Package 'IDF'

April 9, 2021

Type Package

Title Estimation and Plotting of IDF Curves

Version 2.1.0

Date 2021-04-07

Description Intensity-duration-frequency (IDF) curves are a widely used analysis-tool in hydrology to assess extreme values of precipitation [e.g. Mailhot et al., 2007, <doi:10.1016/j.jhydrol.2007.09.019>]. The package 'IDF' provides functions to estimate IDF parameters for given precipitation time series on the basis of a duration-dependent generalized extreme value distribution [Koutsoyiannis et al., 1998, <doi:10.1016/S0022-1694(98)00097-3>].

Author Jana Ulrich [aut, cre],
Laura Mack [ctb],
Oscar E. Jurado [ctb],
Felix S. Fauer [ctb],
Christoph Ritschel [aut],
Carola Detring [ctb],
Sarah Joedicke [ctb]

Maintainer Jana Ulrich <jana.ulrich@met.fu-berlin.de>

Imports stats, evd, ismev, RcppRoll, pbapply, fastmatch

License GPL (>= 2)

Encoding UTF-8

URL https://gitlab.met.fu-berlin.de/Rpackages/idf_package

LazyData true

RoxygenNote 7.1.1

NeedsCompilation no

Repository CRAN

Date/Publication 2021-04-09 09:40:02 UTC

R topics documented:

`IDF-package` *Introduction*

Description

This package provides functions to estimate IDF relations for given precipitation time series on the basis of a duration-dependent generalized extreme value distribution (d-GEV). The central function is `gev.d.fit`, which uses the method of maximum-likelihood estimation for the d-GEV parameters, whereby it is possible to include generalized linear modeling for each parameter. This function was implemented on the basis of `gev.fit`. For more detailed information on the methods and the application of the package for estimating IDF curves with spatial covariates, see Ulrich et. al (2020).

Details

- The **d-GEV** is defined following Koutsoyiannis et al. (1998):

  $$G(x) = \exp[-(1 + \xi(x/\sigma(d) - \tilde{\mu}))^{-1/\xi}]$$

  defined on $\{x : 1 + \xi(x/\sigma(d) - \tilde{\mu} > 0)\}$, with the duration dependent scale parameter $\sigma(d) = \sigma_0/(d + \theta)^\eta > 0$, modified location parameter $\tilde{\mu} = \mu/\sigma(d) \in R$ and shape parameter $\xi \in R$, $\xi \neq 0$. The parameters $\theta \leq 0$ and $0 < \eta < 1$ are duration offset and duration exponent and describe the slope and curvature in the resulting IDF curves, respectively.
- The dependence of scale and location parameter on duration, $\sigma(d)$ and $\mu(d)$, can be extended by multiscaling and flattening, if requested. Multiscaling introduces a second duration exponent η_2, enabling the model to change slope linearly with return period. Flattening adds a parameter τ, that flattens the IDF curve for long durations:

  $$\sigma(x) = \sigma_0(d + \theta)^{-\eta_2} + \tau$$

  $$\mu(x) = \tilde{\mu}(\sigma_0(d + \theta)^{-\eta_1} + \tau)$$
- A useful introduction to **Maximum Likelihood Estimation** for fitting for the generalized extreme value distribution (GEV) is provided by Coles (2001). It should be noted, however, that this method uses the assumption that block maxima (of different durations or stations) are independent of each other.

References

- Ulrich, J.; Jurado, O.E.; Peter, M.; Scheibel, M.; Rust, H.W. Estimating IDF Curves Consistently over Durations with Spatial Covariates. Water 2020, 12, 3119, https://doi.org/10.3390/w12113119
- Demetris Koutsoyiannis, Demosthenes Kozonis, Alexandros Manetas, A mathematical framework for studying rainfall intensity-duration-frequency relationships, Journal of Hydrology, Volume 206, Issues 1–2,1998,Pages 118-135,ISSN 0022-1694, https://doi.org/10.1016/S0022-1694(98)00097-3
- Coles, S.An Introduction to Statistical Modeling of Extreme Values; Springer: New York, NY, USA, 2001, https://doi.org/10.1198/tech.2002.s73

Examples

```
## Here are a few examples to illustrate the order in which the functions are intended to be used.

## Step 0: sample 20 years of example hourly 'precipitation' data
```

