, u = 3, mu = c(3,3), w = 0.01)
```

```
prior <- list(u = c(22,5), mu_mu = c(2,5), mu_eta = c(0.01,0.01),
             eta_mu = c(3,3),eta_eta = c(0.01,0.01))

model2 <- fmgpd(rainfall, k= 2, it = 25000, start = start, var =var, prior = prior)

## End(Not run)
```

---

ggpd

*The GGPD distribution*


---

### Description

Density, distribution function, quantile function and random generation for the GGPD distribution.

### Usage

```
dggpd(x, xi, sigma, u, mu, eta, log = FALSE)

pggpd(q, xi, sigma, u, mu, eta, lower.tail = TRUE)

qggpd(p, xi, sigma, u, mu, eta, lower.tail = TRUE)

rggpd(N, xi, sigma, u, mu, eta)
```

### Arguments

x, q	vector of quantiles.
xi	shape parameter of the tail GPD (scalar).
sigma	scale parameter of the tail GPD (scalar).
u	threshold parameter of the tail GPD (scalar).
mu	mean of the gamma bulk (scalar).
eta	shape of the gamma bulk (scalar).
log	logical; if TRUE, probabilities p are given as log(p).
lower.tail	logical; if TRUE (default), probabilities are $P(X \leq x)$ otherwise $P(X > x)$ .
p	vector of probabilities.
N	number of observations.

### Value

The GGPD distribution is an extreme value mixture model with density

$$f_{GGPD}(x|\xi, \sigma, u, \mu, \eta, w) = \begin{cases} f_{GA}(x|\mu, \eta), & x \leq u \\ (1 - F_{GA}(u|\mu, \eta))f_{GPD}(x|\xi, \sigma, u), & \text{otherwise,} \end{cases}$$

where  $f_{MG}$  is the density of the mixture of Gammas,  $F_{MG}$  is the distribution function of the mixture of Gammas and  $f_{GPD}$  is the density of the Generalized Pareto Distribution, i.e.

$$f_{GPD}(x|\xi, \sigma, u) = \begin{cases} 1 - (1 + \frac{\xi}{\sigma}(x - u))^{-1/\xi}, & \text{if } \xi \neq 0, \\ 1 - \exp\left(-\frac{x-u}{\sigma}\right), & \text{if } \xi = 0, \end{cases}$$

where  $\xi$  is a shape parameter,  $\sigma > 0$  is a scale parameter and  $u > 0$  is a threshold.

dggpd gives the density, pggpd gives the distribution function, qggpd gives the quantile function, and rggpd generates random deviates. The length of the result is determined by N for rggpd and by the length of x, q or p otherwise.

## References

Behrens, Cibele N., Hedibert F. Lopes, and Dani Gamerman. "Bayesian analysis of extreme events with threshold estimation." *Statistical Modelling* 4.3 (2004): 227-244.

## See Also

[fggpd](#)

## Examples

```
dggpd(3, xi = 0.5, sigma = 2, u = 5, mu = 3, eta = 3)
```

---

logLik.evmm

*Log-likelihood Method*

---

## Description

Computation of the log-likelihood of an extreme value mixture model (thus also AIC and BIC are available).

## Usage

```
## S3 method for class 'evmm'
logLik(object, ...)
```

## Arguments

object            an object of class evmm.  
 ...                additional parameters for compatibility.

---

mgamma

*The Gamma Mixture Distribution*

---

## Description

Density, distribution function, quantile function and random generation for the mixture of Gamma distribution.

**Usage**

```

dmgamma(x, mu, eta, w, log = FALSE)

pmgamma(q, mu, eta, w, lower.tail = TRUE)

qmgamma(p, mu, eta, w, lower.tail = TRUE)