---

dgev.d — *d-GEV probability density function*

Description

Probability density function of duration-dependent GEV distribution

Usage

```
dgev.d(q, mut, sigma0, xi, theta, eta, d, eta2 = NULL, tau = 0, ...)
```

Arguments

`q`	vector of quantiles
`mut, sigma0, xi`	numeric value, giving modified location $\tilde{\mu}$, scale offset σ_0 and shape parameter ξ.
`theta`	numeric value, giving duration offset θ (defining curvature of the IDF curve)
`eta`	numeric value, giving duration exponent η (defining slope of the IDF curve)
`d`	positive numeric value, giving duration
`eta2`	numeric value, giving a second duration exponent η_2 (needed for multiscaling). Default: NULL, treated as $\eta_2 = \eta$.
`tau`	numeric value, giving intensity offset τ (defining flattening of the IDF curve). Default: $\tau = 0$.
`...`	additional parameters passed to `dgev`

Details

For details on the d-GEV and the parameter definitions, see IDF-package.

Value

list containing vectors of density values for given quantiles. The first element of the list are the density values for the first given duration etc.

See Also

`pgev.d`, `qgev.d`, `rgev.d`

Examples

```
x <- seq(4,20,0.1)
# calculate probability density for one duration
dgev.d(q=x,mut=4,sigma0=2,xi=0,theta=0.1,eta=0.1,d=1)

# calculate probability density for different durations
ds <- 1:4
dens <- lapply(ds,dgev.d,q=x,mut=4,sigma0=2,xi=0,theta=0.1,eta=0.1)

plot(x,dens[[1]],type='l',ylim = c(0,0.21),ylab = 'Probability Density')
for(i in 2:4){
  lines(x,dens[[i]],lty=i)
}
legend('topright',title = 'Duration',legend = 1:4,lty=1:4)
```

`example` *Sampled data for duration-dependent GEV*

Description

Randomly sampled data set used for running the example code, containing:

- `$xdat`: 'annual' maxima values
- `$ds`: corresponding durations
- `$cov1`, `$cov2`: covariates

d-GEV parameters used for sampling:

- $\tilde{\mu} = 4 + 0.2cov_1 + 0.5cov_2$
- $\sigma_0 = 2 + 0.5cov_1$
- $\xi = 0.5$
- $\theta = 0$
- $\eta = 0.5$
- $\eta_2 = 0.5$
- $\tau = 0$

Usage

```
data('example',package ='IDF')
```

Format

A data frame with 330 rows and 4 variables

`gev.d.diag` *Diagnostic Plots for d-gev Models*

Description

Produces diagnostic plots for d-gev models using the output of the function `gev.d.fit`. Values for different durations can be plotted in different colors of with different symbols.

Usage

```
gev.d.diag(
  fit,
  subset = NULL,
  cols = NULL,
  pch = NULL,
  which = "both",
  mfrow = c(1, 2),
  legend = TRUE,
  title = c("Residual Probability Plot", "Residual Quantile Plot"),
  emp.lab = "Empirical",
  mod.lab = "Model",
  ...
)
```

Arguments

`fit`	object returned by `gev.d.fit`
`subset`	an optional vector specifying a subset of observations to be used in the plot
`cols`	optional either one value or vector of same length as `unique(fit$ds)` to specify the colors of plotting points. The default uses the `rainbow` function.
`pch`	optional either one value or vector of same length as `unique(fit$ds)` containing integers or symbols to specify the plotting points.
`which`	string containing 'both', 'pp' or 'qq' to specify, which plots should be produced.
`mfrow`	vector specifying layout of plots. If both plots should be produced separately, set to `c(1,1)`.
`legend`	logical indicating if legends should be plotted
`title`	character vector of length 2, giving the titles for the pp- and the qq-plot
`emp.lab, mod.lab`	character string containing names for empirical and model axis
`...`	additional parameters passed on to the plotting function

Examples

```
data('example',package ='IDF')

fit <- gev.d.fit(xdat=example$dat,ds = example$d,ydat=as.matrix(example[,c('cov1','cov2')])
                 ,mutl=c(1,2),sigma0l=1)
# diagnostic plots for complete data
gev.d.diag(fit,pch=1)
# diagnostic plots for subset of data (e.g. one station)
gev.d.diag(fit,subset = example$cov1==1,pch=1)
```

---

gev.d.fit *Maximum-likelihood Fitting of the duration-dependent GEV Distribution*

Description

Modified gev.fit function for Maximum-likelihood fitting for the duration-dependent generalized extreme value distribution, following Koutsoyiannis et al. (1998), including generalized linear modeling of each parameter.

Usage

```
gev.d.fit(
  xdat,
  ds,
  ydat = NULL,
  mutl = NULL,
  sigma0l = NULL,
  xil = NULL,
  thetal = NULL,
  etal = NULL,
  taul = NULL,
  eta2l = NULL,
  mutlink = make.link("identity"),
  sigma0link = make.link("identity"),
  xilink = make.link("identity"),
  thetalink = make.link("identity"),
  etalink = make.link("identity"),
  taulink = make.link("identity"),
  eta2link = make.link("identity"),
  init.vals = NULL,
  theta_zero = FALSE,
  tau_zero = TRUE,
  eta2_zero = TRUE,
  show = TRUE,
  method = "Nelder-Mead",
  maxit = 10000,
```

```
  ...
)
```

Arguments

- `xdat` — A vector containing maxima for different durations. This can be obtained from `IDF.agg`.
- `ds` — A vector of aggregation levels corresponding to the maxima in xdat. 1/60 corresponds to 1 minute, 1 corresponds to 1 hour.
- `ydat` — A matrix of covariates for generalized linear modeling of the parameters (or NULL (the default) for stationary fitting). The number of rows should be the same as the length of xdat.
- `mutl, sigma0l, xil, thetal, etal, taul, eta2l` — Numeric vectors of integers, giving the columns of ydat that contain covariates for generalized linear modeling of the parameters (or NULL (the default) if the corresponding parameter is stationary). Parameters are: modified location, scale offset, shape, duration offset, duration exponent, respectively.
- `mutlink, sigma0link, xilink, thetalink, etalink, eta2link, taulink` — Link functions for generalized linear modeling of the parameters, created with `make.link`. The default is `make.link("identity")`.
- `init.vals` — list, giving initial values for all or some parameters (order: mut, sigma0, xi, theta, eta, eta2, tau). If one of those parameters shall not be used (see theta_zero, eta2_zero, tau_zero), the number of parameters has to be reduced accordingly. If some or all given values in init.vals are NA or no init.vals at all is declared (the default), initial parameters are obtained internally by fitting the GEV separately for each duration and applying a linear model to obtain the duration dependency of the location and shape parameter. Initial values for covariate parameters are assumed as 0 if not given.
- `theta_zero` — Logical value, indicating whether theta should be estimated (FALSE, the default) or should stay zero.
- `tau_zero, eta2_zero` — Logical values, indicating whether tau,eta2 should be estimated (TRUE, the default).
- `show` — Logical; if TRUE (the default), print details of the fit.
- `method` — The optimization method used in `optim`.
- `maxit` — The maximum number of iterations.
- `...` — Other control parameters for the optimization.

Details

For details on the d-GEV and the parameter definitions, see IDF-package.

Value

A list containing the following components. A subset of these components are printed after the fit. If `show` is TRUE, then assuming that successful convergence is indicated, the components nllh, mle and se are always printed.

nllh	single numeric giving the negative log-likelihood value
mle	numeric vector giving the MLE's for the modified location, scale_0, shape, duration offset and duration exponent, resp. If requested, contains also second duration exponent and intensity-offset
se	numeric vector giving the standard errors for the MLE's (in the same order)
trans	A logical indicator for a non-stationary fit.
model	A list with components mutl, sigma0l, xil, thetal and etal. If requested, contains also eta2l and taul
link	A character vector giving inverse link functions.
conv	The convergence code, taken from the list returned by optim. A zero indicates successful convergence.
data	data is standardized to standard Gumbel.
cov	The covariance matrix.
vals	Parameter values for every data point.
init.vals	Initial values that were used.
ds	Durations for every data point.