rmgamma(N, mu, eta, w)

```

**Arguments**

x, q	vector of quantiles.
mu	means of the gamma mixture components (vector).
eta	shapes of the gamma mixture components (vector).
w	weights of the gamma mixture components (vector). Must sum to one.
log	logical; if TRUE, probabilities p are given as log(p).
lower.tail	logical; if TRUE (default), probabilities are $P(X \leq x)$ otherwise $P(X > x)$ .
p	vector of probabilities.
N	number of observations.

**Details**

The Gamma distribution has density

$$f_{GA}(x|\mu, \eta) = \frac{(\eta/\mu)^\eta}{\Gamma(\eta)} x^{\eta-1} \exp(-(\eta/\mu)x), \quad x > 0,$$

where  $\mu > 0$  is the mean of the distribution and  $\xi > 0$  is its shape. The density of a mixture of Gamma distributions with  $k$  components is defined as

$$f_{MG}(x|\mu, \eta, w) = \sum_{i=1}^k w_i f_{GA}(x|\mu_i, \eta_i),$$

where  $w_i, \mu_i, \eta_i > 0$ , for  $i = 1, \dots, k$ ,  $w_1 + \dots + w_k = 1$ ,  $\mu = (\mu_1, \dots, \mu_k)$ ,  $\eta = (\eta_1, \dots, \eta_k)$  and  $w = (w_1, \dots, w_k)$ .

**Value**

dmgamma gives the density, pmgamma gives the distribution function, qmgamma gives the quantile function, and rmgamma generates random deviates.

The length of the result is determined by N for rmgamma and by the length of x, q or p otherwise.

**References**

Wiper, Michael, David Rios Insua, and Fabrizio Ruggeri. "Mixtures of gamma distributions with applications." *Journal of Computational and Graphical Statistics* 10.3 (2001): 440-454.

**Examples**

```
dmgamma(3, mu = c(2,3), eta = c(1,2), w = c(0.3,0.7))
```

mgpd

*The MGPD distribution***Description**

Density, distribution function, quantile function and random generation for the MGPD distribution.

**Usage**

```
dmgpd(x, xi, sigma, u, mu, eta, w, log = FALSE)
pmgpd(q, xi, sigma, u, mu, eta, w, lower.tail = TRUE)
qmgpd(p, xi, sigma, u, mu, eta, w, lower.tail = TRUE)
rmgpd(N, xi, sigma, u, mu, eta, w)
```

**Arguments**

x, q	vector of quantiles.
xi	shape parameter of the tail GPD (scalar).
sigma	scale parameter of the tail GPD (scalar).
u	threshold parameter of the tail GPD (scalar).
mu	means of the gamma mixture components (vector).
eta	shapes of the gamma mixture components (vector).
w	weights of the gamma mixture components (vector). Must sum to one.
log	logical; if TRUE, probabilities p are given as log(p).
lower.tail	logical; if TRUE (default), probabilities are $P(X \leq x)$ otherwise $P(X > x)$ .
p	vector of probabilities.
N	number of observations.

**Details**

The MGPD distribution is an extreme value mixture model with density

$$f_{MGPD}(x|\xi, \sigma, u, \mu, \eta, w) = \begin{cases} f_{MG}(x|\mu, \eta, w), & x \leq u \\ (1 - F_{MG}(u|\mu, \eta, w))f_{GPD}(x|\xi, \sigma, u), & \text{otherwise,} \end{cases}$$

where  $f_{MG}$  is the density of the mixture of Gammas,  $F_{MG}$  is the distribution function of the mixture of Gammas and  $f_{GPD}$  is the density of the Generalized Pareto Distribution, i.e.

$$f_{GPD}(x|\xi, \sigma, u) = \begin{cases} 1 - (1 + \frac{\xi}{\sigma}(x - u))^{-1/\xi}, & \text{if } \xi \neq 0, \\ 1 - \exp\left(-\frac{x-u}{\sigma}\right), & \text{if } \xi = 0, \end{cases}$$

where  $\xi$  is a shape parameter,  $\sigma > 0$  is a scale parameter and  $u > 0$  is a threshold.

**Value**

dmgpd gives the density, pmgpd gives the distribution function, qmgpd gives the quantile function, and rmgpd generates random deviates. The length of the result is determined by N for rmgpd and by the length of x, q or p otherwise.

**References**

do Nascimento, Fernando Ferraz, Dani Gamerman, and Hedibert Freitas Lopes. "A semiparametric Bayesian approach to extreme value estimation." *Statistics and Computing* 22.2 (2012): 661-675.

**See Also**

[fmgpd](#)

**Examples**

```
dmgpd(3, xi = 0.5, sigma = 2,5, u = 5, mu = c(2,3), eta = c(1,2), w = c(0.3,0.7))
```

---

plot.evmm

*Plot of Extreme Value Mixture Models*

---

**Description**

Plotting method for objects of class evmm giving an overview of an estimated model.

**Usage**

```
## S3 method for class 'evmm'
plot(x, ...)
```

**Arguments**

x                    an object of class evmm.  
 ...                    additional parameters for compatibility.

**Details**

The plot method for objects of class evmm reports four plots:

- An histogram of the posterior distribution of xi.
- An histogram of the posterior distribution of sigma.
- A line plot of the estimated quantiles, their posterior credibility interval, and the empirical ones.
- A plot of the predictive distribution together with the data histogram.

**Examples**