See Also

IDF-package, IDF.agg, gev.fit, optim

Examples

```
# sampled random data from d-gev with covariates
# GEV parameters:
# mut = 4 + 0.2*cov1 +0.5*cov2
# sigma0 = 2+0.5*cov1
# xi = 0.5
# theta = 0
# eta = 0.5
# eta2 = 0.5
# tau = 0

data('example',package ='IDF')

gev.d.fit(xdat=example$dat,ds = example$d,ydat=as.matrix(example[,c('cov1','cov2')])
,mutl=c(1,2),sigma0l=1)
```

---

gev.d.lik *d-GEV Likelihood*

Description

Computes (log-) likelihood of d-GEV model

Usage

```
gev.d.lik(
  xdat,
  ds,
  mut,
  sigma0,
  xi,
  theta,
  eta,
  log = FALSE,
  tau = 0,
  eta2 = NULL
)
```

Arguments

`xdat`	numeric vector containing observations
`ds`	numeric vector containing corresponding durations (1/60 corresponds to 1 minute, 1 corresponds to 1 hour)
`mut, sigma0, xi, theta, eta, eta2, tau`	numeric vectors containing corresponding estimates for each of the parameters
`log`	Logical; if TRUE, the log likelihood is returned.

Value

single value containing (log) likelihood

Examples

```
# compute log-likelihood of observation values not included in fit
train.set <- example[example$d!=2,]
test.set <- example[example$d==2,]
fit <- gev.d.fit(train.set$dat,train.set$d,mutl = c(1,2),sigma0l = 1
          ,ydat = as.matrix(train.set[c('cov1','cov2')]))
params <- gev.d.params(fit,ydat = as.matrix(test.set[c('cov1','cov2')]))
gev.d.lik(xdat = test.set$dat,ds = test.set$d,mut = params[,1],sigma0 = params[,2],xi = params[,3]
          ,theta = params[,4],eta = params[,5],log=TRUE)
```

`gev.d.params` *Calculate gev(d) parameters from* `gev.d.fit` *output*

Description

function to calculate mut, sigma0, xi, theta, eta, eta2, tau (modified location, scale offset, shape, duration offset, duration exponent, second duration exponent, intensity offset) from results of `gev.d.fit` with covariates or link functions other than identity.

Usage

```
gev.d.params(fit, ydat = NULL)
```

Arguments

`fit`	fit object returned by `gev.d.fit` or `gev.fit`
`ydat`	A matrix containing the covariates in the same order as used in `gev.d.fit`.

Value

data.frame containing mu_tilde, sigma0, xi, theta, eta, eta2, tau (or mu, sigma, xi for gev.fit objects)

See Also

`IDF-package`

Examples

```
data('example',package = 'IDF')
fit <- gev.d.fit(example$dat,example$d,ydat = as.matrix(example[,c("cov1","cov2")])
                 ,mutl = c(1,2),sigma0l = 1)
gev.d.params(fit = fit,ydat = cbind(c(0.9,1),c(0.5,1)))
```

`IDF.agg` *Aggregation and annual maxima for chosen durations*

Description

Aggregates several time series for chosen durations and finds annual maxima (either for the whole year or chosen months). Returns data.frame that can be used for the function `gev.d.fit`.

Usage

```
IDF.agg(
  data,
  ds,
  na.accept = 0,
  which.stations = NULL,
  which.mon = list(0:11),
  names = c("date", "RR"),
  cl = 1
)
```

Arguments

- `data` — list of data.frames containing time series for every station. The data.frame must have the columns 'date' and 'RR' unless other names are specified in the parameter 'names'. The column 'date' must contain strings with standard date format.
- `ds` — numeric vector of aggregation durations in hours. (Must be multiples of time resolution at all stations.)
- `na.accept` — numeric in [0,1) giving maximum percentage of missing values for which block max. should still be calculated.
- `which.stations` — optional, subset of stations. Either numeric vector or character vector containing names of elements in data. If not given, all elements in 'data' will be used.
- `which.mon` — optional, subset of months (as list containing values from 0 to 11) of which to calculate the annual maxima from.
- `names` — optional, character vector of length 2, containing the names of the columns to be used.
- `cl` — optional, number of cores to be used from `parLapply` for parallel computing.

Details

If data contains stations with different time resolutions that need to be aggregated at different durations, IDF.agg needs to be run separately for the different groups of stations. Afterwards the results can be joint together using 'rbind'.