```
## Not run:
data(rainfall)
model1 <- fggpd(rainfall, it = 25000, burn = 5000, thin = 25)
plot(model1)

## End(Not run)
```

---

plot.upper\_bound      *Plot Upper Bounds*

---

### Description

Plotting method for the posterior distribution of the upper bound. No plot is reported if the posterior sample of xi has only positive values (unbounded distribution).

### Usage

```
## S3 method for class 'upper_bound'
plot(x, xlim = c(min(x$bound), max(x$bound)), ...)
```

### Arguments

x                    an object of class upper\_bound.  
 xlim                limits of the x-axis.  
 ...                 additional parameters for compatibility.

### Examples

```
## Not run:
data(rainfall)
model1 <- fggpd(rainfall, it = 25000, burn = 5000, thin = 25)
plot(upper_bound(model1))

## End(Not run)
```

---

plot\_summaries      *Plot Methods for Summaries*

---

### Description

Plotting methods for objects created with quant, ES, return\_level or VaR.

### Usage

```
## S3 method for class 'quant'
plot(x, ylim = NULL, ...)

## S3 method for class 'return_level'
plot(x, ylim = NULL, ...)

## S3 method for class 'VaR'
plot(x, ylim = NULL, ...)

## S3 method for class 'ES'
plot(x, ylim = NULL, ...)
```

**Arguments**

x                    an object of class quant, ES, return\_level or VaR.  
 ylim                limits of the y-axis.  
 ...                 additional parameters for compatibility.

**Details**

Two types of plot can be output: either a line plot in the case the functions quant, ES, return\_level or VaR where called with more than one value for the input values, or an histogram otherwise.

**Examples**

```
## Not run:
data(rainfall)
model1 <- fggpd(rainfall, it = 25000, burn = 5000, thin = 25)
plot(return_level(model1)) ## for line plot
plot(return_level(model1, values = 100)) ## for histogram

## End(Not run)
```

---

 pred

*Predictive Distribution*


---

**Description**

Plot of the predictive distribution of an extreme value mixture model.

**Usage**

```
pred(x, ...)
```

```
## S3 method for class 'evmm'
pred(
  x,
  x_axis = seq(min(x$data), max(x$data), length.out = 1000),
  cred = 0.95,
  xlim = c(min(x$data), max(x$data)),
  ylim = NULL,
  ...
)
```

**Arguments**

x                    the output of a model estimated with extrememix.  
 ...                 additional arguments for compatibility.  
 x\_axis             vector of points where to estimate the predictive distribution.  
 cred                amplitude of the posterior credibility interval.  
 xlim                limits of the x-axis.  
 ylim                limits of the y-axis.



**Details**

Consider an extreme value mixture model  $f(y|\theta)$  and suppose a sample  $(\theta^{(1)}, \dots, \theta^{(S)})$  from the posterior distribution is available. The predictive distribution at the point  $y$  is estimated as

$$\frac{1}{S} \sum_{s=1}^S f(y|\theta^{(s)})$$

**Value**

A plot of the estimate of the predictive distribution together with the data histogram.

**References**

do Nascimento, Fernando Ferraz, Dani Gamerman, and Hedibert Freitas Lopes. "A semiparametric Bayesian approach to extreme value estimation." *Statistics and Computing* 22.2 (2012): 661-675.

**Examples**

```
## Not run:
data(rainfall)
model1 <- fggpd(rainfall, it = 25000, burn = 5000, thin = 25)
pred(model1)

## End(Not run)
```

---

print

*Printing Methods*


---

**Description**

Collection of printing methods for various objects created by `extrememix`.

**Usage**

```
## S3 method for class 'evmm'
print(x, ...)

## S3 method for class 'summary.ggpd'
print(x, ...)

## S3 method for class 'quantile'
print(x, ...)

## S3 method for class 'return_level'
print(x, ...)

## S3 method for class 'VaR'
print(x, ...)

## S3 method for class 'ES'
```

```
print(x, ...)

## S3 method for class 'upper_bound'
print(x, ...)
```

### Arguments

`x` an object created by `extrememix`.  
`...` additional arguments for compatibility.

---

quant	<i>Estimated Quantiles</i>
-------	----------------------------

---

### Description

Computation of posterior quantiles for an extreme value mixture model

### Usage

```
quant(x, ...)

## S3 method for class 'evmm'
quant(x, values = NULL, cred = 0.95, ...)
```

### Arguments

`x` the output of a model estimated with `extrememix`.  
`...` additional arguments for compatibility.  
`values` numeric vector of values of which to compute the quantile.  
`cred` amplitude of the posterior credibility interval.

### Details

For a random variable  $X$  the  $p$ -quantile is the value  $x$  such that  $P(X > x) = 1 - p$ . For an extreme value mixture model this can be computed as

$$x = u + \frac{\sigma}{\xi} ((1 - p^*)^{-\xi} - 1),$$

where

$$p^* = \frac{p - F_{\text{bulk}}(u|\theta)}{1 - F_{\text{bulk}}(u|\theta)},$$

and  $F_{\text{bulk}}$  is the distribution function of the bulk, parametrized by  $\theta$ .

### Value

A list with the following entries:

- `quantiles`: a matrix containing the quantiles, the posterior credibility intervals and the empirical estimate.
- `data`: the dataset used to estimate the quantiles.
- `complete`: a matrix with the quantiles for each value in the posterior sample.

## References

do Nascimento, Fernando Ferraz, Dani Gamerman, and Hedibert Freitas Lopes. "A semiparametric Bayesian approach to extreme value estimation." *Statistics and Computing* 22.2 (2012): 661-675.

## See Also

[ES](#), [return\\_level](#), [VaR](#)

## Examples

```
## Not run:
data(rainfall)
model1 <- fggpd(rainfall, it = 25000, burn = 5000, thin = 25)
quant(model1)

## End(Not run)
```

---

rainfall

*Monthly Maxima Daily Rainfall in Madrid*

---

## Description

Monthly maxima of the daily rainfall (measured in mms) recorded at the Retiro station in the city centre of Madrid, Spain, between 1985 and 2020.

## Usage

```
data(rainfall)
```

## Format

A positive numeric vector of length 414. Observations where the monthly maxima are zero were discarded.

## Source

Instituto de Estadística, Comunidad de Madrid.

---

return_level	<i>Return Levels</i>
--------------	----------------------

---

### Description

Computation of the return levels for an extreme value mixture model

### Usage

```
return_level(x, ...)
```

```
## S3 method for class 'evmm'
return_level(x, values = NULL, cred = 0.95, ...)
```

### Arguments

x	the output of a model estimated with <code>extrememix</code>
...	additional arguments for compatibility.
values	numeric vector of values of which to compute the value at risk.
cred	amplitude of the posterior credibility interval.

### Details

A return level at  $T$  units of time is defined as the  $1 - 1/T$  quantile.

### Value

A list with the following entries:

- `quantiles`: a matrix containing the estimated return levels, the posterior credibility intervals and the empirical estimate.
- `data`: the dataset used to estimate the return levels.
- `complete`: a matrix with the return levels for each value in the posterior sample.

### References

do Nascimento, Fernando Ferraz, Dani Gamerman, and Hedibert Freitas Lopes. "A semiparametric Bayesian approach to extreme value estimation." *Statistics and Computing* 22.2 (2012): 661-675.

### See Also

[ES](#), [quant](#), [VaR](#)

### Examples

```
## Not run:
data(rainfall)
model1 <- fggpd(rainfall, it = 25000, burn = 5000, thin = 25)
return_level(model1)

## End(Not run)
```

---

summary.evmm	<i>Summary Method</i>
--------------	-----------------------

---

**Description**

Posterior estimates and credibility intervals for the parameters of extreme value mixture models.

**Usage**

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

**Arguments**

object	an object of class evmm.
...	additional parameters (compatibility).

---

upper_bound	<i>Upper Bound</i>
-------------	--------------------

---

**Description**

Computation of the upper bound of the distribution

**Usage**

```
upper_bound(x, ...)

## S3 method for class 'evmm'
upper_bound(x, cred = 0.95, ...)
```

**Arguments**

x	the output of a model estimated with <code>extrememix</code> .
...	additional arguments for compatibility.
cred	amplitude of the posterior credibility interval.

**Details**

For an extreme value mixture model with a shape parameter  $\xi < 0$  the distribution is right-bounded with upper limit equal to  $u - \sigma/\xi$ .

**Value**

`upper_bound` returns a list with entries:

- `bound`: a sample from the posterior distribution of the upper limit of the model, taken over the posterior values of  $\xi$  which are negative.
- `prob`: the posterior probability that the distribution is unbounded.
- `cred`: the requested amplitude of the posterior credibility intervals.

## References

Coles, Stuart, et al. An introduction to statistical modeling of extreme values. Vol. 208. London: Springer, 2001.

## Examples

```
## Not run:
data(rainfall)
model1 <- fggpd(rainfall, it = 25000, burn = 5000, thin = 25)
upper_bound(model1)

## End(Not run)
```

---

VaR

*Value-at-Risk*

---

## Description

Computation of the Value-at-Risk for an extreme value mixture model.

## Usage

```
VaR(x, ...)