Value

data.frame containing the annual intensity maxima [mm/h] in '$xdat', the corresponding duration in '$ds', the '$year' and month ('$mon') in which the maxima occurred and the station id or name in '$station'.

See Also

`pgev.d`

Examples

```
dates <- as.Date("2019-01-01")+0:729
x <- rgamma(n = 730, shape = 0.4, rate = 0.5)
df <- data.frame(date=dates,RR=x)

# get annual maxima
IDF.agg(list('Sample'= df),ds=c(24,48),na.accept = 0.01)

##      xdat ds year  mon station
## 0.2853811 24 2019 0:11  Sample
## 0.5673122 24 2020 0:11  Sample
## 0.1598448 48 2019 0:11  Sample
## 0.3112713 48 2020 0:11  Sample

# get monthly maxima for each month of june, july and august
```

```
IDF.agg(list('Sample'=df),ds=c(24,48),na.accept = 0.01,which.mon = list(5,6,7))

# get maxima for time range from june to august
IDF.agg(list('Sample'=df),ds=c(24,48),na.accept = 0.01,which.mon = list(5:7))
```

IDF.plot *Plotting of IDF curves at a chosen station*

Description

Plotting of IDF curves at a chosen station

Usage

```
IDF.plot(
  durations,
  fitparams,
  probs = c(0.5, 0.9, 0.99),
  cols = 4:2,
  add = FALSE,
  legend = TRUE,
  ...
)
```

Arguments

durations	vector of durations for which to calculate the quantiles.
fitparams	vector containing parameters mut, sigma0, xi, theta, eta (modified location, scale offset, shape, duration offset, duration exponent) for chosen station as obtained from gev.d.fit (or gev.d.params for model with covariates).
probs	vector of non-exceedance probabilities for which to plot IDF curves (p = 1-1/(Return Period))
cols	vector of colors for IDF curves. Should have same length as probs
add	logical indicating if plot should be added to existing plot, default is FALSE
legend	logical indicating if legend should be plotted (TRUE, the default)
...	additional parameters passed on to the plot function

Examples

```
data('example',package = 'IDF')
# fit d-gev
fit <- gev.d.fit(example$dat,example$d,ydat = as.matrix(example[,c("cov1","cov2")])
                 ,mutl = c(1,2),sigma0l = 1)
# get parameters for cov1 = 1, cov2 = 1
par <- gev.d.params(fit = fit, ydat = matrix(1,1,2))
```

```
# plot quantiles
IDF.plot(durations = seq(0.5,35,0.2),fitparams = par)
# add data points
points(example[example$cov1==1,]$d,example[example$cov1==1,]$dat)
```

---

pgev.d *d-GEV cumulative distribution function*

Description

Cumulative probability distribution function of duration-dependent GEV distribution

Usage

```
pgev.d(q, mut, sigma0, xi, theta, eta, d, tau = 0, eta2 = NULL, ...)
```

Arguments

q	vector of quantiles
mut, sigma0, xi	numeric value, giving modified location, modified scale and shape parameter
theta	numeric value, giving duration offset (defining curvature of the IDF curve)
eta	numeric value, giving duration exponent (defining slope of the IDF curve)
d	positive numeric value, giving duration
tau	numeric value, giving intensity offset τ (defining flattening of the IDF curve). Default: $\tau = 0$.
eta2	numeric value, giving a second duration exponent η_2 (needed for multiscaling). Default: NULL, treated as $\eta_2 = \eta$.
...	additional parameters passed to pgev

Details

The duration dependent GEV distribution is defined after [Koutsoyiannis et al., 1998]:

$$G(x) = \exp[-(1 + \xi(x/\sigma(d) - \mu_t))^{-1/\xi}]$$

with the duration dependent scale $\sigma(d) = \sigma_0/(d + \theta)^\eta$ and modified location parameter $\mu_t = \mu/\sigma(d)$.

For details on the d-GEV and the parameter definitions, see IDF-package.

Value

list containing vectors of probability values for given quantiles. The first element of the list are the probability values for the first given duration etc.