## S3 method for class 'evmm'
VaR(x, values = NULL, cred = 0.95, ...)
```

## Arguments

<code>x</code>	the output of a model estimated with <code>extrememix</code>
<code>...</code>	additional arguments for compatibility.
<code>values</code>	numeric vector of values of which to compute the value at risk.
<code>cred</code>	amplitude of the posterior credibility interval.

## Details

The Value-at-Risk for level  $q\%$  is defined as the  $1-q/100$  quantile of the model.

## Value

A list with the following entries:

- `quantiles`: a matrix containing the estimated value at risk, the posterior credibility intervals and the empirical estimate.
- `data`: the dataset used to estimate the value at risk.
- `complete`: a matrix with the value at risk for each value in the posterior sample.

## References

Lattanzi, Chiara, and Manuele Leonelli. "A changepoint approach for the identification of financial extreme regimes." *Brazilian Journal of Probability and Statistics*.

**See Also**

[ES](#), [quant](#), [return\\_level](#)

**Examples**

```
## Not run:
data(rainfall)
model1 <- fggpd(rainfall, it = 25000, burn = 5000, thin = 25)
VaR(model1)

## End(Not run)
```

---

 WAIC

*Widely Applicable Information Criteria*


---

**Description**

Computation of the WAIC for an extreme value mixture model.

**Usage**

```
WAIC(x, ...)

## S3 method for class 'evmm'
WAIC(x, ...)
```

**Arguments**

`x` the output of a model estimated with `extrememix`.  
`...` additional arguments for compatibility.

**Details**

Consider a dataset  $y = (y_1, \dots, y_n)$ ,  $p(y|\theta)$  the likelihood of a parametric model with parameter  $\theta$ , and  $(\theta^{(1)}, \dots, \theta^{(S)})$  a sample from the posterior distribution  $p(\theta|y)$ . Define

$$\text{llpd} = \sum_{i=1}^n \log \left( \sum_{s=1}^S p(y_i|\theta^{(s)}) \right)$$

and

$$p_{\text{WAIC}} = \sum_{i=1}^n \text{Var}_{\theta|y}(\log p(y_i|\theta)).$$

Then the Widely Applicable Information Criteria is defined as

$$\text{WAIC} = -2\text{llpd} + 2p_{\text{WAIC}}.$$

Models with a smaller WAIC are favored.

**Value**

The WAIC of a model estimated with `extrememix`

**References**

Gelman, Andrew, Jessica Hwang, and Aki Vehtari. "Understanding predictive information criteria for Bayesian models." *Statistics and computing* 24.6 (2014): 997-1016.

Watanabe, Sumio. "A widely applicable Bayesian information criterion." *Journal of Machine Learning Research* 14.Mar (2013): 867-897.

**See Also**

[DIC](#)

**Examples**

```
## Not run:  
data(rainfall)  
model1 <- fggpd(rainfall, it = 25000, burn = 5000, thin = 25)  
WAIC(model1)  
  
## End(Not run)
```



# Index

- \* **datasets**
  - claim, 4
  - rainfall, 19
- check\_convergence, 3
- claim, 4
- dggpd (ggpd), 10
- DIC, 2, 4, 24
- dmgamma (mgamma), 11
- dmgpd (mgpd), 13
- ES, 2, 5, 19, 20, 23
- extrememix (extrememix-package), 2
- extrememix-package, 2
- fggpd, 2, 6, 9, 11
- fmgpd, 2, 7, 8, 14
- ggpd, 2, 7, 10
- logLik.evmm, 11
- mgamma, 11
- mgpd, 2, 9, 13
- pggpd (ggpd), 10
- plot.ES(plot\_summaries), 15
- plot.evmm, 14
- plot.quant(plot\_summaries), 15
- plot.return\_level(plot\_summaries), 15
- plot.upper\_bound, 15
- plot.VaR(plot\_summaries), 15
- plot\_summaries, 15
- pmgamma (mgamma), 11
- pmgpd (mgpd), 13
- pred, 2, 16
- print, 17
- qggpd (ggpd), 10
- qmgamma (mgamma), 11
- qmgpd (mgpd), 13
- quant, 2, 6, 18, 20, 23
- rainfall, 19
- return\_level, 2, 6, 19, 20, 23
- rggpd (ggpd), 10
- rmgamma (mgamma), 11
- rmgpd (mgpd), 13
- summary.evmm, 21
- upper\_bound, 21
- VaR, 2, 6, 19, 20, 22
- WAIC, 2, 5, 23