See Also

dgev.d, qgev.d, rgev.d

Examples

```
x <- seq(4,20,0.1)
prob <- pgev.d(q=x,mut=4,sigma0=2,xi=0,theta=0.1,eta=0.1,d=1)
```

---

qgev.d — *d-GEV quantile function*

Description

Quantile function of duration-dependent GEV distribution (inverse of the cumulative probability distribution function)

Usage

```
qgev.d(p, mut, sigma0, xi, theta, eta, d, tau = 0, eta2 = NULL, ...)
```

Arguments

`p`	vector of probabilities
`mut, sigma0, xi`	numeric value, giving modified location, modified scale and shape parameter
`theta`	numeric value, giving duration offset (defining curvature of the IDF curve for short durations)
`eta`	numeric value, giving duration exponent (defining slope of the IDF curve)
`d`	positive numeric value, giving duration
`tau`	numeric value, giving intensity offset τ (defining flattening of the IDF curve). Default: $\tau = 0$.
`eta2`	numeric value, giving a second duration exponent η_2 (needed for multiscaling). Default: NULL, treated as $\eta_2 = \eta$.
`...`	additional parameters passed to `qgev`

Details

The duration dependent GEV distribution is defined after [Koutsoyiannis et al., 1998]:

$$G(x) = \exp[-(1 + \xi(x/\sigma(d) - \mu_t))^{-1/\xi}]$$

with the duration dependent scale $\sigma(d) = \sigma_0/(d + \theta)^\eta$ and modified location parameter $\mu_t = \mu/\sigma(d)$.

For details on the d-GEV and the parameter definitions, see IDF-package.

Value

list containing vectors of quantile values for given probabilities. The first element of the list are the q. values for the first given duration etc.

See Also

pgev.d, dgev.d, rgev.d

Examples

```
p <- c(0.5,0.9,0.99)
# calulate quantiles for one duration
qgev.d(p=p,mut=4,sigma0=2,xi=0,theta=0.1,eta=0.3, d=1)

# calculate quantiles for sequence of durations
ds <- 2^seq(0,4,0.1)
qs <- lapply(ds,qgev.d,p=p,mut=4,sigma0=2,xi=0,theta=0.1,eta=0.3)
qs <- simplify2array(qs)

plot(ds,qs[1,],ylim=c(3,20),type='l',log = 'xy',ylab='Intensity',xlab = 'Duration')
for(i in 2:3){
  lines(ds,qs[i,],lty=i)
}
legend('topright',title = 'p-quantile',
       legend = p,lty=1:3,bty = 'n')
```

---

rgev.d — *Generation of random variables from d-GEV*

Description

Generation of random variables following duration-dependent GEV.

Usage

```
rgev.d(n, mut, sigma0, xi, theta, eta, d, tau = 0, eta2 = NULL)
```

Arguments

`n`	number of random variables per duration
`mut, sigma0, xi`	numeric value, giving modified location, modified scale and shape parameter
`theta`	numeric value, giving duration offset (defining curvature of the IDF curve)
`eta`	numeric value, giving duration exponent (defining slope of the IDF curve)
`d`	positive numeric value, giving duration
`tau`	numeric value, giving intensity offset τ (defining flattening of the IDF curve). Default: $\tau = 0$.
`eta2`	numeric value, giving a second duration exponent η_2 (needed for multiscaling). Default: NULL, treated as $\eta_2 = \eta$.

Details

For details on the d-GEV and the parameter definitions, see IDF-package

Value

list containing vectors of random variables. The first element of the list are the random values for the first given duration etc. Note that the random variables for different durations are nor ordered (contrary to precipitation maxima of different durations).

See Also

pgev.d, qgev.d, dgev.d

Examples

```
# random sample for one duration
rgev.d(n=100,mut=4,sigma0=2,xi=0,theta=0.1,eta=0.3,d=1)

# compare randomn samples for different durations
ds <- c(1,4)
samp <- lapply(ds,rgev.d,n=100,mut=4,sigma0=2,xi=0,theta=0.1,eta=0.3)

hist(samp[[1]],breaks = 10,col=rgb(1,0,0,0.5),freq = FALSE
     ,ylim=c(0,0.3),xlim=c(3,20),xlab='x',main = 'Random d-GEV samples')
hist(samp[[2]],breaks = 10,add=TRUE,col=rgb(0,0,1,0.5),freq = FALSE)
legend('topright',fill = c(rgb(1,0,0,0.5),rgb(0,0,1,0.5)),
legend = paste('d=',1:2,'h'),title = 'Duration')
```

Index